ArMTFr: a new permutation-based image encryption scheme

Hassan Elkamchouchi

Electrical Engineering Department,
Faculty of Engineering,
Alexandria University, Egypt
Email: helkamchouchi@ieee.org

Wessam M. Salama

Department of Basic Sciences,
Pharos University, Egypt
Email: wm.hm.1986@gmail.com

Yasmine Abouelseoud*

Engineering Mathematics Department,
Faculty of Engineering,
Alexandria University, Egypt
Email: yasmine.abouelseoud@gmail.com
*Corresponding author

Abstract: In this paper, a new image encryption scheme named (ArMTFr) is proposed. An image is encrypted using a combination of keyed permutations and substitution, where a fractal is XORed with the scrambled image. Fractal images are employed in order to improve the performance of the encryption scheme from the viewpoint of randomisation and to increase the encryption key space, thus boosting its security. The employed permutations are the Arnold map and Mersenne-Twister's permutation algorithm. Before the encryption process starts, histogram equalisation is used to enhance the contrast of the image by transforming the intensity values in it, so that the histogram of the output image approximately matches a uniform histogram. First, greyscale images are considered and then the basic algorithm is extended to handle coloured images. Three representations for coloured images are considered: RGB, YCbCr and HSI colour spaces. The security of the algorithm is enhanced in this case by applying RGB colour channels multiplexing. The experimental results show that the encrypted image has low correlation coefficients among adjacent pixels and a good histogram distribution, as well as resistance to various attacks.

Keywords: correlation; image encryption; histogram equalisation; pixel permutation; Arnold map; fractals.

Reference to this paper should be made as follows: Elkamchouchi, H., Salama, W.M. and Abouelseoud, Y. (2019) 'ArMTFr: a new permutation-based image encryption scheme', *Int. J. Electronic Security and Digital Forensics*, Vol. 11, No. 1, pp.1–28.

Biographical notes: Hassan Elkamchouchi is a Professor of Communications Engineering, Electrical Engineering Department, Faculty of Engineering, Alexandria University. His research interests include wave propagation, antenna design and cryptography. He has numerous publications in respectful journals and conferences in these fields.

Wessam M. Salama is a PhD candidate at the Electrical Engineering Department, Faculty of Engineering, Alexandria University. Her research interests include communication theory and network security.

Yasmine Abouelseoud is an Associate Professor of Engineering Mathematics, Engineering Mathematics and Physics Department, Faculty of Engineering, Alexandria University. Her research interests include optimisation theory and cryptography.

1 Introduction

Security is a major concern in any communications network. Data is transmitted over an open network and thus needs to be protected. Encryption mechanisms are used to ensure confidentiality of the transmitted data and preventing unauthorised access from eavesdroppers. In such mechanisms, a secret piece of information, known as private key, is used to encode data into an unreadable form without the knowledge of this secret piece of information. Among the desirable properties of an encryption algorithm are computational efficiency, a large key space to prohibit exhaustive key space search attack, and resistance to other types of attacks such as known plaintext and chosen plaintext attacks. Steganography is another approach for achieving privacy or confidentiality. In steganography, the original data are embedded into a cover medium, such as images, to hide their existence (Manoj, 2010).

The exchanged information over the network may be in the form of text or multimedia. Securing multimedia data is more challenging compared to text data. Digital images encryption is the main concern of this paper. Digital images representations show redundancy which can be exploited by attackers. Moreover, adjacent pixels are highly correlated and the distribution of the colour levels is not uniform. Furthermore, data contained in a digital is massive and requires efficient algorithms for their encryption (Abd-El-Hafiz et al., 2014). Thus, compression before or after encryption has been investigated by several researchers (Khan and Shah, 2014; Wu and Kuo, 2005; Johnson et al., 2004).

The desirable features of an image encryption scheme include destroying the inherent correlation between adjacent pixels, producing a cipher image with a uniform histogram and sensitivity to slight changes in the plain image and slight changes in the key employed (Li, 2005). In literature, images are encrypted using random pixel positions permutations (Huang, 2012) or using suitable pixel values transformations (Pareek et al., 2011), or even a combination of both (Abd-El-Hafiz et al., 2014; Li et al., 2013; Chen et al., 2015).

This paper presents a new method for image encryption, which combines the use of pixel permutations with fractal images. The building blocks of the proposed scheme are not novel, yet the developed combination is unique. Though the proposed algorithm is simple, however, its performance is very promising with regard to security and computational efficiency. The encrypted images are analysed using statistical measures and differential attack measures. The obtained results show that the proposed scheme successfully meets various requirements of an image encryption scheme and is competitive to other schemes in literature. Both greyscale images and coloured images are considered in this paper.

The rest of the paper is organised as follows. Section 2 reviews some related image encryption schemes in literature. Section 3 presents an overview of the tools used in the proposed scheme. In Section 4, measures for evaluating the robustness of an image encryption scheme are listed. Different security attack models are defined in Section 5. The encryption and decryption modules of the proposed image encryption scheme are described in Section 6. Additionally, key space analysis and preliminary security analysis of the proposed algorithm are provided in this section. Section 7 provides the results of implementing our algorithm on greyscale images. Moreover, an extension of our scheme to coloured images along with the associated results is given in Section 8. Finally, the paper is concluded in Section 9.

2 Related work

There are several classifications for image encryption algorithms. They may be classified as spatial domain techniques (Zhang and Xiao, 2014) or frequency domain techniques. In the second class of techniques, both the finite field cosine transform (FFCT) (Lima et al., 2013; Mikhail et al., 2017), and the discrete fractional random transform (DFRNT) (Guo et al., 2010) have been successfully applied in image encryption. Spatial domain techniques are generally more computationally efficient compared to frequency domain techniques. However, frequency domain techniques provide better levels of security. Some researchers prefer working in a hybrid domain (Yu et al., 2010; El-Latif et al., 2012). This is in an attempt to achieve secure and more computationally efficient algorithms.

Moreover, image encryption techniques can be divided into three categories: pixel position permutation based techniques, pixel value substitution-based techniques and visual transformation techniques. In the first category, the pixel positions of the original image are shuffled according to a re-shared key and a keyed permutation (Zhang et al., 2004). In the second category, the pixel values (or colour levels) are replaced by other values according to some keyed transformation agreed upon between the communicating parties before transmission. In the last category, an image is used as a key or watermarking techniques are employed in encryption (Tiwari et al., 2013).

Yet, another possible classification is one that discriminates between chaotic-based techniques and non-chaotic based techniques. Recently, chaotic maps have been extensively used for image encryption (Corrochano et al., 2005; Cao, 2013) and steganography (Kumar et al., 2015; Ghebleh and Kanso, 2014). Chaotic maps are characterised by sensitivity to slight changes in their parameters and initial seed.

An image enciphering scheme is either a stream cipher or block cipher. In block ciphers, the plaintext is divided into blocks of a fixed size and the encryption function is applied to each block (Mikhail et al., 2017; Ahmad and Alam, 2009). Various modes of chaining the blocks can be used to achieve different tradeoffs among error propagation, security and parallel processing of the blocks. Variants of the advanced encryption standard (AES) have been introduced in literature to suit image encryption (Kamali et al., 2010). Common stream ciphers, like RC4, have been also employed in image encryption in Sasidharan and Philip (2011). A combination of both block ciphering and stream ciphering has also been investigated in Wang et al. (2011).

The proposed scheme is a spatial domain technique, which employs both pixel position transformations as well as pixel value transformation.

3 Background

In this section, the basic tools used as building blocks of the proposed scheme are described. Moreover, different colour spaces for digital images representations are reviewed.

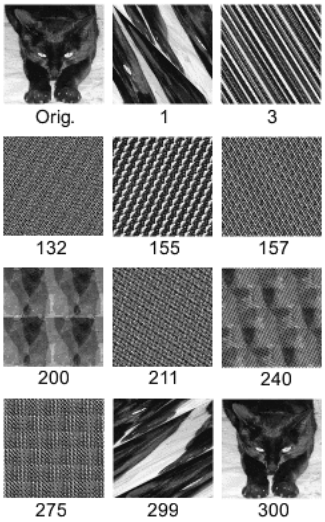
3.1 Arnold cat map

The Arnold cat map is a two-dimensional invertible chaotic map, which is used to shuffle the pixel positions of the plain image (Mishra and Mankar, 2013; Krishnamoorthi and Murali, 2012). Assume the dimension of the original image to be N × N. The coordinates of the pixels constitute the set S = {(x, y) | x, y = 0, 1, 2, …, N – 1}. The 2D Arnold cat map can be described as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \bmod(N) \tag{1}$$

where p and q are positive integers. It is noteworthy that the determinant of the transformation matrix is unity, which simplifies the inversion process. The ordered pair (x′, y′) represents the new position of the pixel originally located at position (x, y) when the Arnold transformation is applied once. The result after applying the Arnold cat map for a number of iterations R will be a shuffled image that contains all of the same pixel values of the original image. Thus, the Arnold cat map parameters are p, q, and the number of iterations R, which can all be used as components of the secret key. It is very efficient to shuffle the pixel positions using the Arnold cat map as it is a linear transformation. After several iterations, the correlation among the adjacent pixels can be destroyed completely. It is noteworthy that the Arnold cat map if applied repeatedly, after a certain number of iterations (C) the original image re-appears, as shown in Figure 1. On the receiving side, to restore the original image, the Arnold map is applied (C – R) times. Clearly, the Arnold map cannot be used solely for secure image encryption (Krishnamoorthi and Murali, 2012).

Figure 1 Results of applying successive iterations of Arnold's map, with p = 1, q = 1, on a picture of 150 × 150 pixels



Note: After 300 iterations, the original image is restored.

3.2 Mersenne-Twister (MT) algorithm

Each column of the image is permutated using a random key. The algorithm used to generate the pseudo-random sequence for re-ordering the pixels is the well-known MT algorithm by Matsumoto and Nishimura (1998). MT is a pseudo random number generator (PRNG) satisfying all the requirements to be certified as an efficient PRNG. It is fast and provides very high-quality pseudorandom numbers with a long period length. The period length is chosen to be a Mersenne prime. MT generates a periodic sequence of w-bit integers that can be regarded as uniform pseudorandom integers between 0 and $2^w - 1$, where w is the word size used. There are basically two standard word sizes employed by the MT algorithm, which are 32-bits and 64-bits. If the required sequence involves numbers of only v bits, v < w, then the leading v bits of each number in the generated sequence are considered and the rest are discarded. CryptMT is a stream cipher that employs MT and has been considered to be cryptographically secure (Matsumoto et al., 2007). It is computationally efficient as all calculations involved are bit shifts, bitwise XOR, bitwise AND and bitwise OR operations.

MT proceeds in two phases: a recurring phase and a tempering phase. The MT algorithm involves several parameters. These parameters include the degree of the recurrence relation (α) used in its definition, and correspondingly the initial seed sequence consisting of $x_0, x_1, \ldots, x_{\alpha-1}$. Moreover, its parameters include an offset used in the recurrence relation (β) and two w × w matrices; one used in the recurring phase and the second in the tempering phase (Matsumoto and Nishimura, 1998). There are standard values for all of the parameters of the MT algorithm, together with suitable initialisation procedures, thus leaving only the initial seed x0 as an input parameter to be fed to the algorithm.

3.3 Fractal images

Fractal images can be generated by iterating a mathematical equation or function for a finite number of times (Addison, 2002). Fractals typically show self-similar patterns, where self-similar refers to the fact that their shapes are repeated at different scales. Fractals may be exactly the same at every scale, or, they may be nearly the same at different scales. Figure 2 gives some examples of fractal images. Due to the frequent use of fractal images in several fields, there are several online applications for their generation, for example http://www.chaospro.de/.

Figure 2 Samples of fractal images

One common technique of fractals generation is the iterated function system (IFS) technique (Barnsley and Demko, 1985). It is a three stages process. The first stage is an initialisation stage. In the second stage, the iteration formula is applied repeatedly until a termination criterion is met. Finally, the last stage involves post-processing on the resulting vector (pixel rendering).

The Mandelbrot set is one possible choice for fractals generation. This is the set of complex numbers C, for which the quadratic function $f(z) = z^2 + C$ does not diverge when iterated from $z = 0$. Mandelbrot set images are created by sampling from the set of complex numbers and determining for each sample point C whether the result of iterating the above function diverges or not. Treating the real and imaginary parts of C as image coordinates, pixels are then coloured according to how rapidly the sequence $z^2_n + C$ diverges, with the colour 0 (black) usually used for points where the sequence converges (Crilly et al., 2012). Alternatively, C may be held fixed and the initial value z_0 is variable instead.

Fractals have found several applications in cryptography. Fractals have been used for pseudo-random numbers generation in Abd-El-Haleem et al. (2013). Moreover, fractals have been used in key stream generation for image encryption in Abd-El-Hafiz et al. (2014) and Mikhail et al. (2017). Furthermore, fractals have been employed by researchers in data hiding (El-Khamy et al., 2008).

3.4 Histogram equalisation

An image histogram depicts the frequency distribution of the colours in the image. The histogram is plotted by examining all pixels in the image and assigning each to a bin depending on the pixel colour level. The height of a bin reflects the frequency of pixels assigned to it. The number of bins, in which the whole colour levels range is divided, is usually in the order of the square root of the number of pixels in the image or simply equal to the number of colour levels used for the image representation (Pizer et al., 1987). An image histogram is an important tool for characterising an image. For instance, the abundance of the blue colour in the histogram of an image reflects the existence of the sky or sea in the scene.

An efficient image encryption scheme should produce a cipher image with a uniform histogram. In this paper, histogram equalisation is used to improve the contrast in an image to make it clearer than the original image. The pixels of an image seem clustered around the middle of the available range of colour levels and histogram equalisation aims to broaden this range.

Histogram equalisation refers to mapping one distribution for colour levels to another distribution which is wider and more uniform distribution spread over the whole range of colour levels. This procedure reduces to finding a suitable transformation T, which maps the gray levels in the original image to their new values in the equalised image.

Let L denote the number of possible colour levels, usually 256, and let p_k denote the proportion of pixels whose colour level is k in the original image. According to Pizer et al. (1987), the required transformation T for equalising the histogram is given by

$$T(k) = \text{floor}\left((L-1)\sum_{i=0}^{k} p_i \right) \tag{2}$$

3.5 Colour information representations

Greyscale images are represented as a two-dimensional array of pixels, whose values range from 0 (black) to 255 (white). On the other hand, coloured images are traditionally represented as three layers/planes of two-dimensional arrays. These three layers (RGB) give the red colour level, the green colour level and the blue colour level, respectively. However, this representation suffers from inherent redundancy. Moreover, medical research proved that the human eye shows different sensitivity to colour and brightness. Consequently, the YCbCr representation of coloured images has been introduced in literature (Jin et al., 2017). The image is decomposed again into three components: Y-component which represents the luminance; the Cb-component which gives the blue chrominance; and Cr-component which gives the red chrominance. The Y or luminance component resembles the greyscale version of the original image. The Cb-component emphasises on bluish objects and the Cr-component emphasises on reddish objects.

The equations used for converting an image from the RGB colour model to the YCbCr colour space are given below (Jin et al., 2017):

$$Y = 0.299\,R + 0.587\,G + 0.114\,B \tag{3}$$

$$Cr = R - Y \tag{4}$$

$$Cb = B - Y \tag{5}$$

Another commonly used colour model is hue-saturation-intensity (HSI) model. It is quite similar to how humans describe and interpret colours. Hue represents the dominant colour as perceived by an observer, while saturation represents the relative purity or amount of white light mixed with the dominant colour (hue). Intensity is related to brightness.

The HSI model decouples the intensity from colour-carrying information (hue and saturation) in a colour image. The equations used for transforming an image from RGB model to HSI space are given by Alvarado-Robles et al. (2017):

$$\mathrm{I} = \frac{(\mathrm{R}+\mathrm{G}+\mathrm{B})}{3}, \tag{6}$$

$$\mathrm{S} = 1 - \frac{3}{(\mathrm{R}+\mathrm{G}+\mathrm{B})}[\min(R,G,B)], \quad \text{if} \quad \mathrm{I} \neq 0 \tag{7}$$

$$\mathrm{H} = \cos^{-1}\left\{\frac{0.5[(\mathrm{R}-\mathrm{G})+(\mathrm{R}-\mathrm{B})]}{\sqrt{(\mathrm{R}-\mathrm{G})^2+(\mathrm{R}-\mathrm{B})(\mathrm{G}-\mathrm{B})}}\right\}, \text{if } \mathrm{S} \neq 0. \tag{8}$$

It is noteworthy that the hue component is not defined if the saturation component S = 0. On the other hand, the saturation component is not defined if the intensity component equals zero. The hue component (H) is normalised as H = H/360°. If H has a negative value, then add 360° to its value before normalisation.

4 Qualitative measures and quantitative measures for evaluation of an image encryption scheme

Evaluating the quality of the results of applying an encryption scheme is very important as it reflects the weaknesses and the good aspects in the cryptosystem employed. When dealing with image encryption, one should be aware that specialised tests are needed to judge the quality of the encrypted image and this fact arises from the nature of images that are more challenging compared to text data (Abd-El-Hafiz et al., 2014; Mikhail et al., 2017).

There are qualitative measures for evaluating the quality of an encrypted image. Visually, the encrypted image should be as distinct as possible from the original plain image and thus conveying no information to eavesdroppers. Moreover, the decrypted image should reflect sensitivity to slight variations in the key. This means that if a wrong key is used which is different from the pre-shared key, the resulting image from decryption should be visually different from the original plain image.

The histogram of the encrypted image should be flat; that is, all colour levels are present in the enciphered image with equal frequencies. Consequently, the plain image features are completely hidden. Moreover, in the encrypted image, the scatter diagram for horizontally adjacent pixels and that for vertically adjacent pixels should reflect weak correlation among them.

The quantitative measures can be divided into two main categories: statistical measures and differential measures. These are elaborated on in the following subsections.

4.1 Statistical measures

These measures are used to give numerical estimates for the correlation among adjacent pixels (horizontally adjacent pixels, vertically adjacent pixels and diagonally adjacent pixels), and the flatness of the histogram.

4.1.1 Correlation coefficient

In the plain image, neighbouring pixels are highly correlated to each other; thus, one of the encryption goals is to destroy this correlation (Abd-El-Hafiz et al., 2014).

The correlation coefficient ρ between two vectors X and Y is computed as follows:

$$\mathrm{cov}(X, Y) = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - \frac{1}{N}\sum_{j=1}^{N} x_j\right)\left(y_i - \frac{1}{N}\sum_{j=1}^{N} y_j\right) \tag{9}$$

$$S^2(X) = \frac{1}{N-1}\sum_{i=1}^{N}\left(x_i - \frac{1}{N}\sum_{j=1}^{N} x_j\right)^2 \tag{10}$$

$$\rho = \mathrm{cov}(X, Y) / \sqrt{S^2(X)S^2(Y)} \tag{11}$$

4.1.2 Entropy

The entropy value for a sequence is a measure of the randomness or uncertainty in this sequence. In digital images, the distribution of colour levels is not uniform. Thus, original images are likely to have low entropy values. On the other hand, an encrypted image should appear as a random sequence minimising predictability. This means that for a good cipher image the entropy value should be high. The entropy of a sequence of elements S_i, represented in 8-bits (pixel colour value) with the corresponding probability $p(S_i)$, is calculated as

$$\mathrm{Entropy} = -\sum_{i=1}^{2^8} p(S_i)\log_2 p(S_i) \tag{12}$$

This measure is used to quantify the flatness of the histogram, instead of just relying on visual testing, which is somewhat subjective.

4.2 Differential measures

A secure image encryption scheme should be sensitive to small variations in the plain image and small changes in the encryption key. In what follows, some of the differential measures employed in the analysis of the results obtained using the proposed scheme are defined.

4.2.1 The mean absolute error

This measure gives the average absolute change in colour levels between the encrypted image and the original image (Lima et al., 2013).

$$\mathrm{MAE} = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}\left|P(i, j) - E(i, j)\right|, \tag{13}$$

where P(i, j) and E(i, j) denote the colour levels of pixel (i, j) in the plain image and encrypted image, respectively. Moreover, M and N stand for height (number of rows) and the width (number of columns) of the image under consideration, respectively.

4.2.2 The mean squared error

The mean squared error (MSE) between two images, $I_1(i, j)$ and $I_2(i, j)$ is given by:

$$\mathrm{MSE} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left[I_1(i,j) - I_1(i,j)\right]^2}{M \times N}. \tag{14}$$

This measure can be used as an estimate for the error resulting due to decrypting with a wrong key, which differs in one bit from the valid key, or decrypting a noisy encrypted image.

4.2.3 The number of pixels change rate

The number of pixels change rate (NPCR) is used to measure the percentage of pixels that are different between two encrypted images whose corresponding plain images are identical except for only one pixel and it is computed as (Wu et al., 2011):

$$D(i,j) = \begin{cases} 0, & E_1(i,j) = E_2(i,j) \\ 1, & E_1(i,j) \neq E_2(i,j) \end{cases}, \tag{15}$$

$$\mathrm{NPCR} = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N} D(i,j) \times 100\%. \tag{16}$$

where E_1 is the encrypted version of the original image and E_2 is the encrypted image of an image that differs in one pixel from the plain image.

4.2.4 The unified average changing intensity

This measure represents the average of the absolute differences between two encrypted images provided that their corresponding two plain images are identical except for only one pixel and it is calculated according to the following equation (Wu et al., 2011):

$$\mathrm{UACI} = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}\left|E_1(i,j) - E_2(i,j)\right| / 255 \times 100\%. \tag{17}$$

5 Security attack models

Some of the attack models are defined in this section, which are later considered in our security analysis of the proposed scheme. These include brute-force attack, known-plaintext and chosen-plaintext attacks (CPAs) mounted to reveal the secret parameters of the algorithm. Moreover, noise attacks are also defined.

5.1 Brute-force attack

In this attack, the intruder tries all possible keys to decrypt the cipher image until a meaningful image appears. If the attacker succeeds in its trials, the system key is no longer a secret and thus the system security is fully jeopardised.

5.2 Known-plaintext attack

The information available to the attacker in this kind is at least one sample of both the plaintext and the corresponding ciphertext, which can be available to it through eavesdropping. The attacker aims to break the algorithm by exploiting this information (Stallings, 2005).

5.3 Chosen-plaintext attack

The attacker feeds a specific plaintext of its choice to the encryption mechanism. The choice of the plaintext is made in such a way that can help the attacker recover the encryption key (Stallings, 2005).

5.4 Additive noise and cropping attacks

An attacker can intercept an encrypted image and modify it, while the authorised recipient receives it and tries to decrypt it. This type of attack is against the integrity of the received message. A successful image encryption scheme should enable the receiver to detect that the encrypted image has been tampered with during transmission.

In the additive noise attack, random noise is added to the intercepted image by the attacker. The salt and pepper noise is applied in our analysis of the security of the proposed scheme. On the other hand, in the cropping attack, one or more areas of the encrypted image are deleted or cropped (Loukhaoukha et al., 2012).

6 The proposed image encryption scheme (ArMTFr)

In this section, the proposed image encryption scheme for greyscale images is described. The two communicating parties should agree on the system key composed of the following items:

- Arnold map parameters p, q, R, C.
- A seed x_0 for the MT-algorithm.
- The index of a fractal image chosen from a pre-shared database of fractals.

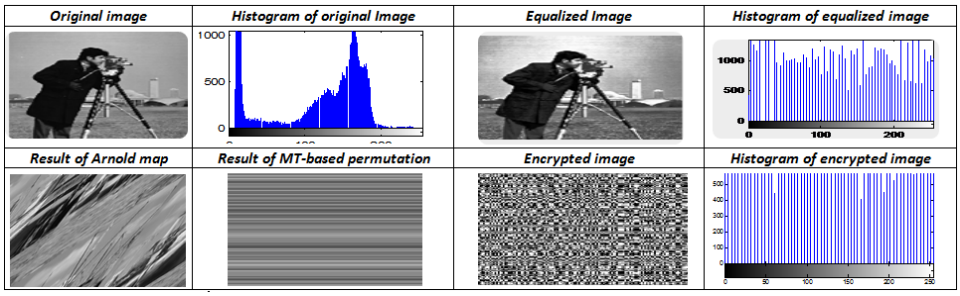
6.1 Encryption algorithm steps

The basic idea of the proposed scheme is to use simple and computationally efficient building blocks in such a way that the combination introduced is both practical with regard to time and capable of achieving an appropriate level of security.

On the sender's side, first, histogram equalisation is applied to the original greyscale image. The aim of this step is to enhance the contrast in the image, as shown in Figure 3. The Arnold cat map is applied to the equalised image for R times. This step produces a scrambled image. However, since the Arnold map is known to possess weak security, the MT algorithm is applied to the columns of the image to obtain random permutations of the pixel values in each column. MT algorithm is known to produce high quality pseudo-random sequences with several adjustable parameters, broadening the key space

and thus combating brute-force attacks. Finally, bitwise exclusive-OR is carried between the image resulting from the MT-based permutation step and the fractal image (F) agreed upon between the two communicating parties. This step further enlarges the key space and additionally aids in producing more uniform histograms for the encrypted image.

Figure 3 Histogram analysis of Cameraman, both the original and encrypted images (see online version for colours)



These steps are summarised in what follows.

a Load the original image (I_m) of size N × N to be encrypted.

b Apply histogram equalisation to I_m to obtain a contrast-enhanced image, (I_{eq}).

c Apply Arnold map iteratively to scramble the pixels of the equalised image (I_{eq}) according to a given key component that specifies the number of iterations, denoted as 'R'. The pixel value in position (x, y) in the plaintext image is assigned to the pixel in position (x′, y′) determined by the map. The result of this step is an image (I_{Ar}).

d For each column in I_{Ar}, random pixels permutation is then applied based on Mersenne-Twister algorithm (I_{MT}).

e A fractal image is XORed with the image resulting from the previous step (I_{MT}) to obtain the final cipher image, I_{Fr}.

6.2 *Decryption algorithm steps*

Upon receiving the encrypted image (I_{Fr}), the recipient recovers the image (I_{MT}) by applying a bitwise XOR operation between the received image and the fractal indicated in the pre-shared key. The inverse permutation obtained using the MT-algorithm is then applied. Finally, the Arnold map is applied (C – R) times to restore the original image, given that the integrity of the encrypted image is preserved.

The decryption process is summarised below.

a $I_{MT} \leftarrow I_{Fr} \oplus F$ where ⊕ denotes bitwise exclusive OR operator applied on each pixel.

b Apply the inverse of the permutation obtained by the MT algorithm.

c Apply the Arnold map (C – R) times.

6.3 Security analysis

In what follows, some security aspects of the proposed ArMTFr scheme are investigated.

6.3.1 Key space analysis

Recall that the image size is $N \times N$. Now, let $N = 2^n$. The system key components and their sizes are given below:

1 p is n bits in length

2 q is n bits in length

3 R takes n bits of storage.

4 MT PRNG initial seed x_0 which is w-bits in length

5 index of fractal of ℓ bits, indicating which of the available 2^ℓ fractals is chosen as a key component.

The total key size is given by $3\,n + w + \ell$. For fairly large images ($1{,}024 \times 1{,}024$), and $w = 64$ bits, to achieve desirable level of security, ℓ should be in the order of 8 bits. To save storage required for the database of fractals, the Mandelbrot set can be employed and thus the corresponding generators of the fractals are only stored.

6.3.2 Resistance to KPA

From the above description of the proposed scheme, it is clear that given a pair (P, C) consisting of the plain image P and the cipher image C, no information is conveyed to the eavesdropper. This is because the attacker has no access to the key components involved in the two permutations applied to the original image as well as the fractal image. Therefore, it resists known-plaintext attacks (KPAs).

6.3.3 Resistance to CPA

It is apparent that a pure black plain image fed to the proposed scheme reveals the fractal part of the key, however, the rest of the key components remain secure. Thus, the algorithm should reject to encrypt such a plain image and consequently it is resistant to CPAs as well.

6.4 Computational time complexity

The three key steps in the proposed encryption process are: iterative application of Arnold map, MT-based pixels permutations on the image columns and bitwise XOR operation with a fractal image.

The Arnold map is applied R times for each of the image pixels and therefore this step is $O(N^2)$. The generation of a single pseudo-random number using MT-algorithm is $O(1)$ and N^2 of these pseudo-random numbers are involved in the second step of the algorithm. Finally, the bitwise XOR operation with a fractal image is also $O(N^2)$. Thus, the whole algorithm is $O(N^2)$.

7 Experimental results and security assessment

We applied our proposed algorithm (ArMTFr) to a sample of 1,024 × 1,024 standard images available from the USC-SIPI Image Database (1997), including Cameraman, Mandrill, Boat, and Peppers. For the fractals, Mandelbrot fractals are readily available from the internet. The performance of the proposed scheme is evaluated using the measures listed in Section 4. Moreover, its computational efficiency is investigated.

7.1 Visual testing

As apparent in Figures 3, 4, 5 and 6, the histogram of ciphered images is flat with all colour levels being equally likely. Moreover, the scatter diagrams shown in Figure 7 demonstrate that in the original image adjacent pixels along the vertical, horizontal and diagonal directions are positively correlated. On the other hand, adjacent pixels in the encrypted image seem uncorrelated.

Figure 4 Histogram analysis of Boat original and encrypted images (see online version for colours)

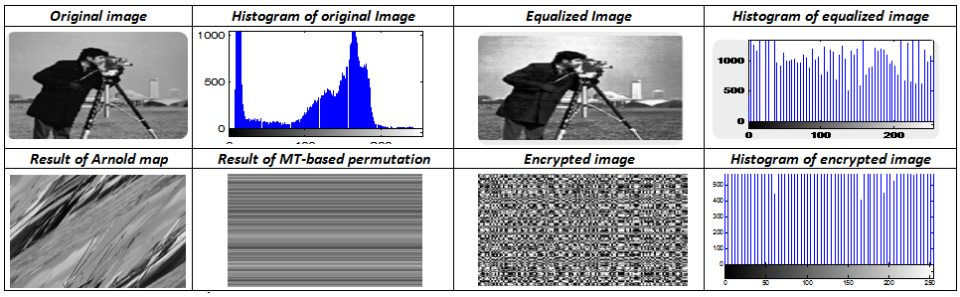


Figure 5 Histogram analysis of Mandrill original and encrypted images (see online version for colours)

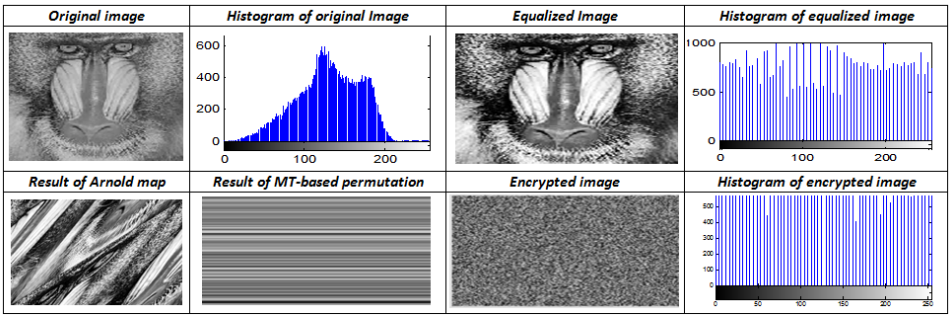


Figure 6 Histogram analysis of Peppers original and encrypted images (see online version for colours)

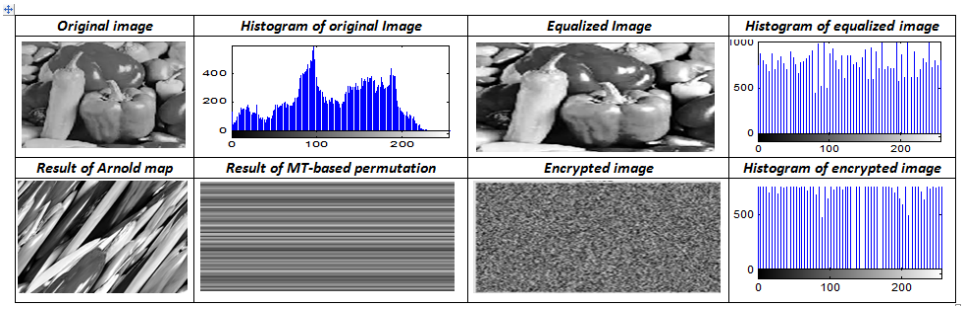


Figure 7 Scatter diagrams for adjacent pixels in original and encrypted images of Cameraman image (see online version for colours)

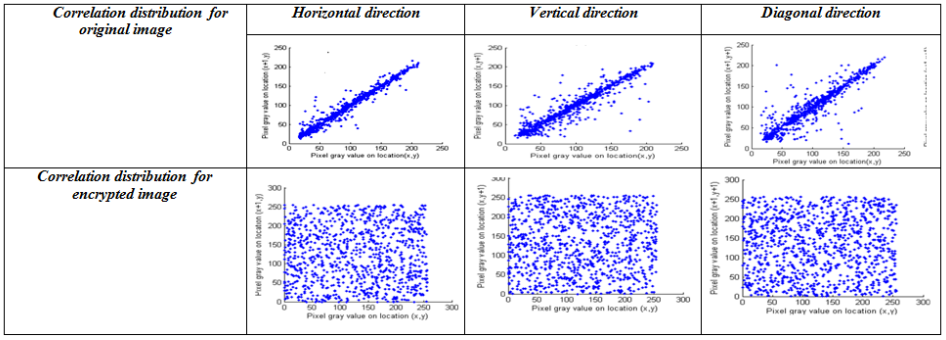
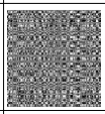
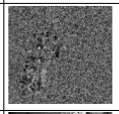
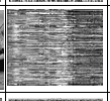
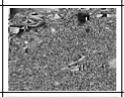
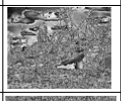
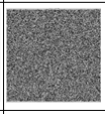
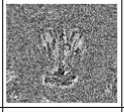
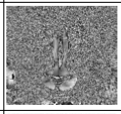
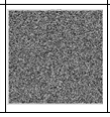
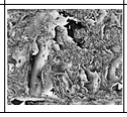
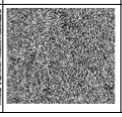


Figure 8 Differential measures for some decrypted images using wrong keys

Original image	Encrypted image with K_1	Decrypted image with correct key K_1	Decrypted image with wrong key K_2	Decrypted image with wrong key K_3	Differential measures Between plain image and decrypted with K_2			Differential measures Between plain image and decrypted with K_3		
					MSE	NPCR	UACI	MSE	NPCR	UACI
					2.49×10^3	99.89	45.96	2.53×10^3	98.99	46.66
					2.61×10^3	97.72	46.88	2.69×10^3	98.88	46.98
					2.49×10^3	99.44	42.79	2.51×10^3	99.87	43.75
					2.88×10^3	98.96	36.66	2.91×10^3	99.48	34.99

Regarding sensitivity to slight variations in the system key (K_1), two tests have been applied. We tried decryption using a wrong key (K_2), which differs in one bit in the fractal index component of the system key and another wrong key (K_3), which differs only in one bit in the Arnold map parameters component of (K_1). The obtained results for several test images are shown in Figure 8. It is apparent that decryption using either of the two wrong keys yields a totally distorted image.

Figure 9 Decrypted images under salt and pepper noise attack

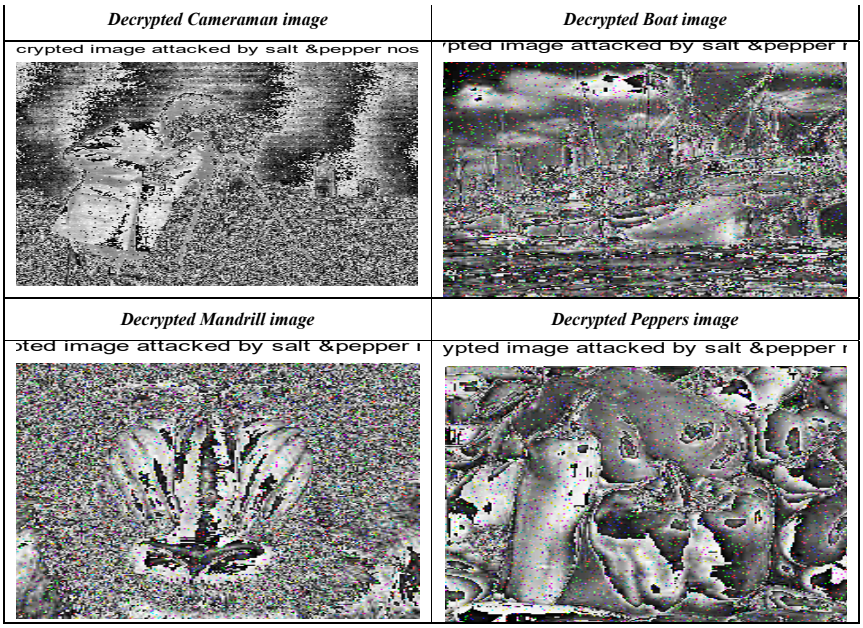
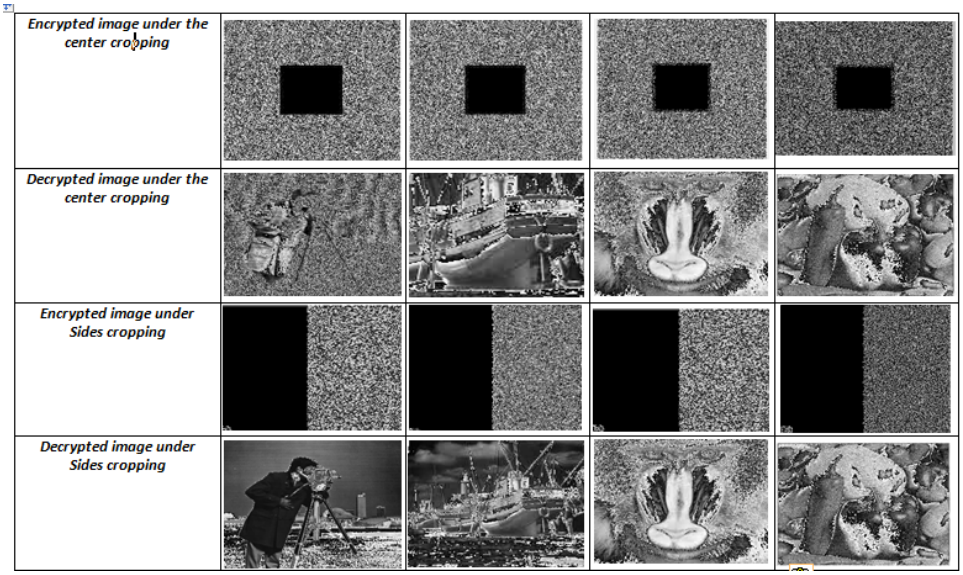


Figure 10 Encrypted images under cropping attack in centre and sides (50% cropping)



Salt and pepper noise attack has been applied to some encrypted images with density 0.05 and the results are shown in Figure 9. The decrypted images are quite corrupted and thus the receiver detects that the received message has been tampered with during transmission. The cropping attack is also detectable as can be seen in Figure 10.

7.2 Numerical measures

In Table 1, the correlation among pixels in the original (plain) images is high as expected in vertical and horizontal directions, as well as along the diagonal. On the other hand, the correlation coefficients along the three directions have decreased in the encrypted image and they are approximately equal to zero as recommended for a secure image encryption mechanism.

Table 1 Correlation coefficients of some 1,024 × 1,024 original and encrypted test images

Image (file name)	*Original image*			*Encrypted image*		
	Horz.	*Vert.*	*Diag.*	*Horz.*	*Vert.*	*Diag.*
Cameraman	0.9936	0.9951	0.9335	–0.0005	–0.0244	–0.0009
Boat	0.9664	0.9623	0.9493	0.0004	0.0564	–0.0359
Peppers	0.9606	0.9815	0.9483	–0.0076	–0.0086	0.1222
Mandrill	0.9073	0.8388	0.9459	0.0276	–0.0236	0.0444

Table 2 Differential tests results and entropy for test images

Image	*MAE*	*NPCR*	*UACI*	*Entropy*	
				Orig.	*Enc.*
Cameraman	89.881	99.98	46.15	7.0097	7.8912
Boat	80.621	96.62	47.48	7.3124	7.8883
Peppers	88.22	99.51	43.01	7.0021	7.8882
Mandrill	57.498	98.62	33.44	7.7061	7.8884

Table 2 shows the entropy results and the obtained values for some differential measures. The entropy increases and approaches its best value which is eight, as suggested by the flat histograms. This proves the increase in uncertainty and randomness in the encrypted images. The obtained values for MAE indicate that the average absolute difference between pixel values in the original image and the encrypted image is quite large as desired. Thus, the encrypted image reveals almost no information about the plain image. It is apparent from the high values of unified average changing intensity (UACI) and NPCR that the proposed scheme is sensitive to small changes in the plain image.

In Figure 8, it is noteworthy that the UACI and NPCR numeric estimates are calculated to measure the differences between the decrypted image and the original image under a wrong key. The high MSE, UACI and NPCR values obtained for all test images indicate that the proposed scheme is sensitive to slight changes in the system key.

Table 3 MSE between original images and the decrypted forms under salt and pepper noise attack

Image	*Density*	*MSE*
Cameraman	0.05	2.37×10^3
Peppers	0.05	2.54×10^3
Boat	0.05	2.40×10^3
Mandrill	0.05	2.88×10^3

In Table 3, the results obtained under the salt and pepper noise attack are shown. The high MSE values between the original images and the decrypted versions under this attack indicate that the proposed scheme withstands this type of attack. The fact that the integrity of the received image is not maintained is detectable.

Table 4 MSE between original images and the decrypted versions under cropping attack

Image	*Centre cropping*	*Sides cropping*
Cameraman	2.88×10^3	2.82×10^3
Peppers	2.84×10^3	2.78×10^3
Boat	2.97×10^3	2.97×10^3
Mandrill	2.09×10^3	2.03×10^3

Table 4 gives the MSE values between the original images and the decrypted under cropping attack, either in their centres or on the image sides. The results shown indicate that the MSE is large enough to make the cropping attack detectable at the receiver's side.

Table 5 Comparison between the proposed scheme and other schemes in literature

Scheme	*Encrypted image corr.*			*UACI*	*Entropy*	*NPCR*
	Horz.	*Vert.*	*Diag.*			
This work (Cameraman)	–0.0005	–0.024	0.0009	46.15%	7.8912	99.98%
Wang et al. (2011)	0.0141	0.0093	0.0035	27.880%	7.9982	99.670%
Lima et al. (2013)	−0.0049	0.0016	0.0022	33.4758%	7.9993	99.6066%
Rakesh et al. (2012)	0.00089	0.00170	—	--	7.9899	--

In Table 5, the comparison between the results obtained using the proposed ArMTFr scheme with those of other schemes in literature shows that the proposed scheme has competitive performance; especially, it achieves higher UACI and NPCR percentages.

7.3 Speed analysis

The proposed encryption algorithm has been implemented using MATLAB 2013 on a personal computer with Intel i2duo 2.20 GHZ processor and 3GB RAM running Windows 7. Table 6 shows the time required for the encryption process and that for the decryption process for some test images. On average, the encryption module processes 11.972 Mbps.

Thus, the proposed scheme is both secure and computationally efficient, which makes it suitable for use in many practical scenarios.

Table 6 Time required for encryption and decryption for some sample images

Image	*Encryption time (sec)*	*Decryption time (sec)*
Cameraman	0.08232	0.08222
Peppers	0.08312	0.08311
Boat	0.08444	0.08441
Mandrill	0.08423	0.08433

8 Extension of the proposed encryption algorithm to coloured images

Coloured images are traditionally represented using three colour channels, red, green and blue channels. The proposed encryption scheme can be directly applied to each colour channel separately. However, in our experiments, to further enhance the security of the encryption scheme, keyed colour channels multiplexing has been applied. A simple multiplexing scheme is employed, where the communicating parties agree on which channels to swap. For example, the red channel pixels values are replaced by their blue counterparts and vice versa. Though simple, yet it is effective as apparent from the results provided in Figure 11. The obtained encrypted images are totally different from the original images. Moreover, decryption using a wrong key yields an image which looks like random noise, conveying no information about the plain image. More complex channel multiplexing schemes, as suggested in Abd-El-Hafiz et al. (2014), can be applied to further improve the results. In Figure 12, the histograms of the three colour channels for sample coloured images are provided to show the diversity of the colour levels distribution in the different channels. The histogram analysis for the corresponding encrypted images is shown in Figure 13. It is clear that they are quite different from those of the plain image showing a uniform distribution of colour levels in the three colour channels; thus, the encrypted images leak no information about the original plain images.

Figure 11 Encrypted and decrypted images with correct, wrong key and channel multiplexing in RGB colour space (see online version for colours)

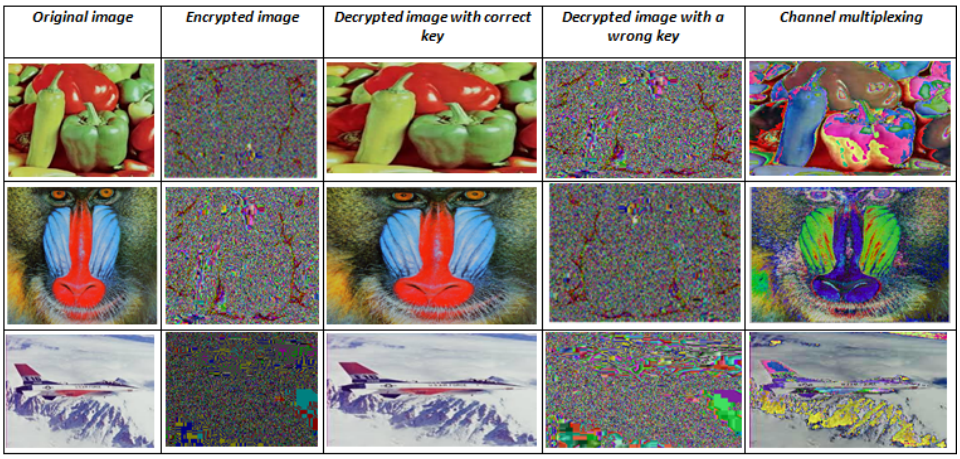


Figure 12 Histogram analysis of Peppers, Mandrill and Airplane images (see online version for colours)

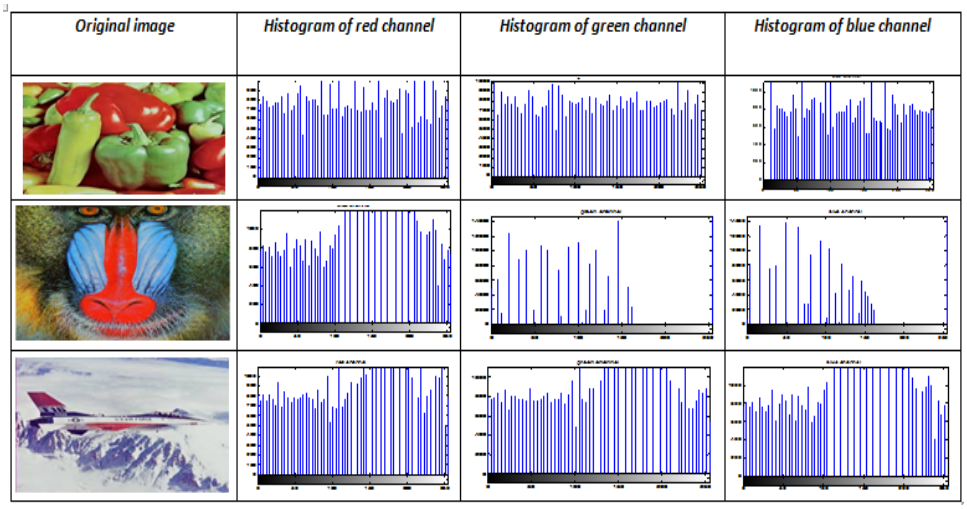
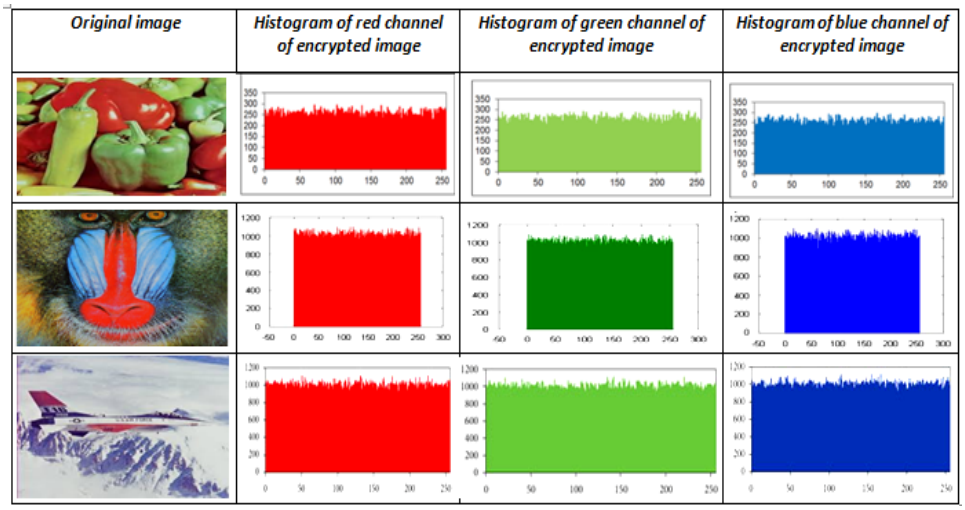


Figure 13 Histogram analysis of Peppers, Mandrill and Airplane encrypted images (see online version for colours)



The correlation coefficients for neighbouring pixels along the three usual directions (vertical, horizontal and diagonal directions) are given in Table 7. It is clear that in the original images, the correlation coefficients; along the three directions in all three colour channels, are high as expected, yet they are nearly zero for encrypted images.

Table 7 Correlation coefficients of some 1,024 × 1,024 original and encrypted test images in RGB colour space

Image	*Pixel correlation coefficients*	*Original image*			*Encrypted image*		
		Horz.	*Vert.*	*Diag.*	*Horz.*	*Vert.*	*Diag.*
Peppers	Red	0.9606	0.9615	0.9533	–0.0076	–0.0081	–0.0117
	Green	0.9828	0.9833	0.9778	–0.0373	–0.0236	0.0344
	Blue	0.9591	0.9598	0.9582	–0.0082	–0.0173	–0.0017
	Avg.	*0.9675*	*0.9682*	*0.9631*	*–0.0177*	*–0.0163*	*0.0071*
Mandrill	Red	0.9321	0.8661	0.8554	–0.0773	0.0010	0.0005
	Green	0.8662	0.7701	0.7348	0.0029	–0.0256	0.0004
	Blue	0.9073	0.8812	0.8398	–0.0250	0.0013	0.0029
	Avg.	*0.90186*	*0.8391*	*0.8100*	*–0.03313*	*0.00126*	*0.00776*
Airplane	Red	0.9736	0.9588	0.9334	–0.0521	–0.0521	–0.050
	Green	0.9588	0.9336	0.9326	–0.0394	0.0352	0.0341
	Blue	*0.9640*	*0.9523*	*0.9246*	*0.0055*	*0.0332*	*–0.0312*
	Avg.	*0.96546*	*0.9492*	*0.9302*	*–0.02877*	*0.00543*	*–0.0157*

Table 8 Differential measures of some test images in RGB colour space

Image		*MAE*	*NPCR%*	*UACI%*
Peppers	Red	80.0033	98.0523	34.0022
	Green	85.8425	98.0597	33.0018
	Blue	86.5335	98.0364	32.9921
	Avg.	*84.1264*	*98.0495*	*33.332*
Mandrill	Red	75.0610	98.0489	32.9987
	Green	86.6564	98.0598	33.0098
	Blue	87.5341	98.0599	33.9871
	Avg.	*83.0838*	*98.0562*	*33.3318*
Airplane	Red	81.5331	98.1540	35.9856
	Green	84.5621	98.1443	34.8879
	Blue	83.2871	98.1365	35.8796
	Avg.	*83.1274*	*98.1449*	*35.584*

The differential measures obtained for sample test images are provided in Table 8. The large values for the MAE measure reflect that the encrypted images are significantly different from their corresponding plain images. Moreover, the scheme still preserves its sensitivity to small variations in the plain image. This is clear from the values obtained for the NPCR and UACI measures.

Table 9 Correlation coefficients of some 1,024 × 1,024 original and encrypted test images in YCbCr colour space

Image	*Pixel correlation coefficients*	*Original image*			*Encrypted image*		
		Horz.	*Vert.*	*Diag.*	*Horz.*	*Vert.*	*Diag.*
Peppers	Y	0.9617	0.9625	0.9633	–0.0066	–0.0071	–0.0112
	Cb	0.9868	0.9853	0.9798	–0.0288	–0.0211	0.0288
	Cr	0.9691	0.9698	0.9682	–0.0077	–0.0163	–0.0015
	Avg.	*0.9743*	*0.9725*	*0.9701*	*–0.01436*	*–0.01483*	*0.00536*
Mandrill	Y	0.9431	0.8871	0.8665	–0.0698	0.0011	0.0005
	Cb	0.8773	0.7751	0.7458	0.00212	–0.0211	0.0004
	Cr	0.9193	0.8999	0.8488	–0.0180	0.0011	0.0019
	Avg.	*0.9132*	*0.8540*	*0.8203*	*–0.02856*	*–0.0063*	*0.0009*
Airplane	Y	0.97896	0.9698	0.94534	–0.0421	–0.0421	–0.0211
	Cb	0.9688	0.9436	0.9426	–0.0384	0.0342	0.0311
	Cr	0.9697	0.9622	0.9356	0.0045	0.0332	–0.0312
	Avg.	*0.9724*	*0.9585*	*0.9411*	*–0.02533*	*0.0084*	*–0.0070*

Table 10 Correlation coefficients of some 1,024 × 1,024 original and encrypted test images in HSI colour space

Image	*Pixel correlation coefficients*	*Original image*			*Encrypted image*		
		Horz.	*Vert.*	*Diag.*	*Horz.*	*Vert.*	*Diag.*
Peppers	Hue	0.9698	0.9656	0.9533	–0.0061	–0.0061	–0.0117
	Saturation	0.9899	0.9900	0.9700	–0.0213	–0.0222	0.0344
	Intensity	0.9643	0.9654	0.9671	–0.0074	–0.0143	–0.0017
	Avg.	*0.9746*	*0.9736*	*0.9634*	*–0.0116*	*–0.0142*	*0.007*
Mandrill	Hue	0.9399	0.8661	0.8554	–0.0673	0.0010	0.0004
	Saturation	0.8862	0.7999	0.8898	0.0022	–0.0166	0.0004
	Intensity	0.9199	0.8918	0.8998	–0.0200	0.0013	0.0029
	Avg.	*0.9153*	*0.8526*	*0.8816*	*–0.02836*	*–0.0047*	*0.0012*
Airplane	Hue	0.9788	0.9677	0.9411	–0.0401	–0.0401	–0.0310
	Saturation	0.9697	0.9411	0.9458	–0.0377	0.0341	0.0333
	Intensity	0.9701	0.9643	0.9321	0.0044	0.0232	–0.0301
	Avg.	*0.9728*	*0.9577*	*0.9396*	*–0.02446*	*0.0057*	*–0.0092*

We have also investigated applying our encryption algorithm to colour images represented in the YCbCr colour space and the HSI colour space. First, channel multiplexing over the RGB channels is performed, and then the result of this step is converted to either of the other two colour spaces. Again, as indicated in Table 9 and Table 10, the proposed scheme succeeds in destroying the correlation between the adjacent pixels. Moreover, there is no resemblance between the original images and the encrypted images, as shown in Figure 14 and Figure 15. Furthermore, decryption using a wrong key, which slightly differs from the correct key, produces a corrupted image.

Figure 14 Encrypted and decrypted images with correct and wrong keys using conversion to YCbCr colour space (see online version for colours)

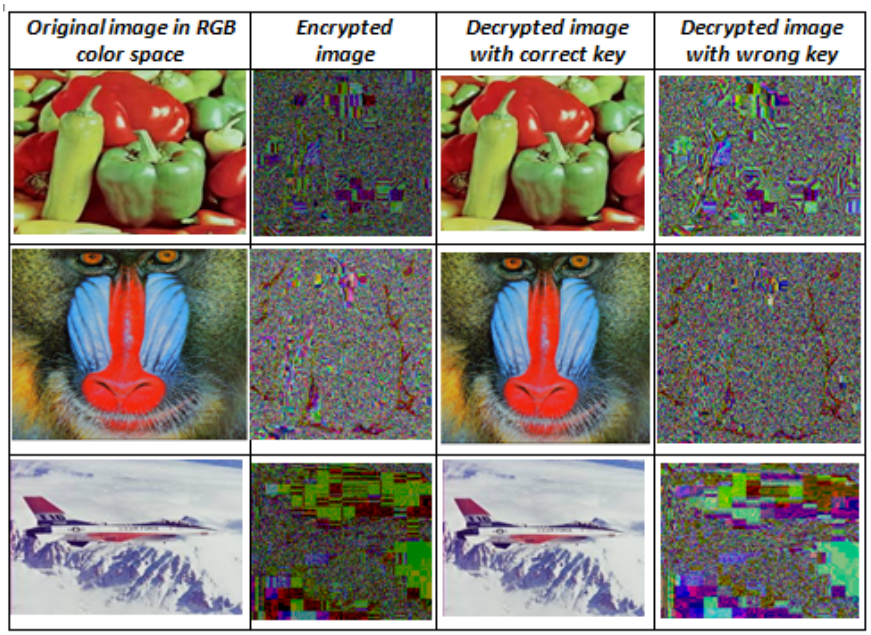


Figure 15 Encrypted and decrypted images with correct and wrong keys using conversion to HSI colour space (see online version for colours)

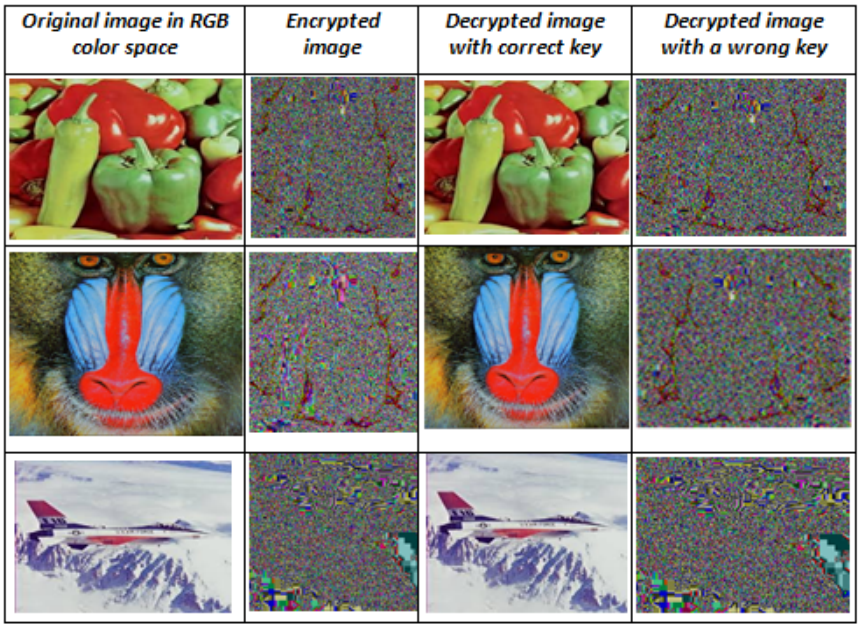


Table 11 Differential measures of some test images in YCbCr colour space

Image		*MAE*	*NPCR%*	*UACI%*
Peppers	Y	81.0122	98.8888	34.5874
	Cb	85.9999	98.7981	33.1421
	Cr	86.8891	99.1111	33.9121
	Avg.	*84.6337*	*98.9326*	*33.8805*
Mandrill	Y	76.1110	98.5678	33.9100
	Cb	86.2178	98.8541	33.8894
	Cr	88.5521	98.9990	34.9944
	Avg.	*83.6269*	*98.8069*	*34.2646*
Airplane	Y	81.9981	98.5789	35.9899
	Cb	85.1241	99.0011	35.9991
	Cr	84.3240	98.7564	35.1254
	Avg.	*83.8154*	*98.7788*	*35.7048*

In Table 11 and Table 12, the differential measures for the proposed scheme based on the YCbCr colour representation and the HSI representation are provided, respectively. Clearly, the proposed scheme is successful in both representations.

Table 12 Differential measures of some test images in HSI colour space

Image		*MAE*	*NPCR%*	*UACI%*
Peppers	Hue	82.1142	99.0011	35.0041
	Saturation	86.0011	99.1002	35.0021
	Intensity	87.4451	98.8874	34.1111
	Avg.	*85.1868*	*98.9962*	*34.7057*
Mandrill	Hue	77.0012	99.8877	34.8811
	Saturation	86.9981	98.8891	33.9941
	Intensity	88.7891	99.0025	34.8871
	Avg.	*84.2628*	*99.2597*	*34.5874*
Airplane	Hue	82.9889	99.8741	36.0101
	Saturation	86.2214	99.7154	36.0001
	Intensity	84.9988	99.8871	35.9981
	Avg.	*84.7363*	*99.8255*	*36.0027*

The processing time of the proposed scheme for the three colours information representations are given for sample test images in Tables 13, 14, 15. It appears that the conversion to HSI colour space is more time consuming compared to the YCbCr colour space. However, the differential measures obtained using the HSI representations are slightly better.

Table 13 Encryption/decryption time for RGB colour space

Image	*Encryption time (sec)*	*Decryption time (sec)*
Peppers	0.123	0.122
Mandrill	0.122	0.124

Table 14 Encryption/decryption time for YCbCr colour space

Image	*Encryption time (sec)*	*Decryption time (sec)*
Peppers	0.128	0.129
Mandrill	0.129	0.130

Table 15 Encryption/decryption time for HSI colour space

Image	*Encryption time (sec)*	*Decryption time (sec)*
Peppers	0.135	0.137
Mandrill	0.138	0.139

Table 16 Comparison of the average NPCR% and UACI% using RGB representation of an image

Scheme	*Encrypted image corr.*			*UACI%*	*Entropy*	*NPCR%*
	Horz.	*Vert.*	*Diag.*			
This work (Airplane)	–0.02877	0.00543	–0.0157	46.15	7.8912	99.98
Mazloom and Eftekhari-Moghadam (2009)	0.007539	0.012878	0.004914	33.5319	--------	99.6083
Huang and Nien (2009)	------	------	------	26.7948	----------	99.5207
Liu and Wang (2011)	----------	----------	----------	33.3756	----------	99.5946
Wei et al. (2012)	----------	----------	----------	33.4055	----------	99.2173

The performance of the extended version of our basic algorithm for handling colour images in RGB model is compared to other related schemes in Table 16. The superiority of the proposed scheme is apparent from the higher values for the UACI and NPCR measures and the comparative values for the average correlation coefficients.

9 Conclusions

In this paper, a new image encryption algorithm based on two keyed random permutations as well as fractals has been proposed. The use of fractals aided in improving the security of the proposed scheme by enlarging the key space and providing a more uniform histogram of the encrypted image.

In the scheme, we aim to reduce the correlation coefficients among pixels through multiple pixels permutations. This algorithm is able to resist attacks from unauthorised users due to its sensitivity to slight changes in the plain image and changes in the system key. This is evident from the results obtained for the differential measures reflecting changes in the encrypted/decrypted images due to small variations in either the plain image or the system key. Security arguments for resisting known-plaintext and CPAs are also provided.

Malicious interception of the encrypted images, breaking the integrity of the received message by modification, is detectable. Both the cropping attack and the salt and pepper noise attack have been mounted on the proposed scheme. The large values for the MSE, between the decrypted image resulting from an attacked image and the original image, indicate the success of the proposed scheme in their detection. Moreover, encrypted images subjected to either of the two attacks produce noisy decrypted images as desired.

Various results are encouraging and suggest that the proposed ArMTFr scheme is secure enough to be adopted for secure image communications. Additionally, it is computationally efficient with an average processing rate of 11.9 Mbps on a personal computer with moderate computational and storage resources.

The proposed scheme has been extended to handle coloured images; where in the latter case, colour channel multiplexing has been applied to further improve the quality of the obtained results. We investigated three standard models for colour information representation. The algorithm succeeded in the three cases to break the correlation among adjacent pixels and showed sensitivity to changes in the system key. An image decrypted using a slightly different key appeared to be corrupted and the MSE value between the original image and that corrupted image was high.

References

Abd-El-Hafiz, S. et al. (2014) 'A fractal-based image encryption system', *IET Image Processing*, Vol. 8, No. 12, pp.742–752.

Abd-El-Haleem, S. et al. (2013) 'Design of pseudo random keystream generator using fractals', *IEEE 20th International Conference on Electronics, Circuits and Systems (ICECS)*, pp.877–880.

Addison, P.S. (2002) *Fractals and Chaos: An Illustrated Course*, CRC Press Book, Taylor and Francis Group, ISBN: 9780750304009.

Ahmad, M. and Alam, M. (2009) 'A new algorithm of encryption and decryption of images using chaotic mapping', *International Journal on Computer Science and Engineering*, Vol. 2, No. 1, pp.46–50.

Alvarado-Robles, G. et al. (2017) 'Segmentation of green areas using bivariate histograms based in hue-saturation type color spaces', *International Conference on Image Analysis and Processing*, Springer, Cham.

Barnsley, M. and Demko, S. (1985) 'Iterated function systems and the global construction of fractals', *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, Vol. 399, No. 1817, The Royal Society.

Cao, Y. (2013) 'A new hybrid chaotic map and its application on image encryption and hiding', *Mathematical Problems in Engineering*, Vol. 2013, pp.1–13, Article ID 728375.

Chen, J-X. et al. (2015) 'Analysis and improvement of a double-image encryption scheme using pixel scrambling technique in gyrator domains', *Optics and Lasers in Engineering*, March, Vol. 66, pp.1–9.

Corrochano, E.B., Mao, Y. and Chen, G. (2005) 'Chaos-based encryption', *Handbook of Geometric Computing*, pp.231–265, Springer, Berlin Heidelberg.

Crilly, A., Earnshaw, R. and Jones, H. (Eds.) (2012) *Fractals and Chaos*, Springer Science & Business Media, ISBN: 978-1-4612-3034-2.

El-Khamy, S. et al. (2008) 'A hybrid fractal-wavelet data hiding technique', *National Radio Science Conference (NRSC)*, pp.1–9.

El-Latif, A., Niu, X. and Amin, M. (2012) 'A new image cipher in time and frequency domains', *Optics Communications*, Vol. 285, Nos. 21–22, pp.4241–4251.

Ghebleh, M. and Kanso, A. (2014) 'A robust chaotic algorithm for digital image steganography', *Communications in Nonlinear Science and Numerical Simulation*, Vol. 19, No. 6, pp.1898–1907.

Guo, Q., Liu, Z. and Liu, S. (2010) 'Color image encryption by using Arnold and discrete fractional random transforms in HIS space', *Optics and Lasers in Engineering*, Vol. 48, No. 12, pp.1174–1181.

Huang, C.K. and Nien, H.H. (2009) 'Multi chaotic systems based pixel shuffle for image encryption', *Optics Communications*, Vol. 282, No. 11, pp.2123–2127.

Huang, X. (2012) 'Image encryption using chaotic Chebyshev generator', *Nonlinear Dynamics*, Vol. 67, No. 4, pp.2411–2417.

Jin, X. et al. (2017) 'Color image encryption in non-RGB color spaces', *Multimedia Tools and Applications*, p.123.

Johnson, M. et al. (2004) 'On compressing encrypted data', *IEEE Transactions on Signal Processing*, Vol.52, No. 10, pp.2992–3006.

Kamali, M.R. et al. (2010) 'A new modified version of advanced encryption standard based algorithm for image encryption', *International Conference on Electronics and Information Engineering (ICEIE)*, 2010, pp.141–145.

Khan, M. and Shah, T. (2014) *A Literature Review on Image Encryption Techniques*, 3D Research Center, Kwangwoon University and Springer-Verlag Berlin Heidelberg, Vol. 5, p.29.

Krishnamoorthi, R. and Murali, P. (2012) 'A new hybrid domain image encryption based on chaos with discrete cosine transform', *IEEE 4th International Conference on Electronics Computer Technology*.

Kumar, A. et al. (2015) 'Image steganography using index based chaotic mapping', *Proceedings of International Conference on Distributed Computing and Internet Technology (ICDCIT)*, Vol. 1, pp.1–4.

Li, C. (2005) *Cryptanalysis of Some Multimedia Encryption Schemes*, PhD thesis, Zhejiang University, Hangzhou.

Li, H. et al. (2013) 'Double-image encryption by using chaos-based local pixel scrambling technique and gyrator transform', *Optics and Lasers in Engineering*, Vol. 51, No. 12, pp.1327–1331.

Lima, J.B., Lima, E.A.O. and Madeiro, F. (2013) 'Image encryption based on the finite field cosine transform', *Signal Processing: Image Communication*, November, Vol. 28, No. 10, pp.1537–1547.

Liu, H. and Wang, X. (2011) 'Color image encryption using spatial bit-level permutation and high-dimension chaotic system', *Optics Communications*, Vol. 284, No. 16, pp.3895–3903.

Loukhaoukha, K. et al. (2012) 'A secure image encryption algorithm based on Rubik's cube principle', *Journal of Electrical and Computer Engineering*, 13pp, Hindawi Publishing, Article ID 173931.

Manoj, I.V.S. (2010) 'Cryptography and steganography', *International Journal of Computer Applications*, 0975–8887, Vol. 1, No. 12, pp.63–68.

Matsumoto, M. and Nishimura, T. (1998) 'Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator', *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, Vol. 8, No. 1, pp.3–30.

Matsumoto, M., Saito, M., Nishimura, T. and Hagita, M. (2007) 'CryptMT stream cipher version 3', *SASC2007 Conference*.

Mazloom, S. and Eftekhari-Moghadam, A.M. (2009) 'Color image encryption based on coupled nonlinear chaotic map', *Chaos, Solitons & Fractals*, Vol. 42, No. 3, pp.1745–1754.

Mikhail, M., Abouelseoud, Y. and Elkobrosy, G. (2017) 'Two-phase image encryption scheme based on FFCT and fractals', *Journal of Security and Communication Networks*, January, 13pp, Article ID 7367518.

Mishra, M. and Mankar, V.H. (2013) 'Chaotic encryption scheme using 1-D chaotic map', *arXiv preprint arXiv: 1312.4042*.

Pareek, K.K.S., Narendra, K. and Patidar, V. (2011) 'A symmetric encryption scheme for colour BMP images', *International Journal of Computer Applications, Special Issue on Network Security and Cryptography*, pp.42–46.

Pizer, S.M. et al. (1987) 'Adaptive histogram equalization and its variations', *Computer Vision, Graphics and Image Processing*, Vol. 39, No. 3, pp.355–368.

Rakesh, S., Kaller, A.A., Shadakshari, B.C. and Annappa, B. (2012) *Multilevel Image Encryption* [online] https://arxiv.org/abs/1202.4871 (accessed 17 January 2018).

Sasidharan, S. and Philip, D.S. (2011) 'A fast partial encryption scheme with wavelet transform and RC4', *International Journal of Advances in Engineering and Technology (IJAET)*, Vol. 1, No. 4, pp.322–331.

Stallings, W. (2005) *Cryptography and Network Security*, 4th ed., Prentice Hall Publishing Inc.

The USC-SIPI Image Database (1997) *University of Southern California, Signal and Image Processing Institute* [online] http://sipi.usc.edu/database/ (accessed 17 January 2018).

Tiwari, N. et al. (2013) 'Digital watermarking using DWT and DES', in *Proceedings of IEEE 3rd Conference on Advanced Computing (IACC)*, pp.1100–1102.

Wang, X. et al. (2011) 'Chaotic encryption algorithm based on alternant of stream cipher and block cipher', *Nonlinear Dynamics: an International Journal of Nonlinear Dynamics and Chaos in Engineering Systems*, Vol. 63, No. 4, pp.587–597.

Wei, X. et al. (2012) 'A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system', *Journal of Systems and Software*, Vol. 85, No. 2, pp.290–299.

Wu, C-P. and Kuo, C-C.J. (2005) 'Design of integrated multimedia compression and encryption systems', *IEEE Transactions on Multimedia*, October, Vol. 7, No. 5, pp.828–839.

Wu, Y. et al. (2011) 'NPCR and UACI randomness tests for image encryption', *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*, Vol. 1, No. 2, pp.31–38.

Yu, Z. et al. (2010) 'A chaos-based image encryption algorithm using wavelet transform', in *2nd International Conference on Advanced Computer Control*, March 2010, Vol. 2, No. 4, pp.217–222.

Zhang, M-R. et al. (2004) 'T-matrix and its applications in image processing', *IEEE Electronics Letters*, Vol. 40, No. 25, pp.1583–1584.

Zhang, Y. and Xiao, D. (2014) 'An image encryption scheme based on rotation matrix bit-level permutation and block diffusion', *Communications in Nonlinear Science and Numerical Simulation*, Vol. 19, No. 1, pp.74–82.