

A Novel Image Encryption Scheme Based on Intertwining Chaotic Maps and RC4 Stream Cipher

Manju Kumari · Shailender Gupta

Received: 4 January 2018 / Revised: 14 February 2018 / Accepted: 19 February 2018 / Published online: 7 March 2018
© 3D Research Center, Kwangwoon University and Springer-Verlag GmbH Germany, part of Springer Nature 2018

Abstract As the systems are enabling us to transmit large chunks of data, both in the form of texts and images, there is a need to explore algorithms which can provide a higher security without increasing the time complexity significantly. This paper proposes an image encryption scheme which uses intertwining chaotic maps and RC4 stream cipher to encrypt/decrypt the images. The scheme employs chaotic map for the confusion stage and for generation of key for the RC4 cipher. The RC4 cipher uses this key to generate random sequences which are used to implement an efficient diffusion process. The algorithm is implemented in MATLAB-2016b and various performance metrics are used to evaluate its efficacy. The proposed scheme provides highly scrambled encrypted images and can resist statistical, differential and brute-force search attacks. The peak signal-to-noise ratio values are quite similar to other schemes, the entropy values are close to ideal. In addition, the scheme is very much practical since having lowest time complexity than its counterparts.

Keywords Intertwining chaotic map · RC4 · Image encryption/decryption · Statistical/differential attacks · Diffusion

1 Introduction

Transmission of digital images (personal and professional) over public data networks have become a necessity in the existing era. The current infrastructure allows us to transmit huge amount of data due to technological advancements and towering bandwidth support. The data flowing through these channels can be accessed by unauthorized users like ransomware. Figure 1 present an approximate total of ransomware family from 2004 till 2016 with a projected total for year 2017 [43]. Therefore, researcher keeps on searching new algorithms in order to make data confidential. Most of the initial algorithms for providing confidentiality focused on encrypting small chunks of data via symmetric [4, 5, 30, 39] and asymmetric key algorithms. These conventional schemes are vulnerable to attacks by unauthorized users [6, 49]. Also, the initial schemes turn out to be ineffective for encryption [10, 42, 46, 48] since, the image possesses high redundancy and correlation among adjacent pixels which motivated various researchers to branch out and construct efficient algorithms for encrypting digital images.

Since 1989, various image encryption [21, 27–51] algorithms are proposed. These algorithms are based on various techniques like chaotic systems with Kolmogorov flow [38], or with total shuffling [50], vector quantization [8], and iterative random encoding [25]. The chaotic based systems attracted a lot of attention of researchers due to their high dependency

M. Kumari (✉) · S. Gupta
Department of Electronics Engineering, YMCA
University of Science and Technology, Faridabad, India
e-mail: manju_mrce027@yahoo.com

S. Gupta
e-mail: shailender81@gmail.com

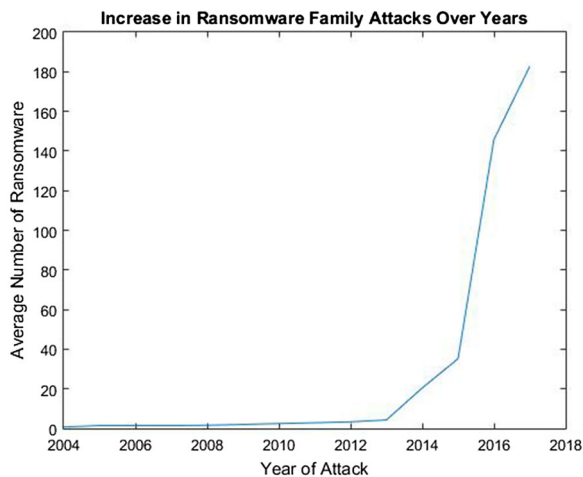


Fig. 1 Annual number of ransomware families

on the initial condition, simplicity, computational efficiency and confusion/diffusion process to make the system more non-linear in nature and hence harder to predict [3, 19, 44, 45] for intruders/hackers.

In this paper, a novel encryption/decryption based on intertwining chaotic maps and RC4 stream cipher has been proposed. The proposed scheme can be used to encrypt/decrypt colored images and provides a high interlinking between the Red (R), Green (G), and Blue (B) planes of the image. So, a minute change in the components of any plane will provide an encrypted image significantly different from the originally encrypted image. Moreover, the RC4 cipher is well known for providing a high key sensitivity and the scheme is not breakable by any attack in a sensible amount of time. All this indicates high security levels for using the scheme in chaotic implementations. Also, the scheme is significantly fast and it ensures high protection against the brute force attack because of its large key spaces available.

The second section is a review of various encryption schemes present in literature. This section contains some of the most popular conventional encryption schemes and a few of modern chaos-based encryption schemes. Third section explains the proposed chaos-based algorithm. The section details about the various components of the scheme which are key generation using intertwining chaotic map, random sequence generation using the RC4 cipher, diffusion and confusion. Also, the algorithm is explained in this section. Fourth section contains the simulation setup parameters used. These metrics

include visual assessment, statistical analysis by histogram and correlation analysis, differential attack analysis, key space analysis, quantitative and time complexity analysis. The fifth section contains the results based on these performance metrics. And finally, the sixth section summarizes and concludes the study.

2 Literature Review

Modern cryptography is shaped by a series of efficient algorithms each one of which had their pros and cons. This section is an effort to summarize the most popular conventional cryptography techniques used for encryption of textual data along with some modern chaos-based techniques used for encrypting images. Table 1 summarizes the cryptography techniques.

The conventional cryptography techniques were originally developed to encrypt textual information that was not suitable for encrypting images due to high correlation between adjacent pixels and high time complexity. These shortcomings were removed by chaos-based encryption. The salient features of these schemes [15, 47] are as follows:

- Simplicity
- Large key space
- Sensitivity to key change
- Resist against differential and statistical attacks
- Low computational time

The chaos based mechanisms are used efficiently in both one-dimensional [31] and multi-dimensional implementations [9, 13]. Some chaotic encryption schemes were designed only to encrypt gray scale images [23, 24, 33], while many others can encrypt color RGB images too [20]. Though, chaos based encryption schemes are better but are still vulnerable to cryptanalysis [41, 51]. This occurs when improper confusion or diffusion is employed or a bad key is used. We are of the opinion that the salient features of chaos based schemes can still be improved by choosing proper confusion and diffusion process and by keeping the process as simple as possible. In addition, for the process to be practically applicable, it should be fast. This paper is an effort for the same. To prove the efficacy of the proposed scheme, it is compared with most of the popular schemes in literature mentioned below:

Table 1 Summary of popular convention and modern chaos-based cryptography techniques

| Year | Method | Author | Advantages | Disadvantages |
|------|--|---|--|--|
| 1553 | Vigenère cipher | G. B. Bellaso | Easy to implement | Uses repeating key Can be completely broken |
| 1975 | Data encryption standard (DES) | H. Feistel | Highly influenced modern cryptography Slight change in plaintext will change cipher text drastically | Small key size Popular cryptanalysis available Linear cryptanalysis Brute force attack |
| 1976 | Diffie–Hellman key exchange | W. Diffie, M. Hellman, and R. Merkle | Originated as alternate for exchanging keys physically Simple to implement | Weak against attacks by funded organizations Man-in-the-middle attack |
| 1977 | RSA | R. Rivest, A. Shamir, and L. Adleman | Used in various practical systems like: Email privacy, Protecting web traffic RSA can sign a message | Time complexity is high RSAs with short key lengths are vulnerable for attacks |
| 1991 | International data encryption algorithm (IDEA) | X. Lai and J. Massey | Modified versions provide higher security, better diffusion, etc. Easy to implement | Comparatively slower algorithm Popular breaks Meet-in-the-middle attack, Narrow-bicliques attack |
| 1993 | Blowfish | B. Schneier | Very fast, except for key changing Slow key change protects against dictionary attacks | Key management is complicated Popular attacks: Birthday attack, Known-plaintext attack (for reduced Blowfish) |
| 1994 | Visual cryptography | M. Naor and A. Shamir | All n shares are required to decrypt the image Simple implementation | Collaborating cheaters can create original message Decrypted image shows resolution loss |
| 1994 | RC4 | R. Rivest | Simple and fast in implementation Variants available to provide additional strength: Spritz [35], RC4A, RC4+ | Highly vulnerable when keys are related or nonrandom Popular attacks: Royal Holloway attack: NOMORE attack |
| 1994 | RC5 | R. Rivest | Higher rounds (18–20) provide high security Concise design [34] Customize secret-key length, group size etc. | Lesser round (12) versions are weak against differential attacks [7] Lesser key size also increases vulnerability |
| 1998 | RC6 | R. Rivest, M. Robshaw, R. Sidney, and Y. L. Yin | Variable key sizes, word length etc. Does extra multiplication not available in RC5 | Possible attacks: Multiple linear attack (on 14 round RC6) [18], χ^2 attack (On 17 round RC6) [26] |
| 1998 | Triple data encryption standard (TDES) | ANS X9.52 | Simple to implement When all keys are different, it provides best key space | Popular attacks: Lucks [40], Sweet32 attack Considerably slower than AES |
| 1998 | Advanced encryption standard (AES) | V. Rijmen and J. Daemen | No computationally feasible attack possible Can be implemented on the cheapest processors | Lesser round AES are susceptible to attacks [11] There is scope of improving the keys of AES |

Table 1 continued

| Year | Method | Author | Advantages | Disadvantages |
|------|--|---|---|--|
| 2012 | Scheme based on intertwining chaotic maps [37] | I. S. Sam, P. Devaraj, and R.S. Bhuvaneshwaran | Diffusion process is highly nonlinear High NPCR and UACI values | Decryption time is higher than the encryption time Decrypted images not provided for comparison |
| 2012 | Scheme based on chaotic function using linear congruences [12] | M. François, T. Grosge, D. Barchiesi, and R. Erra | Produces large key space Transmission security high | Comparatively higher time complexity Decrypted images not provided for comparison |
| 2012 | Scheme based on mixed transform logistic maps [36] | I. S. Sam, P. Devaraj, and R. S. Bhuvaneshwaran | Diffusion process is highly nonlinear High NPCR and UACI values | Protection against new type of attacks is not known Decrypted images not provided for comparison |
| 2015 | Scheme based on Peter De Jong chaotic map and RC4 stream cipher [14] | G. Hanchinamani, and L. Kulkarni | Needs only two rounds for high level of security High NPCR and UACI values | Cannot encrypt RGB images Decrypted image shows changes in pixel values |
| 2016 | Scheme based on chaotic map and Vigenère scheme [2] | R. Bansal, S. Gupta, and G. Sharma | Very fast chaotic algorithm Produces very large key space | Protection against new type of attacks is not known PSNR minutely lesser than other chaotic schemes |

- Scheme Based on a Peter De Jong Chaotic Map and a RC4 Stream Cipher, referred here as Scheme 1.
- Scheme based on chaotic function using linear congruences, referred as Scheme 2.
- Scheme based on mixed transform logistic maps, referred as Scheme 3.
- Scheme based on intertwining chaotic maps, referred as scheme 4.

The next section gives the proposed encryption scheme

3 Proposed Encryption Scheme

The proposed scheme follows an architecture popular in many of the chaos-based cryptography schemes. It contains the following steps:

1. Confusion
2. Diffusion

Confusion is an iterative step of the process. It is the permutation stage in which the pixels of the plain-image are permuted among each other. So, the intensities of the pixel values do not change but their positions are randomized. Diffusion is another

iterative step in which each pixel of the image is modified using different values of the key to provide a high complexity to the process. Basically, these confusion and diffusion steps are repeated iteratively in order to provide randomness. In the proposed work only two iterations were enough to provide a high amount of randomization. The proposed scheme has four major steps. These are:

- Key generation using Intertwining Chaotic Map
- Random sequence generation using RC4
- Confusion
- Diffusion

3.1 Encryption

Figure 2 shows the block diagram of the encryption end of the proposed scheme. As shown, the keys generated using the intertwined chaotic maps are used for the confusion step and as the input for RC4. The RC4 block uses the keys as input and generates random sequences which are used for the diffusion process. The diffusion process is done row-wise and column-wise both in forward and backward directions. Here, a three dimensional image containing M rows and N columns is used. The intensities of Red,

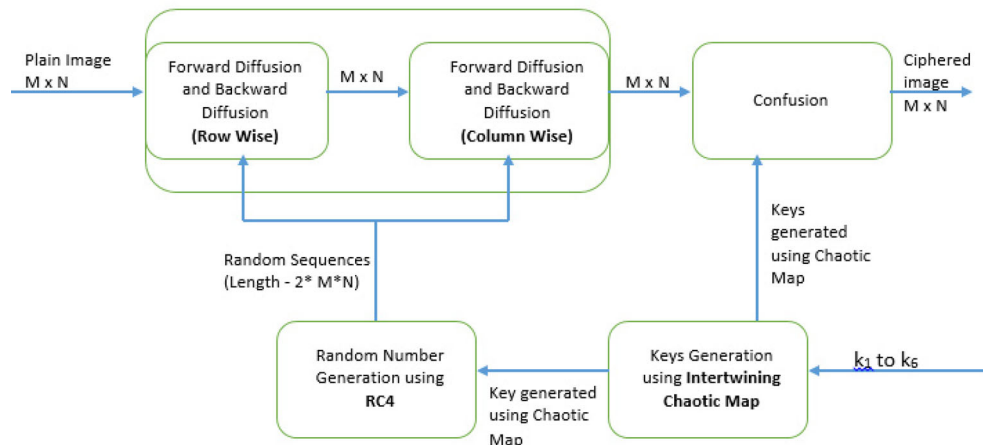


Fig. 2 Block diagram of the encryption end of the proposed scheme

Green, and Blue channels in the image ranges from 0 to 255. The detailed explanation of different steps of the algorithm is given in sections below.

3.1.1 Intertwining Chaotic Map and Key Generation

Chaotic maps are the mathematical functions which provide a dynamic nature to the system. Logistic maps, also known as log maps, and modified logistic maps are used by researchers to obtain dynamic encryption algorithms. The log map provides high randomness, unpredictability and has simple mathematical representation which is given as:

$$x_{n+1} = r * x_n(1 - x_n) \tag{1}$$

where x_n is the initial value and $x_n \in (0, 1)$, and $r \in (0, 4]$. As the function is dependent on the initial conditions and hence provides security than a similar non-chaotic system. Even after these advantages the log maps have an uneven distribution of sequences, stable windows, and a weak key [16], hence are not practical choices for modern encryption schemes. Modified logistic maps, like intertwining chaotic maps, are aimed to make the scheme more practical by removing the mentioned weakness. Mathematically, intertwining logistic maps are given as:

$$x_{n+1} = \text{mod}((\mu * k_4 * y_n * (1 - x_n) + z_n), 1) \tag{2}$$

$$y_{n+1} = \text{mod} \left(\left((\mu * k_5 * y_n) + \left(z_n * \frac{1}{1 + (x_n + 1)^2} \right) \right), 1 \right) \tag{3}$$

$$z_{n+1} = \text{mod}((\mu * (x_n + y_n + k_6) * \sin(z_n)), 1) \tag{4}$$

where $\mu \in (0, 3.999]$, $|k_4| > 33.5$, $|k_5| > 37.9$, $|k_6| > 35.7$.

Figure 3 shows the key generation scheme. The keys k_1 to k_6 were the defined inputs for required for key generation. These keys were used by three paths, namely X path, Y path and Z path, to create three different intertwined chaotic maps, namely $x_{\text{log_diff}}$, $y_{\text{log_diff}}$, and $z_{\text{log_diff}}$. Each path of the chaotic maps used two keys as input parameters and generate sequences which were further used for the confusion and RC4 processes. All these sequences are 1-dimensional vectors.

The sequences generated from the X path and the Y path combined was used for the confusion stage. The element values in both of these sequences range from 0 to $M*N$. Each of the map has a total $M*N$ elements in it, hence making a successful implementation of the confusion process for any image of $M*N$ size. The sequence generated from the Z path was used as the input for RC4 random sequence generation. As the key length required for the RC4 algorithm ranges from 1 to 256 bytes and does not depend on the size of the image, so the sequence generated from the Z path has 256 elements in it, with values ranging from 0 to 256.

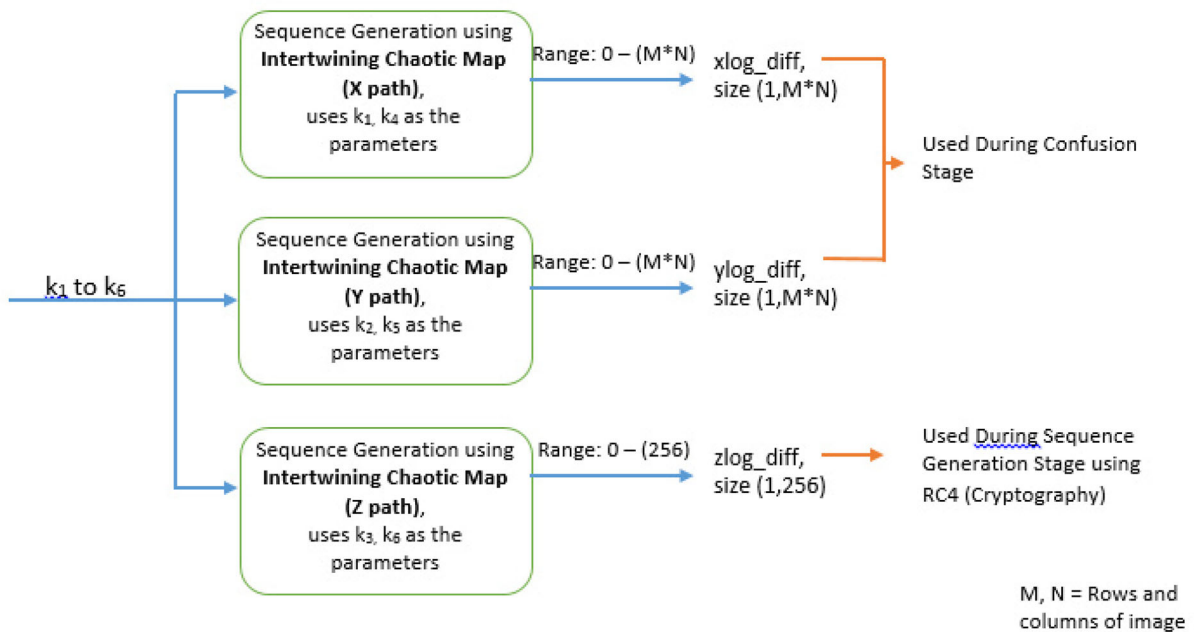


Fig. 3 Key generation using intertwining chaotic map

3.1.2 RC4 and Random Sequence Generation

The RC4 is a symmetric key, stream cipher and known for its speed and simplicity. It was initially developed in 1987 as a trade secret but in 1994 was leaked to public. During the encryption process, the cipher is fed with a key whose length can vary from 1 byte to 256 bytes. This cipher generates a randomized array which is obtained by feeding the key to a pseudorandom byte generator. The output of the pseudorandom generator is known as the random sequences. The random sequences generated in the proposed algorithm had a length of $2*M*N$ and were fed as the inputs of the diffusion process. The decryption process follows similar steps as that of encryption process. Figure 4 shows the block diagram of RC4 cipher end of the proposed algorithm.

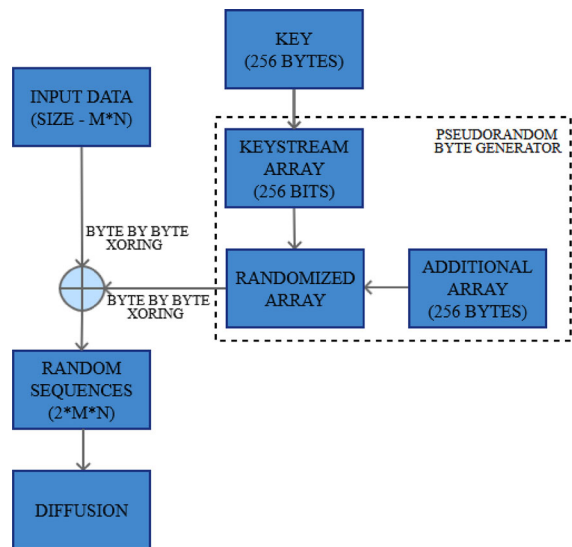


Fig. 4 Block diagram of RC4 cipher

3.1.3 Diffusion

Diffusion is one of the two important stages of the proposed algorithm stages which adds randomness in the pixels of the plain-image. The diffusion property is responsible for spreading the effect of change in a pixel intensity value throughout the entire image. So

any change in the intensity of one pixel value will change the encrypted image drastically and both the images formed will have no or very low resemblance with each other. Figure 5 shows hierarchy of the diffusion process.

As shown in figure above, the diffusion step can be branched into row-wise and column-wise diffusion and both diffusion steps are done in forward and backward directions. In row-wise diffusion, the image is scanned row-wise in alternate directions. This scanning is used to turn the $M \times N$ image into an array form before the actual diffusion steps are used. Then the generated array is diffused in forward followed by backward direction. Once the row-wise diffusion is done, the image is turned back into the $M \times N$ form and then it is scanned column-wise to form another similar array followed by similar forward and backward diffusion steps. Finally, the image is again turned back into the $M \times N$ form and passed to the confusion block.

3.1.4 Confusion

The confusion stage is another step which adds randomness in the pixels of the plain-image. The confusion stage used in the proposed algorithm is a permutation of pixel positions of the original image. The stage shuffles the position of the pixels present in the image while making no changes in the respective pixel intensity values. This stage ensures that an unauthorized user who tries to access the data present in the image will get no useful data about the image as the pixels are moved from their original positions. The

permutation of pixel position helps to achieve lower values of correlation coefficients as the correlation present in the adjacent pixels in original image is removed by the movement. Figure 6 shows the confusion stage of the proposed scheme.

3.2 Decryption

Decryption process follows steps similar to the encryption process. Here, the encrypted image is taken as input and the steps used by the encryption process are used in reverse order.

3.3 Algorithm Explanation for Encryption/Decryption

The algorithm was executed in three separate channels. The three channels are denoted here as R, G and B which are used for the Red, Green and Blue planes. The plane constitutes the red, green, and blue components of the image respectively. Both, the encryption and decryption algorithms are executed for all three

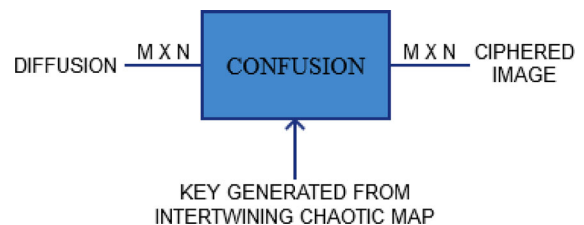


Fig. 6 Confusion stage of the proposed scheme

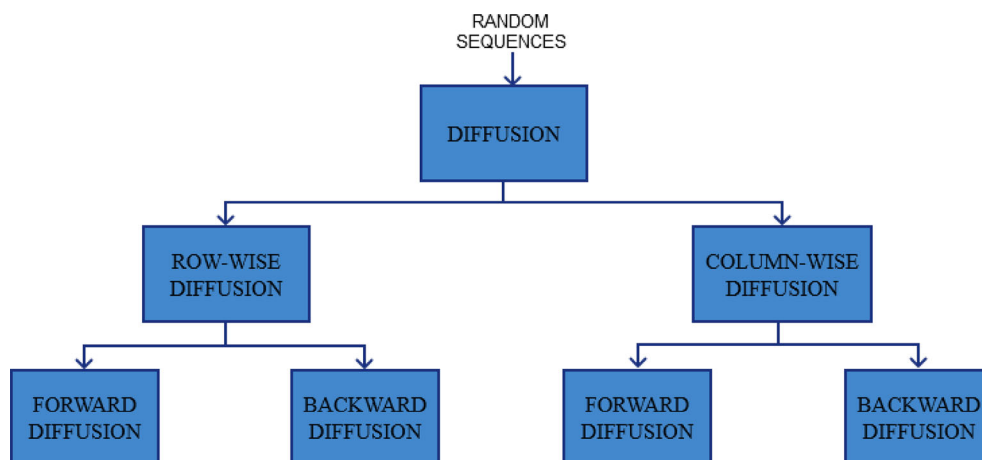


Fig. 5 Hierarchy of the diffusion process

channels separately keeping the same keys and other algorithm variables, i.e., the algorithm is executed three times with the only change of the channel used. The three channels were merged again before providing the encrypted or decrypted images. The steps used for key generation, encryption, and decryption processes are given.

3.3.1 Encryption Algorithm

1. *Step 1* Divide the image into Red, Green and Blue channels.
2. *Step 2* Generate $xlog_diff$, $ylog_diff$, and $zlog_diff$ using intertwining chaotic maps by using the respective keys as defined above.
3. *Step 3* For Red channel, generate a random sequence of length $2*M*N$ by using $zlog_diff$, following the process define above in the respective section.
4. *Step 4* Scan the image row-wise to generate a 1-dimensional array and apply forward diffusion followed by backward diffusion.
5. *Step 5* Convert the array back to $M*N$ form.
6. *Step 6* Repeat steps 4 and 5 for column-wise diffusion.
7. *Step 7* Apply the confusion process.
8. *Step 8* Repeat steps 3–7 + for the Green channel and after it repeat the same for the Blue channel.
9. *Step 9* Recombine the three channels together to get the encrypted image.

3.3.2 Decryption Algorithm

1. *Step 1* Divide the encrypted image into the channels.
2. *Step 2* For Red channel, apply the reverse confusion step.
3. *Step 3* Apply the RC4 decryption stage.
4. *Step 4* Scan the image column-wise to perform reverse diffusion, first in backward then in forward directions for the generated array, and turn the array back into image.
5. *Step 5* Repeat step 4 for row-wise diffusion.
6. *Step 6* Repeat steps 2–5 for green and blue channels.
7. *Step 7* Recombine the three channels to get the decrypted image.

The next section gives the detail of setup parameters used to evaluate the efficacy of our proposed scheme.

4 Simulation Setup Parameters

4.1 Setup Parameters

The proposed algorithm, along with the other algorithms used in the study, was implemented on a personal computer. Table 2 provides the specification of the machine, test plain-images and initial and modified parameters used for the algorithms in the study.

4.2 Performance Metrics Used

4.2.1 Visual Assessment

Visual assessment is one crucial parameter which ensures that no information about the original image can be extracted by looking at the encrypted image. A high pixel scrambling in the encrypted image can distort the visual data of original image and hence can secure a high visual security.

4.2.2 Statistical Attack Analysis

The relation between the pixels of the images can be found by performing their statistical attack analysis. These include histogram and correlation analysis and are described below in detail.

4.2.2.1 Histogram Analysis For a digital image, a histogram is a representation of frequency distribution of pixels of the image in graphical form. In general, an original RGB image has a non-uniform histogram distributed with occasional spikes present in it. On the other hand, the histogram of an ideally encrypted image will be evenly distributed and hence will has no resemblance with the histogram of original image. Histogram analysis is plays a crucial role as the highly secure data encryption schemes, like AES, are vulnerable by histogram based cryptanalysis [1, 17].

4.2.2.2 Correlation Analysis A digital image, in general, has a high level of correlation between the adjacent pixels which provides the effect of image as a

Table 2 Machine and image specifications, initial and modified parameters

| Specifications | |
|--|--|
| Processor | 1.4 GHz dual-core Intel Core i5 |
| Memory | 4 GB of 1600 MHz LPDDR3 |
| Operating system | OS X El Capitan |
| Simulation platform | MATLAB |
| Version | 2015 |
| Size of images | 256 × 256, 512 × 512 |
| Images source | USC-SIPI image database |
| Type | Color images |
| Key used in | |
| Scheme 1 [a, b, c, d, x0, y0] | [1.77, 1.67, − 0.85, 2.1, 0.6, 0.4] |
| Scheme 2 | [4713 654, 84 287, 7487, 1984, 12 314, 10, 74 120, 130 014, 95 210, 1914, 70 553, 2835, 19 800, 299 314, 83 721,610 990, 210, 65 521, 396, 1 109 094, 230 014, 63 010, 10 246] |
| Scheme 3 [k1, k2, k3] | [37.8, 39.8, 37.3] |
| [oddkey1, oddkey2, oddkey3, oddkey4, oddkey5, oddkey6] | [1, 5, 99, 111, 7, 77] |
| Scheme 4 | [33.1,37.3,35.7] |
| Modified key | |
| Scheme 1 [a, b, c, d, x0, y0] | [1.771, 1.67, − 0.85, 2.1, 0.6, 0.4] |
| Scheme 2 | [4714 654, 84 287, 7487, 1984, 12 314, 10, 74 120, 130 014, 95 210, 1914, 70 553, 2835, 19 800, 299 314, 83 721,610 990, 210, 65 521, 396, 1 109 094, 230 014, 63 010, 10 246] |
| Scheme 3 [k1, k2, k3] | [37.81, 39.8, 37.3] |
| Scheme 4 | [33.11,37.3,35.7] |

whole. This correlation is present in horizontal, vertical and diagonal directions. But for an encrypted image there should be no correlation between the adjacent pixels as an unauthorized user can take advantage of this and can reconstruct the originally image either partially or fully. The value of correlation coefficient typically lies between − 1 and + 1, with zero representing no linear relation between the adjacent pixels. Mathematically this can be represented as:

$$r_{\alpha\beta} = \frac{cov(\alpha, \beta)}{\sqrt{D(\alpha)}\sqrt{D(\beta)}} \tag{5}$$

$$E(\alpha) = \frac{1}{N} \sum_{i=1}^N \alpha_i \tag{6}$$

$$D(\alpha) = \frac{1}{N} \sum_{i=1}^N (\alpha_i - E(\alpha))^2 \tag{7}$$

$$cov(\alpha, \beta) = \frac{1}{N} \sum_{i=1}^N (\alpha_i - E(\alpha))(\beta_i - E(\beta)). \tag{8}$$

4.2.3 Differential Attack Analysis

A secure encryption algorithm should provide totally different encrypted image if any of initial conditions, i.e. pixel or key values, is changed even by a single bit. In order to evaluate the robustness of an algorithm against the differential attack two parameters, namely net pixel change ratio (NPCR) and unified average change in intensity (UACI), are used. These parameters are explained below.

In NPCR, two encrypted images are compared on the basis of the pixel difference. The two encrypted images are obtained from the original plain-image and one-pixel modified plain-image encrypted using the same encryption algorithm. Let C1 and C2 be the respective encrypted images. Mathematically, NPCR is given as:

$$NPCR = \frac{\sum_{i=1}^H \sum_{j=1}^W D(i,j)}{W \times H} \times 100\% \quad (9)$$

where H is the height and W is the width of the images. D is a bipolar array of size equal to the images and constitute components 0 and 1 only. Mathematically, each component D(i, j) is given as:

$$D(i,j) = \begin{cases} 0 & C1(i,j) = C2(i,j), \\ 1 & C1(i,j) \neq C2(i,j) \end{cases} \quad (10)$$

UACI is a representation of average intensity difference between the plain and encrypted images. Mathematically, it is given as:

$$UACI = \frac{1}{W \times H} \left[\sum_{i=1}^H \sum_{j=1}^W \frac{|C1(i,j) - C2(i,j)|}{2^L - 1} \right] \times 100\% \quad (11)$$

where L is the number of bits of red, green and blue channels.

There are two conditions defining the efficiency of the differential attack analysis. In both the conditions, an original key and a key obtained by minutely changing the original key are used. According to the first condition, if an image is encrypted to get two ciphered images using both the keys individually, then the ciphered images should be completely different. According to the second condition, if an encrypted image is decrypted using the two keys separately, then only the image obtained by original key should provide the original image with no data loss. On the other hand, the image obtained by the modified key should not provide any information, partially or fully, about the original image.

4.2.4 Key Space Analysis

The key space analysis is done to determine the robustness of an encryption algorithm against the brute-force attack [22, 32]. The analysis evaluates this on the basis of number of key combinations an algorithm produces.

4.2.5 Quantitative Analysis

Quantitative analysis is comparison of image enhancement obtained by the scheme. The parameters of the

analysis included in the study are peak signal-to-noise ratio (PSNR) analysis and information entropy analysis.

4.2.5.1 Peak signal-to-noise ratio (PSNR) analysis PSNR is the ratio between the maximum power component and noise present with a signal. In case of images, the plain-image is considered as the signal and the encrypted one as the noise. Mathematically, it is given as:

$$PSNR = 20 \times \log_{10} \left(\frac{255}{\sqrt{MSE}} \right) \text{dB} \quad (12)$$

where MSE is mean squared error. Mathematically, it is given as:

$$MSE = \frac{1}{WH} \sum_{i=1}^W \sum_{j=1}^H [I(i,j) - K(i,j)]^2 \quad (13)$$

where I represents the pixel values of original and k for the encrypted images and (i, j) is the location.

4.2.5.2 Information Entropy Analysis For an encryption scheme, entropy is the amount of randomness obtained in the pixels of encrypted image when compared to the pixels of plain-image. Mathematically, it is given as:

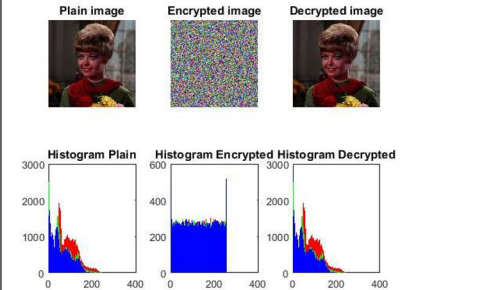
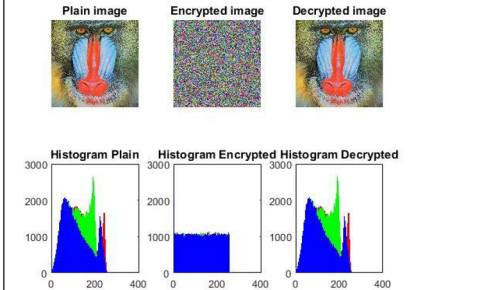
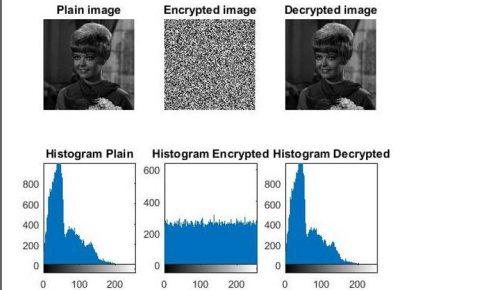
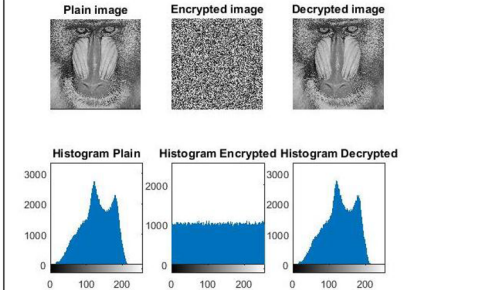
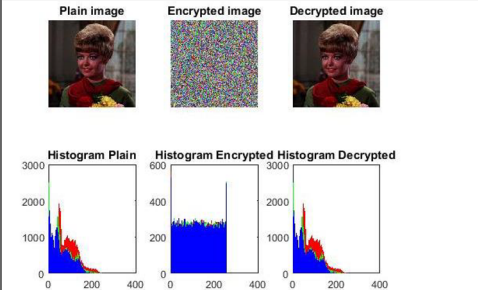
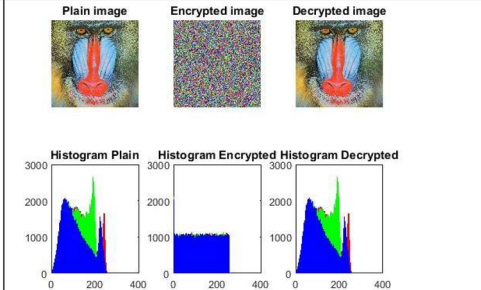
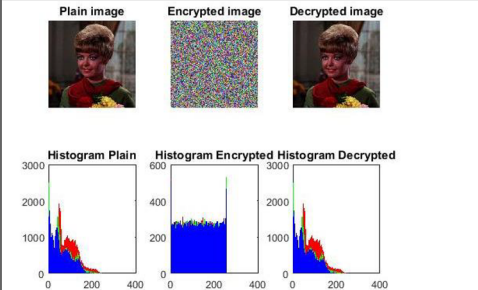
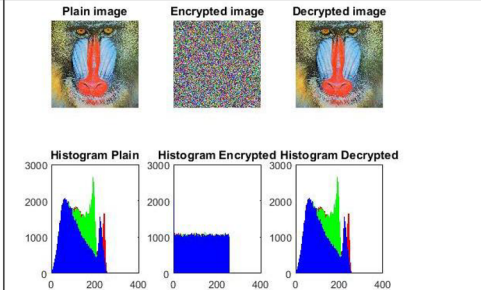
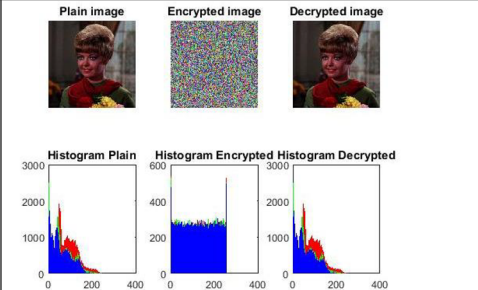
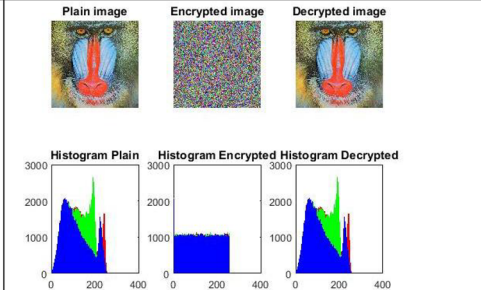
$$H(S) = \sum_{i=0}^{n-1} P(S_i) \log_2 \left(\frac{1}{P(S_i)} \right) \quad (14)$$

where n is the number of symbols, S_i is the pixel values and P(S_i) represents the probability of occurrence of S_i.

4.2.6 Time complexity analysis

Time complexity is the time taken by the system to execute the scheme. This time is the encryption/decryption time of the scheme. It can be calculate manually as well as by using the in-built operations of the software used for implementation. Here, the tic toc function of MATLAB. To get an unbiased comparison, the parameters of Table 2 were kept the same during the study. Any variations in these parameters can affect the time complexity drastically.

Table 3 Visual assessment and histogram analysis of proposed and other schemes

| Image/ scheme | 4.1.01 (Girl) | 4.2.03 (Mandrill) |
|------------------|---|--|
| Proposed |  |  |
| Scheme 1 |  |  |
| Scheme 2 |  |  |
| Scheme 3 |  |  |
| Scheme 4 |  |  |

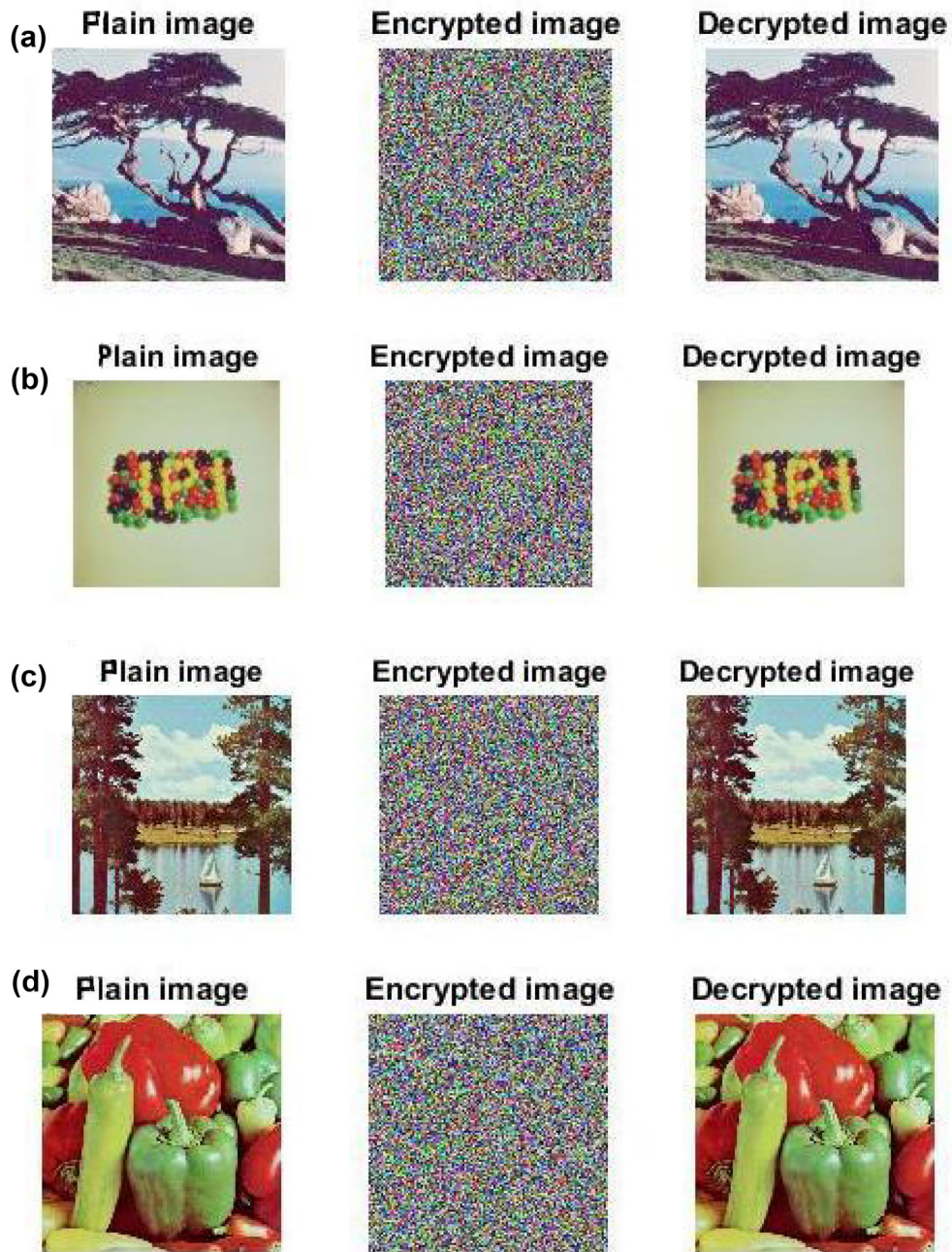


Fig. 7 Visual assessment of encrypted and decrypted images

5 Results

5.1 Visual Assessment

The visual assessment of the encrypted and decrypted images obtained from the proposed scheme can be

done from Fig. 7. The figure shows visual assessment of four test images. From the figure, it can be clearly seen that all the encrypted images obtained are highly scrambled and there is no visual resemblance between them and the original images. So, the information contained in the image, or any part of it, is too complex

Table 4 Correlation coefficient of original and encrypted images

| Correlation/image | Original image | | | Encrypted image | | |
|----------------------|----------------|-----------|-----------|-----------------|-------------|------------|
| | Horizontal | Vertical | Diagonal | Horizontal | Vertical | Diagonal |
| 4.1.01 (Girl) | 0.9652644 | 0.9731423 | 0.9467103 | − 0.001946 | − 0.0103026 | 0.0082849 |
| 4.1.02 (Couple) | 0.9524021 | 0.9507475 | 0.9154162 | 0.0027958 | − 0.0038164 | 0.0021565 |
| 4.1.05 (House) | 0.9339777 | 0.9668360 | 0.9117694 | 0.0162772 | − 0.0038022 | − 0.010902 |
| 4.1.06 (Tree) | 0.9355816 | 0.9570519 | 0.9102446 | 0.0175922 | 0.00564358 | 0.001048 |
| 4.1.07 (Jelly beans) | 0.9766433 | 0.9751024 | 0.9538027 | 0.0035527 | − 3.77E−04 | − 0.005664 |
| 4.2.03 (Mandrill) | 0.8740496 | 0.9205894 | 0.8581495 | − 0.012267 | − 0.0059264 | − 0.004738 |
| 4.2.04 (Lena) | 0.9895723 | 0.9797832 | 0.9714114 | 0.0083380 | 0.00101641 | − 0.00426 |
| 4.2.05 (Airplane) | 0.9615690 | 0.9736587 | 0.9368550 | 0.0138252 | 0.01247294 | − 0.003707 |
| 4.2.06 (Sailboat) | 0.9546872 | 0.9566964 | 0.9400952 | 0.0063655 | − 0.0017267 | 0.0032007 |
| 4.2.07 (Peppers) | 0.9638862 | 0.9627805 | 0.9583060 | 6.17E−04 | − 0.0154179 | − 0.012664 |

Table 5 Comparison of correlation coefficients of proposed and other schemes

| Image/scheme | 4.1.05 (House) | | | 4.2.05 (Plane) | | |
|--------------|----------------|-------------|-------------|----------------|-------------|------------|
| | Horizontal | Vertical | Diagonal | Horizontal | Vertical | Diagonal |
| Proposed | 0.01627721 | − 0.0038022 | − 0.0109025 | 0.01382524 | 0.01247294 | − 0.003707 |
| Scheme 1 | − 0.0064068 | 0.01312022 | − 0.0065064 | 0.00495188 | − 0.0080406 | 0.0136438 |
| Scheme 2 | 0.00298641 | − 0.0064262 | 0.00742554 | − 0.0017259 | 0.00633413 | 1.43E−04 |
| Scheme 3 | 0.00831666 | − 0.0057369 | 0.0146687 | 0.00625204 | 0.01639709 | − 0.004414 |
| Scheme 4 | − 0.0176803 | − 0.0024506 | − 0.0069457 | − 0.0146412 | − 0.013845 | − 0.004889 |

to be resolved by a human eye. Moreover, it can also be seen that the decrypted images are visually same as the plain-images, hence ensuring the reliability of data obtained after decryption.

Additionally, a visual assessment comparison of proposed and other implemented chaotic schemes is provided in Table 3. From the table, it is clear that the encrypted image obtained by proposed scheme has same visual scrambling as of other schemes. The decrypted images obtained by the proposed scheme are also as visually acceptable as the ones obtained from other schemes.

5.2 Statistical Attack Analysis

5.2.1 Histogram Analysis

The histograms of the original, encrypted and decrypted images are provided in Table 3. The

histograms of the proposed and three other schemes contain three colors, representing the frequency distribution of the red, green and blue color plane. The scheme 1 can encrypt the gray-scale images, so the histograms in this scheme do not contain the three planes.

Table 3 also provides the histogram analysis of the proposed and other schemes. In the table, it can be seen that the histograms of encrypted images obtained by the proposed scheme are uniformly distributed and have no resemblance with the histograms of the respective original images. So, the proposed algorithm has a strong resistance for the statistical attacks. Moreover, the histograms of the decrypted images are similar to the histograms of respective original images. Hence there is no measurable data loss during the process.

Table 6 Comparison of correlation plots of original and encrypted images

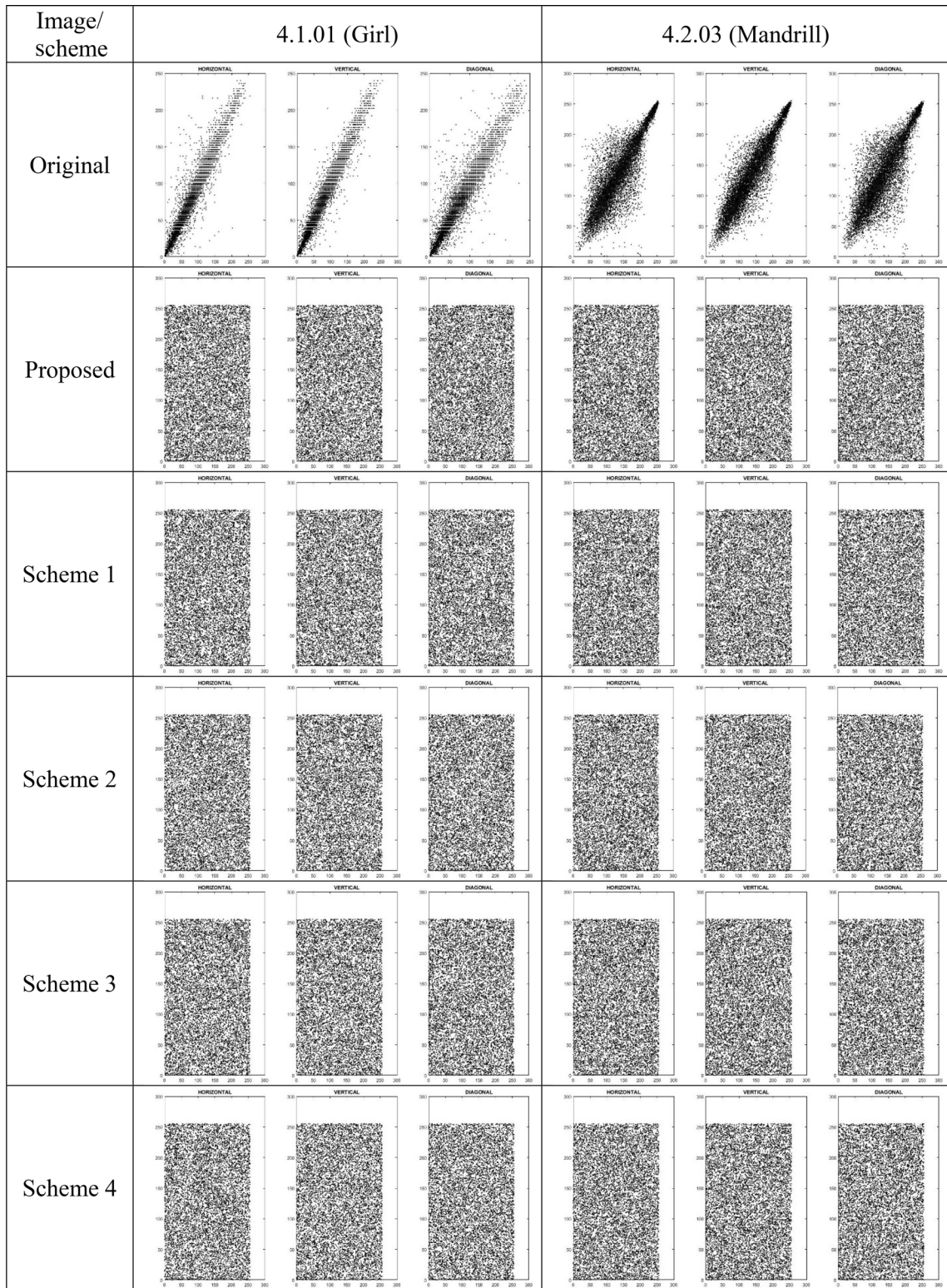


Table 7 NPCR and UACI for one-bit key change

| Scheme/image | Proposed | Scheme 1 | Scheme 2 | Scheme 3 | Scheme 4 |
|----------------------|------------|------------|------------|------------|------------|
| 4.1.01 (Girl) | | | | | |
| NPCR | 0.99597677 | 0.99609375 | 0.99594116 | 0.99597168 | 0.99615987 |
| UACI | 0.35362315 | 0.33445273 | 0.33354753 | 0.33302463 | 0.33421759 |
| 4.1.02 (Couple) | | | | | |
| NPCR | 0.99610901 | 0.99598694 | 0.99571228 | 0.99544779 | 0.99604797 |
| UACI | 0.33523305 | 0.3340253 | 0.33174458 | 0.3335103 | 0.3347666 |
| 4.1.05 (House) | | | | | |
| NPCR | 0.9958547 | 0.99668884 | 0.99606832 | 0.99638875 | 0.99605815 |
| UACI | 0.29576424 | 0.33425167 | 0.33481435 | 0.33452906 | 0.33453648 |
| 4.1.06 (Tree) | | | | | |
| NPCR | 0.99605306 | 0.99629211 | 0.99617004 | 0.99612935 | 0.99587504 |
| UACI | 0.31895481 | 0.33527988 | 0.33431837 | 0.33478773 | 0.33420432 |
| 4.1.07 (Jelly beans) | | | | | |
| NPCR | 0.99591573 | 0.99568176 | 0.99613444 | 0.99690247 | 0.99603271 |
| UACI | 0.30523305 | 0.33613886 | 0.33427272 | 0.33223301 | 0.33454857 |
| 4.2.03 (Mandrill) | | | | | |
| NPCR | 0.99594243 | 0.9961319 | 0.99613444 | 0.99624507 | 0.99611791 |
| UACI | 0.29923347 | 0.33447468 | 0.33469247 | 0.33471343 | 0.33483351 |
| 4.2.04 (Lena) | | | | | |
| NPCR | 0.99618403 | 0.99614716 | 0.99613698 | 0.99595133 | 0.99599457 |
| UACI | 0.30476267 | 0.33578865 | 0.33434783 | 0.33331343 | 0.3347094 |
| 4.2.05 (Airplane) | | | | | |
| NPCR | 0.9960556 | 0.99614716 | 0.9961141 | 0.99634043 | 0.99610774 |
| UACI | 0.3260339 | 0.33557365 | 0.33465736 | 0.33452906 | 0.33438018 |
| 4.2.06 (Sailboat) | | | | | |
| NPCR | 0.99609884 | 0.99596405 | 0.99591446 | 0.99611282 | 0.99599711 |
| UACI | 0.32224115 | 0.33489571 | 0.33431664 | 0.33437593 | 0.334841 |
| 4.2.07 (Peppers) | | | | | |
| NPCR | 0.99614461 | 0.99622726 | 0.99600983 | 0.99617513 | 0.99599838 |
| UACI | 0.32302629 | 0.33533835 | 0.33431664 | 0.3350033 | 0.33479383 |

Table 8 Key space analysis of the proposed and other implemented schemes

| Algorithm | key space |
|-----------|--|
| Proposed | 2^{384} (six keys of 64 bit k_1 to k_6) |
| Scheme 1 | 2^{384} |
| Scheme 2 | $2^{126} \cdot 2^{147}$ |
| Scheme 3 | 2^{192} |
| Scheme 4 | 2^{216} |

5.2.2 Correlation Analysis

The correlation analyses of the original and the encrypted images were performed by taking 10,000

random pixel pairs. These pixel pairs were formed by randomly choosing a pixel from the image and pairing it with an adjacent pixel. Table 4 provides the horizontal, vertical and diagonal correlation coefficients for the original images and the encrypted images obtained by proposed algorithm. It can be seen that the encrypted images provided very low values of correlation coefficients in all the three orientations. Hence, there is very less amount of correlation between the adjacent pixels of the encrypted images.

Table 5 shows a comparison of correlation coefficients of images encrypted by the proposed algorithm and other implemented algorithms. It can be seen that all the correlation coefficients obtained have very low

Table 9 PSRN values obtained using proposed and other schemes

| Scheme/image | Proposed | Scheme 1 | Scheme 2 | Scheme 3 | Scheme 4 |
|----------------------|------------|------------|------------|------------|------------|
| 4.1.01 (Girl) | 7.28321847 | 7.41202708 | 7.28231461 | 7.26577788 | 7.27730719 |
| 4.1.02 (Couple) | 6.23056818 | 6.26459333 | 6.26292911 | 6.22474618 | 6.26166187 |
| 4.1.05 (House) | 8.90073409 | 9.24371439 | 8.89381694 | 8.92478446 | 8.93184627 |
| 4.1.06 (Tree) | 8.17503201 | 8.09784295 | 8.18554431 | 8.14948409 | 8.16039096 |
| 4.1.07 (Jelly beans) | 8.5827447 | 8.46995623 | 8.59851664 | 8.59516241 | 8.58064266 |
| 4.2.03 (Mandrill) | 8.77873443 | 9.52130719 | 8.78169541 | 8.77197294 | 8.77626286 |
| 4.2.04 (Lena) | 8.60451584 | 9.2363062 | 8.62297419 | 8.61374073 | 8.61690171 |
| 4.2.05 (Airplane) | 7.97538627 | 8.00161516 | 7.98241891 | 7.98217713 | 7.98514501 |
| 4.2.06 (Sailboat) | 8.0833475 | 8.23376176 | 8.0828698 | 8.08725184 | 8.07817118 |
| 4.2.07 (Peppers) | 8.06937243 | 8.87192392 | 8.07329404 | 8.07442398 | 8.07371758 |

Table 10 Information entropy of the original and encrypted images

| Scheme/image | Original | Proposed | Scheme 1 | Scheme 2 | Scheme 3 | Scheme 4 |
|----------------------|-----------|-----------|-----------|-----------|-----------|-----------|
| 4.1.01 (Girl) | 6.8981395 | 7.9991809 | 7.9971123 | 7.999005 | 7.9990835 | 7.9991786 |
| 4.1.02 (Couple) | 6.2944978 | 7.9988397 | 7.9970078 | 7.9990167 | 7.9989950 | 7.9991148 |
| 4.1.05 (House) | 7.0686251 | 7.9991077 | 7.9971088 | 7.9990055 | 7.9992297 | 7.9990171 |
| 4.1.06 (Tree) | 7.5370888 | 7.9990680 | 7.9972382 | 7.9987571 | 7.9990427 | 7.9990547 |
| 4.1.07 (Jelly beans) | 6.5834853 | 7.9990668 | 7.9968646 | 7.9991204 | 7.9990223 | 7.998921 |
| 4.2.03 (Mandrill) | 7.7624360 | 7.9997860 | 7.9992402 | 7.9997579 | 7.9997659 | 7.9997663 |
| 4.2.04 (Lena) | 7.7501975 | 7.9997256 | 7.9994449 | 7.9997552 | 7.999788 | 7.9997670 |
| 4.2.05 (Airplane) | 6.6639080 | 7.9997223 | 7.9992539 | 7.9997463 | 7.9997783 | 7.9997781 |
| 4.2.06 (Sailboat) | 7.7621696 | 7.9997621 | 7.9992478 | 7.9997777 | 7.999768 | 7.9997867 |
| 4.2.07 (Peppers) | 7.6698255 | 7.9997709 | 7.9993271 | 7.9997781 | 7.9997684 | 7.9997728 |

values. Hence, the proposed scheme provides a correlation value similar to other schemes in literature.

Table 6 shows the correlation plots of the original and the encrypted images. It shows that the plots obtained from the original images have the elements focused around a central line and doesn't contains significant number of elements elsewhere in the plots. Hence, the plots have a highly non-uniform distribution for all the three orientations. On the other hand, the plots obtained by the encrypted images are uniformly distributed throughout the plot region. The nature was similarly shown by the proposed and other schemes. As there is no resemblance between the two correlation graphs, no information leakage is present and the scheme can withstand the statistical attacks efficiently.

5.3 Differential Attack Analysis

The effect of one-bit key change is observed on the basis of NPCR and UACI parameters. Table 7 shows the NPCR and UACI values obtained for the test images after implementing the proposed and other schemes. For the test images, an average value of 99.60334% is obtained for NPCR and an average value of 31.841% is obtained for UACI. This shows a high sensitivity of the scheme for even a single bit change in the input parameter. The use of row-wise and column-wise forward and backward diffusion along with RC4 plays a critical role for the same as it ensures a high scrambling production. Hence, the proposed scheme has a high resistance against the differential attacks.

5.4 Key Space Analysis

Table 8 shows the key space produced by the proposed and other schemes used in the study. As it can be seen, the proposed scheme provides a very large number of possible key solutions. So, it is impractical for an unauthorized user to retrieve the data by guessing the possible key. Hence, the proposed scheme is efficient enough to resist the brute force attacks.

5.5 Quantitative Analysis

5.5.1 Peak Signal-to-Noise Ratio (PSNR) Analysis

Table 9 shows the PSNR values for different images encrypted using the proposed and other implemented algorithms. It can be seen that the PSNR values obtained from the proposed scheme are similar to that of other schemes present in literature. An average PSNR value of 8.0684 is provided by the proposed scheme for the images used in the study.

5.5.2 Information Entropy Analysis

Table 10 shows the information entropy values of the original and encrypted images. It can be seen that, similar to the other schemes implemented, the entropy values provided by the proposed scheme are

approximating to the ideal value of 8. The average entropy value obtained was 7.9994.

5.6 Time Complexity Analysis

Figure 8 shows the time complexity possessed by the proposed and other schemes implemented for 256*256 images of the SIPI database. It can be seen that the proposed scheme is faster than other proposed schemes and hence reflects the efficiency to be used in practical cases.

6 Conclusion

The following inference can be drawn from the above results:

- The paper proposes an image encryption scheme based on intertwining map and RC4 stream cipher. The proposed scheme has been evaluated on various performance metrics. The results show that the proposed scheme provides highly scrambled encrypted images which have no visual resemblance with the original images. Whereas, the images obtained after decryption were visually alike as of the original images.
- The scheme proved to be strong against the statistical attacks. It is clear from the histograms,

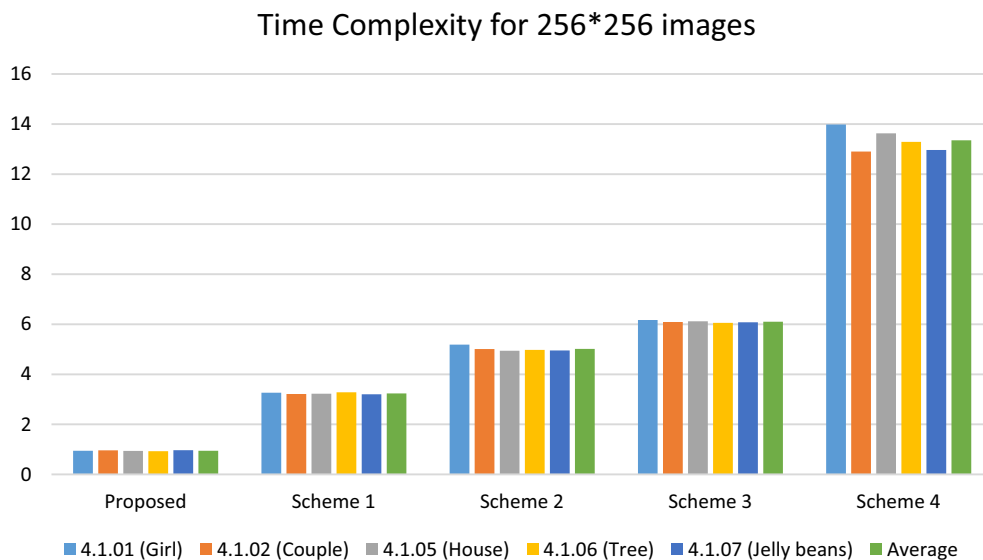


Fig. 8 Time complexity of the proposed and other schemes

which are uniformly distributed and have no resemblance with the histograms of the original images. Also, the encrypted images provide very low values of the correlation coefficients and have uniformly distributed correlation graphs for all three orientations with no resemblance to the original correlation graphs. All this ensures no information leakage and hence ensures security against the statistical attacks.

- The scheme also provides a high resistance against the differential attacks. The NPCR and UACI values obtained are close to the ideal values. An average NPCR of 99.60334% and an average UACI of 31.841% were obtained for a single bit change in the key used. Hence, even for a small change in the value will provide a drastic change in the encrypted/decrypted images and the data in the original image is secured.
- The proposed scheme has a key space of 2^{384} , which makes it resistive against the brute force attack. The scheme provides the PSNR values similar to other schemes, including the ones implemented. Also, the scheme provides an average value of 7.9994 for the information entropy, approximating the ideal value of 8. Finally, the scheme has a faster implementation as compared to the other schemes implemented because of its lower time complexity and hence provides it with an edge over the other schemes.

References

1. Anuradha, K., & Naik, P. P. S. (2015). Medical image cryptanalysis using histogram matching bitplane and adjoin mapping algorithms. *International Journal and Magazine of Engineering, Technology, Management and Research*, 2, 100–105. <http://www.ijmetmr.com/olctober2015/KolakaluriAnuradha-PPeddaSadhuNaik-13.pdf>.
2. Bansal, R., Gupta, S., & Sharma, G. (2016). An innovative image encryption scheme based on chaotic map and Vigenère scheme. *Multimedia Tools and Applications*. <https://doi.org/10.1007/s11042-016-3926-9>.
3. Baptista, M. S. (1998). Cryptography with chaos. *Physics Letters A*, 240, 50–54. http://cmup.fc.up.pt/cmup/murilo.baptista/baptista_PLA1998.pdf.
4. Barker, W. C., & Barker, E. (2012). Recommendation for the triple data encryption algorithm (TDEA) block cipher. National Institute of Standards and Technology. *Special Publication*, 800–867. <http://dx.doi.org/10.6028/NIST.SP.800-67r1>
5. Basu, S. (2011). International data encryption algorithm (IDEA)—A typical illustration. *Journal of Global Research in Computer Science*, 2, 116–118. <http://www.roij.com/open-access/international-data-encryption-algorithm-idea-a-typical-illustration-116-118.pdf>.
6. Biham, E., & Shamir, A. (1993). Differential cryptanalysis of the data encryption standard. Springer-Verlag. <http://www.cs.technion.ac.il/~biham/Reports/differential-cryptanalysis-of-the-data-encryption-standard-biham-shamir-authors-latex-version.pdf>.
7. Biryukov, A., & Kushilevitz, E. (1998). Improved cryptanalysis of RC5. In K. Nyberg (Ed.), *Advances in cryptology—EUROCRYPT'98. EUROCRYPT 1998. Lecture notes in computer science* (Vol. 1403). Berlin: Springer. <https://doi.org/10.1007/bfb0054119>.
8. Chang, C. C., Hwang, M. S., & Chen, T. S. (2001). A new encryption algorithm for image cryptosystems. *Journal of Systems and Software*, 58, 83–91. [https://doi.org/10.1016/S0164-1212\(01\)00029-2](https://doi.org/10.1016/S0164-1212(01)00029-2).
9. Chen, G., Mao, Y., & Chui, C. K. (2004). A symmetric image encryption based on 3D chaotic maps. *Chaos, Solitons & Fractals*, 21, 749–761. <https://doi.org/10.1016/j.chaos.2003.12.022>.
10. Dang, P. P., & Chau, P. M. (2000). Image encryption for secure internet multimedia applications. *IEEE Transactions on Consumer Electronics*, 46, 395–403. <https://doi.org/10.1109/30.883383>.
11. Ferguson, N., et al. (2001). Improved cryptanalysis of Rijndael. In G. Goos, J. Hartmanis, J. van Leeuwen, & B. Schneier (Eds.), *Fast software encryption. FSE 2000. Lecture notes in computer science* (Vol. 1978). Berlin: Springer. https://doi.org/10.1007/3-540-44706-7_15.
12. François, M., Grosge, T., Barchiesi, D., & Erra, R. (2012). A new image encryption scheme based on a chaotic function. *Signal Processing: Image Communication*, 27, 249–259. <https://doi.org/10.1016/j.image.2011.11.003>.
13. Gao, T., & Chen, Z. (2008). A new image encryption algorithm based on hyper-chaos. *Physics Letters A*, 372, 394–400. <https://doi.org/10.1016/j.physleta.2007.07.040>.
14. Hanchinamani, G., & Kulkarni, L. (2015). An efficient image encryption scheme based on a Peter De Jong chaotic map and a RC4 stream cipher. *3D. Research*, 6, 30. <https://doi.org/10.1007/s13319-015-0062-7>.
15. Jawad, L. M., & Sulong, G. (2015). Chaotic map-embedded Blowfish algorithm for security enhancement of colour image encryption. *Nonlinear Dynamics*, 81(4), 2079–2093. <https://doi.org/10.1007/s11071-015-2127-9>.
16. Jianquan, X., Chunhua, Y., Qing, X., Lijun, T. (2009). An encryption algorithm based on transformed logistic map. In *International conference on networks security, wireless communications and trusted computing* (pp. 111–114). <https://doi.org/10.1109/nswctc.2009.201>.
17. Karuvandan, V., Chellamuthu, S., & Periyasamy, S. (2016). Cryptanalysis of AES-128 and AES-256 block ciphers using Lorenz information measure. *The International Arab Journal of Information Technology*, 13, 306–312. <http://ccis2k.org/iajit/PDF/Vol.13,%20No.3/5373.pdf>.
18. Knudsen, L. R., & Meier, W. (2001). Correlations in RC6 with a reduced number of rounds. In G. Goos, J. Hartmanis, J. van Leeuwen, & B. Schneier (Eds.), *Fast software encryption. FSE 2000. Lecture notes in computer science*

- (Vol. 1978). Berlin: Springer. https://doi.org/10.1007/3-540-44706-7_7.
19. Kocarev, L., & Lian, S. (2011). Chaos based cryptography theory algorithms and applications. *Springer-Verlag*. <https://doi.org/10.1007/978-3-642-20542-2>.
 20. Kumar, S., & Sharma, R. K. (2016). Securing color images using two-square cipher associated with Arnold map. *Multimedia Tools and Applications*. <https://doi.org/10.1007/s11042-016-3504-1>.
 21. Kumara, M., Mishra, D. C., & Sharma, R. K. (2014). A first approach on an RGB image encryption. *Journal of Optics and Lasers in Engineering*, 52, 27–34.
 22. Li, C., Li, S., Asim, M., Nunez, J., Alvarez, G., & Chen, G. (2009). On the security defects of an image encryption scheme. *Image and Vision Computing*, 27, 1371–1381. <https://doi.org/10.1016/j.imavis.2008.12.008>.
 23. Liu, F., & Liu, Y. (2016). Research on image encryption based on Lorenz chaotic mapping in finite field. *International Journal of Hybrid Information Technology*, 9(10), 281–288. <https://doi.org/10.14257/ijhit.2016.9.10.28>.
 24. Liu, L., & Miao, S. (2016). A new image encryption algorithm based on logistic chaotic map with varying parameter. *SpringerPlus*, 5, 289. <https://doi.org/10.1186/s40064-016-1959-1>.
 25. Liu, Z., Guo, Q., Xu, L., Ahmad, M. A., & Liu, S. (2010). Double image encryption by using iterative random binary encoding in gyrator domains. *Optics Express*, 18, 12033–12043. <https://doi.org/10.1364/OE.18.012033>.
 26. Lucks, S. (1998). Attacking triple encryption. In S. Vaudenay (Ed.), *Fast software encryption. FSE 1998. Lecture notes in computer science* (Vol. 1372). Berlin: Springer. https://doi.org/10.1007/3-540-69710-1_16.
 27. Matthews, R. (1989). On the derivation of a “chaotic” encryption algorithm. *Cryptologia*, 14, 29–42. <https://doi.org/10.1080/0161-118991863745>.
 28. Mishra, D. C., & Sharma, R. K. (2016). An approach for security of color image data in coordinate, geometric, and frequency domains. *Journal Information Security Journal: A Global Perspective*, 25(4–6), 213–234.
 29. Mishra, D. C., Sharma, R. K., Suman, S., & Prasad, A. (2017). Multi-layer security of color image based on chaotic system combined with RP2DFRFT and Arnold transform. *Journal of Information Security and Applications*, 37, 65–90.
 30. Mousa, A., & Hamad, A. (2006). Evaluation of the RC4 algorithm for data encryption. *International Journal of Computer Science & Applications*, 3, 44–56. <http://www.tmrfindia.org/ijcsa/v3i24.pdf>.
 31. Pareek, N. K., Patidar, V., & Sud, K. K. (2003). Discrete chaotic cryptography using external key. *Physics Letters A*, 309, 75–82. [https://doi.org/10.1016/S0375-9601\(03\)00122-1](https://doi.org/10.1016/S0375-9601(03)00122-1).
 32. Pareek, N. K., Patidar, V., & Sud, K. K. (2006). Image encryption using chaotic logistic map. *Image and Vision Computing*, 24, 926–934. <https://doi.org/10.1016/j.imavis.2006.02.021>.
 33. Rehman, A. U., Khan, J. S., Ahmad, J., & Hwang, S. O. (2016). A new image encryption scheme based on dynamic S-boxes and chaotic maps. *3D Research*, 7(7), 7. <https://doi.org/10.1007/s13319-016-0084-9>.
 34. Rivest, R. L. (1995). RC5 encryption algorithm. *Dr Dobbs's Journal*, 226, 146–148. <http://www.drdoobs.com/security/the-rc5-encryption-algorithm/184409480>.
 35. Rivest, R. L., & Schuldt, J. C. N. (2014). Spritz—a spongy RC4-like stream cipher and hash function. In *Proceedings of the Charles River Crypto Day*, Palo Alto, CA, USA. <https://people.csail.mit.edu/rivest/pubs/RS14.pdf>.
 36. Sam, I. S., Devaraj, P., & Bhuvaneshwaran, R. S. (2012). A novel image cipher based on mixed transformed logistic maps. *Multimedia Tools and Applications*, 56, 315–330. <https://doi.org/10.1007/s11042-010-0652-6>.
 37. Sam, I. S., Devaraj, P., & Bhuvaneshwaran, R. S. (2012). An intertwining chaotic maps based image encryption scheme. *Nonlinear Dynamics*, 69, 1995–2007. <https://doi.org/10.1007/s11071-012-0402-6>.
 38. Scharinger, J. (1998). Fast encryption of image data using chaotic Kolmogorov flows. *Journal of Electronic Engineering*, 7, 318–325. <https://doi.org/10.1117/1.482647>.
 39. Schneier, B. (1994). The Blowfish encryption algorithm. *Dr. Dobbs's Journal*, 19, 38–40. <http://www.drdoobs.com/security/the-blowfish-encryption-algorithm/184409216>.
 40. Shimoyama, T., Takenaka, M., & Koshihara, T. (2002). Multiple linear cryptanalysis of a reduced round RC6. In J. Daemen & V. Rijmen (Eds.), *Fast software encryption. FSE 2002. Lecture notes in computer science* (Vol. 2365). Berlin: Springer. https://doi.org/10.1007/3-540-45661-9_6.
 41. Solak, E., Çokal, C., Yildiz, O. T., & Biyikoğlu, T. (2010). Cryptanalysis of Fridrich's chaotic image encryption. *International Journal of Bifurcation and Chaos*, 10, 1405–1413. <https://doi.org/10.1142/S0218127410026563>.
 42. Tayal, N., Bansal, R., Gupta, S., & Dhall, S. (2016). Analysis of various cryptography techniques: A survey. *International Journal of Security and Its Applications*, 10, 59–92. <https://doi.org/10.14257/ijisia.2016.10.8.07>.
 43. The next tier, 8 security predictions for 2017, <https://www.trendmicro.ae/vinfo/aesecurity/research-and-analysis/predictions/2017>.
 44. Ye, G. (2010). Image scrambling encryption algorithm of pixel bit based on chaos map. *Pattern Recognition Letters*, 31, 347–354. <https://doi.org/10.1016/j.patrec.2009.11.008>.
 45. Yoon, J. W., & Kim, H. (2010). An image encryption scheme with a pseudorandom permutation based on chaotic maps. *Communications in Nonlinear Science and Numerical Simulation*, 15, 3998–4006. <https://doi.org/10.1016/j.cnsns.2010.01.041>.
 46. Younes, M. A. B., & Jantan, A. (2008). Image encryption using block-based transformation algorithm. *International Journal of Computer Science*, 35, 407–415. http://www.iaeng.org/IJCS/issues_v35/issue_1/IJCS_35_1_03.pdf.
 47. Yu, X. Y., Zhang, J., Ren, H. E., Xu, G. S., & Luo, X. Y. (2006). Chaotic image scrambling algorithm based on S-DES. *Journal of Physics: Conference Series*, 48, 349–353. <https://doi.org/10.1088/1742-6596/48/1/065>.
 48. Yun-peng, Z., Zheng-jun, Z., Wei, L., Xuan, N., Shui-ping, C., & Wei-di, D. (2009). Digital image encryption algorithm based on chaos and improved DES. In *IEEE international conference on systems, man, and cybernetics* (pp. 474–479). <https://doi.org/10.1109/icsmc.2009.5346839>.
 49. Zeghid, M., Machhout, M., Khriji, L., Baganne, A., & Tourki, R. (2007). A modified AES based algorithm for image encryption. *International Journal of Computer*,

- Electrical, Automation, Control and Information Engineering*, 1, 745–750. <http://scholar.waset.org/1307-6892/7580>.
50. Zhang, G., & Liu, Q. (2011). A novel image encryption method based on total shuffling scheme. *Optics Communications*, 284, 2775–2780. <https://doi.org/10.1016/j.optcom.2011.02.039>.
51. Zhang, Y., Xiao, D., Wen, W., et al. (2014). Cryptanalysis of image scrambling based on chaotic sequences and Vigenère cipher. *Nonlinear Dynamics*, 78, 235. <https://doi.org/10.1007/s11071-014-1435-9>.