
Attack resistant chaos-based cryptosystem by modified baker map and logistic map

Debanjan Chatterjee

Department of Computer Science and Engineering,
Indian Institute of Technology,
Kanpur, 208016, India
Email: debanjan20@iitk.ac.in

Barnali Gupta Banik*

Department of Computer Science and Engineering,
Koneru Lakshmaiah Education Foundation,
KL Deemed to be University,
Hyderabad, Telangana 500075, India
Email: barnali.guptabanik@icee.org
*Corresponding author

Abhinandan Banik

Department of Computer Science and Engineering,
School of Technology,
GITAM University,
Hyderabad, 502329, India
Email: abanik@gitam.edu

Abstract: In this paper, a new substitution-diffusion type chaos-based cryptosystem is proposed, which can encrypt greyscale images having arbitrary resolution. In substitution, image pixels are permuted using a modified form of the discretised 2-D Baker map, followed by a two-step diffusion approach, which employs a chaotic logistic map. The proposed cryptosystem is resistant to brute force attacks (measured by key-space and key-sensitivity analysis), statistical attacks (tested by histogram and chi-square test) and differential attacks (measured against NPCR, UACI, and hamming distance); the proposed method has also been tested for encryption quality, correlation analysis, entropy analysis, and performance analysis by measuring encryption speed as well as time complexity. Therefore, it is sufficiently secured to be used in real-world applications. To prove the unparalleled outcome of the proposed system, four sets of comparisons have been presented with respect to NPCR and UACI, Encryption throughput, and, lastly, with similar and non-similar existing cryptosystems.

Keywords: data security; encryption; image communication; chaos; sensitivity analysis; statistical analysis.

Reference to this paper should be made as follows: Chatterjee, D., Banik, B.G. and Banik, A. (2023) 'Attack resistant chaos-based cryptosystem by modified baker map and logistic map', *Int. J. Information and Computer Security*, Vol. 20, Nos. 1/2, pp.48–83.

Biographical notes: Debanjan Chatterjee has completed his Bachelor's in Computer Science and Engineering from St. Thomas' College of Engineering and Technology, Kolkata, India. Thereafter, he has worked as a System Engineer in one of the renowned MNC in India. At present, he is pursuing his Masters in Computer Science and Engineering from the Indian Institute of Technology, Kanpur.

Barnali Gupta Banik is an Associate Professor at the Department of Computer Science and Engineering in KL University, Hyderabad, India. She has completed her PhD in Computer Science and Engineering from the University of Calcutta, Kolkata. She has over 12 years of teaching experience and over two years of industrial experience working for MNCs in India and the UK. She has authored several research papers in the information security domain, including 15 Scopus indexed and two SCIE indexed articles as of date.

Abhinandan Banik has received his Master's degree in Telecommunication and Software Engineering from BITS Pilani, India. He has over 15 years of industrial experience working for MNCs in India, USA, UK, and Germany. At present, he is working as an Assistant Professor of the Computer Science and Engineering Department in GITAM University, Hyderabad.

1 Introduction

The extensive development of computer science and technology has led to the digitisation of the world. Hence, the necessity for faster and more secure data communication has become paramount in recent years, as mentioned in Papadimitratos and Haas (2006). In this present era of digital data and readily available internet-enabled devices, in every minute, millions of images are captured across the globe. Most of those images are used for data analysis. As the advent of the Internet has shown ample opportunity for data usage, an intrusion is also apparent. With the proliferation of imagery data, there is a demand for securely storing and transmitting images over the internet. Hence the significance of research in the domain of image security will continue till there are full-proof applications available for preserving the confidentiality, integrity, and availability of imagery data. Generally, there are two approaches to maintaining the confidentiality of a digital image. First is information hiding, which uses techniques like steganography and watermarking, demonstrated in Yu et al. (2007) and Katzenbeisser and Petitcolas (2000).

The alternative approach is to encrypt images. Initially, researchers used conventional encryption techniques such as data encryption standard (DES), RSA, advanced encryption standard (AES), as mentioned in Dang and Chau (2000). Zeghid et al. (2007) and Zhao et al. (2010), however, these methods proved incompetent due to specific intrinsic properties of images like high bulk capacity, high redundancy, the strong correlation among adjacent pixels. Because of disadvantages like low security or high computational delay of traditional encryption schemes, chaos-based cryptosystems were introduced, as stated in Mao and Chen (2005). Therefore, the scope of this research article is to recommend an improvised image encryption algorithm overcoming all pre-existing constraints. Hence an attack resilient cryptosystem is proposed with this.

Chaotic systems have several properties – such as ergodicity, pseudo-randomness, aperiodicity, sensitivity to initial parameters, and conditions that are relevant from a cryptographical point of view. In recent times, several chaos-based cryptosystems have been proposed that are composed of alternate rounds of substitution and diffusion. Usually, 2-D or 3-D chaotic maps are utilised for confusion and diffusion purposes, as mentioned in Run-he et al. (2011). Chaotic maps are simple functions that require modest iteration time; thus, they maintain computational efficiency to be used in real-time applications. Henceforth, in this article, a novel chaotic approach has been suggested for secure image encryption and decryption. Although there are quite a few existing works on baker's map (extensively discussed in Section 2 – 'Literature survey'), however, this proposed method has been enhanced in so many perspectives (details have described in Section 3 – 'Proposed method'), and the outcome of that improvement is reflected in Section 4 – 'Experimental results and quality analysis'.

In image encryption, the original image is enciphered using an encryption algorithm and a key. This encrypted image can be reconstructed with a suitable decryption algorithm and the required key. The combination of the encryption algorithm, decryption algorithm, and set of keys are known as cryptosystems. In this article, a substitution-diffusion type chaos-based cryptosystem is being proposed. Substitution is a method of hiding the relationship between cipher and the key, whereas diffusion is a method of hiding the relationship between the original image and the enciphered image. A modified version of the discretised 2-D Baker map is used in the substitution stage. The substitution process is iterated, depending on a secret key. The proposed mapping is bijective in nature; hence the encryption-decryption technique is free from image distortion. The algorithm contains a two-step diffusion approach, one of them employing a chaotic 2-D logistic map. The seed value for the logistic map is generated by a second secret key; the utilisation of two keys instead of one makes the key-space large enough so that it can resist brute force attacks, as suggested in Knudsen and Robshaw (2011). The algorithm results in a highly scrambled encrypted image having a uniform histogram. Series of experiments have been carried out to determine the response of the proposed method against different cryptanalysis attacks. Cryptanalysis is the technique of breaking the cryptosystem by fraudulent means. The experimental results have proved that the proposed technique is highly resistant to brute-force, statistical, and differential attacks. Hence the objective of this proposed work to build a robust, high-quality chaotic cryptosystem has been fulfilled.

2 Literature survey

Few of chaos-based cryptosystems were proposed initially by Matthews (1989), Wang et al. (2012) and Zhu (2012). However, they lacked specific properties that are fundamental to a secure cryptosystem. Most of them were erroneous due to a lack of robustness and nonlinearity. As a result, they were successfully cryptanalysis, as demonstrated by Li et al. (2012, 2013) and Zhang et al. (2012). The method presented in Alvarez and Li (2006a) provides a framework containing guidelines based on which new chaotic cryptosystems should be designed. It focused mainly on three areas: implementation, key management, and security analysis. In Alvarez and Li (2006b), countermeasures were suggested to enhance the security of such chaos-based algorithms.

In Cheng and Guo (2000), an encryption/decryption algorithm is proposed in which each pixel is bitwise XORed or XNORed to one of the two predetermined keys in a chaotic binary manner. In Kumar and Ghose (2011), the authors have proposed and extended substitution-diffusion-based image encryption scheme using a chaotic standard map and a linear feedback shift register, adding nonlinearity to overcome the weakness of previous techniques.

In Ozorio de Almeida and Saraceno (1991) and Schack and Caves (1992), the authors have illustrated that Baker map exhibits inherent properties to be classified as an effective chaotic system. It was studied in Ye and Zhuang (2010) that Baker map can significantly disorder pixels, thus improving the robustness of the system. Because of such excellent scrambling properties, quite a few cryptosystems were developed using Baker map. In Han et al. (2006), an image encryption scheme for bulk size images using a chaotic Baker map has been proposed. The concept of using an improvised pixel shuffling and different keys in different rounds of iterations resulted in a high encryption efficiency and a large key-space. In Hung et al. (2013), another encryption scheme was proposed which combined Scan patterns with Baker map for free-sized images. The parameters for the generalised 2D Baker map were generated using a chaotic Gauss map. After scrambling, the pixels are transformed using a 3D Chen system.

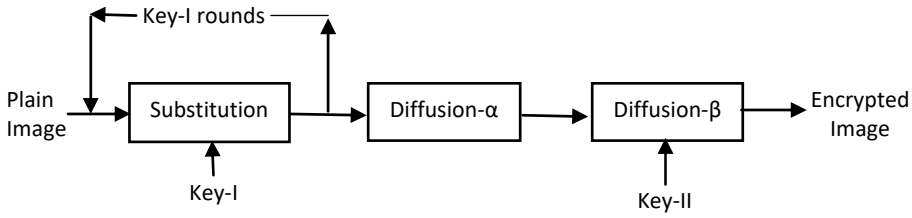
In Subiyakto et al. (2015), the authors have analysed an RGB image encryption technique using a modified Baker map. The modification was based on using several blocks of image and changing the key for every iteration to increase the key-space. The algorithm had decent noise immunity and correlation coefficient values; however, it was still susceptible to key sensitivity and avalanche effect.

Another encryption method has been proposed in Rohith et al. (2014) on a greyscale image, by using a key which has been generated from two separate random sequences. The first pseudorandom sequence is created using a logistic map function. This sequence is then XOR-ed with another random sequence produced from a linear feedback shift register. In Pareek et al. (2006), the authors have proposed a secure cryptographic method that uses an 80-bit secret key to derive the initial conditions for two separate logistic maps which are used for ciphering the image.

3 Proposed method

In this research article, a substitution-diffusion type chaos-based cryptosystem is proposed. Confusion has been achieved by permuting the pixels of the image according to a transformation function in the substitution stage. The pixel values are modified using a two-step diffusion process. The architecture of the proposed encryption algorithm is illustrated in Figure 1.

Figure 1 Encryption model architecture



3.1 Key parameters

In the proposed cryptosystem, substitution and diffusion are two mutually independent stages. The proposed algorithm requires two prerequisite parameters, which also are referred to as secret keys. The first key key-I determines the number of rounds for which the substitution process is repeated. The second key, key-II, is a 20-digit key used to compute the initial parameters for the logistic map used in the Diffusion-β process.

Table 1 Use of keys in the proposed method

Key no.	Key length	Prerequisite for
Key-I	Variable	Substitution
Key-II	20-digit number	Diffusion-β

3.2 Encryption process

Adjacent pixels in an original image have similar pixel values; thus, they have a high correlation. To decorrelate their relationship, pixels should be relocated amongst themselves. Hence, to make encrypted images more resistant to statistical attacks, images need to be scrambled. For this purpose, here improvised version of Baker map has been used, which can perform quick ciphering and deciphering.

3.2.1 Substitution using improvised Baker map

Baker map is a two-dimensional chaotic plot that maps unit square onto itself in a bijective manner. It is called so because it uses a concept similar to the kneading technique applied by Bakers on bread or pasta dough. The unit square or image is divided into two identical vertical rectangles, as shown in Figure 2. Each rectangle undergoes a transformation where its width is doubled, and its height is halved. The two new rectangles formed are placed on top of each other to reform the unit square or shuffled image.

Figure 2 Pictorial representation of the baker map technique

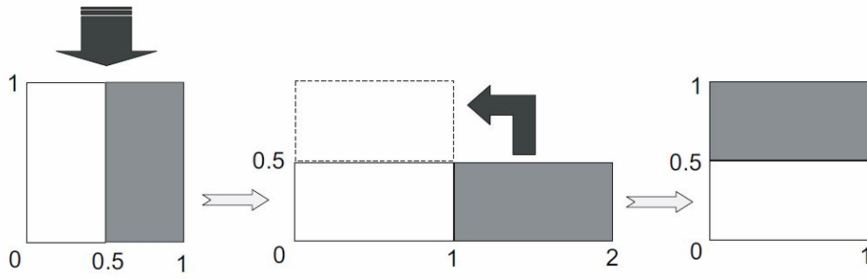


Figure 3 Lena image shuffled using the original discretised Baker map



In Figure 3, the result of Baker map transformation applied to the popular Lena image has been shown. As per Fridrich (1998), Baker map is defined by the following equation (1):

$$(x_{t+1}, y_{t+1}) = \begin{cases} \left(2x_t, \frac{y_t}{2}\right), & \text{if } 0 \leq x_t < \frac{1}{2} \\ \left(2x_t - 1, \frac{y_t + 1}{2}\right), & \text{if } \frac{1}{2} \leq x_t \leq 1 \end{cases} \quad (1)$$

where $x_t, y_t \in [0, 1]$ with $t = 0, 1, 2, 3, \dots$ denotes discrete time. The Baker map needs to be discretised for image pixels to be permuted in a coordinated manner.

In the proposed encryption algorithm, substitution has been achieved by iterating the following three steps for a certain number of rounds given by the value of Key-I.

- a *Segment*: the image is partitioned into two vertical rectangles of equal size in a way similar to the traditional Baker map. Each of the two segments is then compressed using the method explained in the following step. In the algorithm, this step is done in *segment()*.

Input: Greyscale image ($I_{m \times n}$)

Output: 2 greyscale images ($A_{m \times n/2}, B_{m \times n/2}$)

Algorithm: segment($I_{m \times n}$)

Step 1 Find out the size of I and store it in m and n .

Step 2 for $i = 0$ to $m - 1$
 for $j = 0$ to $n - 1$
 if $j < n/2$
 $A(i, j) = I(I, j)$
 else
 $B(I, j - [n/2]) = I(I, j)$
 end if
 end for
end for

b *Compress: the segments get compressed according to the given algorithm*

Input: Greyscale image ($I_{m \times n}$)

Output: Compressed greyscale image ($O_{m/2 \times 2n}$)

Algorithm: compress ($I_{m \times n}$)

Step 1 Find out the size of I and store it in m and n .

Step 2 for $i = 0$ to $m - 1$
 for $j = 0$ to $n - 1$
 if $i \% 2 == 0$
 if $j < n/2$
 $O([i/2], j) = I(I, j)$
 else
 $O([i/2], j + n) = I(I, j)$
 end if
 else
 $O([i/2], j + [n/2]) = I(I, j)$
 end if
 end for
end for

Step 3 End

c *Combine: after the segments are transformed by the above function, the compressed segments are stacked on top of one another and combined to form the ciphered image. Since the concept of segment and combine is like Baker map, hence this technique can be named as improvised Baker map.*

The ciphered image is fed as an input image for the next iteration of the substitution stage. Randomness levels are significantly increased for larger values of key-I. In the algorithm, this step is done in *combine()*.

Input: 2 greyscale images ($A_{m \times n}$, $B_{m \times n}$)

Output: 2 greyscale images ($O_{2 \times m, n}$)

Algorithm: *combine*($A_{m \times n}$, $B_{m \times n}$)

Step 1 Find out the size of A and store it in m and n .

Step 2

```

for i = 0 to 2m - 1
  for j = 0 to n - 1
    if i < m
      O(i, j) = A(I, j)
    else
      O(I, j) = N(i-m, j)
    end if
  end for
end for
end for

```

In Figure 4, the outcome of the improvised Baker map transformation on the same Lena image has been illustrated.

Figure 4 Lena image shuffled using the improvised Baker map



3.2.2 Diffusion process

After the substitution stage, each pixel gets relocated to a different location without any modification in pixel value. As the histogram of the ciphered image is still the same as that of the underlying image, hence, the possibility of being cryptanalysed by statistical attacks remains. Diffusion is a process that changes the value of image-pixels sequentially so that the pixel data of the original image remains concealed. A two-step diffusion process has been used to satisfy this purpose.

- *Step 1: diffusion- α*

This step is a similar technique to the one used in Kumar et al. (2015), where pixels are XOR-ed with one another to obtain a relationship between them. The cipher image generated after this step has a uniform histogram. The scrambled image after substitution is reshaped into a 1-D vector (say $C_{i_{vec}}$). Diffusion takes place according to the following formula:

$$Di_{\alpha}(x) = \begin{cases} C_{i_{vec}}(x) \oplus Di_{\alpha}(x-1), & \text{if } T \neq 1 \\ C_{i_{vec}}(x), & \text{if } T = 1 \end{cases} \quad (2)$$

where $x = 1, 2, 3, \dots, m \times n$. The 1-D vector Di_{α} generated is fed as input to diffusion- β .

In the algorithm, this step is done in *diff1()*.

Input: Grayscale image ($I_{m \times n}$)

Output: Grayscale image ($O_{m \times n}$)

Algorithm: diff1($I_{m \times n}$)

Step 1 Find out the size of I and store it in m and n.

Step 2 Convert $I_{m \times n}$ into a 1D vector ($1 \times mn$)

Step 3 $O(0) = I(0)$

Step 4 for $i = 1$ to $m \times n - 1$

$$O(i) = I(i) \oplus O(i-1)$$

end for

Step 5 Convert $O_{1 \times mn}$ into 2D $O_{m \times n}$

- *Step 2: diffusion- β*

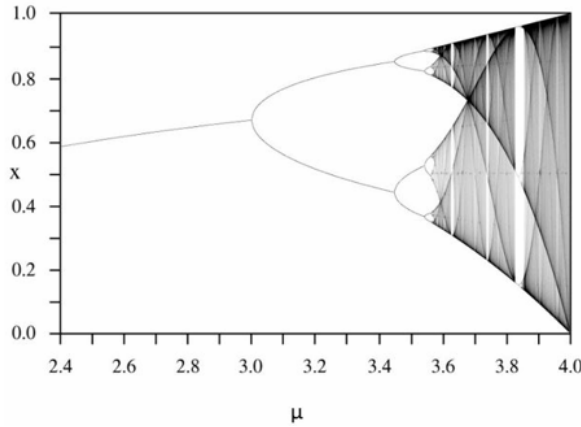
To increase the key-space and key-sensitivity of the proposed algorithm, a second diffusion step is used. This step requires a 20 digit key to generate the seed value for a chaotic logistic map. A Logistic map is mathematically expressed as

$$x_{k+1} = \mu x_k (1 - x_k) \quad (3)$$

where $x_k \in [0, 1]$ and the value of μ should lie between 3.9 and 4.0 to generate a highly random sequence. The bifurcation diagram for the logistic map, as shown in Boeing (2016), is given in Figure 5.

The logistic map function is used to generate a pseudorandom sequence. Each number of the sequence will be used to encrypt each pixel; hence the length of the sequence should be equal to $m \times n$ (the size of the input image).

Figure 5 The bifurcation diagram for the logistic map



The initial values x_0 and μ_0 for equation (3) is generated with the help of a 20-digit key. Key-II will have a form: $d_1, d_2, d_3, \dots, d_{20}$, where each d_i for $i = 1, 2, \dots, 20$ is a decimal number.

$$P = \frac{(d_1 d_2 d_3 d_4)_{10}}{10^4}, \quad Q = \frac{(d_5 d_6 d_7 d_8)_{10}}{10^4}, \quad R = \frac{(d_9 d_{10} d_{11} d_{12})_{10}}{10^4}$$

$$S = \frac{(d_{13} d_{14} d_{15} d_{16})_{10}}{10^4}, \quad T = \frac{(d_{17} d_{18} d_{19} d_{20})_{10}}{10^4}$$

Division by 10^4 is necessary for normalising P, Q, R, S, T .

$$x_0 = (P + Q + 2T) \bmod 1 \tag{4}$$

$$\mu_0 = 3.9 + \frac{(R + S + 2T) \bmod 1}{10} \tag{5}$$

The mod operation is used for extracting the fractional part, thus restricting the values of x_0 and μ_0 within the necessary limits.

The logistic map generates a chaotic sequence. The sequence is converted into an 8-bit binary by multiplying each number with 255 and rounding it to the nearest integer value. It is then converted into an 8-bit binary number. The resulting sequence can be assumed to be a 1-D vector (say Ch), which along with the 1-D vector Di_α generated in the previous step will be used to perform Diffusion- β according to the following formula

$$Di_\beta(x) = Di_\alpha(x) \oplus Ch(x) \tag{6}$$

where $x = 1, 2, 3, \dots, m \times n$. Di_β is reshaped back into the dimensions $(m \times n)$ of the input image to form the final encrypted image. In the algorithm, this step is done in $diff2()$.

Input: 2 greyscale images ($A_{m \times n}$, $Ch_{m \times n}$)

Output: Greyscale image ($O_{m \times n}$)

Algorithm: diff2($A_{m \times n}$, $Ch_{m \times n}$)

Step 1 Find out the size of A and store it in m and n.

Step 2 for i = 0 to m - 1

for j = 0 to n - 1

$O(i,j) = A(i,j) \oplus Ch(i,j)$

end for

end for

Algorithm for complete encryption process:

Input: Original greyscale image ($I_{m \times n}$)

Output: Encrypted greyscale image ($E_{m \times n}$)

Algorithm: encrypt ($I_{m \times n}$, Key-I, Key-II)

for i = 1 to Key-I

$O_{sub} = subst(I_{m \times n})$ /* Substitution algorithm is shown below */

end for

$O_{diff} = diff1(O_{sub})$

$E_{m \times n} = diff2(O_{diff}, Key-II)$

End

The algorithm for substitution function is as follows:

Input: Greyscale image ($I_{m \times n}$)

Output: Substituted greyscale image ($O_{m \times n}$)

Algorithm: subst ($I_{m \times n}$)

$[X,Y] = segment(I_{m \times n})$

$X_c = compress(X)$

$Y_c = compress(Y)$

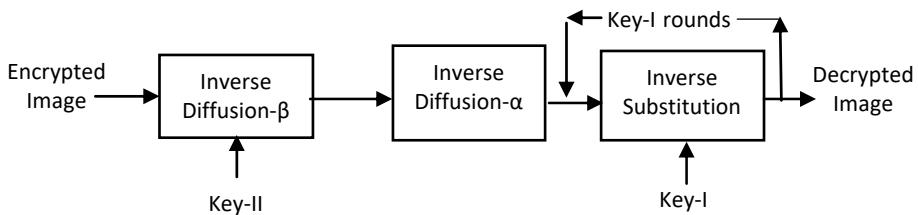
$O_{m \times n} = combine(X_c, Y_c)$

end

3.3 Decryption process

The decryption process is similar to encryption, except that the substitution and diffusion stages are performed in the reverse order. The architecture for the decryption algorithm is shown in Figure 6. It uses the same two keys: key-I and key-II, which was used in the encryption procedure.

Figure 6 Decryption model architecture



3.3.1 Inverse diffusion process

In the decryption procedure, the encrypted image first undergoes an inverse diffusion process. Since the proposed method contains two diffusion steps, diffusion-β is first inverted, followed by diffusion-α.

- *Step 1: inverse diffusion-β*

Key-II is used for generating the seed values of chaotic logistic map, precisely as in diffusion-β. The chaotic map Ch and encrypted image En is converted to 1D vectors. Diffusion-β is inverted according to the given formula:

$$Di_{\beta}(x) = En(x) \oplus Ch(x) \quad (7)$$

where $x = 1, 2, 3, \dots, m \times n$. The 1D vector Di_{β} is fed as input to inverse diffusion-α. In the algorithm, this step is done in $inv_diff2()$.

Input: 2 greyscale images ($A_{m \times n}, Ch_{m \times n}$)

Output: Greyscale image ($O_{m \times n}$)

Algorithm: $inv_diff2(A_{m \times n}, Ch_{m \times n})$

Step 1 Find out the size of A and store it in m and n .

Step 2 for $i = 0$ to $m - 1$

for $j = 0$ to $n - 1$

$$O(i,j) = A(i,j) \oplus Ch(i,j)$$

end for

end for

- *Step 2: inverse diffusion-α*

The following function is applied on the 1-D vector Di_{β} obtained in the previous step in order to inverse the diffusion-α step.

$$Di_{\alpha}(x) = Di_{\beta}(x) \oplus Di_{\beta}(x) \quad (8)$$

where $x = 1, 2, 3, \dots, m \times n$. The 1-D vector Di_{α} is reshaped to form a 2D matrix to form an image. The image undergoes inverse substitution to form the decrypted image. In the algorithm, this step is done in $inv_diff1()$.

Input: Greyscale image ($I_{m \times n}$)

Output: Greyscale image ($O_{m \times n}$)

Algorithm: inv_diff1 ($I_{m \times n}$)

Step 1 Find out the size of I and store it in m and n .

Step 2 Convert $I_{m \times n}$ into a 1D vector ($1 \times mn$)

Step 3 $O(0) = I(0)$

Step 4 for $i = 1$ to $m \times n - 1$
 $O(i) = I(i) \oplus I(i - 1)$
 end for

Step 5 Convert $O_{1 \times mn}$ into 2D $O_{m \times n}$

3.3.2 Inverse substitution process

Similar to the substitution technique in the encryption algorithm, inverse substitution is achieved by iterating the following three steps for key-I number of rounds.

- a *De-segment*: the image is partitioned into two horizontal rectangles of equal size to form two segments. Each segment is then decompressed using the method explained in the following step. In the algorithm, this step is done in *de_segment()*.

Input: Greyscale image ($I_{m \times n}$)

Output: 2 greyscale images ($A_{m \times n/2}, B_{m \times n/2}$)

Algorithm: de_segment($I_{m \times n}$)

Step 1 Find out the size of I and store it in m and n .

Step 2 for $i=0$ to $m-1$
 for $j=0$ to $n-1$
 if $i < m/2$
 $A(i,j) = I(I,j)$
 else
 $B(i-[m/2],j) = I(I,j)$
 end if
 end for
 end for

- b *De-compress*: the segments get decompressed according to the given algorithm.

Input: Grayscale image ($I_{m \times n}$)

Output: Decompressed grayscale image ($O_{m/2 \times 2n}$)

Algorithm: de_compress ($I_{m \times n}$)

Step 1 Find out the size of I and store it in m and n .

Step 2 for $i = 0$ to $m - 1$
 for $j = 0$ to $n - 1$
 if $j < n/4$
 $O(2i, j) = I(I, j)$

```

elseif  $n/4 \leq j \leq 3n/4$ 
     $O(2i + 1, j - \lfloor n/4 \rfloor) = I(i, j)$ 
else
     $O(2i, j - \lfloor n/2 \rfloor) = I(i, j)$ 
end if
end for
end for

```

Step 3 End

- c *De-combine*: Finally, the decompressed segments are placed side by side to and combined to form the deciphered image. In the algorithm, this step is done in *de_combine()*. After key-I iterations, the decrypted image will be obtained.
-

Input: 2 greyscale images ($A_{m \times n}$, $B_{m \times n}$)

Output: 2 greyscale images ($O_{2 \times m, n}$)

Algorithm: *de_combine*($A_{m \times n}$, $B_{m \times n}$)

Step 1 Find out the size of *A* and store it in *m* and *n*.

Step 2 for $i = 0$ to $m - 1$

 for $j = 0$ to $2n - 1$

 if $j < n$

$O(i, j) = A(i, j)$

 else

$O(i, j) = N(i, j - n)$

 end if

 end for

end for

The complete algorithm for the decryption process would be as follows:

Input: Encrypted greyscale image ($I_{m \times n}$)

Output: Decrypted greyscale image ($D_{m \times n}$)

Algorithm: *decrypt* ($I_{m \times n}$, Key-I, Key-II)

$O_{d2} = \text{inv_diff2}(O_{diff}, \text{Key-II})$

$O_{d1} = \text{inv_diff1}(O_{d2})$

for $i = 1$ to Key-I

$O_{isub} = \text{inv_subst}(O_{d1})$ /*Inverse substitution algorithm is described below*/

$O_{d2} = O_{isub}$

end for

$D_{m \times n} = O_{isub}$

end

As referenced above, the *inv_subst()* algorithm functionality is as follows:

```

Input: Greyscale image ( $I_{m \times n}$ )
Output: Inverse substituted greyscale image ( $O_{m \times n}$ )
Algorithm: inv_subst ( $I_{m \times n}$ )
 $[X, Y] = \text{inv\_segment}(I_{m \times n})$ 
 $X_c = \text{de\_compress}(X)$ 
 $Y_c = \text{de\_compress}(Y)$ 
 $O_{m \times n} = \text{inv\_combine}(X_c, Y_c)$ 
end
    
```

4 Experimental results and quality analysis

Using matlab, several experimental tests have been performed to demonstrate the strength and security of the proposed algorithm. A strong cryptosystem should not only resist all types of cryptanalysis attacks but should also preserve the quality of the subject image. Since the proposed algorithm can run on images having an arbitrary resolution, two sets of images have been used - square and rectangular. With each set containing five images. The results are shown in Tables 2 and 3. Unless explicitly mentioned, the key parameters chosen to conduct the required experiments are as follows:

- 1 key-I: 99
- 2 key-II: 93214345678923136629.

Table 2 Test results of square images










Image name with size	Original image	Encrypted image	Decrypted image
F16 512 × 512			
Fishing boat 512 × 512			
Couple 256 × 256			

Table 2 Test results of square images (continued)





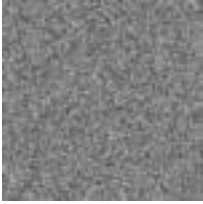
















<i>Image name with size</i>	<i>Original image</i>	<i>Encrypted image</i>	<i>Decrypted image</i>
Peppers 512 × 512			
Chemical plant 256 × 256			

Table 3 Test results of rectangular images

<i>Image name with size</i>	<i>Original image</i>	<i>Encrypted image</i>	<i>Decrypted image</i>
Black bear 256 × 512			
Brandy rose 256 × 512			
Skyline arch 400 × 594			
Pocket watch 1,024 × 768			
Fontaine des Terreaux 768 × 512			

4.1 Decryption quality analysis using image quality metrics

When an image propagates through an image processing pipeline, the quality of the image may get degraded due to distortions. To ensure that the proposed encryption-decryption algorithm is free from distortion, here three well-known image quality metrics have been used to test the quality of the processed image compared to the original one as given in Hore and Ziou (2010). Ten images were used for conducting each metric, out of which five were square images, and the rest were rectangular. Such objective measures can highlight errors that are unperceived by a human observer. The original and decrypted images should be identical for an ideal encryption-decryption algorithm, whereas the encrypted and original image should be highly dissimilar.

4.1.1 Mean-squared error

Mean-squared error or MSE calculates the average of the square of differences between intensities of individual pixel values of two images, which are compared as mentioned in Gulame et al. (2013). The formula to calculate MSE is given in equation (9).

$$MSE = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N [[\beta(m, n) - \gamma(m, n)]^2] \quad (9)$$

where β and γ are the intensity maps of the two compared images, and (m, n) represents the coordinates of the pixels in the map. The dimension of the image is given by $M \times N$. MSE of two identical images will be zero, and a larger value of MSE will mean the two images are highly dissimilar. Table 4 contains the values for MSE between all possible combination pairs of original, encrypted, and decrypted images.

Table 4 Mean-squared error analysis of tested images

<i>Image name</i>	<i>MSE between</i>		
	<i>Original and encrypted</i>	<i>Original and decrypted</i>	<i>Encrypted and decrypted</i>
F16	1.2083×10^4	0	1.2083×10^4
Fishing Boat	7.6300×10^3	0	7.6300×10^3
Couple	1.5350×10^4	0	1.5350×10^4
Peppers	8.4208×10^3	0	8.4208×10^3
Chemical Plant	7.6797×10^3	0	7.6797×10^3
Black Bear	1.0877×10^4	0	1.0877×10^4
Brandy Rose	8.3858×10^3	0	8.3858×10^3
Skyline Arch	1.3260×10^4	0	1.3260×10^4
Pocket Watch	1.0048×10^4	0	1.0048×10^4
Fontaine des Terreaux	1.3091×10^4	0	1.3091×10^4

4.1.2 Peak signal-to-noise ratio

Peak signal-to-noise ratio or PSNR is the ratio of maximum pixel intensity to the power of distortion. PSNR is derived using MSE, and the value is expressed in decibels, as mentioned in Wang et al. (2016).

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} \quad (10)$$

MSE can be computed using equation (9), and the value 255 is used as the greyscale input image is of an 8-bit unsigned integer data type. Higher the PSNR value, greater will be the image quality, hence for two identical images it will be infinity. The results of the PSNR values for the conducted experiments are given in Table 5.

Table 5 Peak-signal-to-noise-ratio analysis of tested images

Image name	PSNR between		
	Original and encrypted	Original and decrypted	Encrypted and decrypted
F16	8.0950	Infinity	8.0950
Fishing Boat	9.3056	Infinity	9.3056
Couple	6.2960	Infinity	6.2960
Peppers	8.8773	Infinity	8.8773
Chemical Plant	9.2773	Infinity	9.2773
Black Bear	7.7655	Infinity	7.7655
Brandy Rose	8.8953	Infinity	8.8953
Skyline Arch	6.9055	Infinity	6.9055
Pocket Watch	8.1099	Infinity	8.1099
Fontaine des Terreaux	6.9611	Infinity	6.9611

4.1.3 Structural similarity index

The Structural Similarity Index or SSIM is another metric to measure the similarity between two given images. MSE and PSNR are computed based on absolute errors of the pixel values, whereas SSIM combines local image structure, luminance, and contrast into a single local quality score. As a result, it is an improvement to the metrics as mentioned earlier. SSIM index between two equal-sized images a and b are measured by

$$SSIM(a, b) = \frac{(2\bar{a}\bar{b}K_1)(2\sigma_{ab} + K_2)}{(\bar{a} + \bar{b}^2 + K_1)(\sigma_a^2 + \sigma_b^2 + K_2)} \quad (11)$$

where \bar{a} and \bar{b} are means of a and b respectively; σ_a and σ_b are the variance of a and b respectively; σ_{ab} are the covariance of a and b . K_1 and K_2 are two variables used for stabilising the division done with the weak denominator, as given in Wang et al. (2004).

SSIM lies between the 0 and 1, with 1 indicating comparison was made between two identical images. Highly dissimilar images will have values closer to 0. Table 6 contains the values for MSE between all possible combination pairs of original, encrypted, and decrypted images.

Table 6 Structural similarity analysis of tested images

<i>Image name</i>	<i>SSIM between</i>		
	<i>Original and encrypted</i>	<i>Original and decrypted</i>	<i>Encrypted and decrypted</i>
F16	0.0104	1	0.0104
Fishing boat	0.0103	1	0.0103
Couple	0.0036	1	0.0036
Peppers	0.0098	1	0.0098
Chemical plant	0.0082	1	0.0082
Black bear	0.0070	1	0.0070
Brandy rose	0.0104	1	0.0104
Skyline arch	0.0069	1	0.0069
Pocket watch	0.0079	1	0.0079
Fontaine des Terreaux	0.0065	1	0.0065

4.2 Security analysis of the proposed cryptosystem

Cryptanalysis is the process decrypting the ciphered message without the knowledge of the cryptographic keys used as given in Li and Zheng (2002). A cryptosystem can only be used for real-world applications if it can rest all types of cryptanalysis attacks. The following analysis has been performed in order to evaluate the strength of the proposed cryptosystem.

4.2.1 Key space analysis

In a brute force attack, the attacker tries to break the cryptosystem by trying out all possible key combinations. Therefore, the key-space should be large enough so that it will take an infeasible amount of time for an attacker to successfully execute a brute force attack, as demonstrated in Monaghan et al. (2007). Here a combination of two secret keys has been used for the substitution and diffusion stage, respectively. The substitution stage uses a key of variable length, so theoretically, the key-space of this algorithm can be infinite. However, for practical usage, this key parameter can be limited to a two- or three-digit number to maintain the speed of the algorithm. The diffusion stage uses a 20-digit key ($10^{20} \approx 2^{67}$). Therefore the overall key-space for the proposed cryptosystem is at least 2^{70} , which makes it impossible for brute force attacks to breach the system.

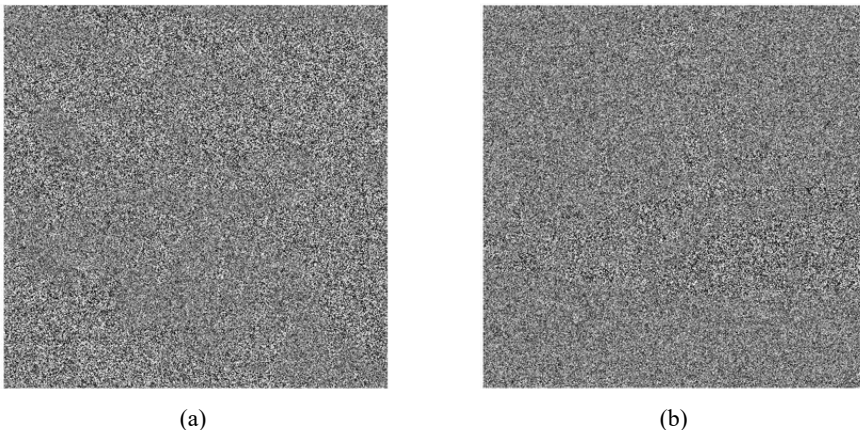
4.2.2 Key sensitivity analysis

For a secure cryptosystem, high key sensitivity is required so that even a slight alteration (a difference of 1 bit) in the encryption and decryption key would result in a completely different image, as mentioned in Guanghai et al. (2014). Key sensitivity is tested in both the encryption and decryption stages. Two encrypted images obtained by a slight change in key-parameter have been experimentally compared, and results are displayed in Table 7.

Table 7 Key sensitivity analysis for encrypted images by slightly changing key parameters

Image name	First set of key parameters		Second set of key parameters		Difference
	Key-I	Key-II	Key-I	Key-II	
F16	99	93214345678923136629	98	93214345678923136629	99.62%
Fishing boat	99	93214345678923136629	89	93214345678923136629	99.60%
Couple	99	93214345678923136629	99	93214345678923136628	99.22%
Peppers	99	93214345678923136629	99	83214345678923136629	99.27%
Chemical plant	99	93214345678923136629	100	93214345678923136629	99.60%
Black bear	99	93214345678923136629	98	93214345678923136629	99.62%
Brandy rose	99	93214345678923136629	89	93214345678923136629	99.64%
Skyline arch	99	93214345678923136629	99	93214345678923136628	99.27%
Pocket watch	99	93214345678923136629	99	83214345678923136629	99.26%
Fontaine des Terreaux	99	93214345678923136629	100	93214345678923136629	99.61%

For the decryption stage, an encrypted image has been decrypted with a slight change in the decryption key parameter, and the results are shown in Figure 7.

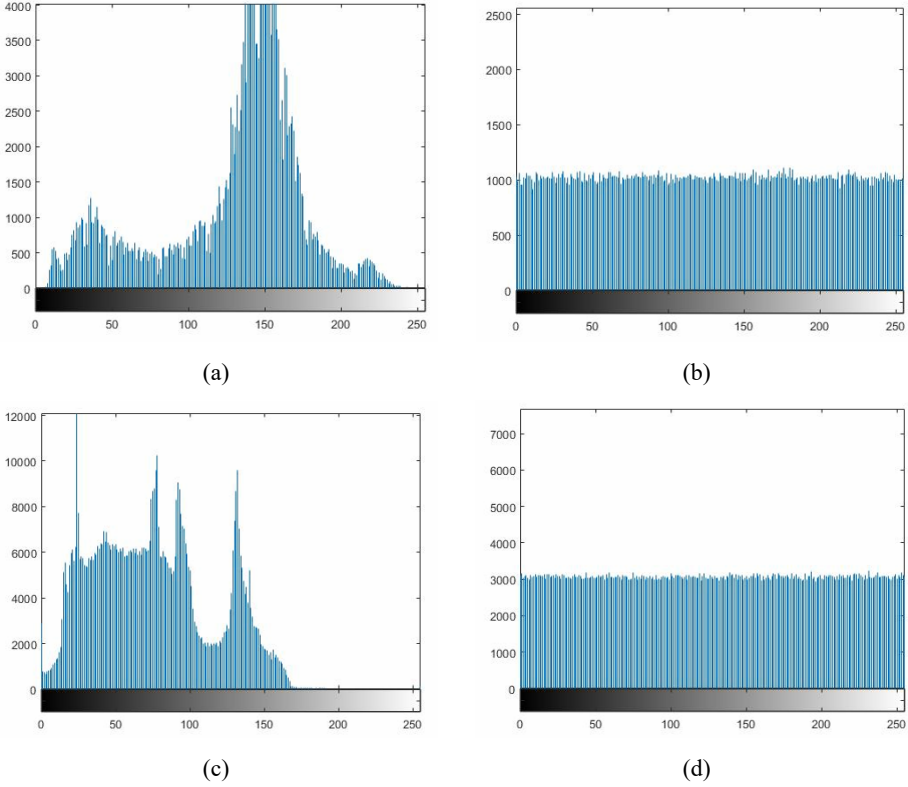
Figure 7 Incorrectly decrypted image, (a) F16 and (b) fishing boat, for one-digit dissimilarity in the decryption key

4.2.3 Statistical analysis based on histogram

The intense confusion and diffusion properties of the algorithm aids in resisting statistical attacks. As mentioned in Junod (2005), attackers perform statistical analysis to establish relationships between the original and encrypted images.

Histogram is a graphical representation of the frequency of each pixel intensity value in an image. Histograms are used to illustrate the pixel distribution of the image. For a robust encryption algorithm, the histogram of an encrypted image should be uniform and entirely different from the histogram of the primary image, as shown in Stoyanov and Kordov (2015a). Examples of histograms of the original and encrypted images have been shown in Figure 8.

Figure 8 Histogram of (a) fishing boat original image (b) fishing boat encrypted image (c) pocket watch original image and (d) pocket watch encrypted image (see online version for colours)



4.2.4 Uniformity analysis by chi-square test

The histogram of the encrypted image is different from the original image histogram, as during encryption, it was approximated by a uniform distribution. The visual analysis is not sufficient on its own to verify uniformity. Hence, the chi-square test is performed to statistically ensure that the histogram of the ciphered images is uniform, as mentioned in Sheela et al. (2018). It is denoted by equation (12).

$$\chi^2 = \sum_{x=0}^{N-1} \frac{(o_x - e_x)^2}{e_x} \tag{12}$$

where N is the number of levels ($N = 256$ in this case), o_x is the observed frequency of each grey level in the histogram of the encrypted image, and e_x is the expected frequency required for uniform distribution. For the encryption to be secure, the chi-square test value should be less than 293, which is the theoretical chi-square value when the number of levels is 256, and the level of significance is 0.05, as stated in Kanafchian and Fathi-Vajargah (2017). Table 8 provides the chi-square test results performed on each of the encrypted images, respectively. Since each of the experimental chi-square value is less than 293, the uniformity of the histograms is verified.

Table 8 Chi-square test results for encrypted images

<i>Image name</i>	<i>Chi-square test score</i>
F16	279.53
Fishing boat	246.83
Couple	254.66
Peppers	268.73
Chemical plant	239.79
Black bear	261.07
Brandy rose	231.72
Skyline arch	235.14
Pocket watch	229.40
Fontaine des Terreaux	270.29

4.2.5 Measurement of encryption quality

Encryption quality (EQ) is measured by the difference of frequency of each grey level before and after the image is encrypted, as shown in Gu and Han (2006). It is defined in equation (13).

$$EQ = \frac{\sum_{x=0}^{255} |f_x(P) - f_x(E)|}{256} \quad (13)$$

where $f_x(P)$ is the frequency value for grey level i in the plain image P , and $f_x(E)$ is the frequency value for grey level i in the encrypted image E .

For comparison, it is necessary to estimate the ideal value of encryption quality. For that purpose, the maximum value of encryption quality represented as EQ_{\max} needs to be calculated. EQ_{\max} is calculated based on the following assumptions:

- 1 For encryption, the worst sample of the plain image is one which has all pixels having the same value that is a full white or an entirely black image. Therefore, the frequency of a specific pixel value x_1 in plain image P is $f_{x_1}(P) = M \times N$, where $x_1 \in \{0, 255\}$ and $M \times N$ is the resolution of the greyscale image. Also, the frequency of all other pixel values except x_1 in the plain image P is $f_{x_2}(P) = 0$, where $x_i \in \{0, 255\}$ and $x_2 \neq x_1$.
- 2 A secure encryption algorithm should produce an encrypted image in which the pixel values are uniformly distributed. Therefore, the frequency of any pixel value x in the encrypted image is $f_x(E) = \frac{M \times N}{256}$, where $x \in \{0, 255\}$ and $M \times N$ is the resolution of the greyscale image.

Using equation (13), we calculate EQ as:

$$EQ = \frac{\left| \frac{M \times N}{256} - M \times N \right| + \left| \frac{M \times N}{256} - 0 \right| \times 255}{256} \quad (14)$$

which on simplifying we get,

$$EQ = \frac{510 \times M \times N}{256^2} \quad (15)$$

As illustrated in below Table 9, on encrypting a full black image, having dimensions 512×512 , using the proposed encryption algorithm, we find the experimental EQ value as calculated by equation (13) to be 2,048, which is the same as the maximum theoretical value of EQ possible as given by equation (15). Hence, the proposed cryptosystem is secure.

Table 9 Result of encryption quality of a greyscale image having the same value in all pixels


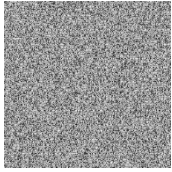
Plain image	Encrypted image	Resolution	Theoretical EQ_{max}	Observed EQ value
		512×512	2,048	2,048

Table 10 reports the observed EQ value on encrypting the following set of images.

Table 10 Encryption quality results

Image name	Resolution	EQ	EQ_{max}
F16	512×512	1069.7	2,040
Fishing boat	512×512	864.06	2,040
Couple	256×256	315.56	510
Peppers	512×512	571.59	2,040
Chemical plant	256×256	245.43	510
Black bear	512×256	228.41	1,020
Brandy rose	512×256	349.66	1,020
Skyline arch	594×400	519.65	1,849
Pocket watch	$768 \times 1,024$	2,742.3	6,120
Fontaine des Terreaux	512×768	1,746.2	3,060

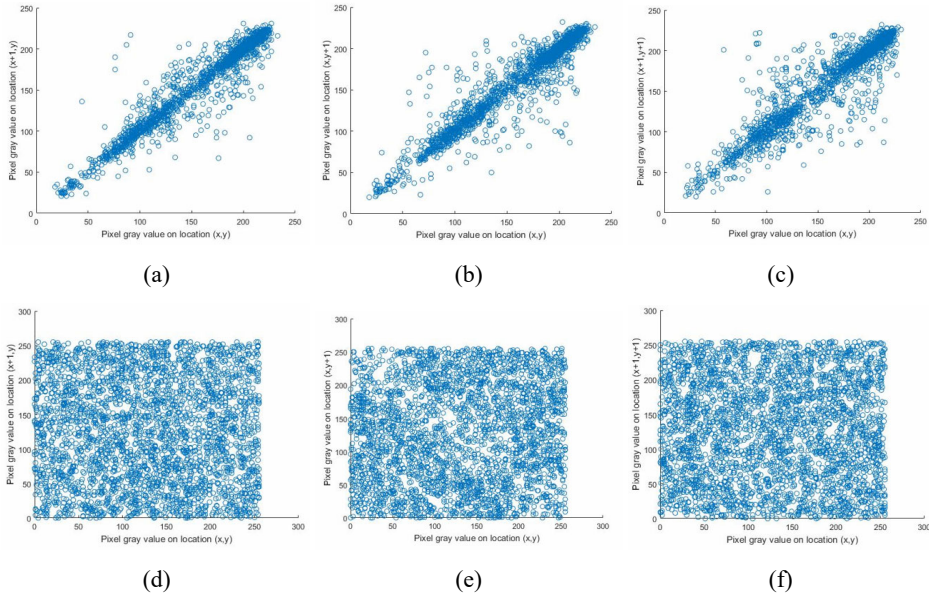
4.2.6 Correlation analysis

In a typical image, two adjacent pixels are highly correlated in horizontal, vertical, and diagonal directions. Correlation coefficients are a measure of the strength of the relationship between two variables (or pixels), as mentioned in Zhang and Liu (2011). It is calculated using equation (16).

$$r_{a,b} = \frac{\sum (a_i - \bar{a})(b_i - \bar{b})}{\sqrt{\sum (a_i - \bar{a})^2 \sum (b_i - \bar{b})^2}} \tag{16}$$

For the original image, the correlation coefficients are close to 1, which is the maximum value possible. For ideal encryption, the ciphered image should have the values of correlation coefficients close to 0. Scatter plots results are provided in Figure 9.

Figure 9 Correlation of plain image’s pixels in (a) horizontal, (b) vertical and (c) diagonal position; correlation of cipher image’s pixels in (d) horizontal, (e) vertical, and (f) diagonal position (see online version for colours)



4.2.7 Entropy analysis

In statistics, entropy refers to disorder or randomness in data. Image entropy can be used as a measure of the texture of the image, as stated in Langford and Hellman (1994). The ideal value of entropy for an encrypted image should be 8. Entropy is calculated as follows:

$$Entropy = \sum_{i=1}^N P_i \log_2 P_i \tag{17}$$

where P_i is the probability of occurrence of intensity value in the given image, and n is the total possible grey levels ($n = 256$). The entropies of the plain images and corresponding ciphered images are given in Table 11.

Table 11 Entropy analysis result for original and encrypted images

Image name	Plain image entropy	Encrypted image entropy
F16	6.7227	7.9992
Fishing boat	7.1914	7.9993
Couple	6.4818	7.9972
Peppers	7.5940	7.9953
Chemical plant	7.0818	7.9974
Black bear	7.7043	7.9986
Brandy rose	7.4207	7.9993
Skyline arch	7.4299	7.9993
Pocket watch	7.1076	7.9998
Fontaine des Terreaux	6.0813	7.9995

4.2.8 Differential analysis by NPCR, UACI and hamming distance

In the differential approach of cryptanalysis, the attacker changes the intensity value of a particular pixel in the original image and detects the alterations in the corresponding encrypted image and tries to obtain a meaningful relationship, as stated in Huang and Nien (2009). Therefore, to resist these types of attacks, a slight change in the original image should result in a significant change in the encrypted image. It is a measure of plain text sensitivity.

- *NPCR*: number of pixels change rate (NPCR) is a measure of the percentage of varying pixels between the two compared images, as mentioned in Wang et al. (2012). It is calculated using the following equations:

$$NPCR = \frac{1}{MN} \sum_{m=1}^N \sum_{n=1}^N G(m, n) \times 100 \quad (18)$$

$$G(m, n) = \begin{cases} 0, & \text{if } I_1(m, n) = I_2(m, n) \\ 1, & \text{if } I_1(m, n) \neq I_2(m, n) \end{cases} \quad (19)$$

where I_1 and I_2 are the encrypted images obtained by encrypting two $m \times n$ size images having a difference of only one-pixel value.

- *UACI*: unified average changing intensity (UACI) computes the mean of absolute difference of corresponding pixel values in two images, as mentioned in Wei et al. (2012).

$$UACI = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N \frac{|I_1(m, n) - I_2(m, n)|}{255} \times 100 \quad (20)$$

where I_1 and I_2 are the encrypted images obtained by encrypting two $m \times n$ size images having a difference of only one-pixel value.

- *Hamming distance (HD)*: hamming distance is another metric used to measure the avalanche effect, i.e., during encryption for a small change (say altering one bit) in the plain image results in a massive change in the encrypted image, as mentioned in Zhu and Sun (2018). It is defined by:

$$HD(E_1, E_2) = \frac{1}{S} \sum_{i=1}^S (E_1(i) \oplus E_2(i)) \quad (21)$$

where $S = m \times n \times 8$, is the image size in bits, and E_1 and E_2 are the two ciphered images. The optimal value of hamming distance is 0.5 or 50%. The results of the differential analysis are shown in Table 12.

Table 12 Calculated NPCR, UACI and hamming distance results for one-bit dissimilarity

Image name	Image size	NPCR (in %)	UACI (in %)	HD (in %)
F16	512 × 512	99.62	33.46	50.02
Fishing boat	512 × 512	99.61	33.42	50.00
Couple	256 × 256	99.69	35.00	49.99
Peppers	512 × 512	99.60	33.65	49.99
Chemical plant	256 × 256	99.62	33.64	49.95
Black bear	512 × 256	99.62	33.58	50.02
Brandy rose	512 × 256	99.62	33.39	49.98
Skyline arch	594 × 400	99.63	33.37	49.91
Pocket watch	768 × 1,024	99.62	33.49	49.99
Fontaine des Terreaux	512 × 768	99.63	33.53	50.01

The experimentally observed values for NPCR, UACI, and hamming distance match the expected theoretical values, as stated in Wu et al. (2011). Hence the proposed cryptosystem can resist differential attacks.

In Tables 13 and 14, the NPCR and UACI results of the proposed method have been compared with other available image cryptosystems, results of which are reported in Behnia et al. (2008). NPCR values are accepted if they are higher than theoretical NPCR critical values and UACI values are accepted if it falls within the acceptance range ($UACI_{\alpha}^{*+}$, $UACI_{\alpha}^{*+}$), where α denotes the significance level. The ‘✓’ indicates the method is within the acceptable range, whereas ‘✗’ indicates the method has failed as the test results are not in an acceptable range.

4.2.9 Performance analysis by measuring time complexity

Substitution, diffusion- α and diffusion- β are the three main functions in the proposed cryptosystem. Therefore, running time for encryption is given by:

$$T_{enc} = T_{subs} + T_{dif\alpha} + T_{dif\beta} \quad (22)$$

However, the substitution phase has the most impact on running time, as it spans for key-I (say k) number of rounds, where each round has $O(mn)$ time complexity. The diffusion- α and the diffusion- β each take $O(mn)$ running time (where $m \times n$ is the image size). Therefore, the encryption process would take $O(kmn)$ running time.

Table 13 NPCR results comparison table

<i>Theoretical NPCR critical values</i>				
NPCR* _{α=0.05} = 99.5693%; NPCR* _{α=0.01} = 99.5527%; NPCR* _{α=0.001} = 99.5341%				
<i>Image size 256 × 256</i>		<i>NPCR test results</i>		
<i>Image cryptosystems</i>	<i>NPCR values reported (in %)</i>	<i>α = 0.05</i>	<i>α = 0.01</i>	<i>α = 0.001</i>
Zhu et al. (2006)	98.669	✗	✗	✗
Huang and Nien (2009)	99.26	✗	✗	✗
[as reported in Liao et al. (2010)]	99.45	✗	✗	✗
	99.13	✗	✗	✗
Zhang et al. (2005)	41.962	✗	✗	✗
Liao et al. (2010)	99.42	✗	✗	✗
	99.54	✗	✗	✓
	99.60	✓	✓	✓
Zhang et al. (2010)	99.66	✓	✓	✓
	99.65	✓	✓	✓
	99.63	✓	✓	✓
Kumar and Ghose (2011)	99.61	✓	✓	✓
Murillo-Escobar et al. (2016)	99.72	✓	✓	✓
Proposed method	99.62	✓	✓	✓

Table 14 UACI results comparison table

<i>Theoretical UACI critical values</i>				
UACI* _{α=0.05} ⁻ = 33.2824%; UACI* _{α=0.01} ⁻ = 33.2255%; UACI* _{α=0.001} ⁻ = 33.1594%				
UACI* _{α=0.05} ⁺ = 33.6447%; UACI* _{α=0.01} ⁺ = 33.7016%; UACI* _{α=0.001} ⁺ = 33.7677%				
<i>Image size 256×256</i>		<i>UACI test results</i>		
<i>Image cryptosystems</i>	<i>UACI values reported (in %)</i>	<i>α = 0.05</i>	<i>α = 0.01</i>	<i>α = 0.001</i>
Zhu et al. (2006)	33.362	✗	✗	✗
Huang and Nien (2009)	21.41	✗	✗	✗
[as reported in Liao et al. (2010)]	23.42	✗	✗	✗
	15.08	✗	✗	✗
Zhang et al. (2005)	33.25	✗	✓	✓
Liao et al. (2010)	27.78	✗	✗	✗
	27.66	✗	✗	✓
	24.94	✗	✗	✗
Zhang et al. (2010)	33.20	✗	✗	✓
	33.31	✓	✓	✓
	34.61	✗	✗	✗
Kumar and Ghose (2011)	38	✗	✗	✗
Murillo-Escobar et al. (2016)	32.821	✗	✗	✗
Proposed method	33.64	✓	✓	✓

Similarly, for decryption, inverse diffusion- β , inverse diffusion- α , and inverse substitution are the three main functions. Running time for decryption is given by:

$$T_{dec} = T_{indif\alpha} + T_{indif\beta} + T_{insubs} \quad (23)$$

where inverse substitution phase spans for key-I (say k) number of rounds, with each round having $O(mn)$ complexity giving the decryption process an $O(kmn)$ running time. A smaller value of k will reduce running time, thereby improving the efficiency of the proposed algorithm.

Running speed (in Mbit/s) of the algorithm has been provided in Table 15. Simulations have been performed on MATLAB version 9.0.0.341360 (R2016a) on a system with hardware specifications – AMD FX 3.8 GHz CPU, 16 GB RAM, and a 64-bit Windows 10 operating system. The image ‘F16’ was used having size: $512 \times 512 \times 8 = 2,097,152$ bits.

Table 15 Running speed of proposed image cryptosystem

<i>Key-I</i>	<i>Key-II</i>	<i>Encryption speed (in Mbit/s)</i>	<i>Decryption speed (in Mbit/s)</i>
99	93214345678923136629	2.5575	2.6887

In Table 16, a comparison has been made with similar existing cryptosystems where encryption throughput has been measured and mentioned in corresponding research articles. It can be noted that the encryption speed of the proposed cryptosystem has been outperformed all similar existing cryptosystems.

Table 16 Comparison of encryption throughput with similar existing cryptosystems

<i>Image cryptosystem algorithm</i>	<i>Encryption speed in Mbit/sec</i>
Proposed method	2.5575
Stoyanov and Kordov (2015b)	1.70
Liu and Tong (2012)	0.4901
Wong et al. (2008).	0.4844

4.3 Comparison

The following sub-sections present detailed discussion when the proposed method is compared with traditional block ciphers, cryptosystems employing Baker map with/without logistic map, and lastly, with other different chaotic image encryption techniques.

4.3.1 Comparison with traditional block ciphers (AES, DES)

Chai et al. (2017) mentioned that traditional block ciphers, such as AES and DES, are not the best choice for image encryption, as these algorithms are mostly applied on plain text or binary sequence. Ye (2014) has documented the reasons behind this. He stated that the large capacity of image leads to abnormal running time for the traditional cryptosystem when applied to the image. He also mentioned that the strong correlation between adjacent pixels in images leads to a security breach. As correlation indicates a high degree of pixel replication, hence it may turn to low encryption efficiency as well as there

could be significant delay in real-time data transmission – all of these factors impacted the choice of AES or DES for image encryption, as suggested by Zhang et al. (2019a). In Koppu and Viswanatham (2017), the authors concluded that all these concerns could be addressed if there are high-end computational resources in terms of hardware and software. In the same regard, Liu and Wang (2010) have stated application of traditional ciphers on image is not appropriate for commercial application software, as it increases the production cost very high.

However, in the recent past, there are few works of image encryption which applies the modified form of AES and DES. One such article is Arab et al. (2019), where the authors have proposed an encryption technique combining the Arnold chaos system with AES. This cryptosystem has a competitive running time and can be implemented with low-level computational resources. However, the closer look on that work reveals that the method uses ten rounds of operation for substitution and diffusion having key size as 128 bits, and it also replaced the column operations by pixel value summations. That is why it decreases the running time compared with the traditional AES. Another example using modified DES is Zhang et al. (2019b), where the authors used chaotic systems, DNA computing, select cipher-text along with the DES encryption algorithm model to reduce the cost of high computational resources and to minimise the vulnerability of DES to brute force attack for image encryption. Hence, it can be concluded that traditional AES and DES are inappropriate for using imagery data.

To compare the performance of the proposed chaotic cryptosystem with the traditional AES and DES encryptions, codes have been executed from Crypto++ library on the same set of 512×512 greyscale images, which have been used earlier in test results. It has been observed that AES takes over 10 minutes to encrypt an image in the same computer where all earlier test results have been obtained. In contrast, the proposed cryptosystem takes less than a second to encrypt the same image (detailed performance outcome of the proposed method has been discussed in Section 4.2.9 – ‘Performance analysis by measuring time complexity’). A similar performance test result of AES encryption has been found in Arab et al. (2019), where the authors have shown that to encrypt the F16 greyscale image having the size of 256×256 , AES algorithm takes 515.2 seconds in an Intel Core i7-6500U 2.50 GHz computer having 8 GB RAM and 64-bit Windows 10 operating system. In another test result, Kumari et al. (2017) have shown that DES is having much higher time complexity as compared to AES; hence DES takes much more extended period to perform encryption and decryption than AES. Thus, it has been proved that the proposed cryptosystem is much faster than the traditional AES and DES encryption algorithms.

Another point worth mentioning here is the key-space. DES has severe weakness in key size, which is 256; hence it can be compromised by brute-force attack, as mentioned in Smid and Branstad (1988). This vulnerability has been overcome in AES encryption; however, the computational cost takes over there. Whereas, the proposed chaotic cryptosystem has been implemented with larger variable key-space 270 in a low-cost computational system, generating much better encryption throughput (details of which are mentioned in Section 4.2.9).

While testing with AES and DES encryption in Crypto++ library, it has been observed that the histograms of encrypted images are not uniformly distributed, and for some of the test images, the histograms are visually spiked. This same fact has also been documented in Kumari et al. (2017). The significance of spiked histogram reveals the pattern of encryption, which can be visually identifiable and henceforth, prone to the

statistical attacks. Whereas the proposed cryptosystem results in a uniform histogram for all the test images signifying it is successful in obscuring visual artefact and is highly robust against statistical attacks.

Hence it has been proved that the proposed chaotic cryptosystem outperforms over traditional DES or AES image encryption techniques.

4.3.2 Comparison with similar chaotic cryptosystems using baker map and logistic map

In Luo et al. (2019), the authors have presented an image encryption algorithm based on two-dimensional Baker map and logistic map. However, the implementation of that technique is entirely different from the current proposed technique. In that article, image encryption process consists of two steps – shuffling and substitution. There the application of the 2D Baker map is only limited in improving the logistic map sequence by periodically changing original parameters and state variables after specific periods; hence more emphasis has been given to determine the iteration periods. The improved logistic map is then applied to both shuffling and substitution. However, the authors have neither discussed how the decryption is accomplished, nor performed decrypted image quality checks with basic image quality metrics like PSNR, MSE, or SSIM. Nonetheless, it can be observed from their test results that the decryption quality is significantly low. Table 17 presents a brief comparison report between this image cryptosystem and the current proposed technique.

Table 17 Comparison of the proposed method with related works which used Baker map with/without logistic map

<i>Cryptanalysis performed</i>	<i>Proposed method</i>	<i>Luo et al. (2019)</i>	<i>Ye and Zhuang (2010)</i>	<i>Han et al. (2006)</i>
Decryption quality analysis (by MSE, PSNR, and SSIM)	✓	✗	✗	✗
Key-space analysis	✓	✓	✓	✓
Key sensitivity analysis	✓	✓	✗	✓
Histogram analysis (visual test)	✓	✓	✓	✓
Uniformity analysis (chi-square test)	✓	✗	✗	✗
Encryption quality analysis	✓	✗	✗	✗
Correlation analysis	✓	✓	✗	✓
Entropy analysis	✓	✓	✗	✓
Differential analysis by NPCR, UACI	✓	✓	✗	✗
Avalanche effect analysis (by hamming distance)	✓	✗	✗	✗
Time complexity analysis	✓	✗	✗	✗

The proposed method outperforms most of the existing chaos-based image cryptosystems since it utilises two separate chaotic maps for pixel and bit shuffling purposes. The algorithm has a relatively time-efficient substitution stage since Baker map is one of the fastest chaotic maps available. The proposed architecture employs a modified version of the Baker map for confusion, along with the two stages of diffusion, where another

chaotic logistic map is used to make the cryptosystem resistant to all kinds of cryptanalysis attacks. In Table 17, the proposed cryptosystem is also compared with two other image cryptosystems presented in Ye and Zhuang (2010) and Han et al. (2006), both of which use different modified versions of the Baker map.

4.3.3 Comparison with non-similar chaotic cryptosystems

As the proposed image encryption algorithm is based on chaotic maps, hence its performance has also been compared with other different types of chaos-based cryptosystems. In Table 18, comparisons have been made between the proposed method and the other non-similar chaos-based cryptosystems, which does not use Baker Map for ciphering the image.

Table 18 Comparison of the proposed method with other chaos-based cryptosystems which does not use Baker map

<i>Cryptanalysis performed</i>	<i>Proposed method</i>	<i>Wong et al. (2008)</i>	<i>Yang et al. (2010)</i>	<i>Wang et al. (2011)</i>	<i>Zhang et al. (2013)</i>	<i>Song et al. (2013)</i>	<i>Chen et al. (2004)</i>	<i>Akhshani et al. (2012).</i>
Decryption quality analysis	✓	✗	✗	✗	✗	✗	✗	✗
Key-space analysis	✓	✗	✓	✓	✗	✓	✓	✓
Key sensitivity analysis	✓	✗	✓	✓	✗	✓	✓	✗
Histogram analysis	✓	✓	✓	✓	✓	✓	✓	✓
Uniformity analysis (by chi-square test)	✓	✗	✗	✗	✗	✗	✗	✗
Encryption quality	✓	✗	✗	✗	✗	✗	✗	✗
Correlation analysis	✓	✓	✓	✓	✓	✓	✓	✓
Entropy analysis	✓	✗	✗	✓	✓	✓	✗	✓
Differential analysis (by NPCR and UACI)	✓	✓	✓	✓	✓	✓	✓	✓
Avalanche effect analysis (by hamming distance)	✓	✗	✗	✗	✗	✗	✗	✓
Time complexity analysis	✓	✓	✗	✗	✗	✗	✗	✗

In Tables 17 and 18, ‘✓’ indicates that experiments were performed and the satisfactory result was obtained, whereas ‘✗’ indicates either experimental details not provided or unsatisfactory results obtained. Image metric analysis is vital as it signifies the quality of the decrypted image and verifies that the proposed algorithm is free from distortion. Other cryptanalysis tests demonstrate the strength and reliability of this encryption-decryption technique.

5 Conclusions

A new chaos-based cryptosystem has been proposed in this paper. The encryption-decryption model architecture is based on a substitution-diffusion scheme. A separate chaotic map has been employed in different stages of this type of image cryptosystem. The improved randomness and efficiency of pixel and bit permutation result in an efficient and robust image ciphering technique.

In the substitution stage, pixel permutation has been achieved with a modified version of a discretised 2D baker map. The enhanced Baker map increases the randomness of the system, therefore improving the strength of the encrypted image. The encryption-decryption process is free from image distortion as pixel shuffling takes place in a bijective manner. The algorithm contains a two-step diffusion process. The first step is a simple XOR operation performed among consecutive pixels, and the second stage uses a chaotic 2-D logistic map. The result of cryptanalysis verifies that the proposed crypto-system is resistant to all types of known attacks, such as brute force, key-sensitivity, statistical, and differential. Therefore, the proposed method is fast, robust and secure enough to be used in real-world applications.

References

- Akhshani, A., Akhavan, A., Lim, S-C. and Hassan, Z. (2012) 'An image encryption scheme based on quantum logistic map', *Communications in Nonlinear Science and Numerical Simulation*, Vol. 17, No. 12, pp.4653–4661 [online] <https://doi.org/10.1016/j.cnsns.2012.05.033>.
- Alvarez, G. and Li, S. (2006b) 'Breaking an encryption scheme based on chaotic baker map', *Physics Letters A*, Vol. 352, Nos. 1–2, pp.78–82 [online] <https://doi.org/10.1016/j.physleta.2005.11.055>.
- Alvarez, G. and Li, S-I. (2006a) 'Some basic cryptographic requirements for chaos-based cryptosystems', *International Journal of Bifurcation and Chaos*, Vol. 16, No. 8, pp.2129–2151 [online] <https://doi.org/10.1142/s0218127406015970>.
- Arab A., Rostami M.J, and Ghavami B. (2019) 'An image encryption method based on chaos system and AES algorithm', *The Journal of Supercomputing*, October, Vol. 75, No. 10, pp.6663–6682 [online] <https://doi.org/10.1007/s11227-019-02878-7>.
- Behnia, S., Akhshani, A., Mahmodi, H. and Akhavan, A. (2008) 'A novel algorithm for image encryption based on mixture of chaotic maps', *Chaos, Solitons & Fractals*, Vol. 35, No. 2, pp.408–419 [online] <https://doi.org/10.1016/j.chaos.2006.05.011>.
- Boeing, G. (2016) 'Visual analysis of nonlinear dynamical systems: chaos, fractals, self-similarity and the limits of prediction', *Systems*, Vol. 4, No. 4, p.37 [online] <https://doi.org/10.3390/systems4040037>.
- Chai, X., Chen, Y. and Broyde, L. (2017) 'A novel chaos-based image encryption algorithm using DNA sequence operations', *Optics and Lasers in Engineering*, January, Vol. 88, pp.197–213, <https://doi.org/10.1016/j.optlaseng.2016.08.009>.
- Chen, G., Mao, Y. and Chui, C.K. (2004) 'Asymmetric image encryption scheme based on 3D chaotic cat maps', *Chaos, Solitons & Fractals*, Vol. 21, No. 3, pp.749–761 [online] <https://doi.org/10.1016/j.chaos.2003.12.022>.
- Dang, P.P. and Chau, P.M. (2000) 'Image encryption for secure Internet multimedia applications', *IEEE Transactions on Consumer Electronics*, Vol. 46, No. 3, pp.395–403 [online] <https://doi.org/10.1109/30.883383>.
- Fridrich, J. (1998) 'Symmetric ciphers based on two-dimensional chaotic maps', *International Journal of Bifurcation and Chaos*, Vol. 8, No. 6, pp.1259–1284 [online] <https://doi.org/10.1142/s021812749800098x>.

- Gu, G. and Han, G. (2006) 'An enhanced chaos based image encryption algorithm', *First International Conference on Innovative Computing, Information, and Control – Volume I (ICICIC'06)* [online] <https://doi.org/10.1109/icicic.2006.46>.
- Guanghui, C., Kai, H., Yizhi, Z., Jun, Z. and Xing, Z. (2014) 'Chaotic image encryption based on running-key related to plaintext', *The Scientific World Journal*, pp.1–9 [online] <https://doi.org/10.1155/2014/490179>.
- Gulame, M., Joshi, K.R. and Kamthe, R.S. (2013) 'A full reference based objective image quality assessment', *International Journal of Advanced Electrical and Electronics Engineering*, Vol. 2, No. 6, pp.13–18.
- Han, F., Yu, X. and Han, S. (2006) 'Improved baker map for image encryption', *2006 1st International Symposium on Systems and Control in Aerospace and Astronautics* [online] <https://doi.org/10.1109/isscaa.2006.1627519>.
- Hore, A. and Ziou, D. (2010) 'Image quality metrics: PSNR vs. SSIM. Presented at the 2010 20th International Conference on Pattern Recognition (ICPR)' [online] <https://doi.org/10.1109/icpr.2010.579>.
- Huang, C.K. and Nien, H.H. (2009) 'Multi chaotic systems based pixel shuffle for image encryption', *Optics Communications*, Vol. 282, No. 11, pp.2123–2127 [online] <https://doi.org/10.1016/j.optcom.2009.02.044>.
- Hung, P.A., Sooraksa, P. and Klomkarn, K. (2013) 'Extended baker map using scan patterns for image encryption', *2013 International Conference on Information Technology and Electrical Engineering (ICITEE)*, October [online] <https://doi.org/10.1109/iciteed.2013.6676223>.
- Cheng, J. and Guo, J.I. (2000) 'A new chaotic key-based design for image encryption and decryption', *2000 IEEE International Symposium on Circuits and Systems. Emerging Technologies for the 21st Century. Proceedings (IEEE Cat No.00CH36353). ISCAS 2000*, Geneva [online] <https://doi.org/10.1109/iscas.2000.858685>.
- Junod, P. (2005) *Statistical Cryptanalysis of Block Ciphers*, EPFL, Lausanne [online] <https://doi.org/10.5075/EPFL-THESIS-3179>.
- Kanafchian, M. and Fathi-Vajargah, B. (2017) 'A novel image encryption scheme based on clifford attractor and noisy logistic map for secure transferring images in navy', *International Journal of E-Navigation and Maritime Economy*, Vol. 6, pp.53–63 [online] <https://doi.org/10.1016/j.enavi.2017.05.007>.
- Katzenbeisser, S. and Petitcolas, F.A.P. (Eds.) (2000) *Information Hiding Techniques for Steganography and Digital Watermarking*, Artech House, MA, USA, ISBN 978-1-58053-035-4.
- Knudsen, L.R. and Robshaw, M.J.B. (2011) 'Brute force attacks', *Information Security and Cryptography*, pp.95–108, Springer Berlin Heidelberg [online] https://doi.org/10.1007/978-3-642-17342-4_5.
- Koppu, S. and Viswanatham, V.M. (2017) 'A fast enhanced secure image chaotic cryptosystem based on hybrid chaotic magic transform', *Modelling and Simulation in Engineering*, Vol. 2017, pp.1–12 [online] <https://doi.org/10.1155/2017/7470204>.
- Kumar, A. and Ghose, M.K. (2011) 'Extended substitution-diffusion based image cipher using chaotic standard map', *Communications in Nonlinear Science and Numerical Simulation*, Vol. 16, No. 1, pp.372–382 [online] <https://doi.org/10.1016/j.cnsns.2010.04.010>.
- Kumar, M., Powduri, P. and Reddy, A. (2015) 'An RGB image encryption using diffusion process associated with chaotic map', *Journal of Information Security and Applications*, Vol. 21, pp.20–30 [online] <https://doi.org/10.1016/j.jisa.2014.11.003>.
- Kumari, M., Gupta, S. and Sardana, P. (2017) 'A survey of image encryption algorithms', *3D Research*, Vol. 8, No. 4 [online] <https://doi.org/10.1007/s13319-017-0148-5>.
- Langford, S.K. and Hellman, M.E. (1994) 'Differential-linear cryptanalysis', *Advances in Cryptology — CRYPTO '94*, pp.17–25 [online] https://doi.org/10.1007/3-540-48658-5_3.

- Li, C., Liu, Y., Xie, T. and Chen, M.Z.Q. (2013) 'Breaking a novel image encryption scheme based on improved hyperchaotic sequences', *Nonlinear Dynamics*, Vol. 73, No. 3, pp.2083–2089 [online] <https://doi.org/10.1007/s11071-013-0924-6>.
- Li, C., Zhang, L.Y., Ou, R., Wong, K-W. and Shu, S. (2012) 'Breaking a novel colour image encryption algorithm based on chaos', *Nonlinear Dynamics*, Vol. 70, No. 4, pp.2383–2388 [online] <https://doi.org/10.1007/s11071-012-0626-5>.
- Li, S. and Zheng, X. (2002) 'Cryptanalysis of a chaotic image encryption method', *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No. 02CH37353)* [online] <https://doi.org/10.1109/iscas.2002.1011451>.
- Liao, X., Lai, S. and Zhou, Q. (2010) 'A novel image encryption algorithm based on self-adaptive wave transmission', *Signal Processing*, Vol. 90, No. 9, pp.2714–2722 [online] <https://doi.org/10.1016/j.sigpro.2010.03.022>.
- Liu, H. and Wang, X. (2010) 'Color image encryption based on one-time keys and robust chaotic maps', *Computers & Mathematics with Applications*, May, Vol. 59, No. 10, pp.3320–3327 [online] <https://doi.org/10.1016/j.camwa.2010.03.017>.
- Liu, Y. and Tong, X-J. (2012) 'A new pseudorandom number generator based on a complex number chaotic equation', *Chinese Physics B*, Vol. 21, No. 9, p.90506 [online] <https://doi.org/10.1088/1674-1056/21/9/090506>.
- Luo, Y., Yu, J., Lai, W. and Liu, L. (2019) 'A novel chaotic image encryption algorithm based on improved baker map and logistic map', *Multimedia Tools and Applications*, Vol. 78, No. 15, pp.22023–22043 [online] <https://doi.org/10.1007/s11042-019-7453-3>.
- Mao, Y. and Chen, G. (2005) 'Chaos-based image encryption', *Handbook of Geometric Computing*, pp.231–265, Springer-Verlag [online] https://doi.org/10.1007/3-540-28247-5_8.
- Matthews, R. (1989) 'On the derivation of a 'chaotic' encryption algorithm', *Cryptologia*, Vol. 13, No. 1, pp.29–42 [online] <https://doi.org/10.1080/0161-118991863745>.
- Monaghan, D.S., Gopinathan, U., Naughton, T.J. and Sheridan, J.T. (2007) 'Key-space analysis of double random phase encryption technique', *Applied Optics*, Vol. 46, No. 26, p.6641 [online] <https://doi.org/10.1364/ao.46.006641>.
- Murillo-Escobar, M.A., Cruz-Hernández, C., Cardoza-Avendaño, L. and Méndez-Ramírez, R. (2016) 'A novel pseudorandom number generator based on pseudorandomly enhanced logistic map', *Nonlinear Dynamics*, Vol. 87, No. 1, pp.407–425 [online] <https://doi.org/10.1007/s11071-016-3051-3>.
- Ozorio de Almeida, A. and Saraceno, M. (1991) 'Periodic orbit theory for the quantized baker's map', *Annals of Physics*, Vol. 210, No. 1, pp.1–15 [online] [https://doi.org/10.1016/0003-4916\(91\)90274-c](https://doi.org/10.1016/0003-4916(91)90274-c).
- Papadimitratos, P. and Haas, Z.J. (2006) 'Secure data communication in mobile ad hoc networks', *IEEE Journal on Selected Areas in Communications*, Vol. 24, No. 2, pp.343–356 [online] <https://doi.org/10.1109/jsac.2005.861392>.
- Pareek, N.K., Patidar, V. and Sud, K.K. (2006) 'Image encryption using chaotic logistic map', *Image and Vision Computing*, Vol. 24, No. 9, pp.926–934 [online] <https://doi.org/10.1016/j.imavis.2006.02.021>.
- Rohith, S., Bhat, K.N.H. and Sharma, A.N. (2014) 'Image encryption and decryption using chaotic key sequence generated by sequence of logistic map and sequence of states of linear feedback shift register', *2014 International Conference on Advances in Electronics Computers and Communications (ICAIECC)*, October [online] <https://doi.org/10.1109/icaecc.2014.7002404>.
- Run-he, Q., Yun, C. and Yu-Zhen, F. (2011) 'Integrated confusion-diffusion mechanisms for chaos based image encryption', *2011 4th International Congress on Image and Signal Processing (CISP)*, October [online] <https://doi.org/10.1109/cisp.2011.6100304>.
- Schack, R. and Caves, C.M. (1992) 'Information and available work in the perturbed baker's map', Presented at the *Workshop on Physics and Computation* [online] <https://doi.org/10.1109/phycmp.1992.615496>.

- Sheela, S.J., Suresh, K.V. and Tandur, D. (2018) 'Image encryption based on modified Henon map using hybrid chaotic shift transform', *Multimedia Tools and Applications*, Vol. 77, No. 19, pp.25223–25251 [online] <https://doi.org/10.1007/s11042-018-5782-2>.
- Smid, M.E. and Branstad, D.K. (1988) 'Data encryption standard: past and future', *Proceedings of the IEEE*, Vol. 76, No. 5, pp.550–559 [online] <https://doi.org/10.1109/5.4441>.
- Song, C-Y., Qiao, Y-L. and Zhang, X-Z. (2013) 'An image encryption scheme based on new spatiotemporal chaos', *Optik – International Journal for Light and Electron Optics*, Vol. 124, No. 18, pp.3329–3334 [online] <https://doi.org/10.1016/j.ijleo.2012.11.002>.
- Stoyanov, B. and Kordov, K. (2015a) 'Image encryption using Chebyshev map and rotation equation', *Entropy*, Vol. 17, No. 4, pp.2117–2139 [online] <https://doi.org/10.3390/e17042117>.
- Stoyanov, B. and Kordov, K. (2015b) 'Novel secure pseudorandom number generation scheme based on two Tinkerbell maps', *Advanced Studies in Theoretical Physics*, Vol. 9, pp.411–421 [online] <https://doi.org/10.12988/astp.2015.5342>.
- Subiyakto, A., Andini, N. and Darlis, D. (2015) 'Security analysis of RGB image encryption based on modified baker map for nanosatellite application', *2015 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET)* [online] <https://doi.org/10.1109/icramet.2015.7380785>.
- Wang, S-Y., Zhao, J-F., Li, X-F. and Zhang, L-T. (2016) 'Image blocking encryption algorithm based on laser chaos synchronization', *Journal of Electrical and Computer Engineering*, pp.1–14 [online] <https://doi.org/10.1155/2016/4138654>.
- Wang, X., Teng, L. and Qin, X. (2012) 'A novel colour image encryption algorithm based on chaos', *Signal Processing*, Vol. 92, No. 4, pp.1101–1108 [online] <https://doi.org/10.1016/j.sigpro.2011.10.023>.
- Wang, Y., Wong, K-W., Liao, X. and Chen, G. (2011) 'A new chaos-based fast image encryption algorithm', *Applied Soft Computing*, Vol. 11, No. 1, pp.514–522 [online] <https://doi.org/10.1016/j.asoc.2009.12.011>.
- Wang, Z., Bovik, A.C., Sheikh, H.R. and Simoncelli, E.P. (2004) 'Image quality assessment: from error visibility to structural similarity', *IEEE Transactions on Image Processing*, Vol. 13, No. 4, pp.600–612 [online] <https://doi.org/10.1109/tip.2003.819861>.
- Wei, X., Guo, L., Zhang, Q., Zhang, J. and Lian, S. (2012) 'A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system', *Journal of Systems and Software*, Vol. 85, No. 2, pp.290–299 [online] <https://doi.org/10.1016/j.jss.2011.08.017>.
- Wong, K-W., Kwok, B.S-H. and Law, W-S. (2008) 'A fast image encryption scheme based on chaotic standard map', *Physics Letters A*, Vol. 372, No. 15, pp.2645–2652 [online] <https://doi.org/10.1016/j.physleta.2007.12.026>.
- Wu, Y., Noonan, J.P. and Aгаian, S. (2011) 'NPCR and UACI randomness tests for image encryption', *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications*, April, pp.31–38.
- Yang, H., Wong, K-W., Liao, X., Zhang, W. and Wei, P. (2010) 'A fast image encryption and authentication scheme based on chaotic maps', *Communications in Nonlinear Science and Numerical Simulation*, Vol. 15, No. 11, pp.3507–3517 [online] <https://doi.org/10.1016/j.cnsns.2010.01.004>.
- Ye, G. (2014) 'A block image encryption algorithm based on wave transmission and chaotic systems', *Nonlinear Dynamics*, February, Vol. 75, No. 3, pp.417–427 [online] <https://doi.org/10.1007/s11071-013-1074-6>.
- Ye, R. and Zhuang, L. (2010) 'The application of an improved baker map in image scrambling and watermarking', *2010 Third International Symposium on Information Processing (ISIP)*, October [online] <https://doi.org/10.1109/isip.2010.85>.
- Yu, Y-H., Chang, C-C. and Lin, I-C. (2007) 'A new steganographic method for color and grayscale image hiding', *Computer Vision and Image Understanding*, Vol. 107, No. 3, pp.183–194 [online] <https://doi.org/10.1016/j.cviu.2006.11.002>.

- Zeghid, M., Machhout, M., Khriji, L., Baganne, A. and Tourki, R. (2007) 'A modified AES based algorithm for image encryption', *World Academy of Science, Engineering and Technology, International Journal of Computer and Information Engineering*, Vol. 1, No. 3, pp.745–750.
- Zhang, G. and Liu, Q. (2011) 'A novel image encryption method based on total shuffling scheme', *Optics Communications*, Vol. 284, No. 12, pp.2775–2780 [online] <https://doi.org/10.1016/j.optcom.2011.02.039>.
- Zhang, L., Liao, X. and Wang, X. (2005) 'An image encryption approach based on chaotic maps', *Chaos, Solitons & Fractals*, Vol. 24, No. 3, pp.759–765 [online] <https://doi.org/10.1016/j.chaos.2004.09.035>.
- Zhang, Q., Guo, L. and Wei, X. (2010) 'Image encryption using DNA addition combining with chaotic maps', *Mathematical and Computer Modelling*, Vol. 52, Nos. 11–12, pp.2028–2035 [online] <https://doi.org/10.1016/j.mcm.2010.06.005>.
- Zhang, Q., Han, J. and Ye, Y. (2019a) 'Image encryption algorithm based on image hashing, improved chaotic mapping and DNA coding', *IET Image Processing*, December, Vol. 13, No. 14, pp.2905–2915 [online] <https://doi.org/10.1049/iet-ipr.2019.0667>.
- Zhang, W., Wong, K., Yu, H. and Zhu, Z. (2013) 'An image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion', *Communications in Nonlinear Science and Numerical Simulation*, Vol. 18, No. 8, pp.2066–2080 [online] <https://doi.org/10.1016/j.cnsns.2012.12.012>.
- Zhang, X., Wang, L., Cui, G. and Niu, Y. (2019b) 'Entropy-based block scrambling image encryption using DES structure and chaotic systems', *International Journal of Optics*, August, Vol. 2019, pp.1–13 [online] <https://doi.org/10.1155/2019/3594534>.
- Zhang, Y., Li, C., Li, Q., Zhang, D. and Shu, S. (2012) 'Breaking a chaotic image encryption algorithm based on perceptron model', *Nonlinear Dynamics*, Vol. 69, No. 3, pp.1091–1096 [online] <https://doi.org/10.1007/s11071-012-0329-y>.
- Zhao, G., Yang, X., Zhou, B. and Wei, W. (2010) 'RSA-based digital image encryption algorithm in wireless sensor networks', *2010 2nd International Conference on Signal Processing Systems (ICSPS)* [online] <https://doi.org/10.1109/icsps.2010.5555601>.
- Zhu, C. (2012) 'A novel image encryption scheme based on improved hyperchaotic sequences', *Optics Communications*, Vol. 285, No. 1, pp.29–37 [online] <https://doi.org/10.1016/j.optcom.2011.08.079>.
- Zhu, C. and Sun, K. (2018) 'Cryptanalyzing and improving a novel color image encryption algorithm using RT-enhanced chaotic tent maps', *IEEE Access*, Vol. 6, pp.18759–18770 [online] <https://doi.org/10.1109/access.2018.2817600>.
- Zhu, C.X., Chen, Z.G. and Ouyang, W.W. (2006) 'A new image encryption algorithm based on general Chen's chaotic system', *Zhongnan Daxue Xuebao (Ziran Kexue Ban)/Journal of Central South University (Science and Technology)*, Vol. 37, No. 6, pp.1142–1148.