Efficient JPEG Encoding Using Bernoulli Shift Map for Secure Communication

Nisar Ahmad
UET: University of Engineering & Technology
Muhammad Usman Younus (usman1644@gmail.com)
Universite Federale Toulouse Midi-Pyrenees Ecole Doctorale Mathematiques Informatique Telecommunications de Toulouse https://orcid.org/0000-0001-9033-1767
Muhammad Rizwan Anjum
IUB: The Islamia University of Bahawalpur Pakistan
Gulshan Saleem
COMSATS University Islamabad - Lahore Campus
Zaheer Ahmed Gondal
COMSATS University Islamabad - Lahore Campus
Sanam Narejo
Mehran University of Engineering and Technology

Research Article

Keywords: Simultaneous encryption and compression, generalized Bernoulli shift map, multimedia security, compression, encryption, chaotic system

Posted Date: June 8th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-485453/v1

Efficient JPEG Encoding Using Bernoulli Shift Map for Secure Communication

Nisar Ahmad[1], Muhammad Usman Younus[2],Muhammad Rizwan Anjum[3], Gulshan Saleem[4], Zaheer Ahmed Gondal[5] , Sanam Narejo [6]

1 Department of Computer Engineering, University of Engineering and Technology Lahore, 54890, Pakistan.
2 Ecole Mathématiques, Informatique, Télécommunications de Toulouse, Université de Toulouse, Toulouse, France.
3 Department of Electronics Engineering, The Islamia University Bahawalpur, Bahawalpur, 63100, Pakistan.
4,5 Department of Computer Science, COMSATS University Islamabad (Lahore Campus), 54890, Pakistan.
6 Department of Computer Systems Engineering, Mehran University of Engineering & Technology (MUET), Jamshoro, Pakistan
*Corresponding Author: Muhammad Usman Younus. Email: usman1644@gmail.com

Abstract

Digital data must be compressed and encrypted to maintain confidentiality and efficient bandwidth usage. These two parameters are essential for information processing in most communication systems. Image compression and encryption may result in reduced restoration quality and degraded performance. This study attempts to provide a compression and encryption scheme for digital data named as Secure-JPEG. This scheme is built on the JPEG compression format, the most widely used lossy compression scheme. It extends the standard JPEG compression algorithm to encrypt data during compression. Secure-JPEG scheme provides encryption along with the process of compression, and it could be altered easily to provide lossless compression. On the other hand, the lossless compression provides less compression ratio and is suitable only in specific scenarios. The paper address the problem of security lacks due to the use of a simple random number generator which can not be cryptographically secure. The improved security characteristics are provided through Generalized Bernoulli Shift Map, which has a chaotic system with demonstrated security. The algorithm's security is tested by several cryptographic tests and the chaotic system's behavior is also analyzed.

Keywords: Simultaneous encryption and compression, generalized Bernoulli shift map, multimedia security, compression, encryption, chaotic system.

1. Introduction

Inexpensive image and video acquisition devices has resulted in significant increase in image and video applications. The development of technology has made images and videos and integral part of our daily life. Compression of digital images and videos is always required due to large storage requirement. Images and videos have large redundancies especially in natural scene imagery which could be compressed to overcome bandwidth and storage limitations. Compression is therefore a vital part of general-purpose acquisition devices. Wireless communication systems is a particular case where compression is

essential for low data rate to ensure power and bandwidth efficiency [1]. Encryption on the other hand is required to ensure data security and confidentiality. Wireless and public networks are at an increased demand for image encryption to protect the data from illicit use. In conventional systems, compression is performed to achieve bandwidth and storage efficiency whereas encryption is performed where data security is required. The proliferation of information technology has facilitated us, but it has resulted in larger number of attacks on personal information and resulting in breach of security. Encryption is normally required when data is exchanged over a public network to ensure data security. When encryption and compression are done separately, decryption and decompression must be done separately, resulting in slower processing and, in some cases, lower image quality. Encryption and decryption are time consuming process and so, it is a major bottleneck in real-time image transmission and communication systems. Similarly, image compression and decompression are a bottleneck. Many functional real-time systems are infeasible due to the computing expense of executing these steps independently [2].

Data confidentiality is becoming a major a concern due to rapid technological advancements. A lot of algorithms are available for encryption but most of them are useful for textual data. Image data encryption, on the other hand, differs from alphanumeric data encryption as it has a high level of redundancy and spatial correlation. As a result, traditional encryption systems like RSA[3] and AES[4] are unsuitable for image or multimedia encryption. Encryption of digital images can be achieved by degrading visual detail with absolute randomness. So, there are few application based encryption algorithms aimed at producing encrypted digital data [5-7]. Due to high-redundancy and bulk-capacity, there is a need of an algorithm which can process both encryption and compression.

The idea of performing simultaneous encryption and compression was acquired from [8]. They have presented a shared key encryption approach for JPEG. The DCT quantized coefficients of JPEG are separated into random looking shares which provide encryption. The basic scheme has some limitations which are covered in some variations of the basic scheme provided in the same paper. In [9], an encryption scheme based on DCT and orthonormal vectors was presented which was JPEG compressible. The scheme achieved reasonable security but contained horizontal correlation which depended on orthonormal vectors. In [10], standard JPEG algorithm is modified by incorporating encryption step on DCT quantized block of 8×8 pixels. The security of [9] and [10] depend completely on random number generator which is cryptographically insecure algorithm [11].

Chaotic systems are believed to be perfect candidates for encryption system. A chaotic system must have three properties: (I) it must be sensitive to initial conditions, (II) it must have dense periodic orbits and (III) it must be topologically mixing. A good random number generator helps in image data encryption and so, a chaotic system serves as a perfect random number generator. In [12], the authors have used a Generalized Bernoulli Shift Map (GBSM) to perform permutation and diffusion to achieve encryption. The scheme is fast and provide cryptographically secure system but lacks the ability to be compressed with a lossy compression scheme. In our approach (Secure-JPEG), we have extended work on our conference paper [10] to make it cryptographically secure by introducing GBSM. The resulting image is visually similar to [9] and [10] and provide a cryptographically secure base. The algorithm modifies the DCT coefficients of JPEG compression scheme based on orthonormal vectors generated using GBSM. Secure-JPEG has two benefits: (I) it provides encryption and compression in a single pass and (II) the quality of decoded image is superior to original scheme [9].

2. Related Work

Various research studies [13-16] have focused primarily on the compression of digital images, dismissing security issues. Due to their intrinsic sensitivity, pseudo-randomness, and ergodicity to initial conditions, the algorithm of chaotic image encryption is becoming more common [3-7, 13, 17]. They have good confusion and diffusion properties that satisfies the cryptographic requirements. On the other hand, these systems cover only image security without taking compression into account. Compression is required in most of the image and video storage and communication scenarios due to high redundancy and volume. This means that digital images need to be both encrypted and compressed. This approach was followed by many researchers in their research, with particular attention paid to data reduction and confidentiality [18-23].

When it comes to achieving encryption and compression of digital files, researchers consider two methods. Encryption and compression were performed in two steps in the first approach [9, 24-27]. Both two stages are entirely independent of one another and manipulating the image in two stages will take longer at times. In this case, the intruder must focus exclusively on cryptoanalysis to compromise the security of the algorithm while ignoring the compression scheme. On contrary, second method involves digital image compression and encryption performed in one stage [20, 28-30]. Since the intruder must consider both compression and encryption algorithm when performing the encryption analysis, so, cumulative encryption and compression results in reduced computational time and increased data security.

In [8], authors have designed a "shared-key JPEG color image encryption algorithm". During quantization process, the algorithm works on DCT coefficients. The method is based on two random 8×8 block which are shared on optical encryption. Each 8×8 block is encrypted and shares randomly generated data that is processed through JPEG for additional computation. The generated shares hold similar characteristics such as identical in size to the original block. The compression does not introduce any error, moreover encryption through share generation is also lossless. To perform decryption of data, DCT coefficients [8] is calculated and then original image was reconstructed. Three extended approaches are also discussed in their work, one for the random distribution of pixels, one to generate asymmetrical shares, and another is to provide more than two shares. These modifications possess varying strengths and limitations when compared to proposed method.

Another scheme which is orthogonal-based color-image encryption has been presented in [9]. The cryptographic arrangement occurs in two steps: the first is partitioning of the 8×8 blocks which are then scrambled using Mersenne Twister [31]. The scrambled 8×8 blocks are then transformed to a frequency domain via DCT function. In the second step, Mersenne Twister produces a random number matrix of the size of the image. This algorithm shows its strength of securely encrypting the data but also introduces spatial correlation. The horizontal spatial correlation was explained through orthogonal matrix. In addition, because of manifold grayscale stretching, the proposed scheme introduces an intensity shift in resultant image. Although the scheme is compression friendly which presents high compression ratio. It also introduced a redundancy into system as DCT is used during encryption as well as during JPEG compression. In addition, encryption and compression generate more quantification errors in contrast to both process in a single process. Simultaneous encryption and compression may also be effective at introducing fewer errors.

In [10], a simultaneous compression and encryption scheme has been designed which is based on standard JPEG algorithm. The standard JPEG is modified by inserting permutation and diffusion steps

just before quantization. The algorithm has shown good compression performance along with demonstrated security characteristics except the Pseudo-Random Number Generator (PRNG). This method uses Mersenne Twister algorithm to generate random number sequences to permute the image pixels blocks and perform diffusion through orthogonal matrices. Mersenne Twister is a PRNG with large period, but it lacks cryptographic security and hence compromise the security of complete encryption scheme. The results of their algorithm is compared with standard JPEG compression algorithm and applying JPEG compression on cipher image obtained through [9].

In [12], the author used Generalized Bernoulli Shift Maps (BSM) method to produce chaotic orbits which were used to encrypt digital images. They used generalized BSM method at two stages. In first stage they scrambled image pixels based upon a chaotic orbit sequence generated by BSM. And in the second stage they generated two chaotic orbits to induce diffusion within image pixel bits. Their method is resistant to resistance to brute-force attack and has strong statistical properties that make differential attacks difficult to implement.

3. Secure-JPEG Based on Bernoulli Shift Map Method

Secure-JPEG is designed to perform both compression and encryption simultaneously on digital images. In this scheme we introduce encryption steps during steps of standard JPEG compression algorithm. More specifically we apply encryption steps before DCT quantization step, and the rest of JPEG algorithm remains the same. Some properties of the algorithms are such that lossy compression (quantization) and entropy coding can provide reasonable compression ratio with very little distortion. The basic algorithmic implementation is presented in [10] which is an extension of [9] by using Mersenne Twister algorithm as a Pseudo-Random Number Generator (PRNG). The Mersenne Twister has a large period so its output seems completely random but it is not cryptographically secure as demonstrated in [11, 31]. As the encryption is used to secure data from cryptographic attacks so a simple PRNG with high period cannot be used in some real time applications. In this proposed scheme, we use PRNG in two steps; one is during image-blocks scrambling which performs permutation and the other is orthogonal matrix multiplication which performs diffusion. Security of both steps is based on this PRNG algorithm so it must be replaced with a cryptographically secure algorithm.

Chaos-based systems are happened to be very sensitive to initial conditions and have high ergodicity, therefore they demonstrate excellent properties of cryptographic security and randomness. Some chaos-based systems provide high randomness and non-predictability, but they have small key space which lend them prone to brute-force attacks.

In our proposed scheme, we are using a chaos-based orbits to generate random numbers namely generalized Bernoulli shift map which has large key space along with properties of randomness. The Bernoulli shift map B0: [0,1] → [0,1] is given by.

$$y_{n+1} = B_0(y_n) \coloneqq 2y \bmod 1 = \begin{cases} 2y & \text{If } y \in [0, \frac{1}{2}] \\ 2y - 1, & \text{If y } \in [\frac{1}{2}, 1] \end{cases} \quad (1)$$

The Bernoulli shift map provides a case for fundamentally nonlinear mechanism because it typically produces deterministic chaos. This basic mechanism may appear in more realistic dynamical systems. In this study, we are employing its generalized version as shown below:

$$y_{n+1} = B(y_n) := \frac{y_n}{a} \; mod \; 1 \tag{2}$$

Where $y_n, y_{n+1} \in [0,1]$ are the states of the map, and $a \in (0,1)$ is the control parameter. As $a = 0.5, B$ becomes the regular Bernoulli shift map. A typical orbit of y_0 obtained from the dynamical system is $\{y_k = B^k(y_0), k = 0,1, \dots\}$ which is presented in Fig. 5 (a) for $a = 1.75$, $y_0 = 0.5$.

Its waveform is quite unusual and shows that the system is chaotic. The control parameter a and the initial condition y_0 can be considered as cipher keys as the map is employed to produce digital data encryption schemes.

The basic working mechanism of Secure-JPEG is demonstrated in Fig. 1. The system is presented with a plaintext image as an input. The encoder module attempts to perform encryption and compression simultaneously based on some secret key. The encoded image can be stored or transmitted through some insecure communication channels. An intruder having access to encoded image can try some attacks to acquire information which is protected through encryption. On the decoder end, the image is received, and process of simultaneous decryption and decompression is performed, and a reconstructed image is obtained as an output.

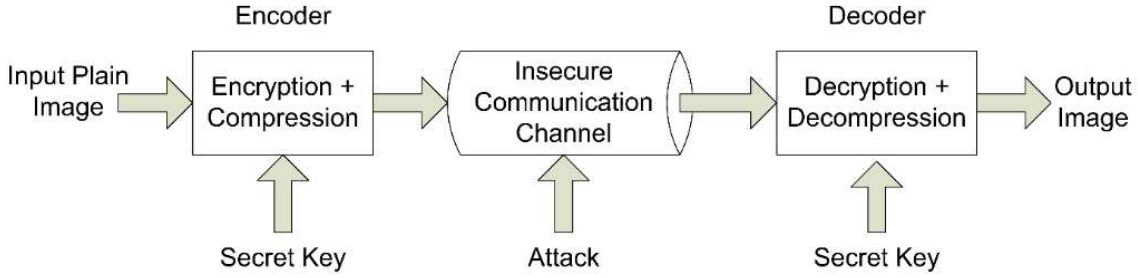


Fig. 1 Secure-JPEG

In this scheme we use Bernoulli shift map method at two stages during encoding process. In first stage, the algorithm uses it to produce a random sequence which is subsequently sorted and used for permutation of non-overlapping blocks$(\beta_1, \beta_2, \cdots \beta_N)$, of 8×8 pixels of the original image, where N is the number of blocks. The conversion of an image matrix into multiple of 8 is performed using zero-padding. Then 8×8 pixels blocks are converted into matrices $(\psi_1, \psi_2, \dots \psi_N)$ using DCT function.

In the second stage of encoding process, random matrices $(\gamma_1, \gamma_2, \dots \gamma_N)$ of 8×8 values are produced using the chaotic PRNG for each DCT transformed block of image. Singular Value Decomposition is used to decompose these randomly generated blocks into U_i, Σ_i & V_i. The encrypted image block is obtained by multiplying left singular vectors U_i with DCT transformed image blocks $(€_1, €_2, \cdots €_N) = (\psi_1 \times U_1, \psi_{2_2} \times U_2, \dots \psi_N \times U_N)$.

These encrypted image blocks are then quantized by dividing with JPEG quantization matrix. The encrypted image blocks are divided with weighted quantization matrix whereas the value of weight defines

the compression quality of the image. The weighted matrix has fewer corresponding values for a better compression quality and higher values when the required compression quality is low. Subsequently, zig-zag scanning is used to transform these 8×8 matrices to row vectors of 64 values. The first values of these vectors are DC values which have higher magnitude, and the remaining are less important AC values with lower magnitudes and most of them become zero when they are divided with the quantization matrix. The DC values of all the image blocks are appended into a separate vector and the AC values are encoded with run-length coding to eliminate most of the zeros. These DC vectors and all the 63 element's AC vectors are converted to binary, and Huffman-dictionary is formed. This dictionary assigns shorter binary codes to the more probable bit sequences and longer codes to less probable sequence, effectively reducing the code length. The obtained binary sequence is compressed and encrypted image data which could be stored in a data file for storage or transmission. The complete process of encryption is depicted in Fig. 2.

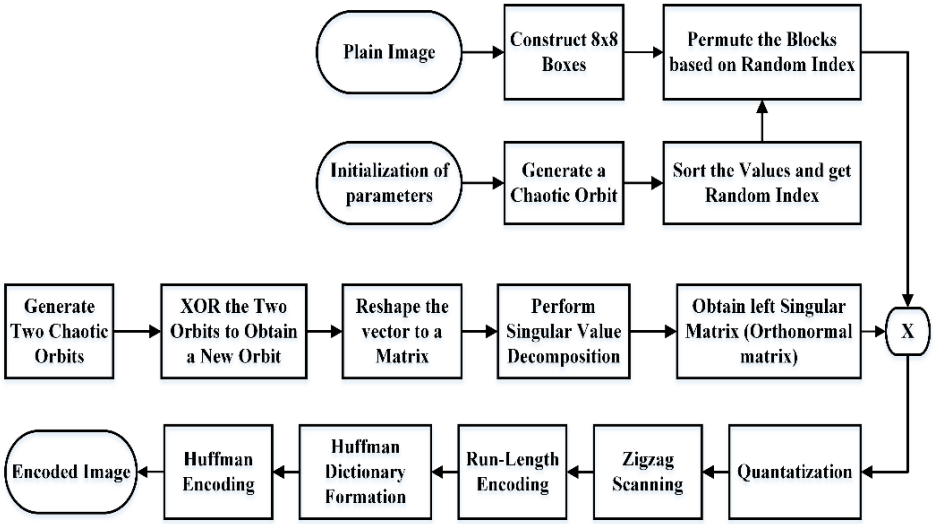


Fig. 2 Encryption scheme according to Secure-JPEG.

Diffie–Hellman key exchange or some other appropriate algorithm may be used to share the encryption key. In reverse order, the encoding algorithm of the system is used in the same way. However, two decoding scenarios have been presented below.

- Scenario-I: The encoded image data will be treated as a JPEG compressed image and the JPEG algorithm is used for the decoding. Huffman decoding is used for data compression and may decode the variable length code. This data is then converted to their true sequence through inverse run-length coding. Then sequenced data is converted from a vector to an 8×8 block. Dequantization regenerates the matrix values with a lossy compression factor error before quantizing. After backward DCT transformation, the image is created, and 8 blocks are combined. In this case, the decoded image data shows a cipher image that the intruder does not have because he does not have the secret key information and encryption algorithm.

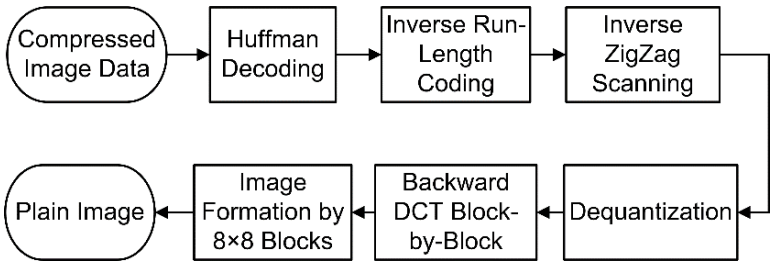


Fig. 3 Scenario-I; Decoder Layout using standard JPEG.

- Scenario-II: It is based on using Secure-JPEG to decode the encrypted image data. In this case user has authorized access to encryption algorithm and so, decode the encoded or encrypted image. The compressed image data is employed to Huffman encoding and its output is a variable length coded data. The resultant data will be converted to the original data sequence by inverse run-length coding. In next step, it is transformed into 8×8 block using inverse Zigzag scanning. Due to lossy compression, an error is produced in data but through dequantization, data is recovered along error. In contrast, 8×8 random matrices $\{\gamma_1, \gamma_2, \dots \gamma_N\}$ are created by the same PRNG using the secret key information. These matrices are further decomposed to $U_i, \Sigma_i \& V_i$ through singular value decomposition. Then inverse of this matrix is calculated through transpose function as the left singular matrix is an orthogonal matrix, and its transposition is equivalent to inverse of its orthogonal matrix. Next step is to multiply both transpose of left singular vector (U_i^T) and dequantized matrices to obtain the decrypted matrices as $(€_1, €_2, \cdots €_N) = (\psi_1 \times U_1^T, \psi_2 \times U_2^T, \dots \psi_N \times U_N^T)$. Then the backward transformation is implemented using a DCT that produces a spatial image. Conversely, the permutation sequence is also formed during encryption, which is used to obtain an inverse permutation sequence. The inverse permutation sequence is then used to do inverse block scrambling. These produced blocks are combined to create a decrypted image using the original image size. Fig. 4 shows the block diagram for this scenario-II.

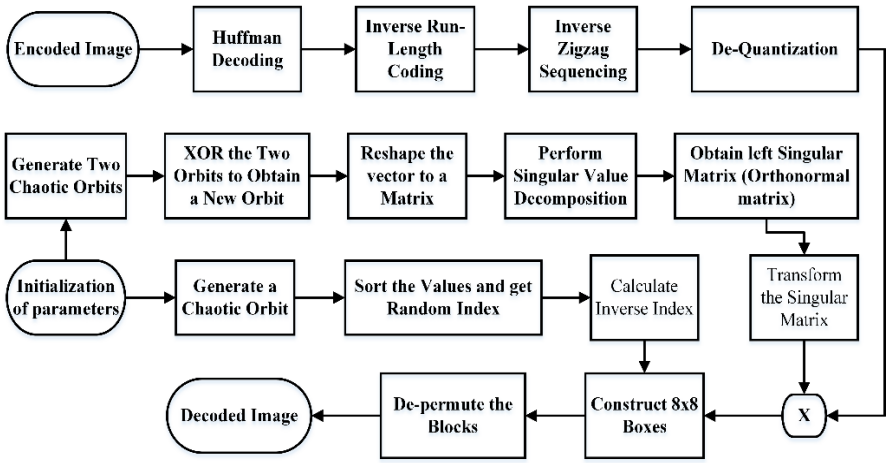
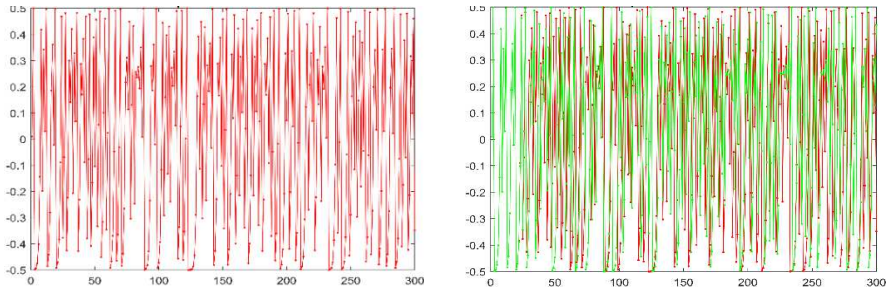


Fig. 4 Scenario-II; Decoder Layout using Secure-JPEG.

4. Results and Discussions

The proposed scheme is an extension of our conference paper [10] which has provided compression and encryption simultaneously. The compression performance is demonstrated in the paper however the encryption is done using PRNG (Mersenee Twister) which is default PRNG of MATLAB. It has large period so can be used in many scenarios, but it lacks the cryptographic security which is necessary for an encryption scheme. To overcome this problem, the proposed modification uses a Generalized Bernoulli Shift Map to generate chaotic orbits which are subsequently used for permutation and diffusion. Chaos based systems has demonstrated cryptographic characteristics and the used scheme has demonstrated its results in [12]. Fig. 5 (a) shows a standard orbit of Generalized Bernoulli Shift Map with initial conditions (x0=1.75, a=0.5 and b=0.25). It could be seen that the values are completely random and chaotic. Fig. 5 (b) provide demonstration of sensitivity to initial conditions. The two orbits (i.e., red, and green) differ each other's by 10^{-12} and their values after few iterations are completely independent and chaotic. Fig. 6 (a) on the other hand shows auto-correlation characteristics and Fig. 6 (b) demonstrates the cross-correlation characters which are required parameters for an ideal chaotic system.



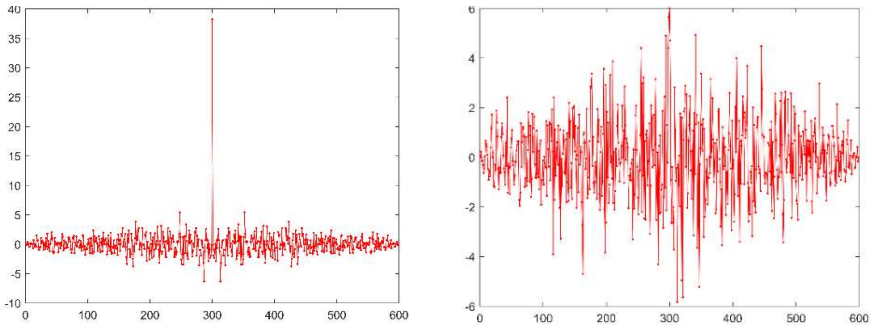
(a) Typical Orbit of Generalized Bernoulli

(b) Difference of two orbits of Generalized

Shift Maps with randomized values

Bernoulli Shift Maps with difference of 10^{-12} in the initial values.

Fig. 5 Random behavior of chaotic orbits and sensitivity to initial conditions

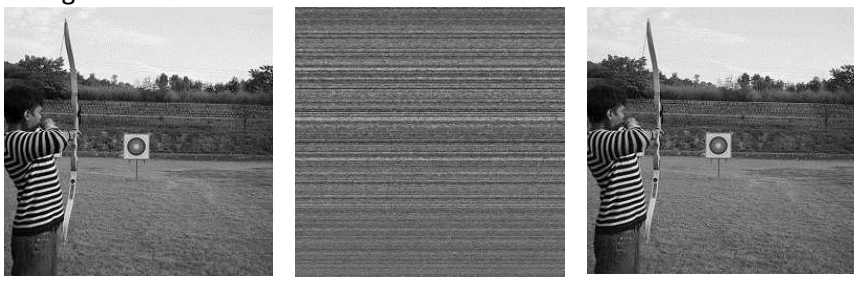


(a) Auto-correlation of an orbit of a Generalized Bernoulli Shift Map

(b) Cross-correlation of two orbits of Generalized Bernoulli Shift Map

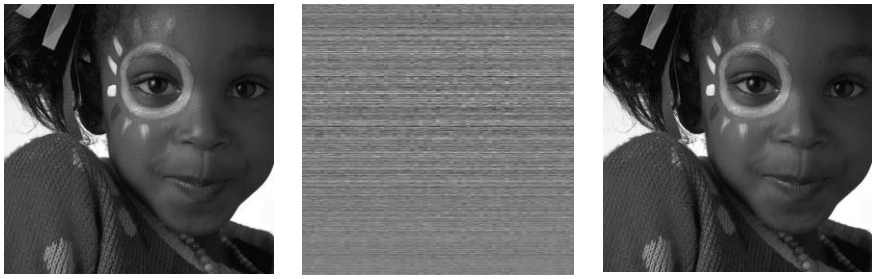
Fig. 6 Auto- and Cross-correlation characteristics of Generalized Bernoulli Shift Map

The chaotic nature of system is demonstrated by Fig. 5 and Fig. 6 which guarantees the cryptographic security of the proposed scheme. Compression performance of the proposed scheme is almost same as the SecureJPEG. At the time of encoding, when compression ratio is chosen as 100%, the decoded image is exact replica of the input image and so only lossless compression is employed. On the other hand, when quality of compression chosen below the 100%, some quantization error is induced (i.e., hence the name lossy compression) but the quality of the decoded image is reasonable for visual inspection as shown in Fig. 7 and Fig. 8 and demonstrated in section 4.6.



(a) (b) (c)

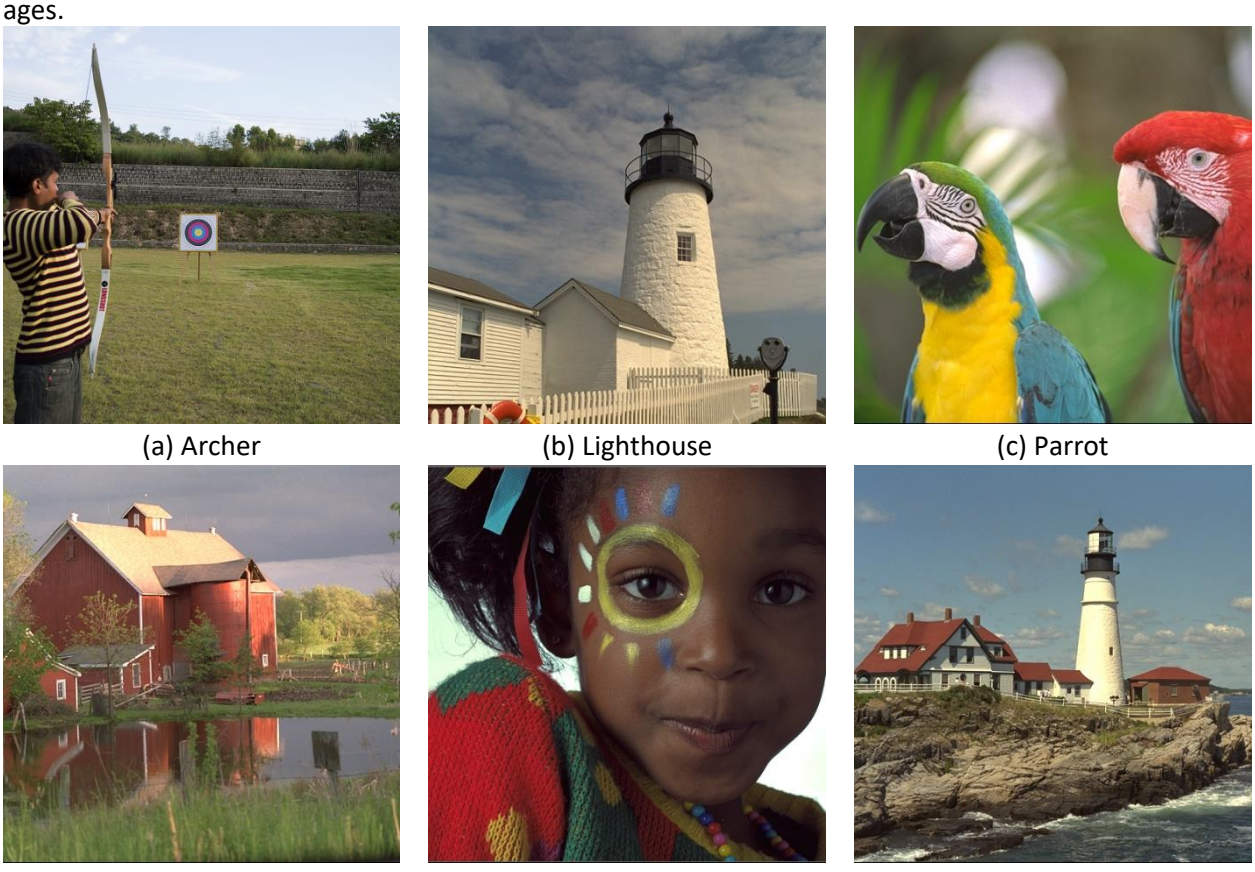
Fig. 7. (a) Input Plain Image-Archer, (b) Standard JPEG Decoding Result of Input Image, (c) Input Image Decoded through Secure-JPEG (Deciphered image)



(a) (b) (c)

Fig. 8 (input Plain Image-Kodim15), (b) Standard JPEG Decoding Result of Input Image, (c) Input Image Decoded through Secure-JPEG (Deciphered image)

Fig. 9 show the test images that are used to demonstrate the encryption performance and quality assessment performance. Altough, graphical results are not reported for all of these images to avoid repetations but the numerical scores are reported and discussion is made based on testing on these six images.



(a) Archer (b) Lighthouse (c) Parrot

(d) House (e) Kodim15 (f) Kodim21

Figure 9 Six test images used for encryption performance and quality assessment

4.1. Correlation Coefficient Analysis

Correlation coefficient analysis is used to determine randomness of pixel values in an image. Natural images have strong correlation in their adjacent pixel values. The security of an image cryptosystem is regarded as high when it produces a highly uncorrelated image. Adjacent pixels correlation is measured in vertical, horizontal, and diagonally adjacent pixels. Below formulas are used to calculate correlation coefficient of two image pixels [11].

$$C.C = \frac{Cov(x,y)}{\sqrt{VAR(x)} \times \sqrt{VAR(y)}} \tag{3}$$

$$VAR(x) = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - E(x)\right)^2 \tag{4}$$

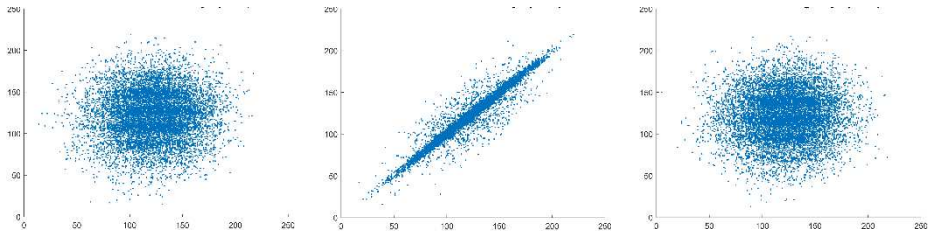
$$Cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \tag{5}$$

Here, x and y are pixel values, C.C. is correlation coefficient and Cov is covariance of pixel values, VAR(x) is variance, δx is standard deviation, E is expected value and N is total number of image pixels.

Table 1 provides the score of correlation calculated with above formulas in three pixels' orientations. The same is calculated and shown graphically for 10,000 adjacent pixels in Fig. 10 and Fig. 11. It is evident that correlation is much low in vertical and diagonal directions, but it is higher in horizontal direction. The higher correlation in horizontal direction is visually apparent in cipher image as well and it is due to orthonormal matrices used for diffusion. This correlation is desirable property and used to achieve good compression ratio.

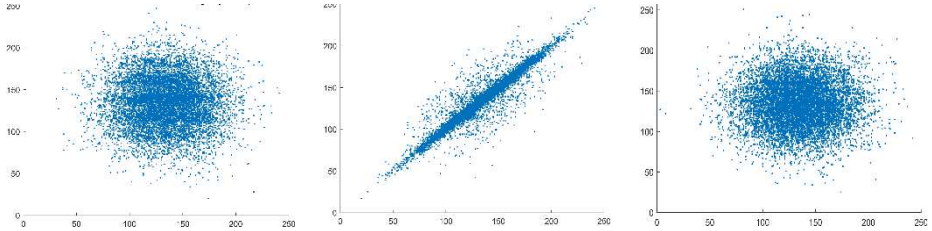
Table 1 Performance measure of Correlation Coefficient of adjacent pixels in vertical, horizontal, and diagonal directions

	Pixel correlation of encrypted image		
Image	**Vertical**	**Horizontal**	**Diagonal**
Archer	0.0054	0.9645	0.0117
Lighthouse	0.0139	0.9775	0.0047
Parrot	0.0070	0.9778	0.0018
House	0.0073	0.9778	0.0124
Kodim15	0.0047	0.9495	0.0053
Kodim21	0.0089	0.9740	0.0222



(a) (b) (c)

Fig. 10 Correlation of 10000 adjacent pixels for Archer Image; (a) vertical direction (b) horizontal direction (c) diagonal direction



(a) (b) (c)

Fig. 11 Correlation of 10000 adjacent pixels for Kodim15 Image; (a) vertical direction (b) horizontal direction (c) diagonal direction

4.2 Information Entropy Analysis

Information entropy is a statistical measure which determines the uncertainty and randomness in an information system. The entropy of a system can be calculated by the below formula [11]:

$$H(s) = -\sum_{i=0}^{2^N-1} P(S_i) \log_2 P(S_i) \quad (6)$$

Here, S is the source, $P(S_i)$ is the probability of occurrence and N is the total number of bits. For an 8-bit grayscale image, the entropy value of an ideally random system should be 8. In practical systems the entropy value is always less the ideal value. However, the encrypted image should have entropy value as close to ideal as possible.

Table 2 performance measure of Information Entropy for encrypted images.

Image	Entropy
Archer	6.89315
Lighthouse	7.01561
Parrot	7.02744
House	7.08853
Kodim15	6.96914
Kodim21	7.08970

Table 2 shows the values of information entropy for encrypted images. It is evident from the values in the table that the cryptosystem provides good randomness in the encrypted images.

4.3 Statistical Analysis

Histogram Analysis

Histogram of image provides the distribution of image pixel values. This distribution can be used to disclose image characteristics and they have been used to decrypt permutation only images. It is therefore taken as an important measure to check the cryptographic security of an image encryption scheme. Histogram plot can be used to visually inspect the statistical characteristics of an encrypted image. Fig. 12 shows the histogram plots for plain and cipher versions of Archer image. The histogram of cipher image is independent of plain image. The test on several images reveled the same results and the histogram remains bell shaped which is due to the property of Central Limit Theorem occurring due to matrix multiplication of DCT transformed image and orthonormal matrix.
Maximum deviation and irregular deviation are numerical measure to ensure statistical randomness [11].

Maximum Deviation

The maximum deviation is a parameter used to verify the statistical randomness of an encryption scheme. It measures the pixel difference in the original and encrypted image. Higher maximum deviation value means more security. The maximum deviation value is calculated using the following formula [11]:

$$D = \frac{d_0 + d_{255}}{2} + \sum_{i=1}^{254} d_i \quad (7)$$

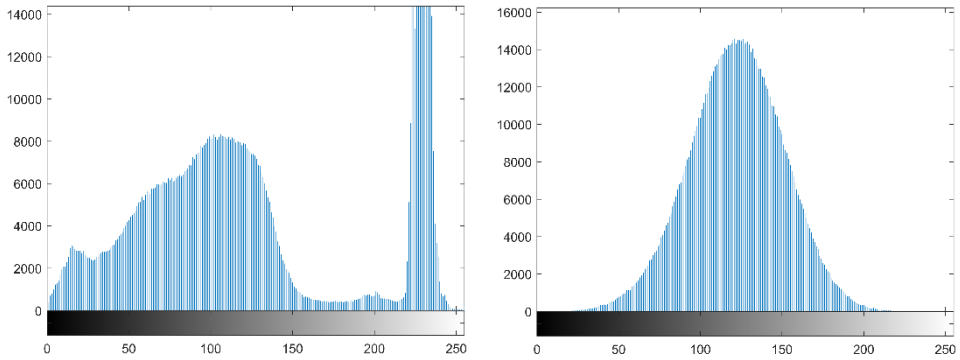
Here, d0 and d255 are difference values of histogram at index 0 and 255 and di is the difference of histograms of original and cipher images.

Irregular Deviation

Maximum deviation sometimes does not appear to be enough as the encryption algorithm should change the value of each pixel randomly to ensure security. a cryptosystem which produces large change in some image pixels and minor change in other image pixels is not statistically secure. Irregular deviation on the other hand, calculate the change in all pixels, the formula to calculate irregular deviation are given below [11]:

$$I_D = \sum_{i=0}^{255} |h_i - M_h| \quad (8)$$

Here, h_i is the difference of histogram values of original and cipher images, Mh is the mean value of h_i and ID is the irregular deviation.
Table 3 provides the scores of maximum deviation and irregular deviations calculated for each plain image and cipher image pair. Higher values of these measures indicate better security towards statistical attacks.



(a) (b)

Fig. 12 Histogram Analysis of Archer Image (a) Original image (b) Encrypted image.

Table 3 Scores of maximum deviation and irregular deviation

Image	Maximum Deviation	Irregular Deviation
Archer	93719	1185776
Lighthouse	67637	303580
Parrot	63120	366090
House	55312	355734
Kodim15	73772	355346
Kodim21	64404	459383

4.4 Differential Analysis

The differential analysis calculates the change in the pixel value of the cipher image in contrast to the plain image. This property tests the diffusion characteristics of the cryptosystem picture. The output image can adjust in an unexpected way to complicate the relationship between a plain image and a cipher image. The number of Pixels Change Rate (NPCR) and Universal Average Change Intensity (UACI) are measurements of the differential characteristics of the cryptosystem image. Table 4 sets out the NPCR and UACI values for the various test images.

Table 4 NPCR and UACI scores for test images

Image	NPCR	UACI
Archer	0.994914	0.240941
Lighthouse	0.992729	0.217997
Parrot	0.993294	0.217973
House	0.994461	0.192507
Kodim15	0.998378	0.311414
Kodim21	0.992165	0.217354

The NPCR scores of Table 4 indicates that the cryptosystem has produced highly random image which is not related to original image in any way. More than 99% of pixels of cipher image are different then the

plain image. UACI on the other hand measure the average change in pixel intensities which should be nearly 30%.

4.5 Keyspace Analysis

Keyspace analysis is used to verify the robustness of the cryptosystem towards brute-force attacks. The cryptosystem must have a large key space to make exhaustive key search infeasible. It must also be very sensitive to secret key and a minor change in secret key should produce an entirely random and different cipher.

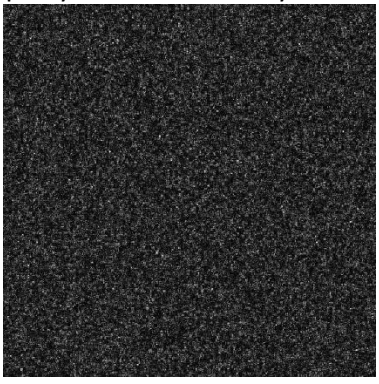
Exhaustive Key Search

Exhaustive keyspace refers to the possible combination of secret keys (initial conditions) which could be provided to produce a randomly generated cipher image. The key space should be sufficiently large to make brute-force attacks infeasible. It is usually defined in term of number of iterations required to check all the combinations of secret key. If it is feasible to try half the number of secret keys in reasonable time the cryptosystem is regarded as insecure.
The cryptosystem uses six variables which we provide with initial conditions to produce different combinations output. A single variable with floating point precision is 252 and for six variables it becomes $(2^{52})^6 = 2^{312}$, which is a large enough key space to make all brute-force attacks infeasible.

Key Sensitive Test

Key sensitivity defines the complete dependence of cryptosystem on exact key. It indicates the dependence of cryptosystem on secret key and demonstrate that a minor change in secret key result in inaccurate decryption. The image in Fig. 13 (a) shows the decrypted image with exact secret key whereas Fig. 13 (b) provides the same image decrypted with a slightly different secret key (difference by 10-12). Hence the test demonstrates the sensitivity of the cryptosystem to secret key.

(a)



(b)

Fig. 13 Key Sensitivity Test (a) Archer-cipher image decrypted with original key (b) Archer-cipher image decrypted with 10^{-12} different secret key.

4.6 Quality of Reproduced Image

The quality of reproduced image is important as compression adds artifacts while employing lossy compression. Moreover, as the encryption is performed in the DCT domain and the inverse DCT result in image values which requires requantatization, the reproduced image will not be exact replica of the input image. Ahmed et al. [32] has discussed the quality of decrypted image which is assessed by different measures such as Peak Signal to Noise Ratio (PSNR), Mean Squared Error (MSE), normalized correlation which are objective methods and tries to measure the change in quality of reproduced image in relation to the original image. It has been demonstrated by [33] that structural distortions are more accurate measure of perceived quality. Their proposed image quality assessment algorithm SSIM [33] and its improvement is considered better alternative to these conventional approaches.

Moreover, no-reference image quality assessment algorithms such as [34-36] are another approach to quality assessment of reproduced image. These methods measure the quality of the image without information of the original image and provide an objective measure of perceptual quality of digital images. We have used SSIM [33] and PIQI [34] as quality assessment measure to assess the quality of the reproduced image.

Table 5 Quality assesment of reproduced image

Image	SSIM	PIQI-Original Image	PIQI-Reproduced Image
Archer	0.9913	0.9286	0.9327
Lighthouse	0.9978	0.9464	0.9282
Parrot	0.9942	0.9693	0.9394
House	0.9952	0.9321	0.9011
Kodim15	0.9967	0.9529	0.9203
Kodim21	0.9962	0.9437	0.9132

In Table 5, SSIM for six images indicates that quality of reproduced image is highly similar to the original image. In case of PIQI, the quality assessment is reported for original as well as reproduced image for comparision as the quality assessment is not relative to original image.

5. Conclusion

This paper intends to improve the security of our conference paper which provides efficient permutation and diffusion in image pixels. The algorithm was focused to perform simultaneous encryption and compression. However, we have used Mersenee Twister (default random number generator of MATLAB) to generate random matrix and permutation sequences. This is not a cryptographically protected random number generator that cannot guarantee the security of the cryptosystem. We replaced it with a chaos-based generator with strong security characteristics and resilience to initial conditions. General-

ized Bernoulli Shift Maps are used to produce a permutation sequence and random matrices that are used for subsequent analysis. The findings of the cryptosystem are illustrated by several experiments, such as correlation coefficient analysis, information entropy analysis, keyspace analysis, mathematical and differential analysis. The current scheme is cryptographically secured and provides good compression. The principal application of this scheme includes areas where compression is a necessary feature otherwise the same scheme can be used in lossless compression format as well.

The proposed strategy can be extended for encryption of video, audio, or other multimedia data where lossy compression is often employed. Wavelet domain processing to integrate it with JPEG2000 is still our future consideration and may be explored for better compression and security characteristics.

Declarations

Funding: There are no funding sources for this project.

Conflicts of Interest: It is declared that there is no financial and other competing conflicts of interests.

Availability of data and material: The data used for experiments is publically available at:
https://github.com/nisarahmedrana/Compression-Friendly-Image-Encryption/upload/main/ImageSet

Code availability: The source code of the project will be made public after acceptance of the manuscript.

Authors' contributions: The contributions of the authors are listed below:

N. Ahmed: Conception, coding, experimentations and writeup.

M. U. Younus: Coding, experimentations, writeup and review.

Muhammad Rizwan Anjum: Writeup and review

G. Saleem: Coding, experimentations, writeup and review.

Z. A. Gondal: Coding and experimentations.

Sadia Anwer: Proof Read and reviewed the manuscript.

References

1. Lu, Q., et al., Low-complexity and energy efficient image compression scheme for wireless sensor networks. Computer Networks, 2008. 52(13): p. 2594-2603.
2. Lian, S., D. Kanellopoulos, and G. Ruffo, Recent advances in multimedia information system security. Informatica, 2009. 33(1).
3. Smith, D.R. and J.T. Palmer, Universal fixed messages and the Rivest-Shamir-Adleman cryptosystem. Mathematika, 1979. 26(01): p. 44-52.
4. Selent, D., Advanced encryption standard. Rivier Academic Journal, 2010. 6(2): p. 1-14.
5. Lian, S., Multimedia content encryption: techniques and applications2008: CRC press.
6. Furht, B., D. Socek, and A.M. Eskicioglu, Fundamentals of multimedia encryption techniques. Multimedia Security Handbook, 2004. 4.
7. Van Droogenbroeck, M. and R. Benedett. Techniques for a selective encryption of uncompressed and compressed images. in Advanced Concepts for Intelligent Vision Systems (ACIVS). 2002.
8. Sudharsanan, S., Shared key encryption of JPEG color images. Consumer Electronics, IEEE Transactions on, 2005. 51(4): p. 1204-1211.

9. Ahmed, N., et al., A Novel Image Encryption Scheme Based on Orthogonal Vectors. Nucleus, 2015. 52(2): p. 71-78.
10. Maqbool, S., et al. Simultaneous Encryption and Compression of Digital Images Based on Secure-JPEG Encoding. in Mexican Conference on Pattern Recognition. 2016. Springer.
11. Ahmed, N., H.M.S. Asif, and G. Saleem, A benchmark for performance evaluation and security assessment of image encryption schemes. International Journal of Computer Network and Information Security (IJCNIS), 2016. 8(12): p. 18-29.
12. Ye, R., An image encryption scheme with efficient permutation and diffusion processes, in Advances in computer science and education applications2011, Springer. p. 32-39.
13. Grigoras, V. and C. Grigoras. Chaos encryption method based on large signal modulation in additive nonlinear discrete-time systems. in Proceedings of the 5th WSEAS international conference on Non-linear analysis, non-linear systems and chaos. 2006. World Scientific and Engineering Academy and Society (WSEAS).
14. Philip, M. and A. Das, Survey: Image encryption using chaotic cryptography schemes. IJCA Special Issue on "Computational Science-New Dimensions & Perspectives" NCCSE, 2011: p. 1-4.
15. Wei-bin, C. and Z. Xin. Image encryption algorithm based on Henon chaotic system. in Image Analysis and Signal Processing, 2009. IASP 2009. International Conference on. 2009. IEEE.
16. Shum, H.-Y., S.B. Kang, and S.-C. Chan, Survey of image-based representations and compression techniques. Circuits and Systems for Video Technology, IEEE Transactions on, 2003. 13(11): p. 1020-1037.
17. Khan, M. and N. Munir, A novel image encryption technique based on generalized advanced encryption standard based on field of any characteristic. Wireless Personal Communications, 2019. 109(2): p. 849-867.
18. Hossein, M., S. Mahmud, and N. Biswas, Image Compression and Encryption. International Journal of ElectroComputational World & Knowledge Interface, 2011. 1(3).
19. Zhou, N., et al., Image compression and encryption scheme based on 2D compressive sensing and fractional Mellin transform. Optics Communications, 2015. 343: p. 10-21.
20. Alfalou, A., C. Brosseau, and N. Abdallah, Simultaneous compression and encryption of color video images. Optics Communications, 2015. 338: p. 371-379.
21. Tong, X.-J., et al., A new algorithm of the combination of image compression and encryption technology based on cross chaotic map. Nonlinear Dynamics, 2013. 72(1-2): p. 229-241.
22. Karmakar, J., D. Nandi, and M. Mandal, A novel hyper-chaotic image encryption with sparse-representation based compression. Multimedia Tools and Applications, 2020. 79(37): p. 28277-28300.
23. Yang, Y.-G., et al., Double image compression-encryption algorithm based on fractional order hyper chaotic system and DNA approach. Multimedia Tools and Applications, 2021. 80(1): p. 691-710.
24. Zhou, J., et al., Designing an efficient image encryption-then-compression system via prediction error clustering and random permutation. Information Forensics and Security, IEEE Transactions on, 2014. 9(1): p. 39-50.
25. Zhou, J., X. Liu, and O.C. Au. On the design of an efficient encryption-then-compression system. in Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. 2013. IEEE.
26. Bansal, R. and M.R. Sharma, Designing an Efficient Image Encryption-Compression System Using A New Haar Wavelet. 2014.
27. Saleem, G., et al., Design and Analysis of a Robust Compression Friendly Image Encryption Scheme. algorithms, 2017. 8(2).
28. Zhu, H., C. Zhao, and X. Zhang, A novel image encryption–compression scheme using hyper-chaos and Chinese remainder theorem. Signal Processing: Image Communication, 2013. 28(6): p. 670-680.
29. Aldossari, M., A. Alfalou, and C. Brosseau, Simultaneous compression and encryption of closely resembling images: application to video sequences and polarimetric images. Optics express, 2014. 22(19): p. 22349-22368.
30. Zhou, N., et al., Novel image compression–encryption hybrid algorithm based on key-controlled measurement matrix in compressive sensing. Optics & Laser Technology, 2014. 62: p. 152-160.
31. Matsumoto, M. and T. Nishimura, Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Transactions on Modeling and Computer Simulation (TOMACS), 1998. 8(1): p. 3-30.
32. Ahmed, N., et al., A novel image encryption scheme based on orthogonal vectors. The Nucleus, 2015. 52(2): p. 71-78.
33. Wang, Z., et al., Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, 2004. 13(4): p. 600-612.
34. Ahmed, N., H.M.S. Asif, and H. Khalid, PIQI: perceptual image quality index based on ensemble of Gaussian process regression. Multimedia Tools and Applications, 2021: p. 1-24.
35. Ahmed, N. and H.M.S. Asif, Perceptual Quality Assessment of Digital Images Using Deep Features. Computing and Informatics, 2020. 39(3): p. 385-409.

36. Khalid, H., M. Ali, and N. Ahmed, Gaussian process-based feature-enriched blind image quality assessment. Journal of visual communication and image representation, 2021: p. 103092.

Biography

Nisar Ahmed his PhD and M.Sc. Computer Engineering from the department of computer engineering, University of Engineering and Technology, Lahore, Pakistan. His areas of interest include Image Quality Assessment, Multimedia Systems and Computer Vision.

Muhammad Usman Younus received his Doctorate and master's degree in engineering from the University of Toulouse (III) Paul Sabatier France and University of Engineering and Technology Lahore Pakistan 2020 & 2014, respectively. His research interests include Wireless Sensor Network, Software Defined Networking, Energy Optimization, Wireless Communication, and Machine Learning. He has more than 15 international journal and conference publications. He is a member of PEC, IEEE, etc. and reviewer of some journals and conferences.

Muhammad Rizwan Anjum received his Ph.D degree from Beijing Institute of Technology, Beijing China in 2015. M. Engg. in Telecommunication and Control Engineering and B.Engg. in Electronic Engineering in 2011 & 2007 respectively from Mehran UET Jamshoro, Pakistan. Presently working as Assistant Professor in the Department of Electronic Engineering, Islamia university of Bahawalpur, Pakistan. He has more than 25 international conferences and journal publications. He is a member of PEC, IEEEP, IEP, IJPE, UACEE, IACSIT, ICCTD, IACSIT, IAENG, etc. and reviewer of several journals and conferences.

Gulshan Saleem is a doctoral student at the department of computer science, COMSATS University Islamabad, Lahore Campus. She has done her MS software engineering from College of Electrical and Mechanical Engineering, National University of Science and Technology, Rawalpindi and her BS software engineering from Fatima Jinnah Women University, Rawalpindi. Her areas of interest include machine learning, computer vision and digital image processing.

Zaheer A. Gondal is a Lecturer in the department of computer science, COMSATS University Islamabad, Lahore Campus. He has done MS in computer science from University of Central Punjab and completed his BS in computer science from University of Central Punjab. His areas of interest are machine learning, data mining, CPN and cyber-physical systems.

Sanam Narejo is currently working as Assistant professor in Department of Computer Systems Engineering, Mehran University of Engineering and Technology (MUET), Jamshoro. She has completed her PhD from Politecnico Di Torino, Italy in 2018. She received her Masters degree in Communication Systems and Networking from MUET. Her research interests include Signal and Image Processing, Machine Learning and Deep Learning Architectures. She has also been a member of Italian Society of Neural Networks (SIREN).

Figures

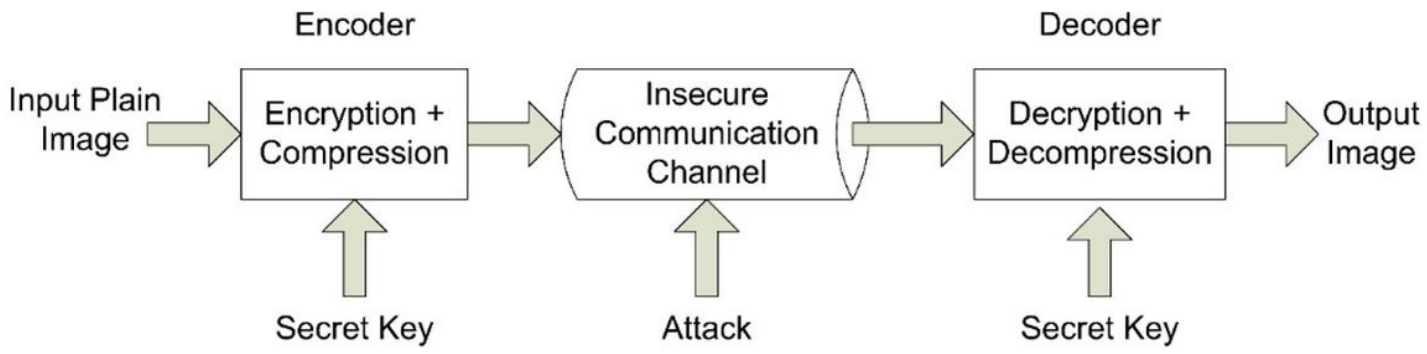


Figure 1

Secure-JPEG

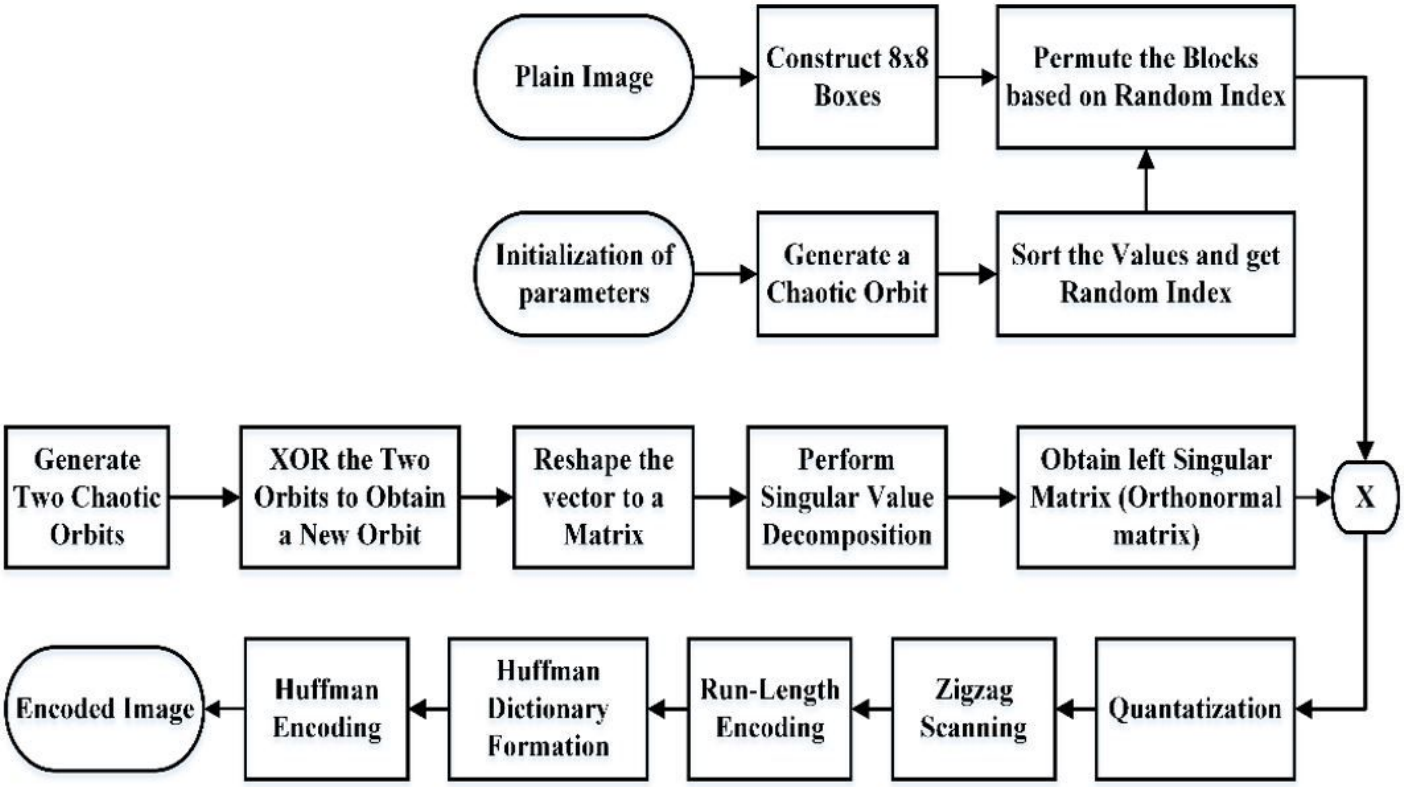


Figure 2

Encryption scheme according to Secure-JPEG.

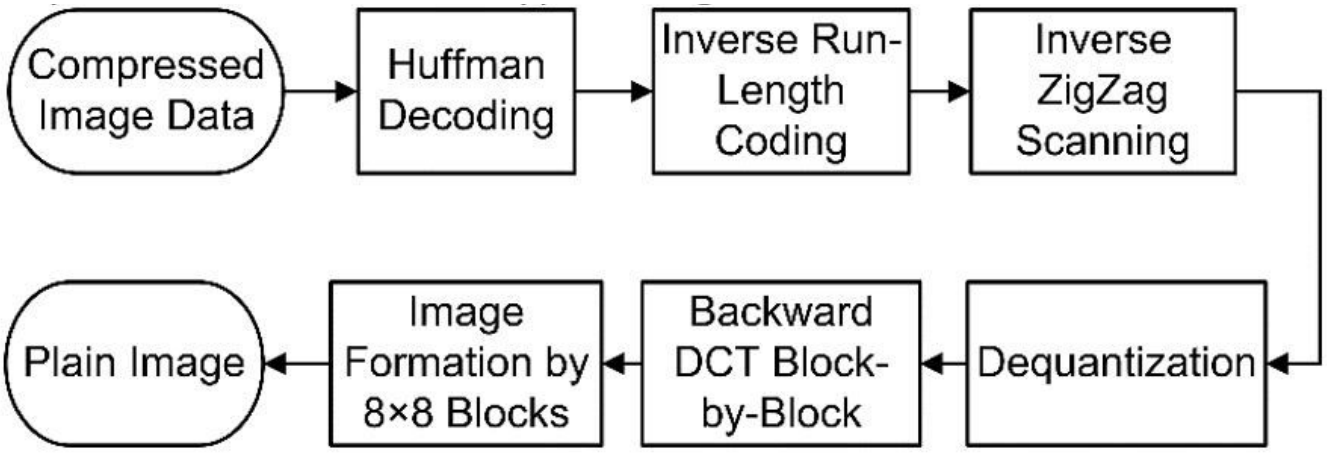


Figure 3

Scenario-I; Decoder Layout using standard JPEG.

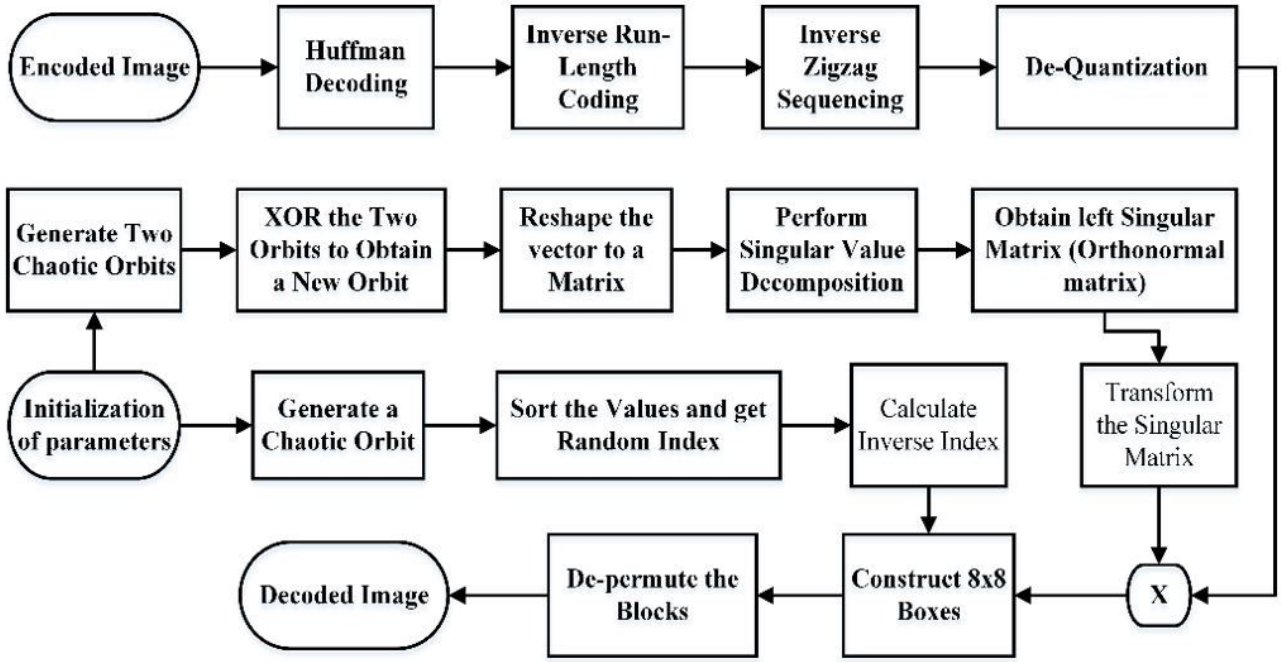


Figure 4

Scenario-II; Decoder Layout using Secure-JPEG.

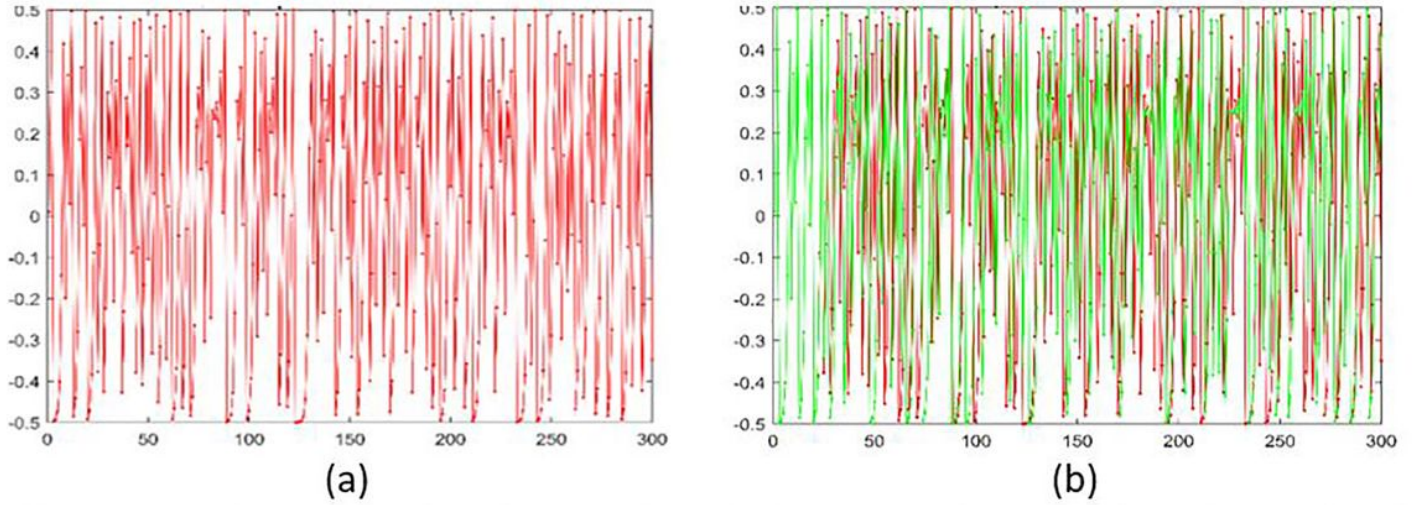


Figure 5

Random behavior of chaotic orbits and sensitivity to initial conditions

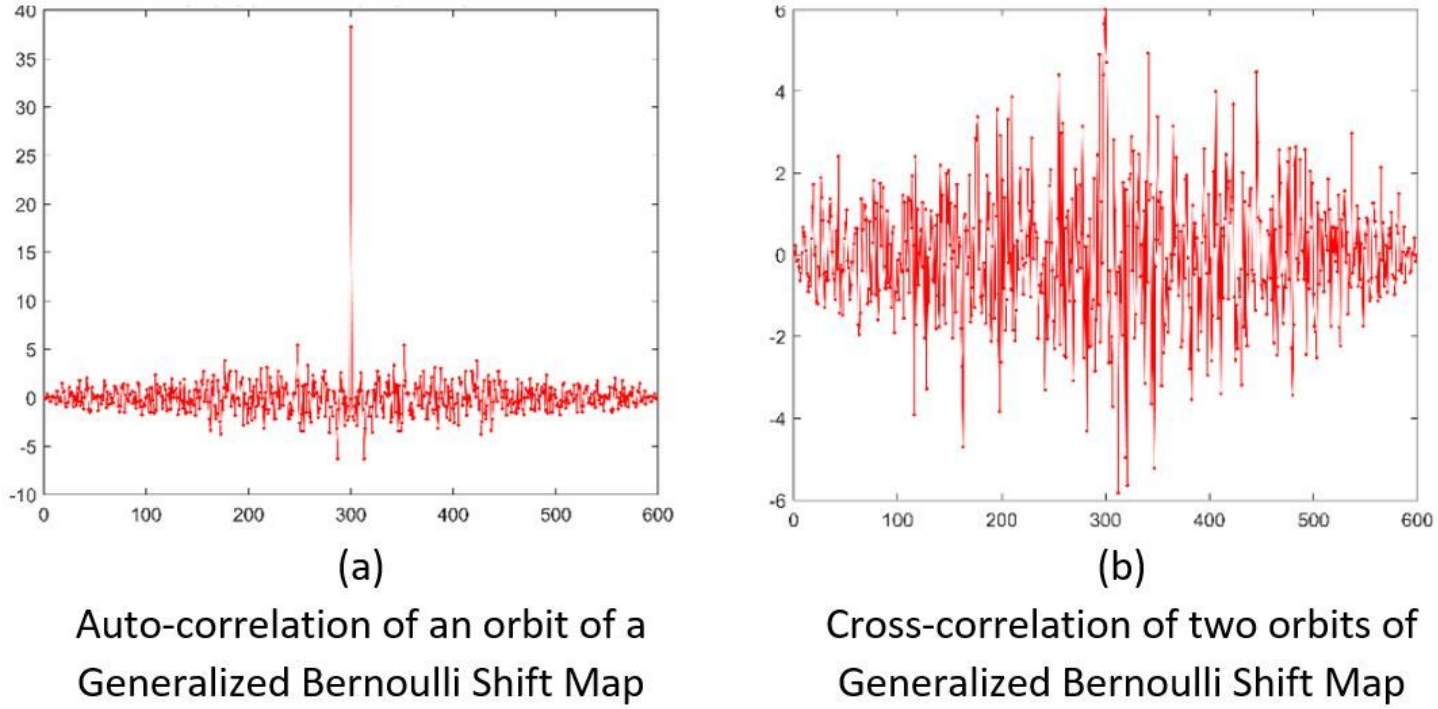


Figure 6

Auto- and Cross-correlation characteristics of Generalized Bernoulli Shift Map

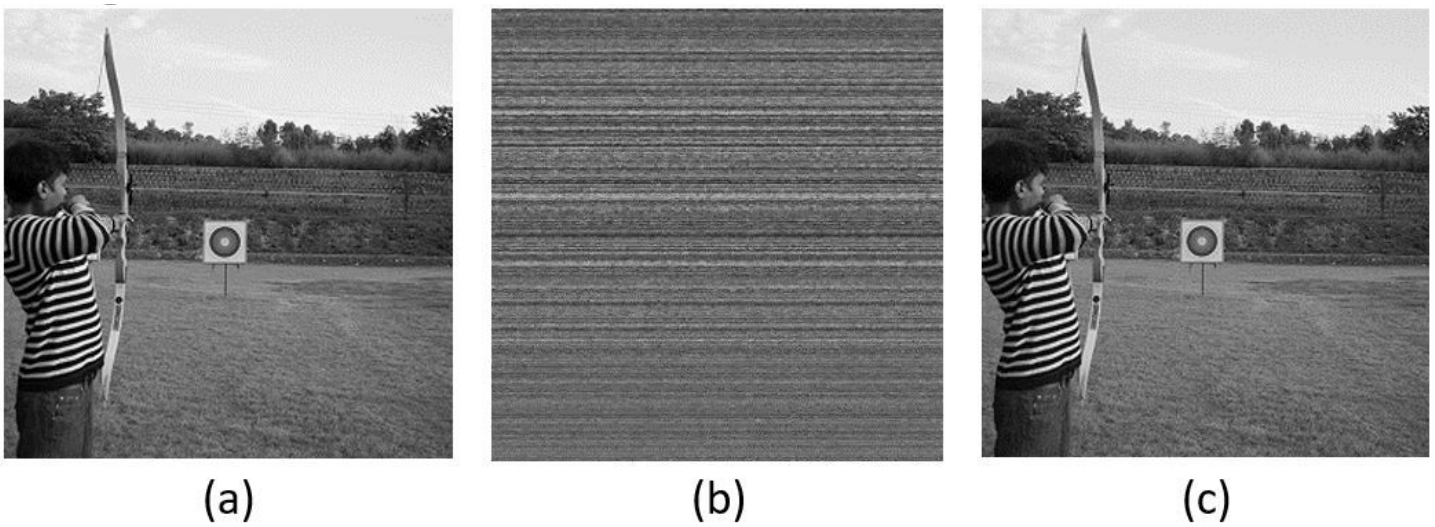


Figure 7

(a) Input Plain Image-Archer, (b) Standard JPEG Decoding Result of Input Image, (c) Input Image Decoded through Secure-JPEG (Deciphered image)

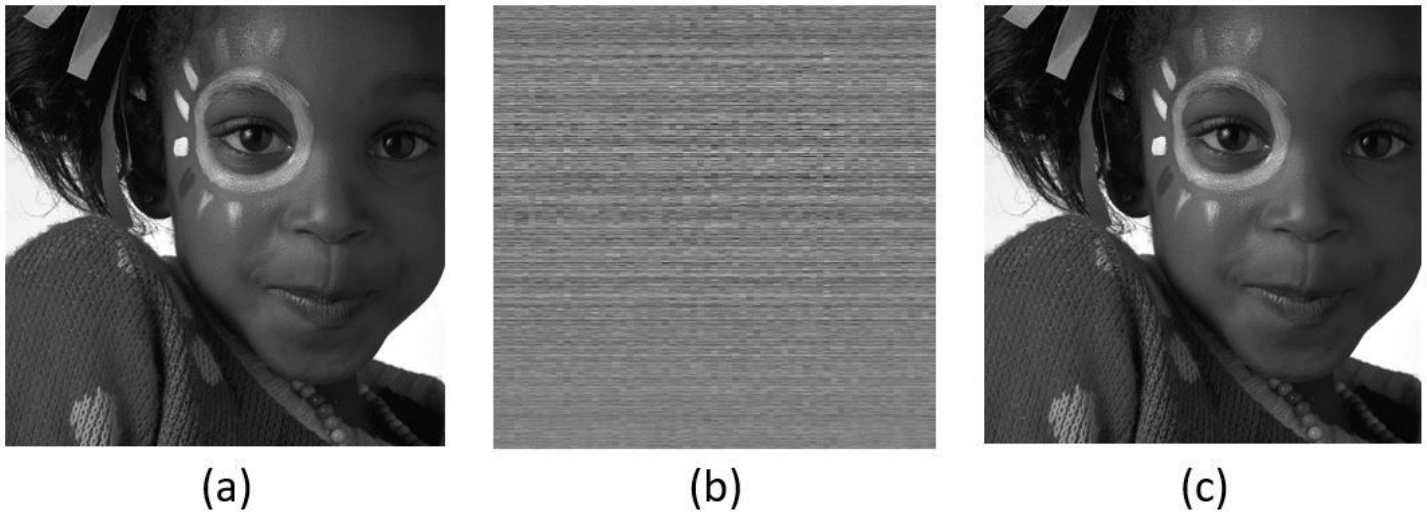


Figure 8

(input Plain Image-Kodim15), (b) Standard JPEG Decoding Result of Input Image, (c) Input Image Decoded through Secure-JPEG (Deciphered image)

Figure 9

Six test images used for encryption performance and quality assessment

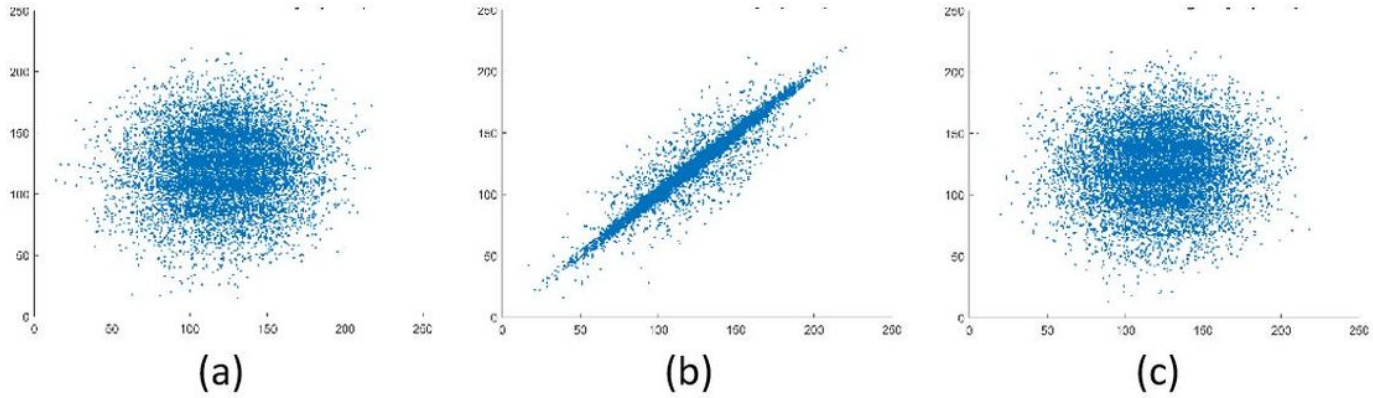


Figure 10

Correlation of 10000 adjacent pixels for Archer Image; (a) vertical direction (b) horizontal direction (c) diagonal direction

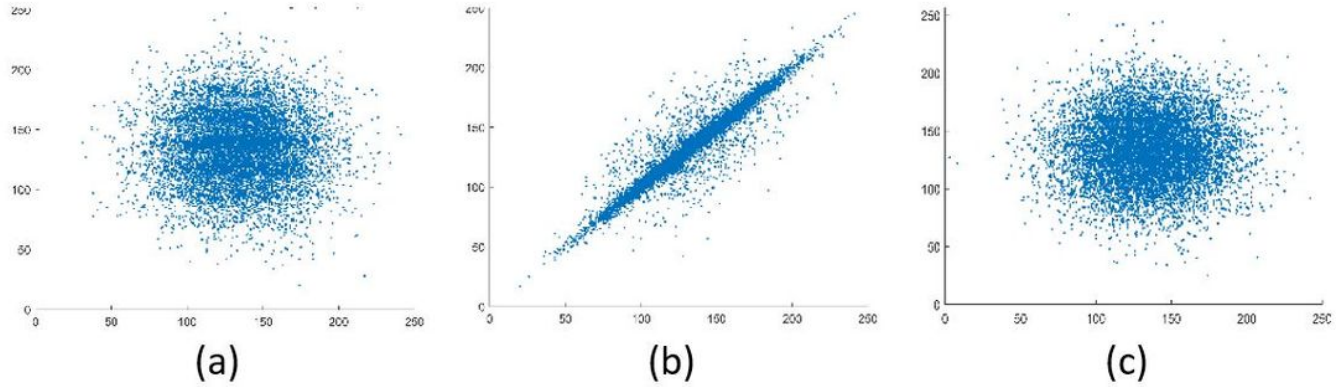


Figure 11

Correlation of 10000 adjacent pixels for Kodim15 Image; (a) vertical direction (b) horizontal direction (c) diagonal direction

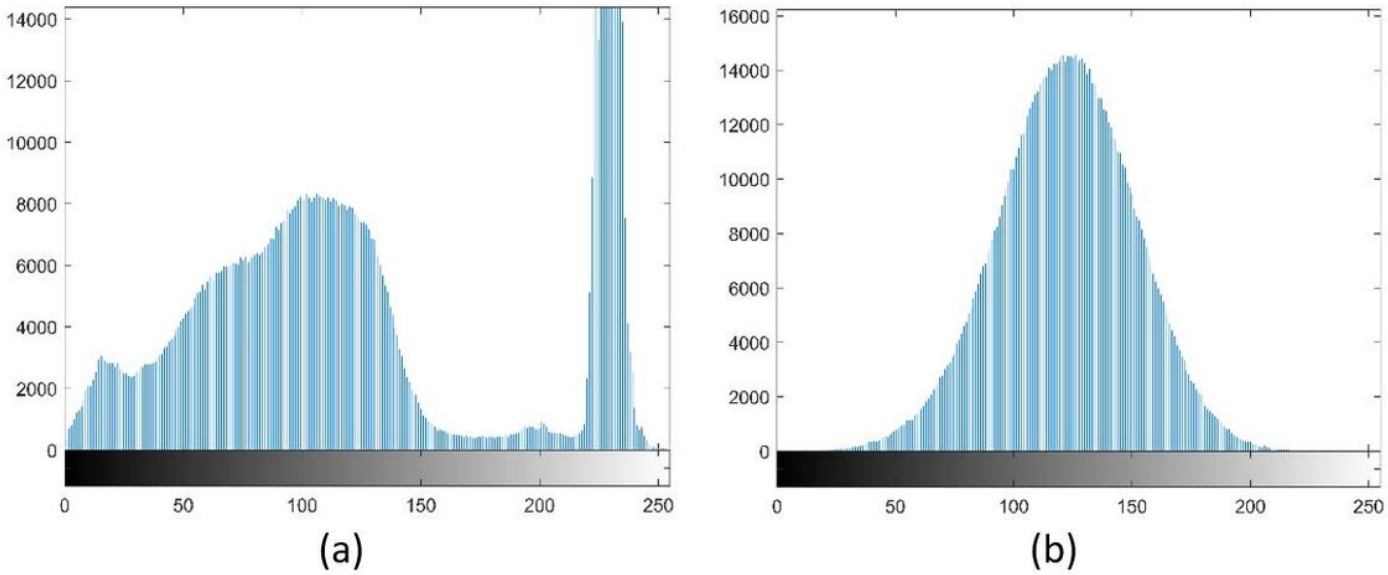


Figure 12

Histogram Analysis of Archer Image (a) Original image (b) Encrypted image.

Figure 13

Key Sensitivity Test (a) Archer-cipher image decrypted with original key (b) Archer-cipher image decrypted with 10-12 different secret key.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- Biography.docx