International Journal of communication and Computer Technologies, 6 (1) (2018)

International Journal of Communication & Computer Technologies

Website: www.ijccts.org
ISSN :2278-9723
Research paper

An improved RSA image encryption algorithm using 1-D logistic map

Yakubu[1] H. J., Aboiyar[2] T., Zirra[3] P. B.

[1]*Department of Mathematics/Statistics/Computer Science, University of Maiduguri, Maiduguri, Nigeria.*
[2]*Department of Mathematics/Statistics/Computer Science, University of Agriculture, Makurdi, Benue, Nigeria.*
[3]*Department of Mathematics and Computer Science, Federal University, Kashere, Gombe, Nigeria.*
***Email:** thejoe_gdf@yahoo.com, hurcha65@gmail.com*

Abstract

The need to have a more secured ways of protecting sensitive images in this modern age of technology has become necessary for ensuring that such images sent via the insecure public network called the Internet are protected from attacks. To this end, various image encryption algorithms have been proposed by many researchers and RSA image encryption algorithm is one of them. Studies on RSA image encryption algorithm reveal that some images when encrypted using the algorithm, gives encrypted images that do expose some features of the plain images on visual inspection no matter the size of primes used. This is due to the fact that digital images have high redundancy and high correlation of image data that make RSA algorithm image dependent. Cryptanalysis has been a major source of concern (in particular the brute-force attack) when designing a cryptosystem since it is assumed that the cryptanalyst knows exactly the design and working of the cryptosystem under study except the secret key. Thus, making the secret key access much more difficult to the cryptanalyst makes the encryption algorithm a more reliable one. In view of these, an improved RSA image encryption algorithm using 1-D logistic map was proposed. The proposed scheme has two stages. In the first stage, the plain-image is shuffled using the chaotic properties of the map and in the second stage; the shuffled image is encryption using the RSA algorithm. The proposed algorithm and the RSA algorithm were tested on two standard test gray scale images: cameraman.tif and clock.tif using four different set of keys. Security analysis such as histogram analysis and correlation coefficient analysis were also carried out on the results obtained from the two methods. The results of the analysis show that the proposed scheme is more secured and stronger against the brute-force attack (encrypted image which does not reveals any hint about the plain image to the attacker and also its key space has double: the two primes, initial condition and control parameter) than the RSA image encryption scheme.

Keywords: *Brute-force attack; Chaos; Cryptanalyst; Image encryption, Public/Private-key,*

1. Introduction

Cryptography came into existence as a means to enable parties to maintain privacy of the information they exchange even in the presence of an adversary with access to the communication channels. It has been used almost since writing was invented [8]. The methodology of concealing the content of messages comes from the Greek words kryptos, meaning hidden and graphikos, meaning writing [10]. Different authors defined cryptography in different ways. Cryptography is also defined as the science of keeping secrets secret [2], and according to [9], cryptography is defined as the design and analysis of mathematical techniques that enables secure communications in the presence of malicious adversaries, while [5] defined cryptography as the art of building encryption schemes that allows secret data exchange over insecure channels. The fundamental and classical goal of cryptography is to provide confidentiality by encryption methods, which is the technique that transforms information to be transmitted into an unreadable and unintelligent form by encryption process so that only authorized persons can correctly recover the information by decryption process [2, 3]. Furthermore, the field of cryptography now encompasses much more than secret communication. It includes message authentication, digital signatures and protocol for exchanging secret keys [1, 10]. Cryptography is basically divided into two categories: (**i**) The Symmetric-key cryptography where the sender and the receiver each have a single secret key that are alike which is used both for encryption and decryption (i.e. $K_e = K_d$). The key must be transmitted from sender to receiver via a separate secret channel. (**ii**) The Asymmetric-key cryptography (also called Public-key cryptosystem) is where each party involved has a pair of different keys that are mathematically linked (K_e, K_d). The encryption key K_e which is made public is different from the decryption K_d that is kept secret (i.e. $K_e \neq K_d$). No additional secret channel is needed for the key transfer [2, 6, and 13]. Symmetric key cryptosystem provides a secured communication channel to each pair of users after agreeing on a common secret key which is being shared between the pair. It also provides confidentiality and data integrity. However, secured delivery of a secret key is observed to be its major setback. Other weaknesses observed are lack of good methods for authentication and non-repudiation [2, 7 and 15]. The public-key cryptosystem which has provided a secured delivery of secret key and also has protocols that provide authentication and non-repudiation was introduced by [7]. Since 1976, numerous public-key encryption algorithms have been proposed; the three most widely used public-key cryptosystems are the Rivest-

Shamir-Adleman (RSA), the ElGamal, and the Rabin cryptosystems but among these, RSA algorithm happens to be the most popular and secured public-key encryption method [2, 4, and 10].

The RSA encryption algorithm was primarily designed for text and its application has now been extended to digital images. Despite the fact that its application on digital images are found not suitable as the chaotic schemes, its application in some areas remain very vital in particular, areas such as medical imaging where every bit of information in the plain-image is important and needs to be recovered in its decrypted image. However, studies have shown that RSA algorithm is image dependent [12], indicating that some encrypted images obtained with the RSA algorithm do expose some hints about their plain images on visual inspection irrespective of the prime size used and with primes below 25 the plain image is completely exposed in the encrypted image. In view of this, an improved RSA image encryption algorithm was proposed using 1-D logistic map for gray scale images.

2. Related literature

Some important tools in algebra and number theory that laid the foundation for building the RSA encryption algorithm are presented in this section.

If $a, b \in \mathbb{Z}$, the set of integers, then $a+b$, $a-b$, and $a \cdot b$, all belongs to $\mathbb{Z}$. However, not all a/b belong to $\mathbb{Z}$. This leads to the fundamental concept of divisibility.

Definition 2.1: *Let $a, b \in \mathbb{Z}$, with $b \neq 0$. We say that b divides a, or that a is divisible by b, if there is a number $c \in \mathbb{Z}$ such that $a = b \cdot c$. $b|a$ to indicate that b divides a.*

Proposition 2.1 (Hoffstein *et al*., 2008):
Let $a,b,c \in \mathbb{Z}$, then the following hold:

1. *If $a|b$ and $b|c$ then $a|c$.*
2. *If $a|b$ and $b|a$, then $a = \pm b$.*
3. *If $a|b$ and $a|c$ then $a|(b+c)$ and $a|(b - c)$.*

Definition 2.2: *A common divisor of $a,b \in \mathbb{Z}$ is a number $d \in \mathbb{Z}^+$ such that $d|a$ and $d|b$.*

Definition 2.3: *The greatest common divisor of $a,b \in \mathbb{Z}$ is the largest number $d \in \mathbb{Z}^+$ such that $d|a$ and $d|b$. It is denoted by gcd(a,b).*

An efficient method for computing gcd has been developed based on the concept of long division with remainder and is called the Division Algorithm.

Definition 2.4 *(Division Algorithm): Let $a,b \in \mathbb{Z}^+$, then $a|b$ has a quotient q and remainder r means that*

$$a = b \cdot q + r; \qquad 0 \leq r < b \tag{1}$$

If *d* is any common divisor of *a* and *b*, then it is clear from (1) that *d* is also a divisor of *r*. Similarly, if *e* is a common divisor of *b* and *r* then from (1) again *e* is a divisor of *a*. Hence, gcd(*a*,*b*) = gcd(*b*,*r*). We repeat the process, dividing *b* by *r* to get another quotient and remainder $b = r \cdot q' + r'$ with $0 \leq r' < r$. The same reasoning shows that the gcd(*b*,*r*) = gcd(*r*,*r*'). Continuing this process, the remainder eventually gets to 0, at which the final value of gcd(*s*,0) = *s* is equal to the gcd(*a*,*b*).

Theorem 2.1 (Extended Euclidean Algorithm): *Let $a,b \in \mathbb{Z}^+$. Then the equation $au + bv = \gcd(a,b)$ always has a solution in integer's u and v (Hoffstein et al., 2008).*

Definition 2.5: *An integer $p > 1$ is called a **prime number** or simply **prime** if 1 and p are the only divisors of p otherwise, p is called **composite**.*

Definition 2.6: *Let $a,b \in \mathbb{Z}$, we say that a and b are relatively prime (also called coprime) if $\gcd(a,b) = 1$.*

Definition 2.7: *Let $n \in \mathbb{Z}^+$ and $n \geq 2$, we say that $a,b \in \mathbb{Z}$ are congruent modulo n if $n|(a - b)$. Written as $a \equiv b \pmod{n}$; where n is called the modulus.*

Proposition 2.2 (Hoffstein *et al*., 2008): *Let $n \in \mathbb{Z}^+$ and $a_1, a_2, b_1, b_2 \in \mathbb{Z}$ so, If $a_1 \equiv a_2 \pmod{n}$ and $b_1 \equiv b_2 \pmod{n}$, then $a_1 \pm b_1 \equiv a_2 \pm b_2 \pmod{n}$ and $a_1 \cdot b_1 \equiv a_2 \cdot b_2 \pmod{n}$.*

Proposition 2.3 (Hoffstein *et al*., 2008, Delfs and Knebl, 2007): *Let $n \in \mathbb{Z}^+$ and $a \in \mathbb{Z}$ then $a \cdot b \equiv 1 \pmod{n}$ for some $b \in \mathbb{Z}$ if and only if $\gcd(a,n) = 1$. If such an integer b exists, we say that b is the (multiplicative) inverse of a modulo n.*

Exponentiation is also an operation that occurs often in cryptography, we often have to compute a power α^e or a modular power $\alpha^e \bmod n$. This can be done efficiently by the fast exponentiation algorithm. The idea is that if the exponent *e* is a power of 2, say $e = 2^k$, then we can exponentiate by successive squarings: that is, $\alpha^e = \alpha^{2^k} = ((((: : : (\alpha^2)^2)^2 : : :)^2)^2)^2$:. In this way, we compute α^e by *k* squarings. For example, $\alpha^{16} = \alpha^{2^4} = (((\alpha^2)^2)^2)^2$. However, if the exponent is not a power of 2, we use the binary expansion of the exponent *e* to convert the calculation of α^e into a succession of squaring and multiplications. For example, to evaluate α^{29} using this approach, first we write 29 in binary form: $29 = 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$; it then follows that $\alpha^{29} = \alpha^{1 \cdot 2^4} \cdot \alpha^{1 \cdot 2^3} \cdot \alpha^{1 \cdot 2^2} \cdot \alpha^{0 \cdot 2^1} \cdot \alpha^{1 \cdot 2^0} = ((((\alpha^2 \cdot \alpha)^2 \cdot \alpha)^2 \cdot \alpha^0)^2 \cdot \alpha$. Thus, only four squarings and three multiplications are needed to compute $\alpha^{29} \bmod n$ as compared to naive approach. It is important that the reduction modulo *n* be done at each squaring or multiplication to avoid large intermediate integers. Based on the above idea, if then follows that, for any $\alpha \in \mathbb{Z}_n$ and $e > 0$, $\alpha^e \bmod n$ can be computed efficiently.

Proposition 2.3 (Hoffstein *et al*., 2008, Delfs and Knebl, 2007): *Let p be a prime and let $e \in \mathbb{Z}^+$ satisfying gcd(e,p-1) = 1. Then Proposition 2.3 tells us that e has an inverse modulo $p-1$ say $de \equiv 1 \pmod{p-1}$ then the congruence $x^e \equiv c \pmod{p}$ has the unique solution $x \equiv c^d \pmod{p}$.*

Proposition 2.6 (Hoffstein *et al*., 2008, Delfs and Knebl, 2007): *Let p and q be two distinct primes and let $e \in \mathbb{Z}^+$ satisfying gcd(e,(p−1)(q−1)) = 1. From Proposition 2.3 we see that e has an inverse modulo $(p-1)(q-1)$ say $de \equiv 1 \pmod{(p-1)(q-1)}$, then the congruence $x^e \equiv c \pmod{pq}$ has the unique solution $x \equiv c^d \pmod{pq}$.*

3. Methodology

3.1. The RSA algorithm

The RSA algorithm was first described in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman of the Massachusetts Institute of Technology and in 1998 it became the first public-key cryptosystem published that could function as both a key agreement mechanism and as a digital signature [13, 14 and 17]. In this algorithm, each communicating party uses two different but mathematically linked keys called the public-key and the private-key. The

public-key is made public for encryption purpose, whereas the private-key must be kept secret by the owner for decryption purpose [1, 10, 12 and 13]. Either of the keys can be used for encryption, the opposite key from the one used to encrypt a message is used to decrypt it. The security of the RSA encryption function depends on the tremendous difficulty of factoring, but the equivalence is not proven. Multiplying two large primes is easy but determining the two prime factors from the product is considered infeasible due to time it would take even with today's Super computers [2]. This algorithm provides a method for assuring the confidentiality, integrity, authenticity and non-reputability of electronic communications and data storage. These attributes have made RSA become the most widely used asymmetric-key encryption algorithm which had stood the test of time to this day, where it is used in cryptographic applications from banking, and e-mail security to e-commerce on the Internet [1]. The RSA algorithm was primarily designed for text and now has been extended to digital images.

A digital image is an image possessing both spatial (layout) and intensity (colour) information that are finite and discrete. Thus, one can therefore consider digital image as a large array of integer numbers represented by dots, each of which has some measure of brightness associated with it and these dots are called picture elements or simply pixels [19]. The application of the RSA algorithm to digital images goes from pixel to pixel until the entire image is encrypted or decrypted. The detail algorithms for the RSA image encryption/decryption as well as key generation are presented:

3.2. The RSA image encryption algorithm

3.2.1. Key generation algorithm

1. Generate two distinct large primes' p and q,
2. Compute the modulus n as $n = pq$ and $\varphi(n) = (p-1)(q-1)$,
3. Chose public exponent e to be relative prime to $\varphi(n)$, with $1 < e < \varphi(n)$,
4. Form the pair (n,e) and publish it as public-key,
5. Find an integer d with $1 < d < \varphi(n)$ such that $ed \equiv 1 \bmod \varphi(n)$,
6. Form the pair (n,d) and keep it secret as secret-key.

3.2.2. Encryption algorithm

1. Load the image M.
2. Convert the image to double
3. Encrypt the image with the public-key to obtain the cipher-image using $c_i = E_{(n,e)}(m_i) = m_i^e (\bmod\ n)$.
4. Convert the cipher-image to integer.
5. Save the cipher-image in a file.
6. Display the encrypted image.

3.2.3. Decryption algorithm

1. Load the cipher-image file
2. Convert the cipher-image to double.
3. Decrypt the cipher-image with the secret-key to obtain the image using $m_i = D_{(n,d)}(c_i) = c_i^d (\bmod\ n)$
4. Convert the decrypted image to integer.
5. Save the decrypted image in a file
6. Display the decrypted image.

It is important to **note** that in computing the private exponent d, we use the relation $de \equiv 1 \pmod{L}$, where L is the lowest common multiple of $p-1$, and $q-1$. Table 1 presents a small sample of encryption and decryption of pixel values ranging from 0 to 9 using the RSA algorithm.

Table 1: A small sample of encrypted/decrypted pixels' values ranging from 0 to 9 using the RSA algorithm.

Key Pair Public key: n = 55, e = 3 Private key: n = 55, d = 7			**Key Pair Generation** Primes: p = 5, q = 11 Modulus: n = pq = 55 Public exponent: e = 3 Private exponent: $d = 3^{-1} \bmod 20 = 7$			
Mes sage	**Encryption** $c = m^3 \bmod n$		**Decryption** $m = c^7 \bmod n$			
m	$m^2 \bmod n$	$m^3 \bmod n$	$c^2 \bmod n$	$c^3 \bmod n$	$c^6 \bmod n$	$c^7 \bmod n$
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	4	8	9	17	14	2
3	9	27	14	48	49	3
4	16	9	26	14	31	4
5	25	15	5	20	15	5
6	36	51	16	46	26	6
7	49	13	4	52	9	7
8	9	17	14	18	49	8
9	26	14	31	49	36	9

3.3. Chaotic dynamical behaviour of the 1-D logistic map

One of the most studied examples of a one-dimensional system capable of various dynamical regimes including chaos is the 1-D logistic map. It is a representation of an idealized population model and is defined by the equation

$$x_{n+1} = f(x_n) = \mu x_n(1 - x_n) \tag{2}$$

where $x_n \in [0,1]$ and it represents the population at year n, and hence x_0 represents the initial population at year 0. Crucial to the behaviour of the map is the control parameter $\mu \in [0,4]$ whose dynamical behaviour is very complicated and it represents a combined rate for reproduction and starvation. Slight changes in the parameter, "μ", of the map can cause the iterated map to change from stable and predictable behaviour to unpredictable behaviour which is called chaos [6, 11]. Figure 1 shows the bifurcation diagram of the 1-D logistic map. Analysis of the map shows that when the control parameter μ is in the interval [0, 3.57], the system remained stable and predictable, for $3.57 < \mu \leq 4.1$ the system becomes chaotic and for values of $\mu > 4.0$, all the orbits escape to infinity.

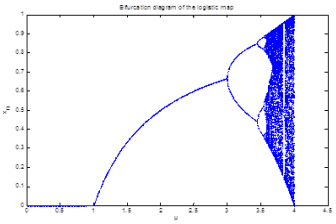


Fig. 1: Bifurcation diagram of the logistic map for $\mu \in [0,4]$

3.4. Proposed improved RSA image encryption algorithm using 1-D logistic map

In this algorithm, the key generation for encryption/decryption is the same with that of the RSA image encryption algorithm. Two additional keys are required in this algorithm: initial condition and control parameter for shuffling the plain-image. After the key generation, the plain image is shuffled using the chaotic properties of the 1-D logistic map. The shuffled image is then encrypted with the RSA algorithm to obtain encrypted image. The cipher-image is obtained by first applying the RSA decryption algorithm on the

encrypted image to obtain a confused image. The pixels' value of the confused image is then reposition to their original position using the same chaotic properties of the map used in shuffling the plain-image. The details of the algorithm are presented:

3.4.1. Key generation algorithm

1. Generate two distinct large primes' p and q,
2. Compute the modulus n as $n = pq$ and $\varphi(n) = (p-1)(q-1)$,
3. Chose public exponent e to be relative prime to $\varphi(n)$, with $1 < e < \varphi(n)$,
4. Form the pair (n,e) and publish it as public-key,
5. Find an integer d with $1 < d < \varphi(n)$ such that $ed \equiv 1 \bmod \varphi(n)$,
6. Form the pair (n,d) and keep it secret as secret-key.

3.4.2. Encryption algorithm

1. Read the Plain-image I.
2. Obtain the image dimension as a and b.
3. Compute the number of Pixels in I as N
4. Read the initial condition and control parameter (x_1, and μ),
5. Evaluate the logistic map up to N-1 times to generate vector X.
6. Confuse the vector X with the round function.
7. Sort the vector X to obtain its index.
8. Use the index to scramble the plain-image I.
9. Convert the scrambled image I to double and called it M.
10. Encrypt the scrambled image M with the public-key to obtain the cipher-image using $c_i = E_{(n,e)}(m_i) = m_i^e (\bmod n)$.
11. Convert the cipher-image to integer
12. Save the cipher-image in a file named C.
13. Display the encrypted image.

3.4.3. Decryption algorithm

1. Load the cipher-image file C.
2. Convert the cipher-image to double.
3. Decrypt the cipher-image with the secret-key to obtain the scrambled image Q using $m_i = D_{(n,d)}(c_i) = c_i^d (\bmod\ n)$
4. Obtain the image dimension of Q as a and b.
5. Compute the number of Pixels in Q as N.
6. Read the initial condition and control parameter (y_1, and μ),
7. Evaluate the logistic map up to N-1 times to generate vector Y.
8. Confuse vector Y with the round function.
9. Sort vector Y to obtain its index.
10. Use the index to reposition pixels' values of the scrambled image back to their original position to obtain an image D1.
11. Convert the image D1 to integer to obtain the decrypted image.
12. Save the decrypted image in a file.
13. Display the decrypted image.

It is important to **note** that y1 and x1 must be equal and the μ must be the same in both the encryption and decryption algorithms.

4. Results and discussion

In carrying out the practical aspect of this work, two standard test digital gray scale images (cameraman.tif and clock.tif) were used as our data for encryption, both of size 256x256 and stored with TIF file format. The codes were implemented in MATLAB to simulate the algorithms for the RSA and the proposed RSA. Both algorithms were applied to the same plain-images (see Fig. 2) using four different pairs of public and private keys ranging from shorter to longer key length.

First, the RSA image encryption algorithm was applied to the plain images and the results are shown in Fig. 3 while Fig. 5 and 6 show the results obtained from the application of improved RSA image encryption algorithm on the plain images. Visual inspection of the cipher images in Fig. 3 reveals that the cipher images have exposed some features of the plain–images under consideration irrespective of the prime size used thereby aiding the attacker with some information about the plain images. However, the situation is completely different when the cipher images in Fig. 4 and 5 were observed, here, no any feature of the plain-images is revealed. This indicates that all the features of the plain-images are completely hidden on visual inspection. Hence, the proposed scheme is effective. Decrypted images with either the RSA or the proposed improved RSA are exact replica of the plain-images as can be seen clearly from their respective histograms shown in Figs. 6 to 9.

(a) (b)

Fig. 2: Plain-images: (a) Cameraman.tif, (b) Clock.tif

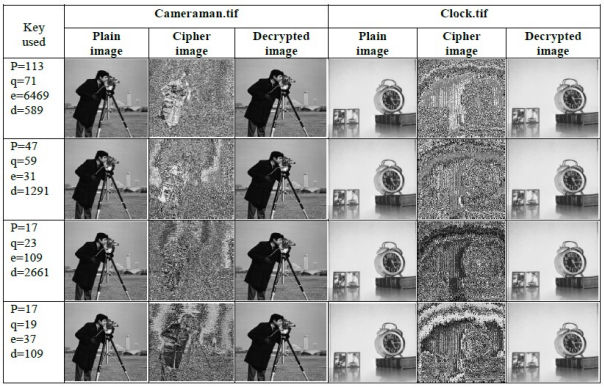


Fig. 3: Plain, cipher and decrypted gray images of cameraman and clock using RSA

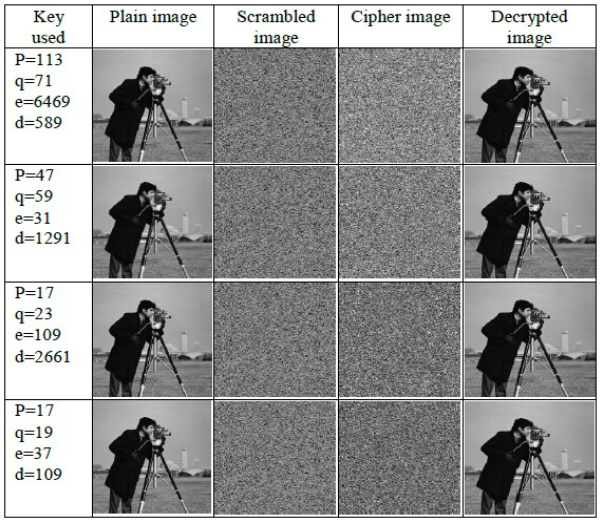


Fig. 4: Plain, scrambled, cipher and decrypted gray image of cameraman using improved RSA

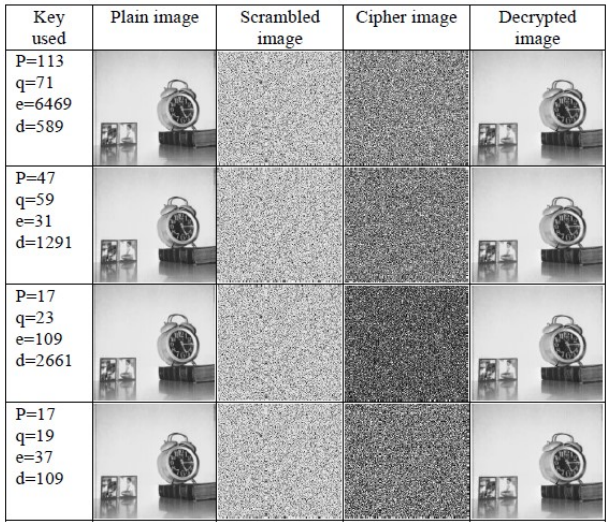


Fig. 5: Plain, scrambled, cipher and decrypted gray image of clock using improved RSA

5. Security analysis

When an encryption algorithm is applied to an image, it is expected that its pixel values change when compared with the original image. A good encrypted image must be composed of totally random patterns that do not reveal any of the features of the original image [2, 3]. To test the robustness of the proposed scheme, security analysis such as the histogram uniformity analysis and the correlation coefficient analysis were performed.

5.1. Histogram uniformity analysis

For image encryption algorithm to be considered worthy of use, the histogram of the encrypted image should satisfy these two properties [3]:

1. It must be totally different from the histogram of the original image.

2. It must have a uniform distribution, which means that the probability of occurrence of any gray scale value is the same.

Figures 6 to 9 show the histograms of the plain, scrambled, cipher and decrypted gray images of cameraman and clock. From these figures, one can see clearly that the histograms of the encrypted images are completely different from their histograms of their plain images which satisfied condition **1** of the histogram uniformity analysis. Also observed is that the histograms of the encrypted images have uniform distribution. Thus, condition **2** is also satisfied indicating that the proposed scheme is worthy of use.

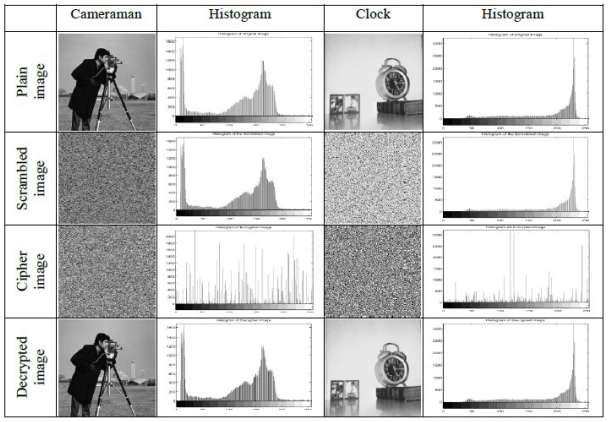


Fig. 6: Histogram of the plain, scrambled, cipher and decrypted gray images of cameraman and clock with p=113, q=71, e=6469 and d=589.

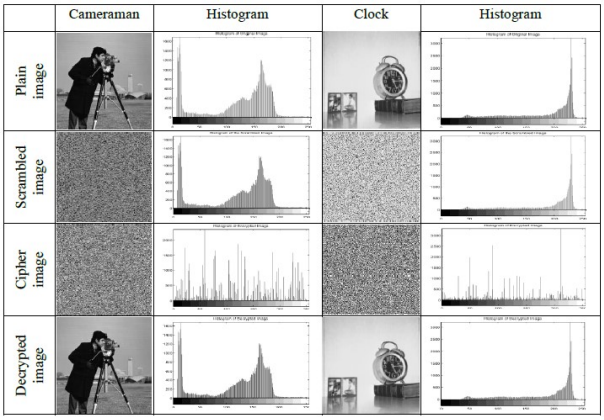


Fig. 7: Histogram of the plain, scrambled, cipher and decrypted gray images of cameraman and clock with p=47, q=59, e=31 and d=1291

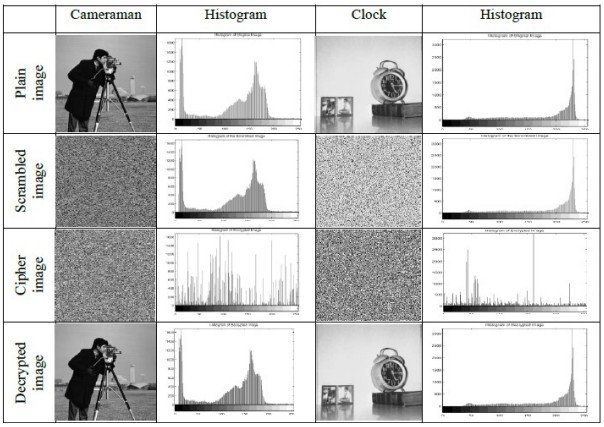


Fig. 8: Histogram of the plain, scrambled, cipher and decrypted gray images of cameraman and clock with p=17, q=23, e=109 and d=2661.

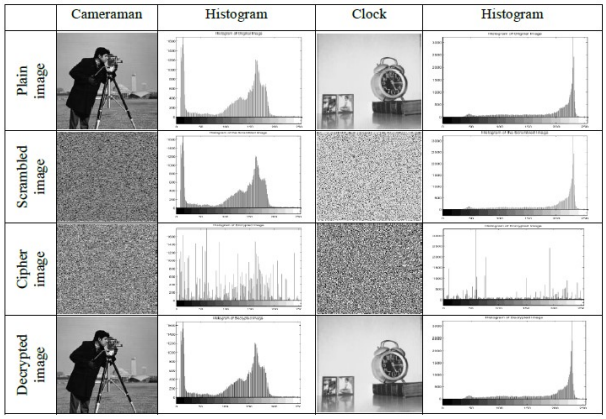


Fig. 9: Histogram of the plain, scrambled, cipher and decrypted gray images of cameraman and clock with p=17, q=19, e=37 and d=109

5.2. Correlation coefficient analysis

One useful metric for assessing the encryption quality of any image encryption scheme is the correlation coefficient between adjacent pixels of the cipher-image. In the proposed encryption algorithm, we analyzed the correlation between two *vertically* adjacent pixels, two *horizontally* adjacent pixels and two *diagonally* adjacent pixels in the cipher-image. We also obtained the same correlation coefficients in the plain-image for comparison purposes. This metric is calculated as follows:

$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \quad (3)$$

where x and y are the values of two adjacent pixels in the cipher-image. In numerical computations, the following discrete formulas can be used:

$$E(x) = \frac{1}{L}\sum_{l=1}^{L} x_l \quad D(x) = \frac{1}{L}\sum_{l=1}^{L} (x_l - E(x))^2 \quad (4)$$

$$Cov(x, y) = \frac{1}{L}\sum_{l=1}^{L} (x_l - E(x))(y_l - E(y)) \quad (5)$$

where L is the number of pixels involved in the calculations. The closer the value of r_{xy} to zero, the better the quality of the encryption algorithm will be [3, 16 and 18]. In carrying out the practical aspect of this work, 3000 pixels out of 65536 pixels in each image were considered and the results of the analysis are presented in Fig. 10 and 11.

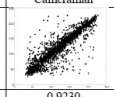
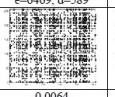
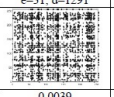
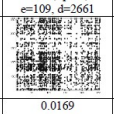
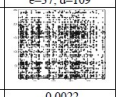
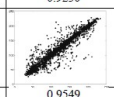
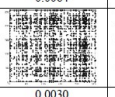
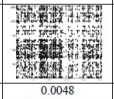
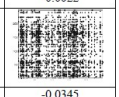
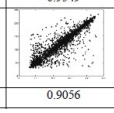
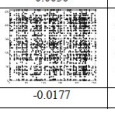
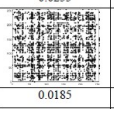
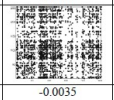
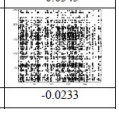
	Plain-Image of Cameraman	Cipher Image of Cameraman			
		P=113, q=71, e=6469, d=589	P=47,q=59, e=31, d=1291	P=17, q=23, e=109, d=2661	P=17, q=19, e=37, d=109
Horizontal					
Corr.	0.9230	0.0064	0.0039	0.0169	-0.0022
Vertical					
Corr.	0.9549	0.0030	0.0255	0.0048	-0.0345
Diagonal					
Corr.	0.9056	-0.0177	0.0185	-0.0035	-0.0233

Fig. 10: Correlation between adjacent pixels of the plain and cipher gray image of cameraman.

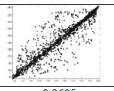
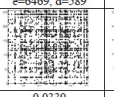
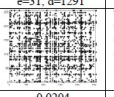
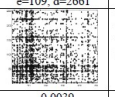
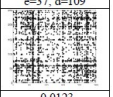
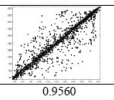
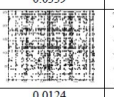
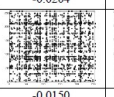
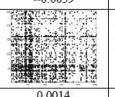
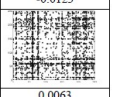
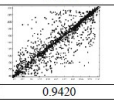
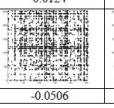
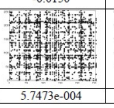
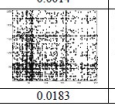
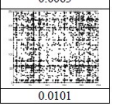
	Plain-Image of Clock	Cipher Image of Clock			
		P=113, q=71, e=6469, d=589	P=47,q=59, e=31, d=1291	P=17, q=23, e=109, d=2661	P=17, q=19, e=37, d=109
Horizontal					
Corr.	0.9605	0.0339	-0.0204	-0.0039	-0.0123
Vertical					
Corr.	0.9560	0.0124	-0.0150	0.0014	0.0063
Diagonal					
Corr.	0.9420	-0.0506	5.7473e-004	0.0183	0.0101

Fig. 11: Correlation between adjacent pixels of the plain and cipher gray image of clock.

From these Figures, we can see that the plain images are highly correlated while the cipher images have almost zero correlation among the adjacent pixels in all the three directions (vertical, horizontal and diagonal) as these can be seen clearly from their respective correlation coefficient values. These results indicate that the attacker would not be able to figure out any information about the plain image from the relationships between the pixels in the cipher images. Thus, the proposed scheme is effective.

6. Conclusion

To improve the security of image transmission, a more secured cryptosystem was proposed in this paper which was achieved by utilizing the rich chaotic properties of a modified 1-D logistic map to shuffle the image. The encrypted image is obtained using the RSA image encryption algorithm on the shuffled image. The proposed scheme and the RSA algorithm were both tested on two standard test gray scale images (Cameraman.tif and Clock.tif). Security analysis such as Histogram uniformity analysis and Correlation coefficient analysis were performed on the proposed scheme. From the experimental results obtained, the proposed scheme is highly secured and stronger against the brute-force attack.

References

[1] Taki El-Deen, A. E., El-Badawy, E. A., and Gobran, S. N., "Digital Image Encryption Based on RSA Algorithm", *Journal of Electronics and Communications Engineering*, vol. 9, No.1, (2014), pp.: 69-73.

[2] Delfs, H., and Knebl, H., "Introduction to Cryptography-Principles and Applications", 2nd Ed., Springer Berlin Heidelberg, New York, USA, (2007), pp.: 1-65.

[3] Abd-El-Samie, E. F., Ahmed, H. E. H., Elashry, F. I, Shahieen, H. M., Faragallah, S.O., El-Rabaie, M. E., and Alshebeili, A. S., Image Encryption- A Communication Perspective,1st Ed., CRC Press, London, (2014), pp.: 1-86.

[4] Chandel, J. S., and Patel, P., "A Review: Image Encryption with RSA and RGB Randomized Histograms", *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, N0. 11, (2013), pp.:4397 – 4401.

[5] Goldreich, O., Foundations of Cryptography-Basic Techniques, 2nd Ed., Cambridge University Press, UK, (2004), pp.: 1-63.

[6] Alligood, K. T., Sauer, T. D., and Yorke, J. A., CHAOS-An Introduction to Dynamical Systems, Springer-Verlag, New York, Inc. USA, (1996), pp.: 1-147.

[7] Diffie, W., and Hellman, M. E., "New Directions in Cryptography", IEEE Transactions on Information Theory, vol. 22, No. 6, (1976), pp.: 644-654.

[8] Ballare, M., and Rogaway, P., Introduction to Modern Cryptography- Principles and Protocols. 1st Ed., CRC Press Book, California, USA, (2005), pp.: 1-35.

[9] Hankerson, D., Menezes, A. and Vanstone, S., *Guide to Elliptic Curve Cryptography*, Springer-Verlag Inc., New York, USA, (2004), pp.: 10-15.

[10] Hoffstein, J., Pipher, J. and Silverman, J. H., An Introduction to Mathematical Cryptography, 1st Ed,. Springer Science + Business Media, New York, USA, (2008), pp.: 10-65.

[11] Biswas, R. H. "One Dimensional Chaotic Dynamical System". *Journal of Pure and Applied Mathematics: Advances and Applications*, vol. 10, N0.1 (2013), pp.: 69-101.

[12] Yakubu, H. J. and Aboiyar, T., "Comments on Rivest-Shamir-Adleman (RSA) Image Encryption Algorithm", *Nigerian Journal of Pure and Applied Sciences*, vol. 8, No. 1, (2016). Pp.: 216-223

[13] Kaliski, B., The Mathematics of the RSA Public-key Cryptosystem. (2012), http://www.mathaware.org/mam/06/Kaliski.pdf.

[14] Kokarev, L., and Makraduli, J., "Public-Key Encryption based on Chebyshev Polynomials", *Circuits, Systems and Signal Processin,* vol. 24, No. 5, (2005), pp.: 497-517.

[15] Stinson, D. R., Cryptography Theory and Practice, 3rd Ed., Chapman & Hall/CRC. New York, (2006), pp.: 1-186.

[16] Sathishkumar,G. A., Bagan, K. B., and Sriraam, N., "Image Encryption Based on Diffusion and Multiple Chaotic Maps", *International Journal of Network Security and its Applications*, vol. 3, No. 2, (2011), pp.: 181 – 194.

[17] Mishra, M., and Mankar, V. H., "Chaotic Encryption Scheme Using 1-D Chaotic Map", *International Journal of Communications, Network and System Sciences*, vol. 4, No.10, (2011), pp.: 452 – 455.

[18] Ramahrishnan, S., Elakkiya, B., Geetha, R., and Vasuki, P., "Image Encryption Using Chaotic Maps in Hybrid Domain", *International Journal of Communication and Computer Technologies*, vol. 2, No. 5, (2014), pp.: 44 – 48.

[19] Gonzalez, R. C., Woods, R. E. and Eddins, S. L., *Digital Image Processing Using MATLAB*, 2nd Ed., Gatesmark Publishing-LLC., USA, (2009), pp.: 5-35.