Multimedia Tools and Applications
https://doi.org/10.1007/s11042-023-15958-x

Quad-color image encryption based on Chaos and Fibonacci Q-matrix

Shaima Safa aldin Baha Aldin[1] · Mahmut Aykaç[2] · Noor Baha Aldin[3]

Received: 13 April 2022 / Revised: 20 March 2023 / Accepted: 29 May 2023

Abstract
The Information technology requires the transmission of daily-life images that may reach to millions or even more. Thus, securing them becomes an urgent matter using the encryption technique. Where, a secret key is used for converting the original image into a noisy one and restoring it back using the same key. Confusion and Diffusion are the wildly used steps in such a technique. Therefore, a new algorithm is presented in this work that uses a fusion, segmentation, random assembling, hyperchaotic and Fibonacci Q-matrix (FQ-matrix). A novel fusion method is designed for fusing four color images into four different sequences according to their contained information. Then the resulted four images are each divided into four segments to be assembled randomly into one image using a random-key; which confused later using a six-dimensional hyperchaotic system and diffused using the FQ-matrix. The performance and robustness of the proposed algorithm have been computed based on different tests; where it proved its powerful capability in securing the transmitted images.

Keywords Quad-color fusion · Segmentation · Random assembling · FQ-matrix · Hyperchaotic system

1 Introduction

In the recent era, a lot of data were spread throughout the networks; due to the fast development of information technology and popularity of internet [13]. Digital image is the most effective type of information that is exchanged daily; where it may expose to an illegal

✉ Shaima Safa aldin Baha Aldin
shaima.safaaldin@nahrainuniv.edu.iq

Mahmut Aykaç
maykac@gantep.edu.tr

Noor Baha Aldin
noor.aldin@hku.edu.tr

[1] Continuing Education Center, Al-Nahrain University, Baghdad, Iraq

[2] Department of Electrical and Electronics Engineering, Gaziantep University, Gaziantep, Turkey

[3] Department of Electrical and Electronics Engineering, Hasan Kalyoncu University, Gaziantep, Turkey

Published online: 10 June 2023

interception, distortion during its journey throughout the network [11]. Therefore, securing them becomes an attractive matter for many researchers [14].

In order to preserve the image secrecy, many security methods were presented to prevent unpermitted users from accessing the image's content [23]. There are three types of Image security; which are watermarking, data hiding, and encryption. Hiding a private message unnoticeably inside a cover image represents the data hiding [2]. Inserting a noticeable data inside another image defines the watermarking technique [8, 16]. Converting an image to a noisy one using a key that also returns it back using it is the technique of encryption [3, 7].

Other researchers use the fusion technique not for securing the data but for automatic rendering the aging faces; which includes the prototype and physical approaches [20]. The first approach is simpler than the second one that averages the people faces to learn the features in the same age group like bi-level dictionary learning [21], while the second approach depends on the alteration of physical factors over time such as mouth, hair, wrinkles. There are so many techniques that dealt with image encryption such as quantum, chaos, DNA, Fibonacci Q-matrix and compressive sensing [25, 27, 31, 34, 35]. In general, encrypting an image includes two steps diffusion and permutation. The first step changes the pixels' values; while the second one exchanges the pixels' locations for breaking the tough correlation between them and hiding the significant and effective details inside the plain images [27]. The Chaotic-related algorithms deals with intrinsic characteristics like random behavior, sensitivity, and non-periodicity for managing the parameters and initial conditions to reach a successful encryption [12, 15]. Chai et al. [4] has described such a technique and classified it into two key types; which are the 1D chaotic maps for low-dimensional system and hyperchaotic maps for a high-dimensional system. Despite the simplicity of low-dimensional chaotic maps, they have a small key-space; which in turn accomplishes a low-security level; while Hua et al. [9] depends on 2D chaotic maps for image encryption. Hyperchaotic methods are utilized as a substitute for the low-dimensional chaotic systems and their limitations; where it has surpassed the low dimension systems regarding the initial conditions, unpredictability, randomness, and nonlinearity; because of its large key-space [22]. A new technique was proposed by Khan, M. et al. [10] based on a hybrid method that mixed chaotic systems with Brownian motion. Encrypting Images using Fractional Fourier Transform based on the Chen's hyperchaotic is what Yu et al. [30] proposed. A novel 3D hyperchaotic map was proposed by Yan W [26] for the encryption process; which has combined with random dynamic encryption for gray scale images in [19].

In [6] a novel algorithm of single-color image encryption was proposed that depends on DNA mutation and fractional order laser chaotic system. The researchers in [18] suggested a bit-plane Decay and Genetic Operators based on chaos system for a gray scale image encryption. A single-color encryption system was suggested by the authors in [17] based on hyper-chaotic system and cellular automata. A multi-image encryption (MIE) was suggested by the authors in [32] using a chaotic and DNA coding. However, at the pixel-level, their encoding is not dynamic. In [24] a novel triple-image encryption was introduced with a hiding technique that combines a compressive sensing (CS) and 2D chaotic system with a 3D discrete cosine transform (DCT). However, it works with Grey-level images, triple and two-dimensional chaotic system. Although a secure and fast MIE algorithm was introduced in [5] based on the index of image-matrix and DNA sequence, the perception approach does not quite differ from encrypting a single-image.

In [28] a nonlinear optical encryption was proposed for multi-image based on improved pixel adaptive diffusion and quaternion discrete fractional Hartley transform; whose key is nearly associated with the original picture. Although the authors in [1] have improved the security by suggesting a combination of elliptic curve cryptosystem (ECC) and chaos system for a three-dimensional (3D) scrambling, the result has a specific blocking effect. An MIE algorithm was proposed in [33] based on a dynamic DNA coding and 3D scrambling model, but depends on a 2D Chaotic Henon–Sine map (2D-HSM). The authors in [29] designed a new improved 3D continuous chaotic system; but for a double Grey-level image encryption.

However, the related works have some limitations and weakness. The initial conditions in some of them for chaotic map were not related to the input image. Also, its key-space is low and less-sensitive to them. Some of the works could not restore the plain images after data-cut and noise attacks or resisting the statistical attacks in which their histograms are non-uniform. Others improved the security robustness but with a scarification in system complexity or speed. Although, some recent works have dealt with multiple images, they did not deal with bit-level fusion of four colorful images based on the amount of information carried inside the pixels and their speed were low compared with our proposed scheme. These weaknesses, limitations, slow speed in processing and time complexity were the motivation for proposing a new encryption algorithm that utilizes a four colored randomly fused images that confused using a 6D-HyperC system and diffused using a FQ-matrix within 2-rounds. The proposed algorithm reduces the time and space complexity for real time applications of multiple RGB images with improving the system robustness. Since, the number of generated keys has minimized when encrypting multiple images at the same time; which in turn reduces the processing overhead and computation time. Also, such a technique helps in improving the resistance capability to different attacks. The contribution of our works is demonstrated as follows: First, a new four-color fusion technique is introduced that depends on the pixels' information. Second, the resulted four fused images are segmented each into four sub-blocks that are combined altogether using a random key. Third, scrambling the pixels-positions in the resulted image is performed based on the 6D HyperC system; where three randomly selected sequences from this 6D HyperC system are used for permitting the input image. Last, diffusing the confused sub-blocks of images is implemented using the FQ-matrix. The organization of this paper is as follows. Section 2 introduces the preliminary works. Section 3 introduces the proposed scheme. Section 4 presents the results of experimental performance tests. Section 5 presents the conclusions of this work.

2 Preliminary works

2.1 Six-dimensional Hyperchaotic method

The non-linearity of Hyperchaotic functions leads to unpredictable responses as shown in the mathematical analysis of its dynamic behavior. Such a thing leads to a more complexity in system compared with the low-dimension chaotic functions. In addition to that, it contains at least two positive Lyapunov exponents regarding the one in low-dimension methods. The mathematical computation is shown as follows [6]:

$$\begin{aligned} \dot{t_1} &= i(t2 - t1) + t4 - t5 - t6 \\ \dot{t_2} &= jt1 - t2 - t1\ t3 \\ \dot{t_3} &= -kt3 + t1t2 \\ \dot{t_4} &= lt4 - t2t3 \\ \dot{t_5} &= mt6 + t3t2 \\ \dot{t_6} &= nt1 \end{aligned} \tag{1}$$

Where, the state variables are referred by $\left(\dot{t_1} \ldots \dot{t_6}\right)$, the constants' values are (i, j, k, l, m, n) = (10, 83, 28, −1, 8, 3) respectively; which are selected precisely for ensuring the system has 2 positive Lyapunov exponents (Lyap-Expos).

2.2 FQ-matrix

The mathematical representation of Fibonacci series (FQ_i) is presented below [35]:

$$FQ_i = FQ_{i-1} + FQ_{i-2}, i > 1 \tag{2}$$

Where, FQ_1 and $FQ_2 = 1$.

The ith power of FQ-matrix is G^i. The determinant $Det(G^i)$ and inverse G^{-i} of FQ-matrix are calculated as follows:

$$G = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \tag{3}$$

$$G^i = \begin{bmatrix} FQ_{i+1} & FQ_i \\ FQ_i & FQ_{i-1} \end{bmatrix} \tag{4}$$

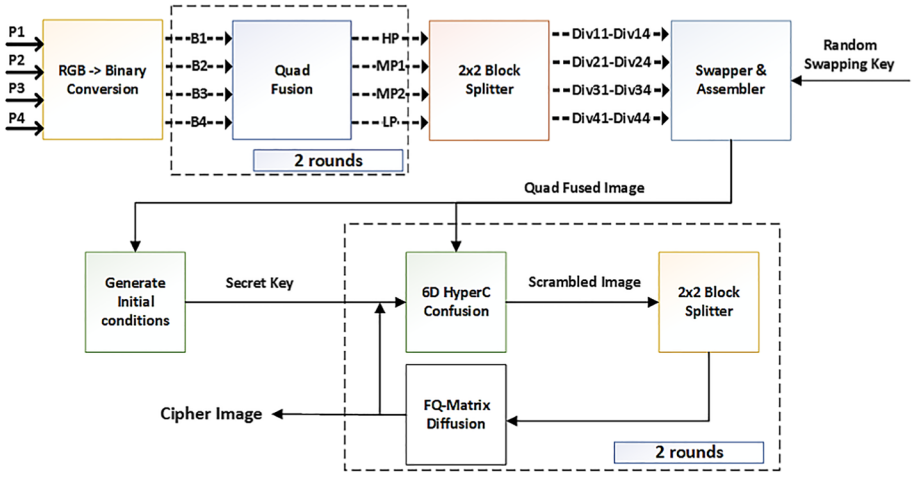


Fig. 1 Encryption Model

$$Det\left(G^{i}\right) = FQ_{i+1}FQ_{i-1} - FQ^{2}{}_{i} = (-1)^{i} \tag{5}$$

$$G^{-i} = \begin{bmatrix} FQ_{i-1} & -FQ_{i} \\ -FQ_{i} & FQ_{i+1} \end{bmatrix} \tag{6}$$

3 The proposed scheme

The new algorithm of image encryption has used a quad fusion, segmentation, random concatenation, 6D hyperchaotic (6D-hyperC) system and FQ-matrix. In the fusion step, the four-color input images are divided according to their pixel's information and reassembled into four new fused ones. The utilization of a 6D-hyperC system has improved the encryption performance; due to the complexity of its high-dynamic operation and the two positive Lyap-Expos. The simplicity and fast processing of FQ-matrix has simplified the diffusion of scrambled image. The proposed scheme is demonstrated in Fig. 1.

3.1 Quad fusion

Step 1. Select the four colorful plaintext images Im1, Im2, Im3, and Im4 of size $L \times W$.

Step 2. The input image Im1, Im2, Im3 and Im4 are converted to 8-bit binary form as shown in Eq. (7); which in turn used for constructing the 32 Boolean matrices (B_{11}, B_{21},B_{31}, B_{41}, B_{12}, B_{22}, B_{32}, B_{42}, B_{13}, B_{23}, B_{33}, B_{43}, B_{14}, B_{24}, B_{34}, B_{44}, B_{15}, B_{25}, B_{35}, B_{45}, B_{16}, B_{26}, B_{36}, B_{46}, B_{17}, B_{28}, B_{37}, B_{47}, B_{18}, B_{28}, B_{38}, B_{48}), as shown in the following equations:

$$\begin{cases} \begin{cases} \mathrm{Img}_1 = dec2base\left(Im_1, 2, 8\right) \\ \mathrm{Ims}_1 = str2num\left(\mathrm{Img}_1(:)\right) \\ rs_1 = reshape\left(\mathrm{Ims}_1, [], 8\right) \end{cases} \\ \begin{cases} \mathrm{Img}_2 = dec2base\left(Im_2, 2, 8\right) \\ \mathrm{Ims}_2 = str2num\left(\mathrm{Img}_2(:)\right) \\ rs_2 = reshape\left(\mathrm{Ims}_2, [], 8\right) \end{cases} \\ \begin{cases} \mathrm{Img}_3 = dec2base\left(Im_3, 2, 8\right) \\ \mathrm{Ims}_3 = str2num\left(\mathrm{Img}_3(:)\right) \\ rs_3 = reshape\left(\mathrm{Ims}_3, [], 8\right) \end{cases} \\ \begin{cases} \mathrm{Img}_4 = dec2base\left(Im_4, 2, 8\right) \\ \mathrm{Ims}_4 = str2num\left(\mathrm{Img}_4(:)\right) \\ rs_4 = reshape\left(\mathrm{Ims}_4, [], 8\right) \end{cases} \end{cases} \tag{7}$$

$$\begin{cases} B_{11} = rs_1(:,1), B_{21} = rs_2(:,1), B_{31} = rs_3(:,1), B_{41} = rs_4(:,1) \\ B_{12} = rs_1(:,2), B_{22} = rs_2(:,2), B_{32} = rs_3(:,2), B_{42} = rs_4(:,2) \\ B_{13} = rs_1(:,3), B_{23} = rs_2(:,3), B_{33} = rs_3(:,3), B_{43} = rs_4(:,3) \\ B_{14} = rs_1(:,4), B_{24} = rs_2(:,4), B_{34} = rs_3(:,4), B_{44} = rs_4(:,4) \\ B_{15} = rs_1(:,5), B_{25} = rs_2(:,5), B_{35} = rs_3(:,5), B_{45} = rs_4(:,5) \\ B_{16} = rs_1(:,6), B_{26} = rs_2(:,6), B_{36} = rs_3(:,6), B_{46} = rs_4(:,6) \\ B_{17} = rs_1(:,7), B_{28} = rs_2(:,7), B_{37} = rs_3(:,7), B_{47} = rs_4(:,7) \\ B_{18} = rs_1(:,8), B_{28} = rs_2(:,8), B_{38} = rs_3(:,8), B_{48} = rs_4(:,8) \end{cases} \tag{8}$$

Where, the conversion from decimal to a binary is defined *as dec2base*(), and the conversion from ASCII string into numeric is defined as *str2num*(). The 8-bit binary form of plain images ($Im_1 \ldots Im_4$) are represented by ($rs_1 \ldots rs_4$).

Step 3. The resulted matrices are cross-combined to produce low-bit *LP*, high-bit *HP*, medium-bit MP1 and MP2 Boolean images, as demonstrated in Fig. 2.

$$\begin{cases} HP = \left[B_{11}, B_{21}, B_{31}, B_{41}, B_{12}, B_{22}, B_{32}, B_{42}\right] \\ MP_1 = \left[B_{13}, B_{23}, B_{33}, B_{43}, B_{14}, B_{24}, B_{34}, B_{44}\right] \\ MP_2 = \left[B_{15}, B_{25}, B_{35}, B_{45}, B_{16}, B_{26}, B_{36}, B_{46}\right] \\ LP = \left[B_{17}, B_{28}, B_{37}, B_{47}, B_{18}, B_{28}, B_{38}, B_{48}\right] \end{cases} \tag{9}$$

We choose" lena"," peppers", "baboon" and "airplane" images as inputs to the fusion process, as demonstrated in Fig. 3.

3.2 Segmentation and random assembling

After fusing the four images (2-rounds), the resulted images are segmented each into four equal divisions 2×2. $\{Div_{11}, Div_{12}, Div_{13}, Div_{14}\}$ are the divisions belong to the first fusion image HP. $\{Div_{21}, Div_{22}, Div_{23}, Div_{24}\}$ are the divisions belong to the second fusion image MP1. $\{Div_{31}, Div_{32}, Div_{33}, Div_{34}\}$ are the divisions belong to the third fusion image MP2. $\{Div_{41}, Div_{42}, Div_{43}, Div_{44}\}$ are the divisions belong to the fourth fusion image LP.

The resulted divisions are assembled altogether randomly into one single image using a random key. Where the random key defines the positions of each division inside the assembled image. The resulted 16 divisions are first arranged into one single combination array (Com) as shown in Eq. (10) and re-organized using the random key:

$$\begin{aligned} Com = \Big[&Div_{11}, Div_{12}, Div_{13}, Div_{14}, Div_{21}, Div_{22}, Div_{23}, Div_{24}, \\ &Div_{31}, Div_{32}, Div_{33}, Div_{34}, Div_{41}, Div_{42}, Div_{43}, Div_{44}\Big] \end{aligned} \tag{10}$$

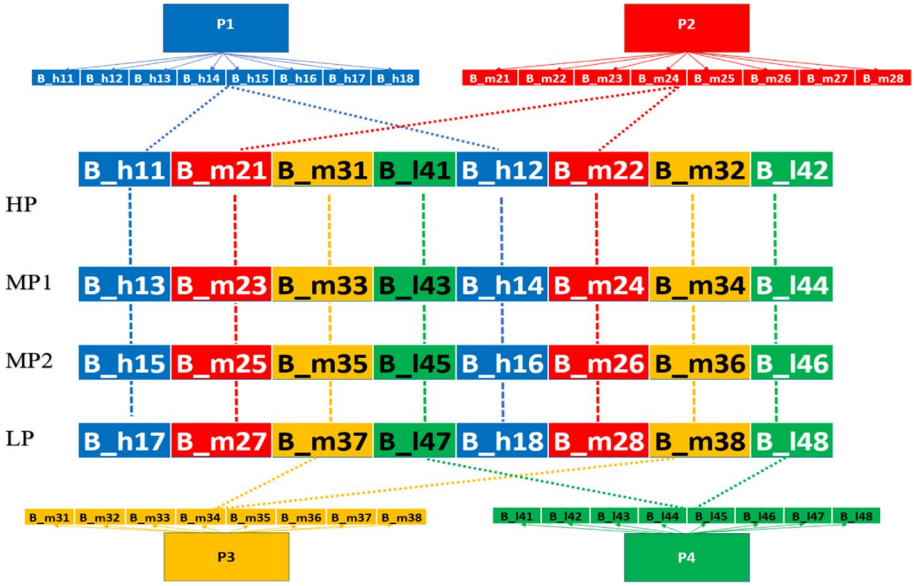


Fig. 2 Fusion images construction process

Then the re-arranged vector is divided into 4 sequences (C_1, C_2, C_3, and C_4) concatenated into two sub-blocks 2×2 dimensions to generate the final assembled image (*Fin*) as shown in the equation below:

$$Fin = \left[\{C_1, C_2\}, \{C_3, C_4\}\right] \tag{11}$$

Figure 4 demonstrates the result of segmentation and random assembling. For example, if the random key = RK = [{15, 2, 8, 10}, {1,6,16,3}, {9,14,4,12}, {5,11,7,13}], then the rearrangement of the input divisions according to their positions in C array will be as follows:

$$\begin{cases} C_1 = \left[\{Dv_{43}, Dv_{12}\}, \{Dv_{24}, Dv_{32}\}\right] \\ C_2 = \left[\{Dv_{11}, Dv_{22}\}, \{Dv_{44}, Dv_{13}\}\right] \\ C_3 = \left[\{Dv_{31}, Dv_{42}\}, \{Dv_{14}, Dv_{34}\}\right] \\ C_4 = \left[\{Dv_{21}, Dv_{33}\}, \{Dv_{23}, Dv_{41}\}\right] \end{cases} \tag{12}$$

3.3 Chaos and Fibonacci Q-matrix

This stage modifies the pixels' ordering and values in stages known as confusion and diffusion stages. The confusion step applies a 6D-HyperC system that computes the initial condition based on the final assembled image; which is converted to a vector V and iterates from (0.32853 x LW) to (LW/3) for generating a new vector of three chosen sequences (k1, k3, and k5), as shown in the equations below. The result is then sorted for confusing

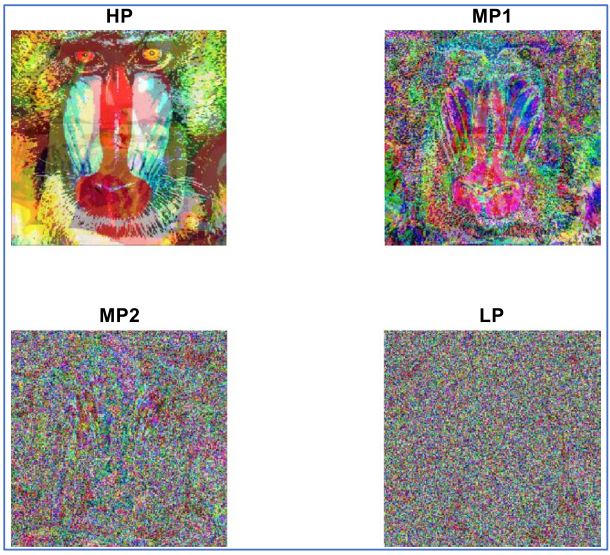


Fig. 3 Quad-Color Fusion Images

the quad-fused image and segmented into 2×2 blocks to be diffused using a FQ-matrix. The confusion and diffusion procedures are repeated twice to construct the cipher-image as shown in Fig. 5.

$$inital\ key\ (k1) = \frac{\sum_{j=1}^{LxW} V_j + (LW)}{2^{23+}(LW)} \tag{13}$$

$$k_i = \mathrm{mod}\left(k_{i-1} \times 10^6, 1\right), i = (2 \ldots 6) \tag{14}$$

Where, LW represent the product of L by W (length and width of an image), k_i defines the key sequence and V defines the image's vector.

3.4 Decryption process

The decryption procedures are demonstrated in Fig. 1b; where the final assembled four-color images can be restored using the encrypted image throughout the following steps that are twice-repeated:

Step 1. A diffusion formula FQ^{-10} is applied to the encrypted image (C) after segmenting it into blocks (2×2), as shown below:

$$\begin{bmatrix} SC'_{a,b} & SC'_{a,b+1} \\ SC'_{a+1,b} & SC'_{a+1,b+1} \end{bmatrix} = \begin{bmatrix} C_{a,b} & C_{a,b+1} \\ C_{a+1,b} & C_{a+1,b+1} \end{bmatrix} \begin{bmatrix} 34 & -55 \\ -55 & 89 \end{bmatrix} \bmod 256 \tag{15}$$

Where, a and b = (1:3: … L) and (1: 3…: W) respectively.
A matrix to vector conversion V is applied then to the above scrambled image SC'.

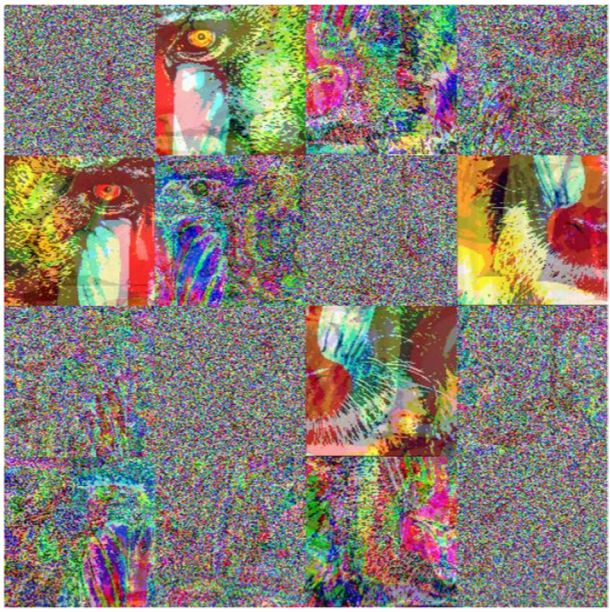


Fig. 4 Randomized Quad Fusion Image (512×512)

Step 2. The encryption key 'k' is utilized for restoring each pixel to its original location as follows:

$$X(k_i) = V_i, i = 1 : \ldots L * W \tag{16}$$

Step 3. A vector to matrix conversion is applied then to X for obtaining the decrypted quad fusion image.

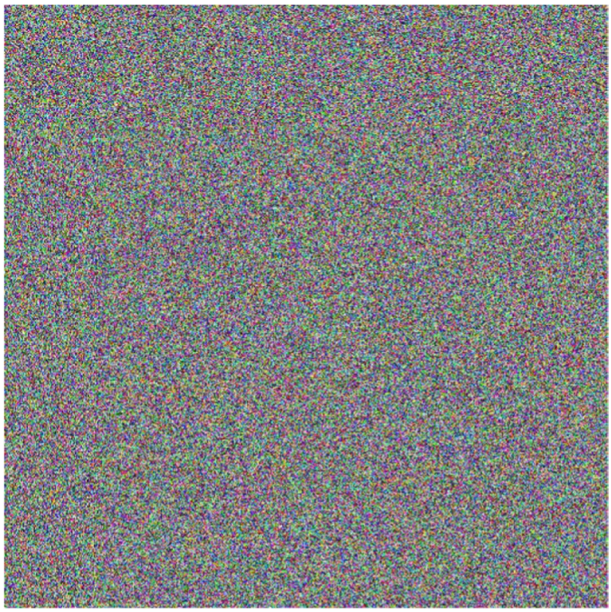
Step 4. The resulted image is then segmented into 4 sequences of 2×2 sub-blocks; which in turn repositioned into their real locations based on the randomly generated key during the encryption process.

Step 5. The re-ordered sequences are then applied to the quad diffusion process to restore the four plain color images.

4 Experimental tests and performance analyses

The reliability of encryption techniques depends mainly on the Security-level. Since its weakness opens the way for illegal person to review the secret data and trying to retrieve the plain-text using several techniques. Thus, to check the robustness of our proposed scheme, it has tested upon various analyses like histogram, peak signal to noise ratio (PSNR), correlation coefficient, entropy, differential attacks, key space, key sensitivity, complexity, speed, and NIST statistical analyses. All the performed testes have been executed using MATLAB (R2021a) with a Laptop computer of Intel(R) Core (TM) i7-8565U @ 1.80GHz GHz CPU and 16 GB RAM. The plaintext images are a dataset from Signal Image and Processing Institute (SIPI) of 256×256 pixels, Color (24 bits/pixel) ("peppers", "baboon", "lena", "airplane").

Fig. 5 Cipher Image



4.1 Histogram analysis

The pixels' occurrences are what the histogram analyze. Thus, it is helpful in evaluating the encryption performance. The cipher-image's histogram has reached to unity; which means the image's information are hidden well. A standard Baboon image of size $256 \times 256 \times 3$ was selected that separated into 3 channels (Red, Green, Blue). The plain-image's histograms show sharp peaks, and the result of random quad fusion process shows fewer sharp peaks that approaches uniformity, while it is distributed equally with no sharp-peaks in the cipher images throughout the entire region as demonstrated in Fig. 6a, b, c respectively.

4.2 Entropy

It is a measurable quantity that determine the randomness in the image's content as shown below:

$$Ent(\mathrm{Im}) = \sum_{j=1}^{2^t-1} \Pr(\mathrm{Im}_j) \log_2 \frac{1}{\Pr(Im_j)} \tag{17}$$

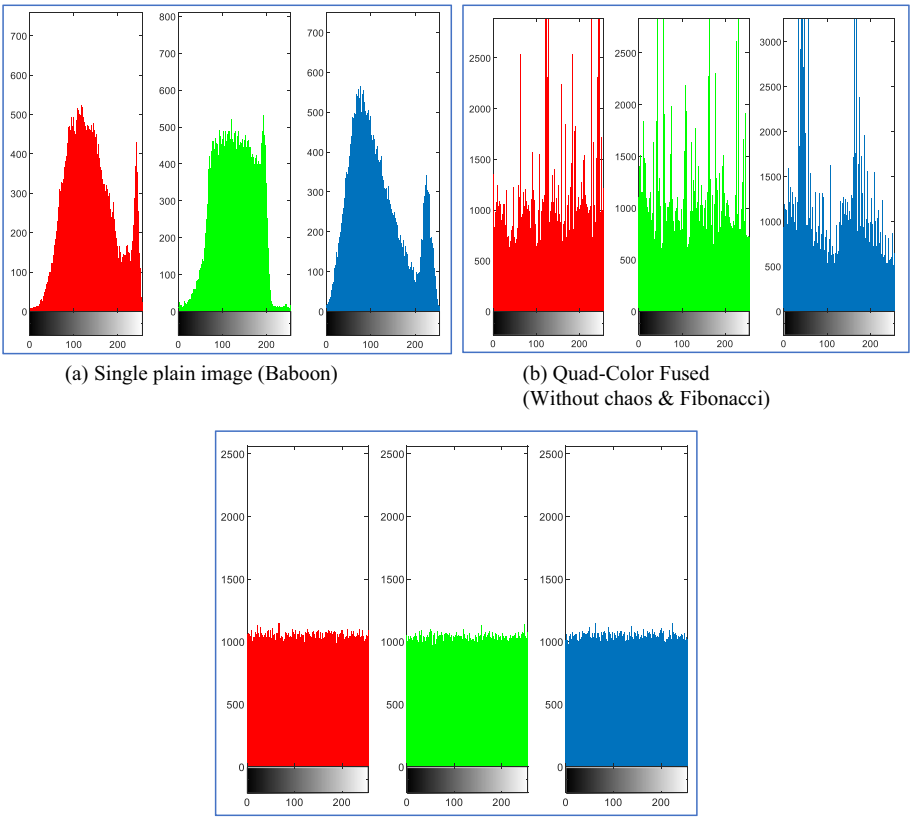
Where, t refers to the total pixels, $Pr(Im_j)$ is the occurrence probability of Im_j. The ideal entropy of an 8-bit image is 8 and the obtained entropy using our proposed scheme (quad) is 7.9998 as shown in Table 1. That means the suggested scheme has significantly increased the random distribution of pixel within the input-image; which in turn results in a more security-level and difficulty for the attackers to crack the algorithm.

4.3 Differential attack

This type of attacks detects the change of ciphertext image from plaintext image. Since a good diffusion characteristic means any minor-changes in the plaintext-image produces a completely different cipher-image; which in turn leads to a robust encryption scheme. Therefore, to compute this change, a two widely used analyses were utilized known as UACI and NPCR (Unified Average Change Intensity and Number of Pixels Changing Rate) that calculated as follows:

$$NPCR = \frac{1}{L \times W} \sum_{a=1}^{L} \sum_{b=1}^{W} DF(a,b) \times 100\% \tag{18}$$

$$UACI = \frac{1}{L \times W} \sum_{a=1}^{L} \sum_{b=1}^{W} \frac{\left| Cp_2(a,b) - Cp_1(a,b) \right|}{255} \times 100\% \tag{19}$$



(a) Single plain image (Baboon)

(b) Quad-Color Fused (Without chaos & Fibonacci)

(c) Quad-Color Cipher Image

Fig. 6 Histogram of Three cases for three channels. **a** single plain image (Baboon) **b** Quad-Color Fused Only **c** Quad-Color Cipher Image

Table 1 Average Entropy

Input	Value
Plain Images	7.2000
Proposed Cipher	7.9998
[24]	7.9967
[28]	7.9959
[1]	7.9996
[33]	7.9994
[29]	7.9993

$$DF(a,b) = \begin{cases} 0, Cp_2(a,b) = Cp_1(a,b) \\ 1, Cp_2(a,b) \neq Cp_1(a,b) \end{cases} \tag{20}$$

The symbols Cp_1 and Cp_2 refer to the chipper image encrypted without and with one-pixel modified within an input image respectively. Table 2 demonstrates the obtained results regarding other algorithms. Referring to the table, the value of NPCR and UACI are greater than other methods that reached to (99.62%) and 33.51% respectively; which shows that our scheme has the capability of resisting differential attack.

4.4 Noise and data cut (cropping) attacks

Transferring images throughout a network make them susceptible to noise or cropping. Thus, a good encryption method must have the capability of defeating these attacks. In order to determine that, a PSNR was implemented for determining the performance of encryption method; in which it measures the corruption in pixels' values inside the deciphered image compared with the plain image as shown below:

$$PSNR = 10 \ \log_{10} \frac{(Int-1)^2}{MSE} \tag{21}$$

$$MSE = \frac{1}{L \times W} \sum_{a=1}^{L} \sum_{b=1}^{W} \left| I_{original}(a,b) - I_{degraded} \right|^2 \tag{22}$$

Where, MSE is the mean squared error, Int represents the number of maximum possible intensity levels within an image and, $I_{degraded}$ and $I_{original}$define the matrix of

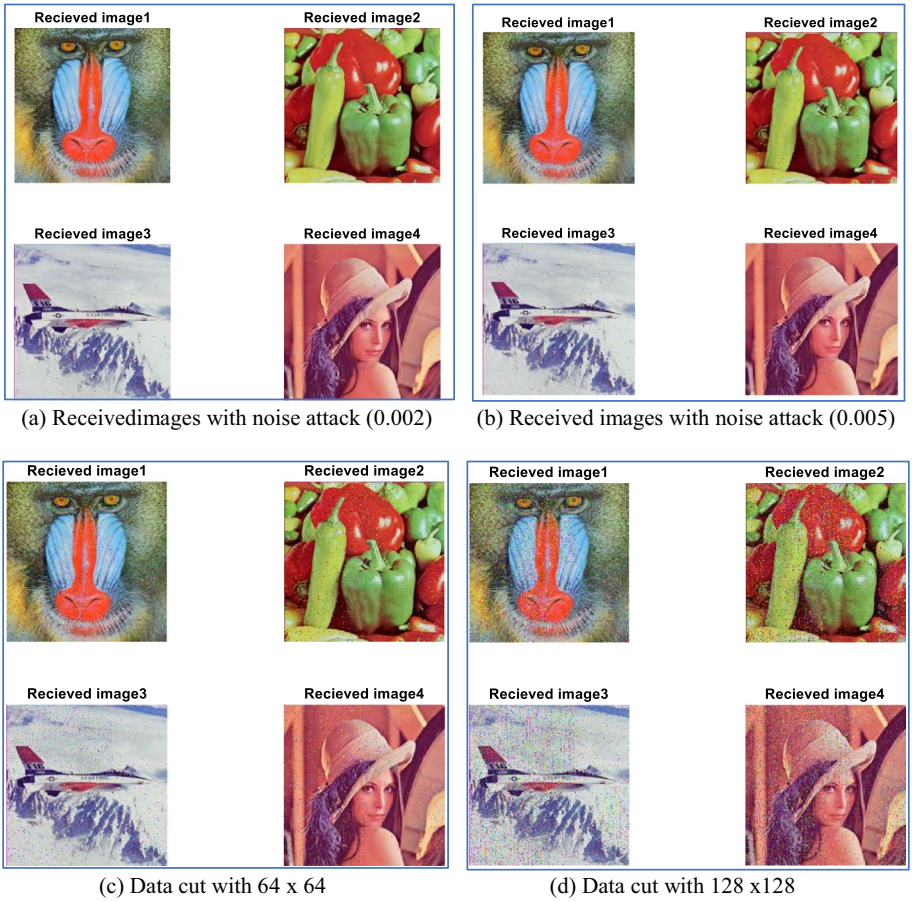
Table 2 Average NPCR & UACI

Method	NPCR	UACI
Proposed	99.6200	33.5104
[32]	99.6155	33.5558
[24]	99.1030	–
[5]	99.1841	33.5284
[28]	99.6534	33.6772
[1]	99.9100	33.5016
[33]	99.6060	33.5126
[29]	99.6554	33.5287

Table 3 PSNR (dB.) of noise and data cut attacks

Standard RGB	Baboon	Peppers	Airplane	Lena
Salt and Pepper (noise = 0.002)	34.21	33.63	33.10	33.76
Salt and Pepper (noise = 0.005)	30.001	30.51	28.86	30.78
Cropping (64 × 64)	24.0	23.1	22.5	23.63
Cropping (128 × 128)	18.1	17.2	17.0	18.0

degraded and original images. A greater PSNR value indicates a better deciphered image quality. In this test, an encrypted image has corrupted using two different attacks (salt & pepper) of (0.002, 0.005) levels and data cut attack of (128 × 128, 64 × 64) cuts; which then decrypted using the proposed technique. The PSNR results of suggested scheme are shown in Table 3; which shows its robustness against these types of attacks; since PSNR are higher than 33db for a noise level = 0.002 and has reached to 30.53db in average for a noise level of 0.005. Moreover, the PSNR values have reached to 24db and around 18db for a data cut of (64 × 64 and 128 × 128) size. Despite the



(a) Receivedimages with noise attack (0.002)

(b) Received images with noise attack (0.005)

(c) Data cut with 64 x 64

(d) Data cut with 128 x128

Fig. 7 Noise and Data-cut attacks

(a) Decrypted Quad Image/ original key (b) Decrypted QuadImage/ modified key

Fig. 8 Key sensitivity

reduction in PSNR values of data cut (128 × 128), the deciphered image can be recognized as shown in Fig. 7.

4.5 Key-space

The size of key-space has a main-role in the encryption stage. Hence a key size greater than 2^{100} makes the encryption method stronger to brute-force attacks. The proposed scheme has different security keys: ($t_1 \ldots t_6$), (i, j, k, l, m, n) and N_0. By assuming the accuracy of initial value = 10^{16}, then the total key-space is larger than $N_0 \times 10^{96}$, which in turn proves the robustness of this scheme against brute-force attack.

4.6 Key sensitivity

A high-sensitivity of secret-keys leads to a successful encryption scheme. Therefore, a slight-change in the initial conditions of used secret key leads to a great modification in the decrypted image. In this test, the quad-color fused-images encrypted in two cases. The first one has an original initial condition of ($x_1 \ldots x_6 = 0.1$) and the second one has the same ($x_1 \ldots x_5 = 0.1$) but modified x_6 in one small difference about (1×10^{-16}) that means ($x_6 = 0.1000000000000001$). The results of decryption process in these two cases

Table 4 Comparison of Correlation coefficients (CC): H, V, and D

Dir	[32]	[5]	[28]	[1]	[33]	[29]	Proposed
H	−0.003	0.0034	0.0009	−0.0036	−0.0003	0.0040	0.0024
V	−0.004	0.0015	0.0016	0.0016	0.0011	−0.0044	−0.0007
D	0.002	0.0008	0.0007	0.0058	0.0013	−0.0012	−0.0039

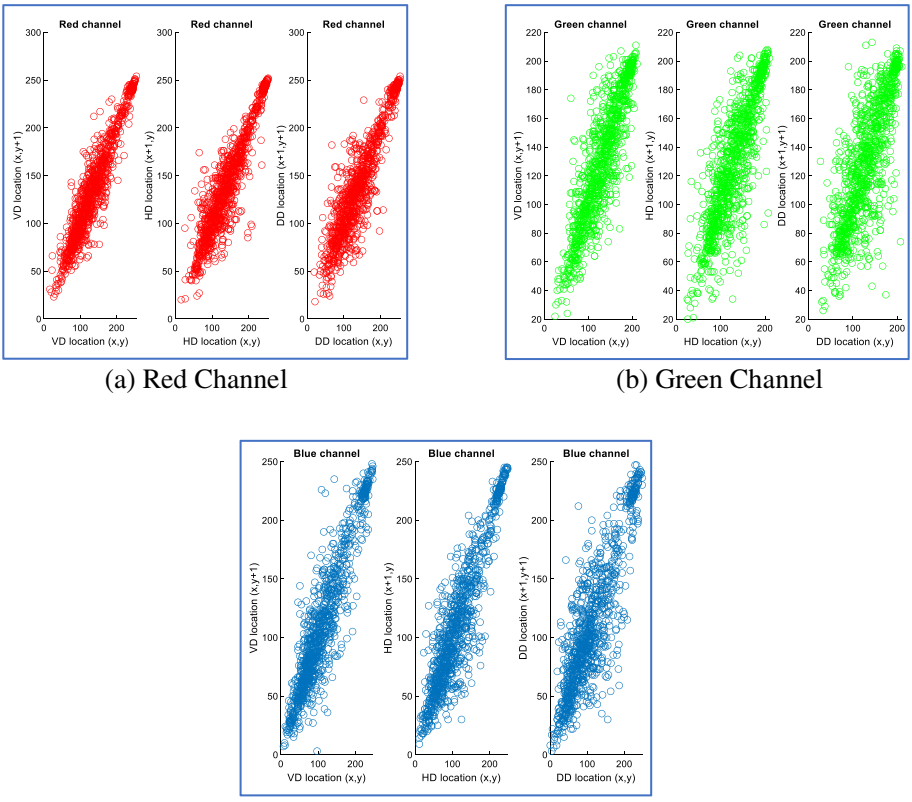
are demonstrated in Fig. 8; which shows its failure in restoring the original image when modifying the key slightly.

4.7 Correlation

In general, there is a great correlation between the adjacent pixels of plain images in three direction (diagonal D, horizontal H, and vertical V). Thus, minimizing it to zero or less increases the robustness of encryption algorithm that is calculated as follows:

$$r_{a,b} = \frac{Ex\Big((b - Ex(b))(a - Ex(a))}{\sqrt{Var(b)Var(a)}} \tag{23}$$

$$Ex(a) = \frac{1}{N}\sum_{j=1}^{N} a_j \tag{24}$$



(a) Red Channel

(b) Green Channel

(c) Blue Channel

Fig. 9 Baboon Image Correlation for 3 channels and 3 directions

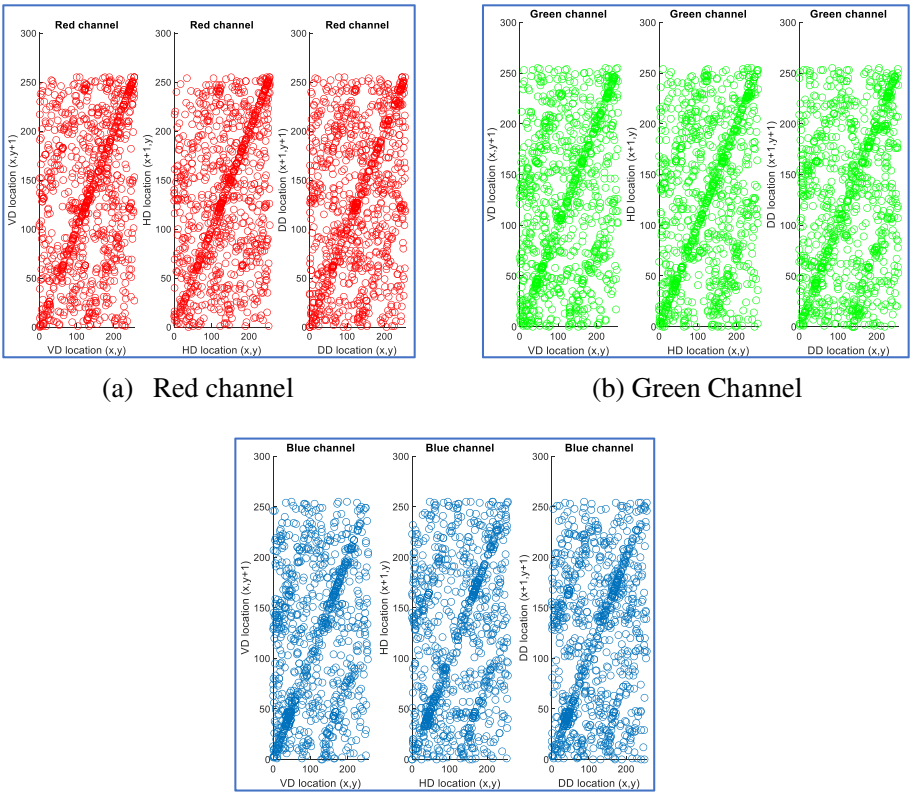
$$Var(a) = \frac{1}{N} \sum_{j=1}^{N} \left(\left(a_j - Ex(a) \right) \right)^2 \tag{25}$$

where N defines the total of nearby pixels; and $Var(a)$ and $Ex(a)$ are the variance and expectation of a, respectively.

The plain image (Baboon) shows a linear relationship among pixels as shown in Fig. 11 and its CC value has approached to unity as shown in Table 4. The suggested technique has broken the correlation among pixels; where the CC has been reduced to zero or negative value as shown in Table 4 and Figs. 9, 10 and 11 for Quad-Color plain bit level fusion and final Quad-Color Cipher image.

4.8 Complexity analysis

The size of input image M×N should be considered when computing the complications of the presented algorithm; assuming that (m) represents the number of pixels. Therefore, the complexity can be computed by the following defined operations: binary conversion, image fusion, 2×2 blocks splitter, randomized assembler, secret key



(a) Red channel

(b) Green Channel

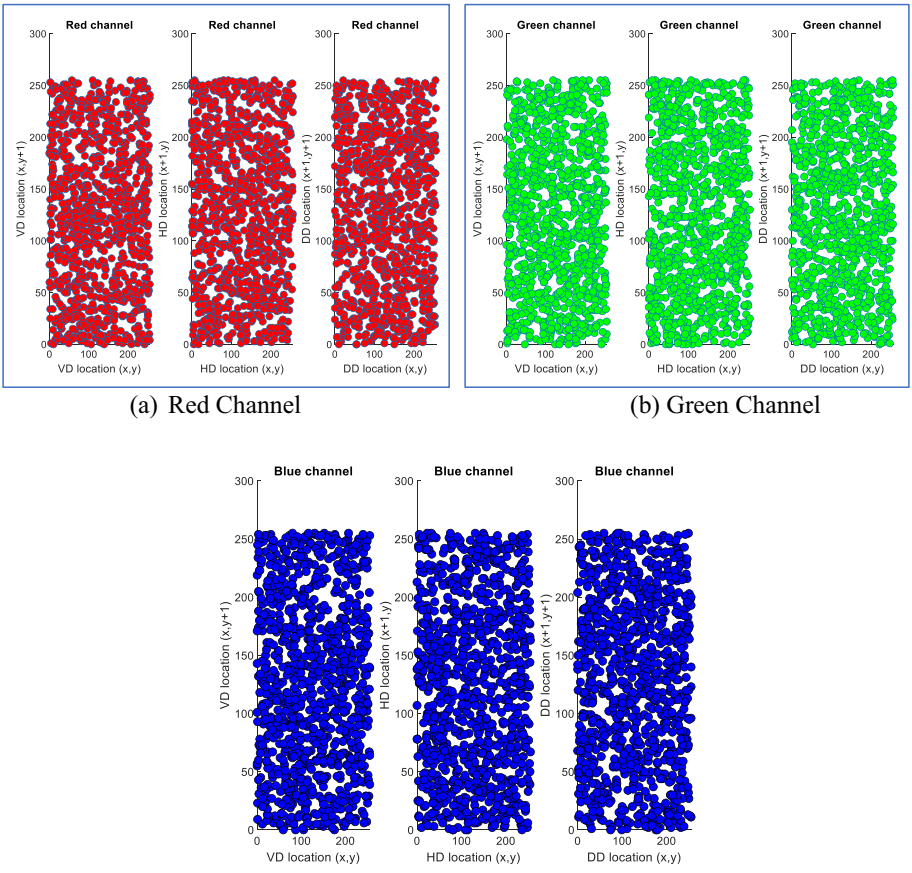
(c) Blue Channel

Fig. 10 Plain Quad-Color Fusion without Chaos and Fibonacci

generation, 6D_HyperC confusion and Fibonacci diffusion. The complexity of binary conversion = $O(n^2)$ and that of image fusion equals to $O(n^2)$. The 2×2 blocks splitter and randomized assembler are equal to $O(2n^2)$. The generation of secret-key has $O(n^2)$. The complexity of 6D-HyperC confusion and Fibonacci diffusion = $O(4n^2)$. Therefore, the overall complexity is $O(9n^2)$.

4.9 NIST statistical test

An NIST SP 800–22 of 15 sub-tests has applied to our scheme for checking the uniform distribution of cipher coordinates. It identifies the randomness of the encryption approach; where a good encryption should produce a high randomness that exceeds 0.01 for a p value in all sub-tests. According to the results shown in Table 5 of a cipher image (512×512), the algorithm has successfully passed all the statistical sub-tests.



(a) Red Channel (b) Green Channel

(a) Blue Channel

Fig. 11 Three Correlation channels of Final Quad-Color Cipher Image in three directions

Table 5 NIST statistical tests

Test Name		P Value	Pass rate	Result
Approximate Entropy		0.122325	1	Success
Frequency		0.122325	0.8	Success
Block frequency		0.017912	0.9	Success
Cumulative Sums	Forward	0.017912	0.8	Success
	Reverse	0.122325	0.8	Success
Runs		0.534146	1	Success
Longest Run		0.030913	1	Success
Serial	p-value1	0.213309	0.9	Success
	p-value2	0.017912	0.9	Success
Rank		0.534146	1	Success
Linear Complexity		0.017912	1	Success
Non-Overlapping Template		0.035174	1	Success
Overlapping Template		0.163426	1	Success
FFT		0.027423	1	Success
Maurer's "Universal Statistical"		0.939495	1	Success
Random Excursions	x = −9	0.398446	0.8	Success
	x = −8	0. 384,864		Success
	x = −7	0. 421,008		Success
	x = −6	0. 401,031		Success
	x = −5	0. 504,582		Success
	x = −4	0. 742,174		Success
	x = −3	0. 649,739		Success
	x = −2	0. 933,253		Success
	x = −1	0. 816,458		Success
	x = 1	0. 884,660		Success
	x = 2	0. 853,812		Success
	x = 3	0. 726,095		Success
	x = 4	0. 503,548		Success
	x = 5	0. 411,058		Success
	x = 6	0. 484,041		Success
	x = 7	0. 303,018		Success
	x = 8	0. 222,067		Success
	x = 9	0. 360,314		Success

4.10 Speed analysis

There are many factors that should be taken into consideration when analysing the elapsed encryption time such as system program capability, programming language, hardware and software environment, etc. Therefore, a comparison has been made with some related approaches including their hardware environments as demonstrated in Table 6. According to the obtained results, the speed of encryption is fast enough for a real-time demand compared with the other schemes.

Table 6 The Time consumption (sec) Encryption Algorithm

Scheme	No./Channel	Computer Configuration	Speed (sec)
[32]	Quad/Gray	Core i5–7500@2.50 GHz CPU and 4GB RAM	23.7
[24]	Triple/Gray	Core i5 processor and 8GB RAM	0.7374
[5]	Quad/Gray	Core i7-3537U@2.00 GHz CPU and 4GB RAM	1.89
[28]	Quad/Gray	Core i7 laptop @ 2.3 GHz CPU and 8GB RAM	1.86
[1]	Octagonal/Colour	Intel (R) W-2133@ 3.60 GHz CPU and 32GB RAM	1.78
[33]	Quad/Gray	M-5Y71@1.20 GHz CPU and 8GB RAM	1.71
Proposed method	Quad/Colour	Intel(R) i7-8565U @ 1.80GHz CPU and 16GB RAM	1.03

5 Conclusions

In this paper, a colorful quad images encryption/decryption algorithm has been proposed based on pixels' information of input images. Where a new fusion scheme was designed for splitting and fusing the inputs according to their binary bits within 2-rounds. Then they were segmented and randomly combined using a random key for increasing the security level. Moreover, two steps of 2-rounds were implemented to the randomly shuffled quad-fused image consists of a confusion (6D HyperC) and diffusion (FQ-matrix).

The time and space complexity has been reduced when dealt with the quad-color images instead of single image. The proposed scheme has been tested upon different metrics for evaluating its strength. The obtained results showed its robustness and resistance against various attacks. Increasing the size of input images, dimension of chaotic maps and a more complicated fusion approach could be our future steps.

References

1. Aasawari S et al (2020) Multiple images encryption based on 3D scrambling and hyper-chaoticSystem. Inf Sci 550:252–267. https://doi.org/10.1016/j.ins.2020.10.031
2. Abdel-Aziz MM et al (2021) Improved data hiding method for securing ColorImages. Multimed Tools Appl 80:12641–12670. https://doi.org/10.1007/s11042-020-10217-9
3. Artiles JA et al (2019) Image encryption using BlockCipher and chaotic sequences. Signal Process Image Commun 79:24–31. https://doi.org/10.1016/j.image.2019.08.014
4. Chai X et al (2018) An image encryption algorithm based on chaoticSystem and compressiveSensing. Signal Process 148:124–144. https://doi.org/10.1016/j.sigpro.2018.02.007
5. Enayatifar R et al (2019) Index-based permutation-diffusion in multiple-image encryption using DNA sequence. Opt Lasers Eng 115:131–140. https://doi.org/10.1016/j.optlaseng.2018.11.017
6. Hao J et al (2022) A novel color image encryption algorithm based on the fractional order laser ChaoticSystem and the DNA mutation principle. Multimed Tools Apps 81:559–587. https://doi.org/10.1007/s11042-021-11431-9
7. He Y et al (2018) A new image encryption algorithm based on two-dimensional spatiotemporal ChaoticSystem. Neural Comput Appl 32:247–260. https://doi.org/10.1007/s00521-018-3577-z
8. Hosny KM et al (2019) Resilient ColorImage watermarking using accurate quaternion radial substituted Chebyshev moments. ACM Trans Multimed Comput Commun Appl 15:1–25. https://doi.org/10.1145/3325193
9. Hua Z et al (2019) Cosine-transform-based chaotic system for image encryption. Inf Sci 480:403–419. https://doi.org/10.1016/j.ins.2018.12.048
10. Khan M et al (2019) A novel chaotic image encryption technique based on multiple discreteDynamical maps. Multimed Tools Apps. 78:26203–26222. https://doi.org/10.1007/s11042-019-07818-4
11. Laiphrakpam DS et al (2017) Medical image encryption based on improved ElGamal encryption technique. Optik 147:88–102. https://doi.org/10.1016/j.ijleo.2017.08.028

12. Li M et al (2019) Cryptanalysis of a Novel Bit-Level Color ImageEncryption Using Improved 1D ChaoticMap. IEEE Access 7:145798–145806. https://doi.org/10.1109/ACCESS.2019.2945578
13. Li N et al (2020) Reversible DataHiding for JPEG images based on pairwiseNonzero AC coefficient expansion. Signal Process 171:107476. https://doi.org/10.1016/j.sigpro.2020.107476
14. Li Y et al (2021) Image encryption based on a single-round dictionary and ChaoticSequences in CloudComputing. Concurr Comput Pract Exp 33:1. https://doi.org/10.1002/cpe.5182
15. Luo Y et al (2018) A parallel image encryptionAlgorithm based on the piecewiseLinear chaotic-Map and hyper-chaotic map. Nonlinear Dyn 93:1165–1181. https://doi.org/10.1155/2013/537934
16. Molina-Garcia J et al (2020) An effective fragileWatermarking scheme for ColorImage tampering detection and self-recovery. Signal Process Image Commun 81:115725. https://doi.org/10.1016/j.image.2019.115725
17. Niyat AY et al (2017) Color image encryption based on hybrid hyper-chaotic system and CellularAutomata. Opt Lasers Eng 90:225–237. https://doi.org/10.1016/j.optlaseng.2016.10.019
18. Premkumar R et al (2022) An efficient Chaos-based image encryption technique using BitplaneDecay and GeneticOperators. Sens 22:8044. https://doi.org/10.3390/s22208044
19. Ran B et al (2022) Image security based on three-dimensional ChaoticSystem and RandomDynamic selection. Entropy 24:958. https://doi.org/10.3390/e24070958
20. Shu X et al (2015) Personalized AgeProgression with AgingDictionary. In proceedings of the 2015 IEEE ICCV, Washington, DC, USA, pp 3970–3978. https://doi.org/10.1109/ICCV.2015.452
21. Shu X et al (2018) Personalized AgeProgression with bi-level AgingDictionary learning. IEEE Trans Pattern Anal Machine Intell 40(4):905–917. https://doi.org/10.1109/TPAMI.2017.2805122
22. Wang J et al (2019) A new six-dimensional hyperchaotic system and its secure communicationCircuit implementation. Int J Circ Theor Appl 47:702–717. https://doi.org/10.1002/cta.2617
23. Wang X et al (2020) Image encryption algorithm for synchronously updating Boolean networks based on matrix semi-tensor product theory. Inf Sci 507:16–36. https://doi.org/10.1016/j.ins.2019.08.041
24. Wang X et al (2021) A novel triple-image encryption and HidingAlgorithm based on chaos, compressiveSensing and 3D DCT. Inf Sci 574:505–527. https://doi.org/10.1016/j.ins.2021.06.032
25. Wu J et al (2018) ImageEncryption using 2D Hénon-sine map and DNA approach. Signal Process 153:11–23. https://doi.org/10.1016/j.sigpro.2018.06.008
26. Yan W et al (2021) A Three-Dimensional InfiniteCollapseMap with ImageEncryption. Entropy 23:1221. https://doi.org/10.3390/e23091221
27. Ye G et al (2020) Image encryption and hidingAlgorithm based on compressiveSensing and randomNumbers insertion. SignalProcess. 172:107563. https://doi.org/10.1016/j.sigpro.2020.107563
28. Ye H et al (2020) Multi-image compression-encryption scheme based on QuaternionDiscrete fractional HartleyTransform and improved PixelAdaptive diffusion. Signal Process 175:107652–107666. https://doi.org/10.1016/j.sigpro.2020.107652
29. Ye G et al (2022) Double image encryption algorithm based on compressiveSensing and ellipticCurve. AEJ 61(9):6785–6795. https://doi.org/10.1016/j.aej.2021.12.023
30. Yu S-S et al (2020) Optical image encryptionAlgorithm based on phase-truncated short-time fractionalFourier transform and hyper-chaotic system. Opt LasersEng 124:105816
31. Zhang Y (2018) The image encryptionAlgorithm based on chaos and DNAcomputing. Multimed Tools Apps 77:21589–21615. https://doi.org/10.1007/s11042-017-5585-x
32. Zhang X et al (2019) Multiple-image EncryptionAlgorithm based on DNA encoding and ChaoticSystem. Multimed Tools Apps 78:7841–7869. https://doi.org/10.1007/s11042-018-6496-1
33. Zhang X et al (2021) Multiple-image encryption algorithm based on the 3Dscrambling model & dynamicDNA coding. Opt LaserTech 141:30–3992. https://doi.org/10.1016/j.optlastec.2021.107073
34. Zhou N et al (2017) Quantum image encryptionScheme with iterative generalized ArnoldTransforms and quantum image cycle shift operations. Quantum Inf Process 16:164. https://doi.org/10.1007/s11128-017-1612-0
35. Zhou T et al (2020) Logarithmic encryptionScheme for cyber–physical systems employing FibonacciQ-matrix. Futur Gener Comput Syst 108:1307–1313. https://doi.org/10.1016/j.future.2018.04.008