Multimedia Tools and Applications (2024) 83:48563–48584
https://doi.org/10.1007/s11042-023-17166-z

Multiple colour image encryption using multiple parameter FrDCT, 3D Arnold transform and RSA

Vandana Guleria[1] · **Yashavant Kumar**[1] · **D. C. Mishra**[2]

Received: 13 June 2022 / Revised: 27 April 2023 / Accepted: 11 September 2023 /
Published online: 3 November 2023

Abstract
We introduce a novel image encryption and decryption algorithm for multiple images incorporating multiple parameter fractional discrete cosine transform (MPFrDCT), 3D Arnold transform and RSA cryptosystem. Before encryption, the images are changed into their indexed formats by removing their color maps. The indexed formats of the images are taken as the red, green and blue channel of an RGB image. Firstly, the RGB image is taken as the input of 3D Arnold transform. The 3D Arnold transform not only dislocates the pixel positions, but also changes the pixel values. Mathematically, the 3D map performs both permutation as well as substitution. The distorted image is now encrypted using RSA cryptosystem which is a public key cryptosystem. The RSA cryptosystem makes the image secure in public domain as the hard problem is the factorization of large primes which is unbreakable. Lastly, the domain of the encrypted image is changed to frequency domain using MPFrDCT. If the secret keys are known to an unauthorized person, the encryption algorithm is still secure as the security of the presented cryptosystem depends upon the secret keys and the arrangements of the secret keys. The proposed image encryption algorithm is storage efficient. The statistical and simulation analysis are conducted to evaluate the robustness of the presented encryption and decryption processes.

Keywords RSA Cryotosystem · Image encryption · Image decryption · 3D Arnold transform and multi-parameter fractional discrete cosine transform

✉ Vandana Guleria
vandana.math@gmail.com

Yashavant Kumar
kumar.yashavant154@gmail.com

D. C. Mishra
deepiitdelhi@gmail.com

[1] Department of Mathematics, Birla Institute of Technology Mesra, Ranchi, India

[2] Department of Mathematics, Govt. P.G. College Jaiharikhal, Uttarakhand, India

1 Introduction

Due to vast development of network and communication technologies, exchange of digital images over the public channel has increased. The major issues in transmitting images over the public channel are storage and security of the images.

The security of the images can be achieved by developing image encryption algorithm. Several color image encryption algorithms are introduced in this research field [18, 22, 40, 44] using Mellin transform, random phase encoding and Arnold transform etc. Optical image encryption algorithm [28] has been introduced in fourier transform domain. Researchers have also developed image encryption algorithms [10, 11, 19, 21, 41] in fractional Fourier transform domain which is the generalization of Fourier transform. Image encryption algorithms based on choatic mapping combined with Hartley transform domain [8, 23], Gyrator transform [1–5, 31] and wavelet transform [6, 9, 26] have been proposed.

Multiple images are taken together to make the image encryption algorithms storage efficient. This concept was introduced by Situ and Zhang [32] incorporating wavelet multiplexing. Later, multiple image encryption algorithms [14, 17, 20, 30, 37, 39] are developed in fractional Fourier transform domain, Arnold transform, Gyrator transform and Fresnel domain. At the same time, double image encryption algorithm [29, 43] using multi-parameter fractional Fourier transform are proposed. The addition of chaotic map, dual pixel scrambling random phase encoding etc. adds an extra layer of security to the image encryption algorithms due to their scrambling behavior. Nonlinear amplitude and phase truncation based multiple image encryption algorithm [34, 35] in Fourier transform domain are also presented by different researchers. Nowadays, many researchers [12, 13, 16, 24, 25, 36, 42] are working in this field.

Our contribution We propose a novel technique to encrypt multiple images together in a multi-parameter frequency domain. Initially, three indexed images are extracted from three RGB images. The three indexed images are treated as red, green and blue plane of an RGB image. The 3D Arnold transform is applied on the constructed RGB image. This transform permutes as well as changes the pixel values of the image. After this step, each pixel value is encrypted using RSA cryptosystem. This improves the security of the image as the hard problem in RSA cryptosystem is factoring of large primes. Lastly, MPFrDCT is applied on each encrypted pixel value. The encrypted image is a single image which makes the image encryption algorithm storage efficient as it is convenient to transform single image as compared to multiple images. The decryption process is the reverse of the encryption process. After decryption, three images are recovered from the single encrypted image. Comparison, statistical and security analyzes are done to testify the proposed encryption algorithm.

Novelity The presented image encryption algorithm is multi-layered secure in comparison to existing similar image encryption algorithms which are only single layered secure. The security in time, frequency and co-ordinate domain is developed using RSA cryptosystem, Arnold 3D cat map and multi-parameter FrDCT. The secret parameters in the scheme are the large primes in RSA, multiple parameters of MPFrDCT, secret keys and the arrangements of secret keys. If an unauthorized person has access to secret keys still he is unable to get the original image as he does not know about the arrangements of secret keys. The presented image encryption algorithm is efficient in comparison to existing similar techniques in terms of time complexity, storage complexity and communication complexity. Therefore, the presented scheme is novel.

In Section 2, we have described the building blocks briefly. These are 3D Arnold transform, MPFrDCT, RSA cryptosystem and Chirikov standard map. The encryption and decryption algorithm is presented in Section 3. Simulation is conducted in Section 4. In Section 5, we briefly compare our proposed algorithm with the similar techniques. Finally, the conclusion is done in Section 6.

2 Preliminaries

The fundamental building blocks RSA cryptosystem, MPFrDCT, discrete Chirikov standard map and 3D Arnold transform are briefly explained in this Section.

2.1 3D Arnold transform

The 2D Arnold transform permutes the position of the pixel in an image of size $M \times M$ [30, 33] using the following map.

$$\begin{pmatrix} r_1' \\ r_2' \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \end{pmatrix} (\text{mod } M), \tag{2.1}$$

where (r_1', r_2') is the new pixel position after permutation and (r_1, r_2) is the old pixel position before permutation. The Arnold transfer only shifts the pixel positions from one position to another position. The intensity values of the pixels remain unchanged.

The 3D Arnold transform permutes and substitutes the image pixels using the following map.

$$\begin{pmatrix} r_1' \\ r_2' \\ r_3' \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 0 \\ z_1 & z_2 & 1 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}, \tag{2.2}$$

where the (mod M) is applied in the calculation of first two rows and (mod 256) is applied in the calculation of last row. The 3D Arnold transform first permutes the pixel position and then the substitution operation is done to change the pixel values. The third parameter

$$r_3' = z_1 \times r_1 + z_2 \times r_2 + r_3 (\text{mod } 256),$$

where r_3' is the pixel intensity after mapping and r_3 is the pixel intensity before mapping. The inverse 3D Arnold map is given as below.

$$\begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & 0 \\ -z_1 & -z_2 & 1 \end{pmatrix} \begin{pmatrix} r_1' \\ r_2' \\ r_3' \end{pmatrix}, \tag{2.3}$$

If the 3D Arnold transform is applied on the image t times, then the same image can be obtained using inverse 3D Arnold transform t times. The parameter t is kept secret.

2.2 Assymetric key cryptosystem: RSA cryptosystem

An asymmetric key cryptosystem is a public key cryptosystem in which two different keys are used for image encryption and decryption process. One key is called public key and another is secret key. The secret key is kept hidden with the decrypter and the public key is made available for everyone. There is mathematical relation between the two keys. The RSA cryptosystem involves following steps.

1. Firstly, two large different primes p and q are selected at random.
2. Generate the integer n by computing $n = p * q$. Compute

$$\phi(n) = (p-1)(q-1), \text{ where}$$

$$\phi(n) = \{x \in \mathbb{N} : (x, n) = 1 \text{ and } x < n\}$$

   is the number of positive integers less than n and relatively prime to n.
3. The encryption key e is the integer less than and relatively prime to $\phi(n)$, where, $(e, \phi(n)) = 1$ and $1 < e < \phi(n)$. The encryption key is made available for everyone and is known as public key.
4. The decryption key d is the inverse of encryption key e modulo $\phi(n)$, i.e, $d = e^{-1} (\text{mod } \phi(n))$. The decryption key is kept secret and is known as secret key.
5. The plaintext message $y \in \mathbb{Z}_n$ is encrypted by computing $y^e (\text{mod } n) = c$, where $\mathbb{Z}_n = \{0, 1, 2, ..., n-1\}$, c is called ciphertext.
6. The ciphertext c is decrypted by computing $c^d (\text{mod } n) \equiv y^{ed} (\text{mod } n) \equiv y (\text{mod } n)$, as $ed \equiv 1 (\text{mod } \phi(n))$.

The RSA cryptosystem is secure as long as the attacker does not know about the factorization of n. So, the hard problem in RSA cryptosystem is factoring of large primes. This hard problem is also known as factoring problem. It states that it is computationally infeasible to factorize $n = p * q$, where p and q are large primes. RSA cryptosystem withstands frequency analysis attack indirectly by its encryption algorithm. In frequency analysis attack, the frequency distribution of ciphertext is analyzed to guess the corresponding plaintext. This attack is more effective where messages are replaced by a fixed pattern like in substitution cipher. The frequency of the ciphertext reveals the original plaintext whereas in RSA cryptosystem the ciphertext is obtained by taking the large power of the plaintext. The resulting ciphertext is a complex mathematical transformation rather than a simple substitution or transposition of the plaintext. The ciphertext looks like a random string of numbers and reveals nothing about the plaintext. Moreover, the key size in RSA is nearly 2048 bits or more which makes it computationally infeasible to factorize the modulus and break the encryption algorithm using brute force or other known attacks.

In the proposed encryption algorithm, an RGB image with size $M \times N \times 3$ is encrypted using RSA cryptosystem. Each channel is a two dimensional matrix, say X, of size $M \times N$, where $X = [X_{i,j}], i = 1, 2, \ldots, M, j = 1, 2, \ldots, N$. Each matrix has $M * N$ elements which are called pixels. Each pixel $X_{i,j}$ is encrypted using the encryption key e as

$$C_{i,j} = X_{i,j}^e (\text{mod } n).$$

Decrypt $C_{i,j}$ using decryption key d as follows.

$$C_{i,j}^{d}(\text{mod } n) = X_{i,j}.$$

2.3 Chirikov standard map

This invertible map is useful in generating $\{\eta(j)\}_{j=1,2,...}$ and $\{\delta(j)\}_{j=1,2,...}$ which are random sequences using initial guesses $\{\eta_0\}, \{\delta_0\} \in [0, 2\pi)$. The two sequences are mathematically generated as follows.

$$\eta_{j+1} = (\eta_j + \delta_j)(\text{mod } 2\pi) \tag{2.4}$$

$$\delta_{j+1} = (\eta_j + \rho \sin(\eta_j + \delta_j))(\text{mod } 2\pi), \tag{2.5}$$

where $\rho > 0$ is a control parameter and $\eta_j, \delta_j \in [0, 2\pi), \forall i$.

2.4 Multi-parameter fractional discrete cosine transform (MPFrDCT)

The DCT is an operator $\mathcal{C} : \mathbb{R}^N \to \mathbb{R}^N$. The input and output of this map are both real vector of size N. The kernel matrix $\mathbf{C}$ for $\mathcal{C}$ is given as below.

$$\mathbf{C} = \left\| \frac{1}{\sqrt{M}} \varepsilon_k \cos(2\pi \frac{l(2n+1)}{4M}) \right\|. \tag{2.6}$$

where $n, l = 0, 1, ..., M-1, \varepsilon_0 = 1, \varepsilon_k = \sqrt{2}, l \geq 1$.

The properties of the matrix $\mathcal{C}$ are as follows.

1. $\mathbf{C}$ is a unitary matrix.
2. $\mathbf{C}$ has M orthonormal eigenvectors x_l with the property $x_m^* x_l = \delta_{ml}$, where

$$\delta_{ml} = \begin{cases} 1, & \text{if m=l} \\ 0, & \text{otherwise.} \end{cases} \tag{2.7}$$

3. The eigenvalue λ_l corresponds to eigenvector x_l where $\lambda_l = e^{j\varphi_l}$, $0 < \varphi_l < \pi \in \mathbb{R}$ and it lies on the unit circle.
4. $\mathbf{C}$ is diagonalized as follows.

$$\mathbf{C} = X \Lambda X^* = \sum_l X_n e^{j\varphi_l}, \tag{2.8}$$

where Λ is a diagonal matrix with diagonal entries as λ_l, $X_l = x_l x_l^*$ is unitary and $X_m X_l = \delta_{ml} X_m$ and $\sum_l X_l = I$

The map $\mathcal{C}_a : \mathbb{R}^N \to \mathbb{R}^N$ is an extension of the operator $\mathcal{C}$ and it is called fractional discrete cosine transform (FrDCT) [7], where "fraction" $a \in \mathbb{R}$. The operator $\mathcal{C}_a$ possess the following properties.

1. $\mathcal{C}_{a+b} = \mathcal{C}_a \mathcal{C}_b$, this property is called additive property.
2. $\mathcal{C}_1 = \mathcal{C}$, i.e, for $a = 1, \mathcal{C}_a = \mathcal{C}$.

The kernel matrix $\mathbf{C}_a$ for FrDCT is given below.

$$\mathbf{C}_a = X\Lambda^a X^*. \tag{2.9}$$

Alternatively,

$$\mathbf{C}_a = 2\mathrm{Re}\left[\sum_{l=1}^{K} X_l \lambda_l^a\right] + Y_1(1)^a + Y_{-1}(-1)^a, \tag{2.10}$$

where $X_l = x_l x_l^*, K = {}^{(M-\varrho_1-\varrho_{-1})}/_2$, 1 and -1 are the eigenvalues with the algebraic multiplicities ϱ_1 and ϱ_{-1}, Y_1 is the total number of ϱ_1 matrices X_l for eigenvalue 1, and Y_{-1} is the total number of ϱ_{-1} matrices X_l for eigenvalue -1.

For instance, take $M = 4M_0$, $M_0 \in \mathbb{Z}$, (2.10) becomes

$$\mathbf{C}_a = 2\mathrm{Re}\left[\sum_{l=1}^{M/2} X_l \lambda_l^a\right] = 2\mathrm{Re}\left[\sum_{l=1}^{M/2} X_l e^{j\omega_l a}\right] = \sum_{l=1}^{M/2}\Big(A_l \cos(\omega_l a) + B_l \sin(\omega_l a)\Big), \tag{2.11}$$

where $\omega_l = \varphi_l + 2\pi q_l,\ l = 1, 2, ..., \frac{M}{2}, 0 < \varphi_l < \pi,\ A_l = 2\mathrm{Re}[X_l], B_l = 2\mathrm{Im}[X_l],\ q_l$ is the random sequence generated as explained in Section 2.3. For FrDCT, $\mathbf{q_l} = (q_1, q_2, ..., q_{M/2})$, is the random sequence . Both $\mathbf{q_l}$ and a are secret.

Multi-parameter fractional discrete cosine transform (MPFrDCT) of fractional order a, b for an image $I_{M,N}$ of size $M \times N$ is defined as follows:

$$I'_{M,N} = C_a I_{M,N} C_b^T,$$

where, C_b^T is the notation for the transpose of C_b. The inverse MPFrDCT is computed a follows:

$$I_{M,N} = C_{-a} I'_{M,N} C_{-b}^T.$$

3 Description of the proposed algorithm

In the design of an image encryption algorithm, a security system is developed by incorporating 3D Arnold transform, RSA cryptosystem and MPFrDCT. Firstly three RGB images are taken and converted into their indexed formats after removing the colour maps. The obtained three indexed images are named as $\mathtt{I}_1$, $\mathtt{I}_2$ and $\mathtt{I}_3$ and are treated as red, green and blue channel of an RGB image. The image encryption and decryption processes are pictorially displayed in Figs. 1 and 2 respectively.

Step 1 (Arnold 3D Transform): Firstly, Arnold 3D map is applied on each channel of an RGB image. The input matrices are $\mathtt{I}_1$, $\mathtt{I}_2$ and $\mathtt{I}_3$. The mechanism involved in this step is given below.

1. The Arnold 3D map is applied t_1, t_2 and t_3 times on $\mathtt{I}_1$, $\mathtt{I}_2$ and $\mathtt{I}_3$. The three matrices AR, AG and AB are obtained as output.
2. The matrices AR, AG and AB are treated as red, green and blue channel of an RGB image image.

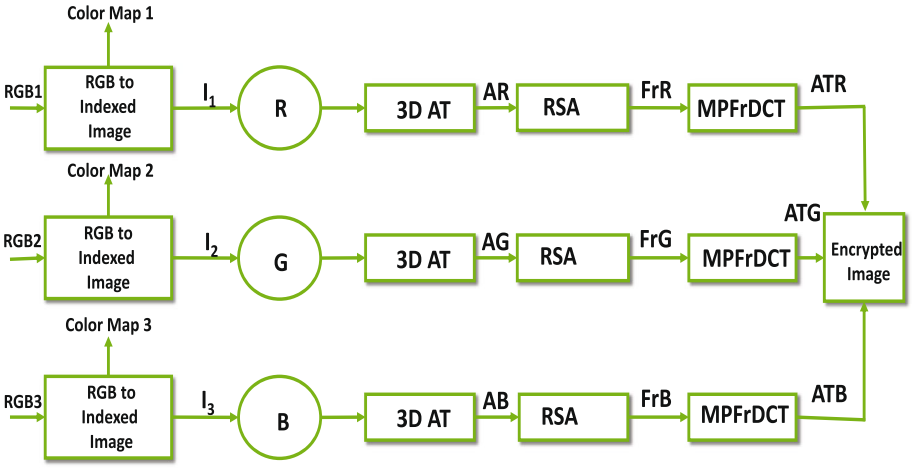

Fig. 1 Encryption Process

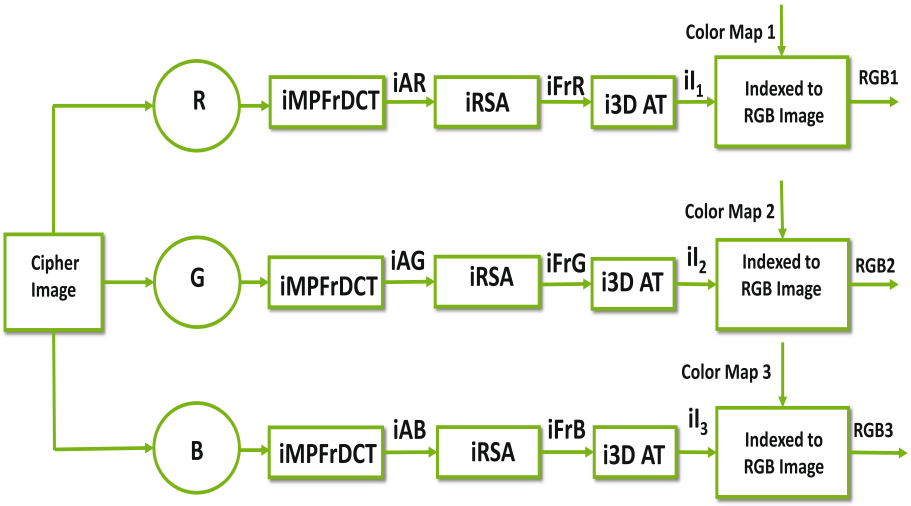

Fig. 2 Decryption Process

Step 2 (RSA Cryptosystem): Secondly, each pixel intensity of each channel of the RGB image is encrypted as discused in Section 2.2 using RSA Cryptosystem. The public key and secret key for each component are given as below.

1. For red channel input matrix is AR, the public and secret keys are $p_r, q_r, n_r = p_r * q_r$, $\phi(n_r) = (p_r - 1)(q_r - 1), e_r, d_r = e_r^{-1} (\text{mod}\, n_r)$, where $(e_r, \phi(n_r)) = 1$ and $1 < e_r < \phi(n_r)$.
2. For green channel input matrix is AG, the public and secret keys are $p_g, q_g, n_g = p_g * q_g$, $\phi(n_g) = (p_g - 1)(q_g - 1), e_g, d_g = e_g^{-1} (\text{mod}\, n_g)$, where $(e_g, \phi(n_g)) = 1$ and $1 < e_g < \phi(n_g)$.

3. For blue channel input matrix is AB, the public and secret keys are $p_b, q_b, n_b = p_b * q_b$, $\phi(n_b) = (p_b - 1)(q_b - 1), e_b, d_b = e_b^{-1} (\text{mod } n_b)$, where $(e_b, \phi(n_b)) = 1$ and $1 < e_b < \phi(n_b)$.

The output matrices of this stage are FrR, FrG and FrB.

Step 3 (Multi-Parameter Fractional Discrete Cosine Transform (MPFrDCT)): The MPFrDCT is applied on the output matrices FrR, FrG and FrB as explained in Section 2.4. The initial control parameters are generated according to the conditions explained in Section 2.3.

1. The initial parameters x_0, y_0 and ρ are taken randomly.
2. The two random sequences **s** and **t** are generated following the () and (2.42.5) $(1000 + M/2)$ times. Each random sequence is of length $M/2$. The sequences **s** and **t** are divided by π to generate sequences $\mathbf{s}'$ and $\mathbf{t}'$. Both the sequences lie in the range [0, 2).
3. Generate the sequences $\mathbf{q_1}$ and $\mathbf{q_2}$ as follows.

$$q_1(j) = \begin{cases} 0, \text{ if } 0 \leq \mathbf{s}'(\mathbf{j}) \leq 1 \\ 1, \ 1 < \mathbf{s}'(\mathbf{j}) \leq 2 \end{cases} \tag{3.1}$$

$$q_2(j) = \begin{cases} 0, \text{ if } 0 \leq \mathbf{t}'(\mathbf{j}) \leq 1 \\ 1, \ 1 < \mathbf{t}'(\mathbf{j}) \leq 2 \end{cases} \tag{3.2}$$

where $j = 1, 2, ..., M/2$.

4. Apply MPFrDCT with parameters a and b.

The output of this step are ATR, ATG and ATB. The encryption algorithm encrypts three images together and produces a single encrypted image. The single encrypted image saves storage and communication cost. The decryption algorithm is just the reverse of the encryption algorithm. In decryption, firstly inverse MPFrDCT is applied on the encrypted image with parameters $-a$ and $-b$. Now the partially encrypted image is decrypted using RSA cryptosystem. The secret keys used in the step are d_r, d_g, and d_b such that $e_r d_r \equiv 1 (\text{mod } \phi(n_r))$, $e_g d_g \equiv 1 (\text{mod } \phi(n_g))$ and $e_b d_b \equiv 1 (\text{mod } \phi(n_b))$. Lastly, the inverse Arnold 3D map is applied t_1, t_2 and t_3 times on each channel of the partially encrypted image. The inverse secret keys for RSA cyptosystem are d_r, d_g and d_b. Finally, colour maps are added to each channel of the decrypted images to get back the original RGB color images.

4 Simulation results

Simulation analysis is done to prove that the real world tests are conducted using the proposed encryption scheme. The three experimental images for simulation are Baboon, Lena and Peppers from top to bottom given in Fig. 3(a). Each image is an RGB image of size $512 \times 512 \times 3$. Firstly, the colour maps are extracted from these three images to produce three indexed images I_1, I_2 and I_3. Three indexed images I_1, I_2 and I_3 are combined together to produce an RGB image of size $512 \times 512 \times 3$. The public and secret keys for red channel are $p_r = 59, q_r = 61, n_r = 3599$, $\phi(n_r) = 3480, e_r = 17, d_r = 1433$. The public and secret

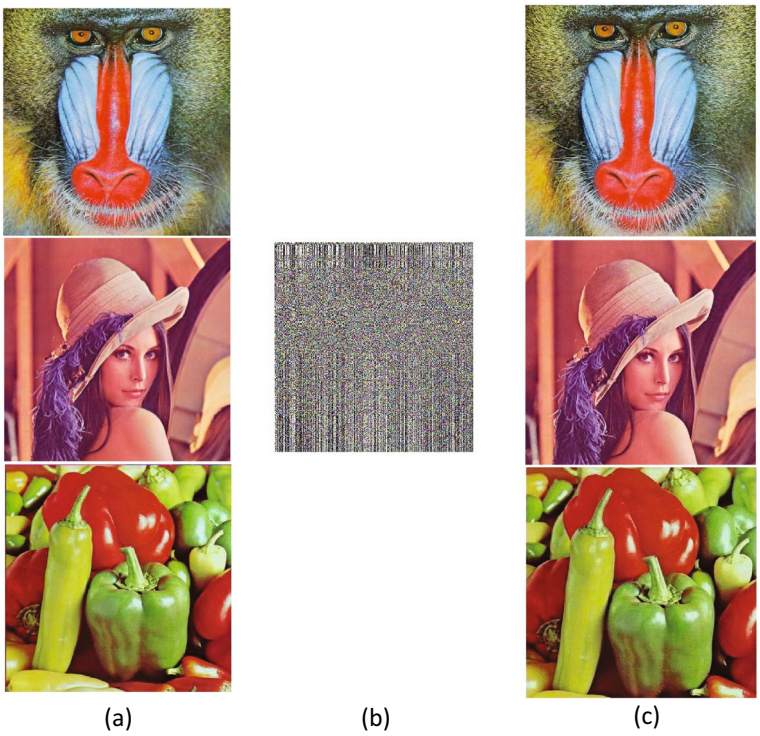

Fig. 3 Encryption/Decryption Results

keys for green channel are $p_g = 73, q_g = 79, n_g = 5767, \phi(n_g) = 5616, e_g = 19, d_g = 3547$. The public and secret keys for blue channel are $p_b = 83, q_b = 89, n_b = 7387$, $\phi(n_b) = 7216, e_b = 23, d_b = 1255$. Control parameters and initial guesses are set as $x_0 = 0.5489$, $y_0 = 0.4517$ and $\rho = 0.3587$ to generate random sequences $\mathbf{q}_1$ and $\mathbf{q}_2$. The MPFrDCT parameters are $a = 0.5762$ and $b = 0.3982$. The parameters for Arnold 3D transform are set as $t_1 = 23$, $t_2 = 50$ and $t_3 = 87$ for red, green and blue channel respectively. These parameters are kept secret. The single encrypted image of three color images is Fig. 3(b). The parameters for decryption process are taken as follows. The inverse Arnold 3D parameters are $t_1 = 23$, $t_2 = 50$ and $t_3 = 87$. The inverse parameters for MPFrDCT are $-a$ and $-b$. The correctly decrypted images are displayed in Fig. 3(c).

4.1 Security analysis

Security analysis is a tool to check the robustness of the encryption technique. The secret keys for the decryption of the encrypted image are $d_r = 1433, d_g = 3547, d_b = 1255$, $t_1 = 23$, $t_2 = 50$, $t_3 = 87$, $x_0 = 0.5489, y_0 = 0.4517$, $\rho = 0.3587$, $a = 0.5762$ and $b = 0.3982$. The images will be recovered correctly if the decrypter uses these secret keys in correct order. The encryption technique is very sensitive to secret keys. By sensitivity, we mean if we do small changes in the secret keys, the decrypter would not be able to get the original images. The sensitivity parameters are $d_r, d_g, d_b, , x_0, y_0, \rho, t_1, t_2, t_3, a$, and b. The sensitivity analyzes are done briefly in this Section to prove our claim.

4.1.1 Key space analysis

To check the robustness of the algorithm, key space analysis plays a vital role. There is a direct relationship between the robustness and the key space size. The secret keys involved in the proposed encryption algorithm are (i) the iteration numbers of Arnold 3D map t_1, t_2 and t_3, (ii) initial parameters x_0 and y_0, (iii) control parameter ρ, (iv) decryption keys d_r, d_g, d_b in RSA cryptosystem and (v) fractions a and b. The hard problem in RSA cryptosystem is factoring algorithm, i.e., the factorization of n_r, n_g and n_b. If n_r, n_g and n_b, all are 1024 bits long, the factorization is completely infeasible. The control parameter ρ and initial guesses x_0, y_0 are very small parameters. If we take the precision 10^{-14}, the key would be approximately of size 10^{70}. Also, the iteration numbers are integers and very large to withstand the exhaustive attack.

4.1.2 Key sensitivity analysis

This test is conducted to demonstrate that the secret parameters are very sensitive. The term sensitivity means the small changes in the secret parameters would not allow decrypter to decrypt images correctly. Figure 3(a) shows the three input colour images. The single encrypted image is given in Fig. 3(b). Correctly decrypted images using correct secret keys with their correct arrangements are displayed in Fig. 3(c).

Sensitivity analysis-I This sensitivity analysis is executed using incorrect keys d_r, d_g and d_b. The incorrect keys are $d_r = 3547, d_g = 1255, d_b = 1433$. The three decrypted images using these secret keys are given in Fig. 4a(i)-(iii).

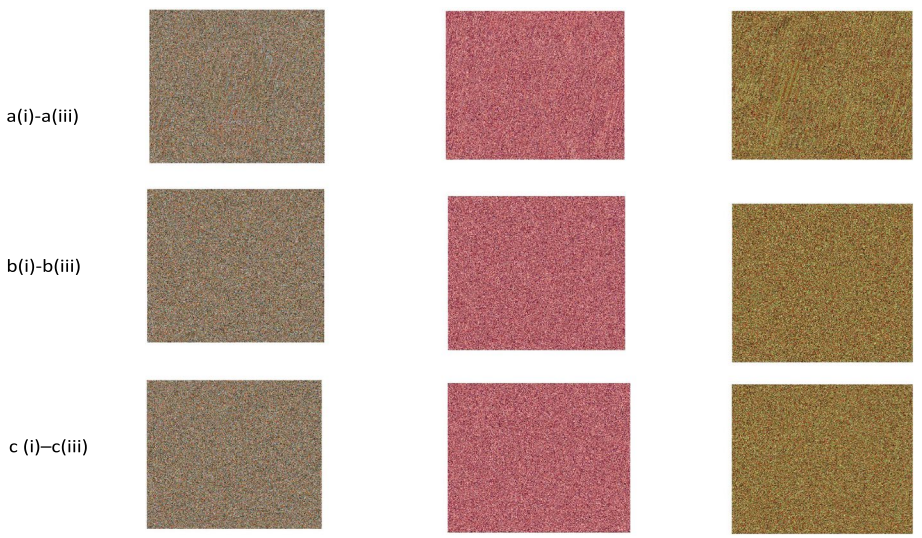

Fig. 4 **a**(i)-a(iii) is Sensitivity Analysis-I using wrong $d_r = 3547, d_g = 1255$ and $d_b = 1433$, **b**(i)-b(iii) is Sensitivity Analysis-II using wrong iteration numbers $t_1 = 50, t_2 = 87$ and $t_3 = 23$, **c**(i)-c(iii) is Sensitivity Analysis-III using wrong fractions $a = 0.3982$ and $b = 0.5762$

Sensitivity analysis-II This sensitivity analysis is executed with incorrect iteration numbers of 3D Arnold map. The iteration parameters are $t_1 = 50, t_2 = 87$ and $t_3 = 23$. The three decrypted images using these secret keys are shown in Fig. 4b(i)-(iii).

Sensitivity analysis-III This sensitivity analysis is executed with random parameters of MPFrDCT. The random parameters are $a = 0.3982$ and $b = 0.5762$. The other secret parameters are not changed. The three decrypted images using these secret keys are shown in Fig. 4c(i)-(iii).

Sensitivity analysis-IV This sensitivity analysis is executed with incorrect initial parameters $x_0 = 0.4517$ and $y_0 = 0.6489$. The three decrypted images using these parameters are displayed in Fig. 5(a). The incorrect decrypted images with incorrect control parameter $\rho = 0.8375$ are shown in Fig. 5(b).

4.1.3 Robustness against CPA and CCA

The proposed encryption algorithm should resist chosen-plaintext attack (CPA) and chosen-ciphertext attack (CCA). In CPA, the attacker has access to plaintexts of his choice. He received ciphertexts for the selected plaintexts. In CCA, the attacker has access to ciphertexts of his choice. He also received the corresponding plaintexts. The attacker tries to develop a relationship between the ciphertexts and plaintexts. If he succeed in developing relationship, he would be able to guess the secret keys. The security of this cryptosystem depends upon the secret parameters as well as their arrangements. Due to vast key space, it is practically impossible to develop a relation between the plaintexts and ciphertexts. Hence the proposed encryption scheme is CPA as well as CCA secure.

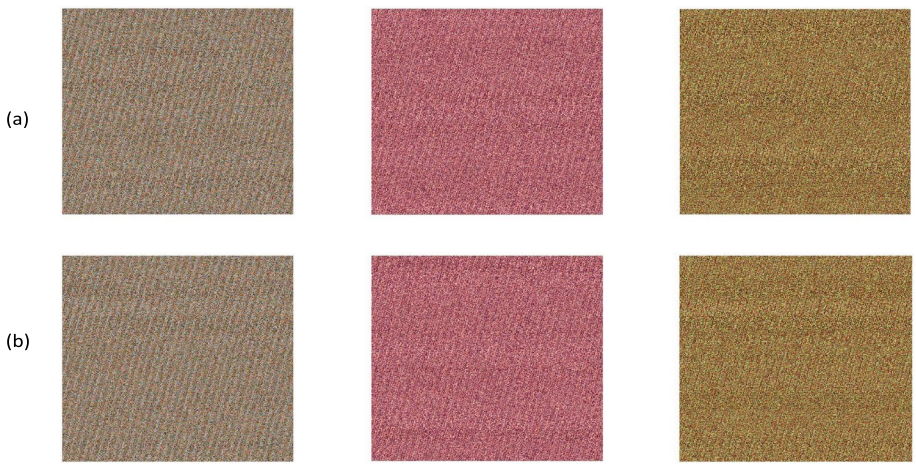

Fig. 5 Sensitivity Analysis-IV using (**a**) wrong initial guesses $x_0 = 0.4517$ and $y_0 = 0.6489$ and (**b**) wrong control parameter $\rho = 0.8375$

4.2 Statistical analysis

The statistical analysis is conducted to effectively present the results of the proposed encryption algorithm. Entropy analysis, peak signal to noise ratio (PSNR), mean square error (MSE), correlation coefficients and histogram analysis are conducted in this Section. These tests are briefly explained as given below.

4.2.1 Entropy analysis

The randomness is measured by entropy. For an input y, the entropy $H(y)$ is given by the following equation.

$$H(y) = -\sum_{x=0}^{2^n-1} p(y_x) \log_b p(y_x), \tag{4.1}$$

where logarithm of the probability $p(y_x)$ of the event y_x to the base b is taken. The base b is 2 or 10. We have computed the entropy values of original Baboon, Lena , Peppers color images Fig. 3(a), encrypted image Fig. 3(b), decrypted Baboon, Lena, Peppers color images Fig. 3(c) and are given in Table 1. If the entropy of the encrypted images is the same as the entropy of the original image, then we can say that the algorithm decrypts the images correctly. It is clear from the Table 1 that the entropy of decrypted images matches with the entropy of original images. So original images are successfully recovered after encryption. If the randomness in encrypted images is less than 8, it shows the proposed algorithm can withstand any type of attack. As the randomness in encrypted images is less than 8, the proposed algorithm is secure against any type of attack.

4.2.2 PSNR, MSE and correlation analysis

The PSNR is the ratio of the maximum power of the signal to the power of corrupting noise. Mathematically it is expressed as a logarithmic quantity in decibel scale. It is computed between the input and output image using the following mathematical equation.

$$\text{PSNR} = 10.\log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20.\log_{10}\left(\frac{MAX_I^2}{\sqrt{MSE}}\right) \tag{4.2}$$

$$= 20.\log_{10}\left(MAX_I\right) - 10.\log_{10}\left(MSE\right), \tag{4.3}$$

Table 1 Entropy Analysis

S.No.	Image	Red component	Green component	Blue component
1.	Fig. 3(a) Baboon	7.7529	7.4640	7.7733
2.	Fig. 3(a) Lena	7.2634	7.5899	6.9854
3.	Fig. 3(a) Peppers	7.3519	7.5899	7.0911
4.	Fig. 3(b)	1.0042	1.0027	1.0020
5.	Fig. 3(c) Baboon	6.7655	6.6660	6.4992
6.	Fig. 3(c) Lena	6.2967	6.5945	6.1075
7.	Fig. 3(c) Peppers	6.3924	6.6869	6.4058

Table 2 PSNR, MSE and CC of decrypted images

S.No.	Image	PSNR	MSE	CC
1.	Baboon Image	Inf	0	1
2.	Lena Image	Inf	0	1
3.	Peppers Image	Inf	0	1

Table 3 PSNR, MSE and CC of single encrypted image

S.No.	Image	PSNR	MSE	CC
1.	Baboon	-18.1422	4.2393e+06	0.0014
2.	Lena	-22.4582	1.1452e+07	-4.2710e-04
3.	Peppers	-24.5817	1.8674e+07	0.0013

where MAX_I represents the maximum pixel intensity of the image. The higher the PSNR, the better the quality of the decrypted image.

The mean squared error (MSE) is calculated between input and output image using the following equation. It tells us how how close a regression line is to a set of points.

$$\text{MSE} = \frac{1}{M \times N} \sum_{m=1}^{M} \sum_{n=1}^{N} \left[\left| \phi(m\Delta\alpha, n\Delta\beta) - \phi_0(m\Delta\alpha, n\Delta\beta) \right|^2 \right], \tag{4.4}$$

where $M \times N$ is the size of an RGB image, ϕ_0 and ϕ are input and output image, $\Delta\alpha$ and $\Delta\beta$ are the sizes of pixels. The zero MSE indicates that the error is minimum in correctly decrypted images.

The correlation coefficient (CC) is also computed between input and output image. Practically, the value of this coefficient is between -1 and 1. The value 1 indicates the strong correlation between the two images whereas -1 represents the weak correlation. Weak correlation is also known as negatively correlated. If the value is 0, it shows no relationship between the input and output image.

The PSNR, MSE and CC of the three input images, encrypted image, decrypted images, incorrect decrypted images using wrong iteration numbers $t_1 = 50, t_2 = 87$ and $t_3 = 23$, wrong decrypted images using incorrect RSA secret keys $d_r = 3547, d_g = 1255$ and $d_b = 1433$, wrong decrypted images using wrong fractions $a = 0.3982$ and $b = 0.5762$, wrong decrypted images using wrong initial guesses $x_0 = 0.4517$ and $y_0 = 0.6489$ and wrong decrypted images using wrong control parameter $\rho = 0.8375$ are given in Tables 3–8. From the Table 2, we see that PSNR is INf, MSE is 0 and CC is 1. These values prove that the decrypted image is of good quality and is same as the original image as the error is zero. Lower PSNR, Higher MSE and zero CC values in Table 3 indicates that nothing can be retrieved from the encrypted image. The values of PSNR, MSE and CC in Tables 4, 5, 6, 7, 8 prove that the decrypted image is not of good quality.

Table 4 PSNR, MSE and CC of incorrectly decrypted images with wrong iteration numbers $t_1 = 50, t_2 = 87$ and $t_3 = 23$

S.No.	Image	PSNR	MSE	CC
1.	Baboon Image	-6.3787	2.8245e+05	-0.0020
2.	Lena Image	-4.4484	1.8112e+05	0.0047
3.	Peppers Image	-5.9496	2.5589e+05	-9.6540e-04

Table 5 PSNR, MSE and CC of incorrectly decrypted images with wrong RSA secret keys $d_r = 3547, d_g = 1255$ and $d_b = 1433$

S.No.	Image	PSNR	MSE	CC
1.	Baboon Image	8.0757	1.0128e+04	0.0471
2.	Lena Image	7.9838	1.0344e+04	0.0500
3.	Peppers Image	7.9655	1.0388e+04	0.0327

Table 6 PSNR, MSE and CC of incorrectly decrypted images with wrong fractions $a = 0.3982$ and $b = 0.5762$

S.No.	Image	PSNR	MSE	CC
1.	Baboon Image	7.8472	1.0675e+04	-0.0019
2.	Lena Image	7.7470	1.0924e+04	-5.1423e-04
3.	Peppers Image	7.8591	1.0646e+04	0.0012

Table 7 PSNR, MSE and CC of incorrectly decrypted images with wrong initial guesses $x_0 = 0.4517$ and $y_0 = 0.6489$

S.No.	Image	PSNR	MSE	CC
1.	Baboon Image	7.8775	1.0601e+04	0.0030
2.	Lena Image	7.7504	1.0915e+04	-0.0011
3.	Peppers Image	7.8531	1.0660e+04	6.9752e-04

Table 8 PSNR, MSE and CC of incorrectly decrypted images with wrongm control parameter $\rho = 0.8375$

S.No.	Image	PSNR	MSE	CC
1.	Baboon Image	7.8775	1.0601e+04	0.0030
2.	Lena Image	7.7504	1.0915e+04	-0.0011
3.	Peppers Image	7.8531	1.0660e+04	6.9752e-04

4.2.3 Correlation analysis

The correlation coefficient gives an idea about the relationship betweens the adjacent pixels. The correlation coefficients are computed for each channel of an RGB image following the (4.5). Table 9 represents the correlation coefficient of red, green and blue channel of three experimental images Baboon, Lena and Peppers. The Table 9 clearly shows that the pixel values are linearly correlated. The correlation coefficients of the single encrypted image are also computed and given in Table 10. The values are approximately zero. This means pixel values do not share any relation. Lastly, correlation coefficients of decrypted images are given in Table 11. The pixel values are close to 1.

Table 9 Correlation Coefficients of original images

	Baboon			Lena			Peppers		
	R	G	B	R	G	B	R	G	B
Horizontal	0.9145	0.8852	0.9332	0.9777	0.9716	0.9504	0.9802	0.9905	0.9818
Vertical	0.8603	0.7954	0.8788	0.9869	0.9868	0.9795	0.9779	0.9934	0.9792
Diagonal	0.8431	0.7722	0.8566	0.9674	0.9666	0.9380	0.9658	0.9868	0.9594

Table 10 Correlation Coefficients of encrypted image

	Encrypted Image		
	R	G	B
Horizontal	-0.0347	-0.0366	-0.0301
Vertical	0.0137	0.0541	0.0058
Diagonal	0.0119	8.6231e-05	-0.0058

Table 11 Correlation Coefficients of decrypted images

	Baboon			Lena			Peppers		
	R	G	B	R	G	B	R	G	B
Horizontal	0.9193	0.8644	0.9092	0.9699	0.9656	0.9251	0.9593	0.9831	0.9653
Vertical	0.8374	0.7842	0.8609	0.9780	0.9760	0.9549	0.9631	0.9822	0.9631
Diagonal	0.8368	0.7520	0.8486	0.9638	0.9492	0.9150	0.9529	0.9738	0.9513

$$C = \frac{\sum_{m=1}^{M}\sum_{n=1}^{N}(v(m,n)-\overline{v})(\mu(m,n)-\overline{\mu})}{\sqrt{\left(\sum_{m=1}^{M}\sum_{n=1}^{N}(v(m,n)-\overline{v})^2\right)\left(\sum_{m=1}^{M}\sum_{n=1}^{N}(\mu(m,n)-\overline{\mu})^2\right)}}, \tag{4.5}$$

where v and μ are the input and output image, $\overline{v}$ and $\overline{\mu}$ are the mean values of input and output image respectively.

4.2.4 Histogram analysis

This statistical analysis represents the number of pixels in an image at different intensity values. The histogram of each component of an RGB image is computed. The histogram of the input Baboon colour image is given in Fig. 6(a). Figure 6(b) and (c) represent the histogram of Lena color image and Peppers color image respectively. The histogram of the single encrypted image is displayed Fig. 6(d). The histogram of the encrypted image is different from the histogram of input images Baboon, Lena and Peppers. This confirms that the proposed encryption algorithm is free from statistical attacks.

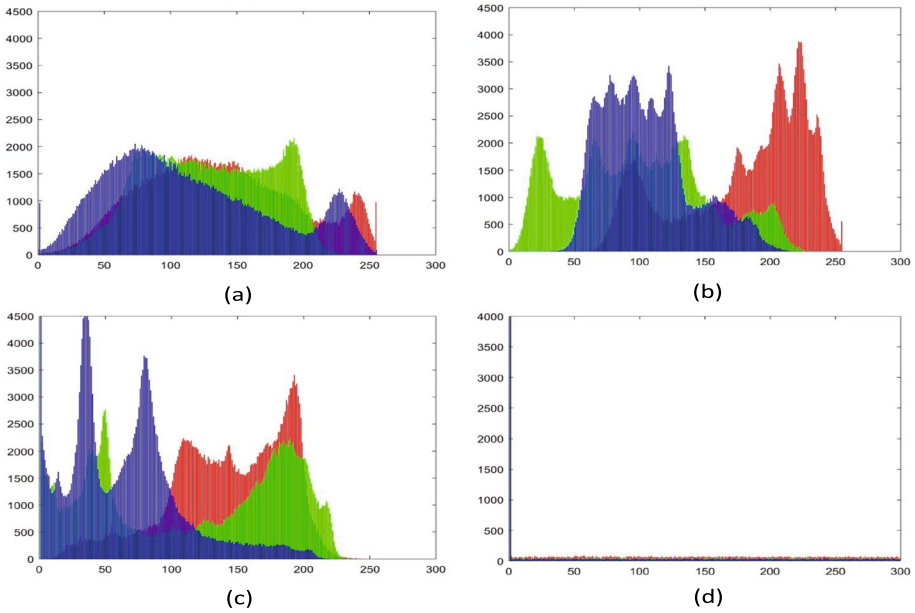

Fig. 6 Histogram analysis of (**a**) Baboon, (**b**) Lena, (**c**) Peppers and (**d**) encrypted image

4.2.5 Occlusion attack

The robustness of the technique is tested by conducting the Occlusion attack by cropping encrypted image 25% and 50% from all the four sides, i.e, left, right, top and bottom. The 25% cropped images from all the four sides are shown in Fig. 7(a), (b), (c) and (d). The three decrypted images after 25% cropping are displayed in Figs. 7(e), 8(f), (g) and 9(h). The 50% cropped images from lall the four sides are given in Fig. 10(a), (b), (c) and (d). The decrypted

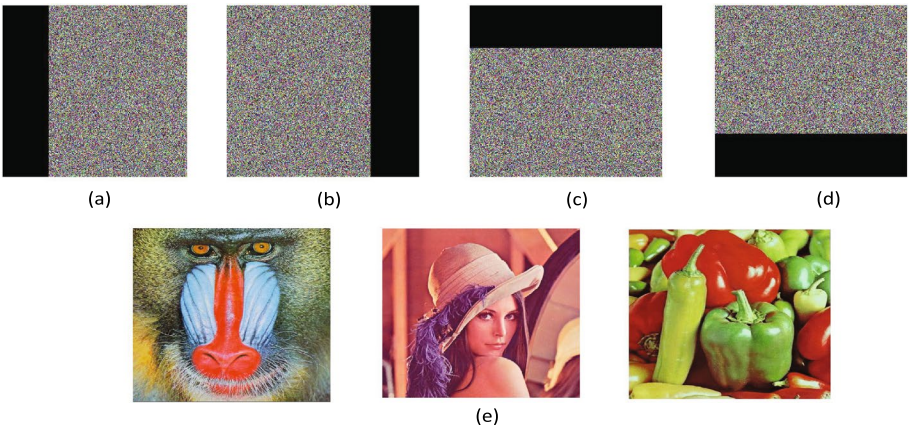

Fig. 7 (**a**), (**b**), (**c**) and (**d**) 25% Cropped Images from all the four sides (**e**) fully decrypted image corresponding to Fig. 7(a)

Fig. 8 (**f**) fully decrypted image corresponding to Fig. 7(b); (**g**) fully decrypted image corresponding to Fig. 7(c)

Fig. 9 (**h**) fully decrypted image corresponding to Fig. 7(d)

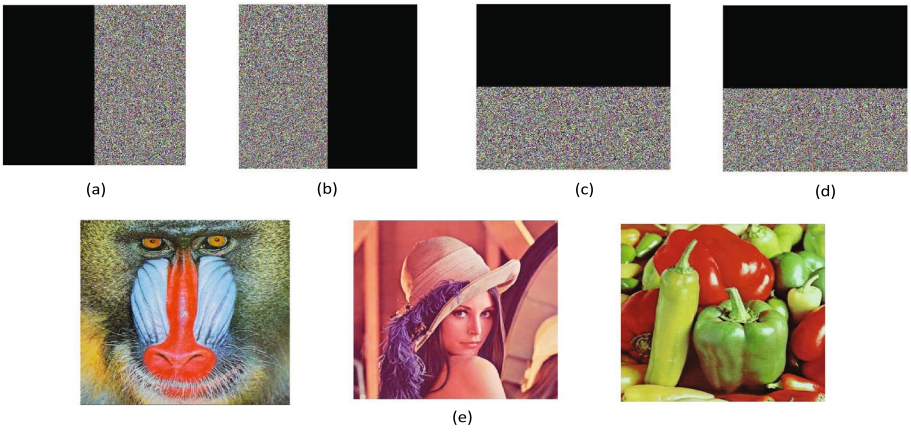

Fig. 10 (**a**), (**b**), (**c**) and (**d**) 50% Cropped Images from all the four sides (**e**) fully decrypted image corresponding to Fig. 10(a)

Fig. 11 (**f**) fully decrypted image corresponding to Fig. 10(b); (**g**) fully decrypted image corresponding to Fig. 10(c)

Fig. 12 (**h**) fully decrypted image corresponding to Fig. 10(d)

images are given in Figs. 10(e), 11(f), (g) and 12(h), respectively. It is visually clear that the proposed encryption algorithm has successfully defended the cropping attack.

5 Comparison

We have done the comparison of our proposed encryption scheme with the existing similar schemes. Several researchers have given their contribution in the field of multiple image encryption algorithm. Some key differences are illustrated in Table 12.

1. Joshi et al. [14], Wang and Zhao [34] and Liu et al. [20] have developed image encryption algorithm for multiple images. The building blocks used are double random phase and fractional Fourier transform.
2. Yong et al. [39] introduced a novel scheme using rotation multiplexing method. Wang and Zhao [35] proposed multiple image encryption technique in Fourier domain.
3. Later discrete fractional Fourier transform and Fourier transform became the epicenter of research. The image encryption algorithms in these domains are given in [29, 43]. These schemes have used chaotic maps, pixel scrambling techniques, fractional orders, and the random phase masks.

Table 12 Comparison analysis

S.No.	Authors [12, 15, 27, 30, 36, 38, 39]	Our approach
1.	Only secret keys are required to decrypt the images correctly.	Secret keys with their correct arrangements are required to decrypt the images correctly.
2.	The images would be recovered correctly if an unauthorized person possess the secret keys.	An unauthorized person would not be able to recover the images correctly if he knows the secret keys.
3.	The proposed scheme does not possess multi-layer security.	The proposed scheme possess multi-layer security.
4.	The presented encryption scheme is either secure in time or in frequency domain.	The presented encryption scheme is secure in time, frequency and coordinate domain.
5.	The proposed encryption algorithm are developed using fractional Fourier transform or Fourier transform or Gyrator transform with random phase encoding or chaotic map.	The presented encryption algorithm is proposed incorporating RSA, 3D Arnold and multiple parameter fractional discrete cosine transform.
6.	Cryptographic primitives are not used in image encryption algorithm.	RSA cryptosystem which is a powerful asymmetric key cryptosystem is used in image encryption algorithm.

4. The multiple image encryption techniques are also given in Gyrator transform domain. Shi et al. [30] proposed the scheme in geometric and frequency domain. The secret key parameters are Arnold and wavelet transform.
5. Wu et al. introduced the concept of encryption of four images together using discrete fractional Fourier transform. The fractional orders are taken as the secret keys. Zhang and Xiao [42] presented his contribution using Chirikov standard map, discrete fractional random transform and chaotic logistic maps. The security is based on pixel scrambling operation, fractional orders and random phase masks.
6. Wu et al. [36] introduced triple color image encryption scheme. The building blocks incorporated are FrDCT, Arnold transform and cyclic shift. The secret keys are iteration numbers and control parameter.
7. Wu et al. [38] proposed an image encryption algorithm incorporating random fractional discrete cosine transform with the dependent scrambling and diffusion. The proposed algorithm can withstand common classical attacks.
8. Qiu et al. [27] proposed double-image encryption algorithm using discrete fractional angular transform with fractional Fourier transform. The algorithm is applied on two grey scale images.

6 Conclusions

We have introduced multi-layer secure image encryption technique using MPFrDCT, Arnold 3D transform and RSA cryptosystem. Three images are encrypted jointly to produce a single encrypted image. The single encrypted image is easy to transfer over the public channel. It saves communication and storage complexity. In the scheme, firstly three indexed images are generated by removing the color maps. The three indexed formats are taken as the three channel, i.e., R, G and B of an RGB image. Firstly, Arnold 3D map is registered on the RGB image. Later RSA cryptosystem is enforced to encrypt each pixel of the RGB image. It is a public key cryptosystem. Lastly, MPFrDCT is incorporated on the RGB image. The decryption is just revere of these steps. The security of the scheme depend upon secret keys as well as their proper arrangements. The robustness of the scheme is tested by simulation analysis. A detailed comparison is also done with the existing similar schemes.

Funding There is no funding by any organization.

Declarations

Conflicts of interest There is no Conflicts of interests.

References

1. Abuturab MR (2012) Color image security system using double random-structured phase encoding in gyrator transform domain. Appl Opt 51(15):3006–3016
2. Abuturab MR (2012) Securing color image using discrete cosine transform in gyrator transform domain structured-phase encoding. Opt Lasers Eng 50(10):1383–1390
3. Abuturab MR (2012) Securing color information using arnold transform in gyrator transform domain. Opt Lasers Eng 50(5):772–779

4. Abuturab MR (2013) Color image security system based on discrete hartley transform in gyrator transform domain. Opt Lasers Eng 51(3):317–324
5. Abuturab MR (2013) Noise-free recovery of color information using a joint-extended gyrator transform correlator. Opt Lasers Eng 51(3):230–239
6. Antonini M, Barlaud M, Mathieu P, Daubechies I (1992) Image coding using wavelet transform. IEEE Trans Image Process 1(2):205–220
7. Cariolaro G, Erseghe T, Kraniauskas P (2002) The fractional discrete cosine transform. IEEE Trans Signal Process 50(4):902–911
8. Chen L, Zhao D (2006) Optical image encryption with hartley transforms. Opt Lett 31(23):3438–3440
9. Chen L, Zhao D (2008) Image encryption with fractional wavelet packet method. Optik-Int J Light Electron Opt 119(6):286–291
10. Hahn J, Kim H, Lee B (2006) Optical implementation of iterative fractional fourier transform algorithm. Opt Express 14(23):11103–11112
11. Hennelly B, Sheridan JT (2003) Optical image encryption by random shifting in fractional fourier domains. Opt Lett 28(4):269–271
12. Joshi AB, Kumar D, Gaffar A, Mishra D (2020) Triple color image encryption based on 2d multiple parameter fractional discrete fourier transform and 3d Arnold transform. Opt Lasers Eng 133:106139
13. Joshi AB, Kumar D, Mishra DC (2021) Security of digital images based on 3d Arnold cat map and elliptic curve. Int J Image Graph 21(01):2150006
14. Joshi M, Singh K et al (2008) Color image encryption and decryption for twin images in fractional fourier domain. Opt Commun 281(23):5713–5720
15. Kumar D, Joshi AB, Mishra VN (2020) Optical and digital double color-image encryption algorithm using 3d chaotic map and 2d-multiple parameter fractional discrete cosine transform. Results Opt 1:100031
16. Kumar M, Mishra D, Sharma R (2014) A first approach on an rgb image encryption. Opt Lasers Eng 52:27–34
17. Li H, Wang Y, Yan H, Li L, Li Q, Zhao X (2013) Double-image encryption by using chaos-based local pixel scrambling technique and gyrator transform. Opt Lasers Eng 51(12):1327–1331
18. Liu H, Nan H (2013) Color image security system using chaos-based cyclic shift and multiple-order discrete fractional cosine transform. Opt Laser Technol 50:1–7
19. Liu S, Mi Q, Zhu B (2001) Optical image encryption with multistage and multichannel fractional fourier-domain filtering. Opt Lett 26(16):1242–1244
20. Liu Z, Dai J, Sun X, Liu S (2009) Triple image encryption scheme in fractional fourier transform domains. Opt Commun 282(4):518–522
21. Liu Z, Liu S (2007) Random fractional fourier transform. Opt Lett 32(15):2088–2090
22. Liu Z, Xu L, Liu T, Chen H, Li P, Lin C, Liu S (2011) Color image encryption by using Arnold transform and color-blend operation in discrete cosine transform domains. Opt Commun 284(1):123–128
23. Liu Z, Zhang Y, Liu W, Meng F, Wu Q, Liu S (2013) Optical color image hiding scheme based on chaotic mapping and hartley transform. Opt Lasers Eng 51(8):967–972
24. Mishra D, Sharma H, Sharma R, Kumar N (2017) A first cryptosystem for security of two-dimensional data. Fractals 25(01):1750011
25. Mishra DC, Sharma R, Suman S, Prasad A (2017) Multi-layer security of color image based on chaotic system combined with rp2dfrft and arnold transform. J Inf Secur Appl 37:65–90
26. Prasad A, Kumar M, Choudhury DR (2012) Color image encoding using fractional fourier transformation associated with wavelet transformation. Opt Commun 285(6):1005–1009
27. Qiu T, Dai WH, Chen SH, Zhou H, Gong LH (2022) Double-image encryption algorithm based on discrete fractional angular transform and fractional fourier transform. Opt Appl 52(4)
28. Refregier P, Javidi B (1995) Optical image encryption based on input plane and fourier plane random encoding. Opt Lett 20(7):767–769
29. Shan M, Chang J, Zhong Z, Hao B (2012) Double image encryption based on discrete multiple-parameter fractional fourier transform and chaotic maps. Optics Commun 285(21–22):4227–4234
30. Shi X, Zhao D, Huang Y, Pan J (2013) Multiple color images encryption by triplets recombination combining the phase retrieval technique and Arnold transform. Opt Commun 306:90–98
31. Singh N, Sinha A (2009) Gyrator transform-based optical image encryption, using chaos. Opt Lasers Eng 47(5):539–546
32. Situ G, Zhang J (2005) Multiple-image encryption by wavelength multiplexing. Opt Lett 30(11):1306–1308
33. Sui L, Gao B (2013) Color image encryption based on gyrator transform and Arnold transform. Opt Laser Technol 48:530–538

34. Wang X, Zhao D (2011) Double-image self-encoding and hiding based on phase-truncated fourier transforms and phase retrieval. Opt Commun 284(19):4441–4445
35. Wang X, Zhao D (2011) Multiple-image encryption based on nonlinear amplitude-truncation and phase-truncation in fourier domain. Opti Commun 284(1):148–152
36. Wu J, Guo F, Liang Y, Zhou N (2014) Triple color images encryption algorithm based on scrambling and the reality-preserving fractional discrete cosine transform. Optik-Int J Light Electron Opt 125(16):4474–4479
37. Wu J, Luo X, Zhou N (2013) Four-image encryption method based on spectrum truncation, chaos and the modfrft. Opt Laser Technol 45:571–577
38. Wu J, Zhang M, Zhou N (2017) Image encryption scheme based on random fractional discrete cosine transform and dependent scrambling and diffusion. J Mod Opt 64(4):334–346
39. Yong-Liang X, Su X, Li S, Liu X, Zeng S (2011) Key rotation multiplexing for multiple-image optical encryption in the fresnel domain. Opt Laser Technol 43(4):889–894
40. Zhang S, Karim MA (1999) Color image encryption using double random phase encoding. Microw Opt Technol Lett 21(5):318–323
41. Zhang Y, Zheng CH, Tanno N (2002) Optical encryption based on iterative fractional fourier transform. Opt Commun 202(4–6):277–285
42. Zhang Y, Xiao D (2013) Double optical image encryption using discrete chirikov standard map and chaos-based fractional random transform. Opt Laser Technol 51(4):472–480
43. Zhong Z, Chang J, Shan M, Hao B (2012) Double image encryption using double pixel scrambling and random phase encoding. Optics Communications 285(5):584–588
44. Zhou N, Wang Y, Gong L, Chen X, Yang Y (2012) Novel color image encryption algorithm based on the reality preserving fractional mellin transform. Opt Laser Technol 44(7):2270–2281