


ITC 3/49 Information Technology and Control Vol. 49 / No. 3 / 2020 pp. 346-369 DOI 10.5755/j01.itc.49.3.25901	Secured Color Image Compression Based on Compressive Sampling and Lü System	
	Received 2020/04/22	Accepted after revision 2020/06/08
	 http://dx.doi.org/10.5755/j01.itc.49.3.25901	

HOW TO CITE: Krishnan, K. S., Jaison, B., Raja, S. P. (2020). Secured Color Image Compression Based on Compressive Sampling and Lü System. *Information Technology and Control*, 49(3), 346-369. <https://doi.org/10.5755/j01.itc.49.3.25901>

Secured Color Image Compression Based on Compressive Sampling and Lü System

K. Sundara Krishnan*

Department of Computer Science and Engineering; Alagappa Chettiyar Government College of Engineering and Technology; Karaikudi, Tamilnadu, India; phone: +917708795039; e-mail: sundarakrishnank@gmail.com

B. Jaison

Department of Computer Science and Engineering; RMK Engineering College; Chennai Tamilnadu, India; phone: +919840024357; e-mail: bjn.cse@rmkec.ac.in

S. P. Raja

Department of Computer Science and Engineering; Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology; Chennai, Tamilnadu, India; phone: +919486181212; e-mail: avemariaraja@gmail.com

*Corresponding author: sundarakrishnank@gmail.com

An efficient and secured approach is vital for the transmission of sensitive and secret images over the unsecure public Internet. In this paper, a secured color image compression method based on compressive sampling and the Lü system is proposed. Initially, the plainimage is sparsely represented in a transform basis. Compressive sampling measurements are obtained from these sparse transform coefficients by employing an incoherent sensing matrix. Permutation-substitution operations are performed on pixels based on the Lü system to upgrade security levels. Keys are obtained from the input image to add input sensitivity in the scheme. Lastly, a fast and efficient greedy algorithm is utilized for sparse signal reconstruction. The experimental outcome and analysis reveal that the proposed system offers a larger key space, strong input sensitivity, low correlation coefficients and producing visually good reconstructed images.

KEYWORDS: Security, Compressed Sensing, Sensing Matrix, Permutation, Substitution, Lü system.

1. Introduction

Rapid developments in Internet infrastructure and applications have facilitated easy data transmission, particularly of social network applications that create and share large volumes of information, mostly in the form of digital images. Images, which are usually large-sized and have a propensity for redundancy, may also contain sensitive and valuable information. Transmitting confidential images over the unsecure public Internet is fraught with risks. In this regard, two challenges are to be addressed: the confidentiality of sensitive and secret images, and their redundancy. Compression best represents images in a condensed version with the assurance of good visual quality. Compressive sensing is the latest in imaging technology to offer efficient compression. Given that encryption protects image content, highly sensitive chaotic systems are tailor-made for image encryption. Thus, simultaneous compression and encryption resolves multimedia communication challenges.

1.1. Related Work

In recent years, much research has been reported on combined compressive sampling (CS) and chaotic system frameworks in joint compression and encryption techniques. Liya et al. designed a cryptosystem based on block sparse sampling with a permutation-diffusion structure for encryption. The discrete transform coefficients are classified into high, low and medium frequency components and compressed simultaneously using compressive sampling. The encryption is carried out using a one-dimensional logistic map. Such a low-dimensional map keeps the design simple, produces a small key size, and provides weaker security than high-dimensional systems [16]. In [14], a hybrid compression-encryption (CE) scheme was designed using a 1D cascade map in which the Arnold transform minimizes the block effect in the measurement process. Ponuma et al. [23] proposed an image cipher based on compressive sampling, where parallel compressive sampling and masking are employed to resist common attacks. Ponuma et al. [24] again proposed a second image cipher using compressive sensing with a rotated sensing matrix for measurement observation. In [37], the authors introduced a new analysis sparse representation-based hybrid compression and encryption algorithm with a

fixed dictionary. Their scheme produced considerable compression results, though security is not a primary concern. A 2D compressive sampling-based cipher with the fractional random transform, with a logistic map used for a Hadamard matrix construction, was presented in [5]. Tongfeng et al. [30] presented a novel hybrid chaotic map-based cryptosystem using the Fibonacci-Lucas transform, which is robust against cropping attacks. The methods above are based on 1D chaotic maps that have small key spaces and are, therefore, susceptible to bruteforce attacks. It is worthwhile, therefore, to use higher-dimensional chaotic systems.

Xinsheng et al. [32] proposed simultaneous compression and encryption, based on sparse Bayesian learning and the Arnold cat map. The sparse representation is realized by applying the discrete cosine transform while the SBL is adopted for compression. The Arnold cat map is employed for permutation at bit-level cubes. The scheme obtained adequate results in regard to security, though with poor reconstruction quality, because high-frequency coefficients are coarsely quantized in the DCT. In [36], Yaqin et al. presented a CE scheme with compressive sampling and a new chaotic system that offers excellent reconstruction. A secure method for data transmission using chaotic compressive sampling with the Bernoulli sensing matrix was put forward in [9], and performed well against malicious attacks. Qiaoyun et al. [26] proposed a fast image cryptosystem using a hyper-chaotic modulation map, with two sensing matrices for measurement calculation, that is fast and efficient against known plaintext attacks. In [13], Liahua et al. designed a CE method using the fractional transform and Chen's chaotic system, in which the discrete fractional random transform is used for sparse representation. A novel compressive sampling-based combined compression and encryption method was developed by Shuqin et al. [28]. This algorithm depends entirely on two matrices: a scrambling matrix for permutation operations and a Gaussian random matrix for sensing the small and arbitrary number of measurements. Since the lightweight algorithm designed for encryption only includes the permutation process, the absence of the diffusion process results in low resistance against

differential attacks. In [12], Junxin et al. reported that the 3D cat map-based compressive sampling with a structurally random matrix produced good information entropy results.

Miao et al. [19] designed a CE algorithm using an integer wavelet transform and set partitioning in hierarchical trees to produce good lossless compression performance. In [17], Kumar et al. presented an encryption algorithm, followed by a compression algorithm, for images based on Huffman coding and singular-value decomposition. In [31], Xiao et al. designed a CE scheme using a discrete cosine transform dictionary and compressive sensing. Brindha et al. [2] proposed lossless compression and encryption using the Chinese remainder theorem and hash table, where the Henon map is used for scrambling. The scheme obtained good NPCR and UACI values. An encryption and compression technique for color video images, with a rational analysis undertaken on video sequences, was introduced in [1]. Nanrun et al. [20] proposed a new image CE algorithm using a key-controlled sensing matrix in compressive sampling that produced acceptable compression and security results but high computational complexity. Nanrun et al. [22] introduced a hyperchaotic system-based nonlinear encryption approach that achieves confusion using a cycle shift operation. In [21], Nanrun et al. proposed a fractional Mellin transform and 2D compressive sampling-based cryptosystem where CE is achieved simultaneously by observing measurements in two directions where, as the dimension increases, the complexity of the algorithm also increases correspondingly. Xiuli et al. [3] presented a compressive sampling and memristive chaotic system-based image CE algorithm with zigzag confusion. Some research has been based on cellular automata that produce random patterns from simple rules. In [33], Xiuli et al. introduced cellular automata and a compressed sensing-based CE algorithm in which a sensing matrix is constructed by a key-controlled chaotic map to produce good security results.

Zhang et al. [38] designed a joint image data compression and encryption scheme by employing cellular automata and set splitting in hierarchical-trees. It performs three round of permutation and substitution operation in encryption. In each round of operation, different iterated chaotic system is used to generate

keys. Priya et al. [25] developed an enhanced version of logistic map and a simple image cipher technique. In this algorithm, block permutation, zigzag transformation operations are performed in the confusion phase and pixel substitution is performed in the diffusion phase based on the keys obtained from enhanced logistic map and input image. This scheme is able to resist fifty percentage against the occlusion attack. A color image cryptosystem based on Arnold-Tent chaotic maps and Walsh-Hadamard transform (WHT) was proposed by Sneha et al. [29]. Initially, the WHT transform is applied on the color components of the input image. In the encryption process, the Arnold map is used for permutation and Tent map based key sequences are used for substitution.

An image cipher using Fourier transform and CS was presented by Miao et al. [18]. Initially, the input image is divided equally with same dimensions and then random measurements are obtained. Here, the Arnold transform is employed for permutation and Chen map based 2D fractional transform is utilized in the diffusion process. A joint optical-image compression and encryption scheme by using Rivest-Shamir-Adleman algorithm and CS was proposed by Lihua et al. [15]. The logistic-Tent map is utilized to permute the pixel location and intensity values are substituted by the DNA sequences. Xiuli et al. [34] and Chen et al. [7] presented a meaningful image combined compression and encryption technique based on CS. These schemes perform well against the known and chosen plaintext attacks.

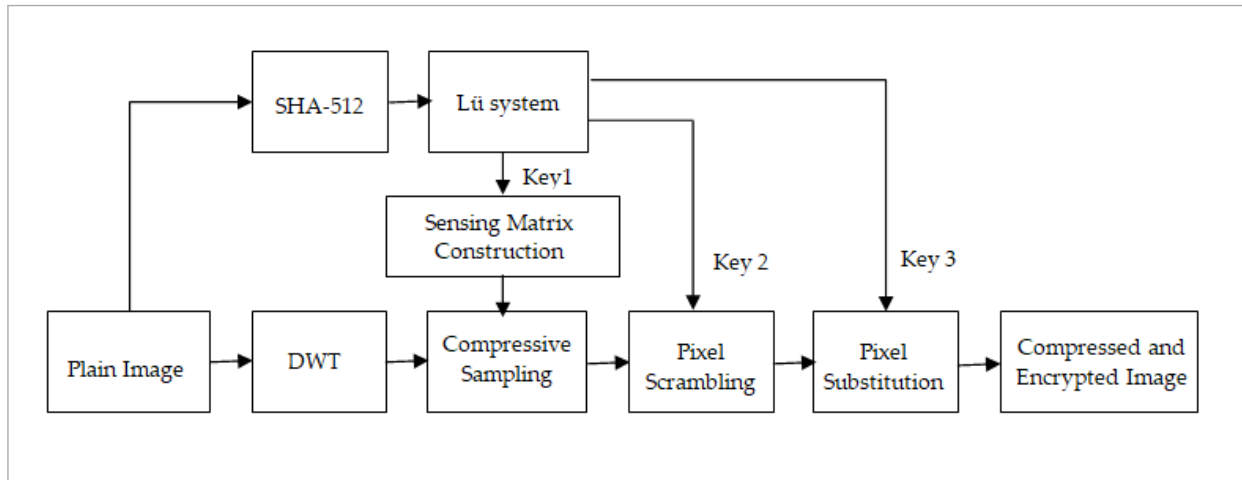
Several methods in the literature survey have executed compression and encryption on grayscale images. Both permutation and substitution operations are performed in the proposed system. The discrete wavelet transform is utilized for sparse representation and the three-dimensional chaotic system for key generation. All of these measures enhance the performance of this system over those described in the references.

1.2. Motivation and Justification

Digital images provide a rich source of information in several domains. Images that carry sensitive and confidential information must be efficiently and securely transmitted. Typically, images are large-sized and highly redundant. Efficient transmission requires

Figure 1

Outline of the Proposed Work



that redundancy be minimized by compression and confidentiality provided by encryption. Compressive sensing is a promising piece of compression technology which significantly minimizes the number of data points to be transmitted for accurate reconstruction. The discrete wavelet transform (DWT) is a good transform for images, especially in terms of efficiently capturing abrupt changes in the images concerned. Chaotic systems are well known for their sensitivity and randomness-state sequences that are most useful in encryption. A combination of permutation and substitution greatly enhances the security of encryption algorithms. The orthogonal matching pursuit is an efficient sparse signal reconstruction algorithm that reconstructs images as accurately as possible. Inspired and motivated by the work above, this paper proposes a secured color image compression approach. The proposed scheme produces good results, thus demonstrating that it executes compression-encryption effectively.

1.3. Outline of the Proposed Work

The proposed scheme broadly comprises six steps: 1) keys are generated from the 512-bit hash of the input image, 2) a sensing matrix is constructed using a chaotic random sequence, 3) the input image is sparsely represented in the DWT, 4) measurements are obtained by applying compressed sensing, 5) measurements are scrambled, based on index sequences,

and 6) the scrambled measurements are substituted using the XOR operation directed by true random sequences. The framework of the proposed scheme is shown in Figure 1.

1.4. Contribution

The contributions of our work include the following: a) Sensing matrix construction: the Lü system-based sensing matrix that is constructed effectively adds the restricted isometric property, and resolves the problem of transmitting the whole sensing matrix to the reconstruction side; b) Dynamic key selection: the starting seeds, dynamically obtained from the input image, enhance key sensitivity and resolve issues with fixed keys; and c) Intra-color channel permutation: the independent permutation of color components significantly minimizes the correlation association and averts statistical attacks.

1.5. Organization of the Paper

The rest of the paper is organized as follows. Section 2 discusses compressive sensing, the discrete wavelet transform, the orthogonal matching pursuit, the Lü system and the secure hash algorithm-512. Section 3 presents the proposed secure compression scheme. Section 4 discusses the experimental setup. Section 5 presents the results and analyzes the performance of the proposed system. Section 6 concludes the paper.

2. Basic Knowledge

2.1. Compressive Sensing

Compressive sensing is an emerging imaging technology which recreates natural images with a high probability of accuracy from fewer measurements/data points than used by conventional schemes. As a result, compressive sensing possesses inherently strong compression and weak encryption characteristics, with its underlying principles being sparsity and incoherence. Sparsity implies that a signal/image is considered sparse when its information or content is reflected in few data points. Incoherence means that the data points are sampled randomly [6]. CS comprises three core processes: sparse representation of a signal/image, measurement calculation, and sparse signal/image reconstruction. Fortunately, all natural signals and images are sparse in themselves or in an appropriate transform. Most coefficients in sparse representation are very close to zero, with few large ones. A signal that is K -sparse indicates that it has K non-zero coefficients. A signal that is sparse in nature requires no sparse representation, and random measurements can be observed directly, as in Equation (1),

$$y = \Phi x, \quad (1)$$

where x is a sparse signal sized $N \times 1$, Φ is the sensing matrix sized $M \times N$ and y is the number of measurements sized $M \times 1$. This process is also known as linear dimensionality reduction. Each measurement, y_i , is the information-preserving projection of x_i on rows (Φ^T_i) of the sensing matrix in Equation (2),

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} \Phi^T_1 \\ \Phi^T_2 \\ \vdots \\ \Phi^T_M \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}. \quad (2)$$

The process of reconstructing x from y is an ill-posed inverse problem. Algorithms proposed to resolve the optimization problem include the gradient projection for sparse reconstruction, split Bregman iteration, backpropagation, smooth l^0 and orthogonal matching pursuit (OMP). Signals that are not sparse in nature are sparsely re-represented in an appropriate orthogonal transform basis using Equation (3), where ψ is

the transform basis sized $N \times N$ and α is the sparse coefficient vector sized $N \times 1$,

$$x = \psi \alpha \quad (3)$$

the random measurements are calculated by Equation (4):

$$y = \Phi \psi \alpha. \quad (4)$$

In the proposed scheme, we have used the discrete wavelet transform (DWT) as the sparse transform basis (ψ) and the orthogonal matching pursuit (OMP) as the signal reconstruction algorithm, both of which are explained below. The measurement matrix (Φ) constructed is based on the chaotic system described in the next section.

2.1.1. Discrete Wavelet Transform

A wavelet is a rapidly decaying wave-like oscillation that has zero mean. Unlike sinusoids which extend to infinity, a wavelet exists for a finite duration. The key concept of wavelets is scaling, which refers to stretching or shrinking signals in time. A stretched wavelet helps capture the slowly varying changes in a signal while a compressed wavelet helps capture abrupt changes. A wavelet transform produces good frequency resolutions for low-frequency components that are, basically, the average intensity values of images. It also produces high temporal resolutions for high-frequency components that are, intrinsically, edges of images. The two major wavelet transforms are the continuous wavelet transform (CWT) and discrete wavelet transform (DWT). Images have smooth regions, interrupted by edges or abrupt changes. The abrupt changes are often the most interesting parts of the data, both perceptually and in terms of the information they provide. The wavelet transform represents these abrupt, well-localized changes efficiently, making it ideal for sparsifying natural images. In the proposed scheme, we have used the discrete wavelet transform to represent images sparsely [3].

2.1.2. Orthogonal Matching Pursuit

The OMP is an efficient and fast sparse signal recovery algorithm presented in [7]. It reconstructs the sparse signal, \bar{x} as closely as possible from the input measurement vector, y , and sensing matrix, Φ ($y = \Phi \bar{x}$). It works in iteration as follows. The OMP considers the sensing matrix as a set of N columns [$\Phi_1 \Phi_2 \dots \Phi_N$]

Step1: It finds the column Φ_j that has the largest projection on y as Equation (5) and build the basis matrix (A). From this basis matrix, it calculates the first best measurement \bar{x}_1 as in Equation (6).

$$i_1 = \arg \max_{1 \leq j \leq N} |\Phi_j^T y|. \tag{5}$$

$$A_1 = [\Phi_{i_1}]. \tag{6}$$

$$\bar{x}_1 = (A_1^T A_1)^{-1} A_1^T y. \tag{7}$$

the residue after the first iteration is

$$r_1 = (y - A_1 \bar{x}_1). \tag{8}$$

Step2: It finds the column Φ_j that has the largest projection on residue r_1 as Equation [8] and expand the basis matrix (A). From this basis matrix it calculates the next best measurement \bar{x}_2 as in Equation (11).

$$i_2 = \arg \max_{1 \leq j \leq N} |\Phi_j^T r_1|. \tag{9}$$

$$A_2 = [\Phi_{i_1} \Phi_{i_2}]. \tag{10}$$

$$\bar{x}_2 = (A_2^T A_2)^{-1} A_2^T y \tag{11}$$

the residue after the second iteration is

$$r_2 = (y - A_2 \bar{x}_2). \tag{12}$$

Repeat Step2 until the difference between the residue in successive iterations is less than the stopping criterion.

2.2. Lü Chaotic System

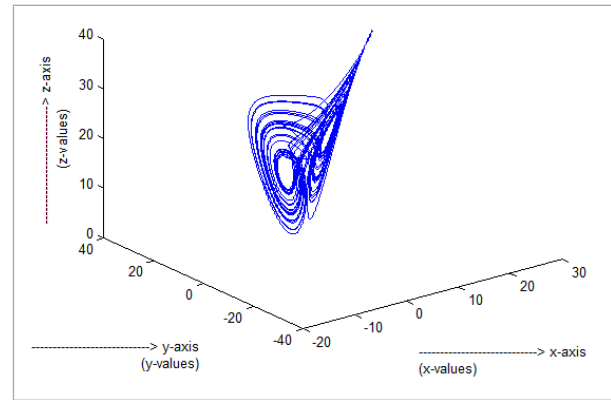
The motivation underlying the use of a chaotic system in our proposed scheme is its extreme sensitivity to initial conditions and its capacity for producing true random sequences. Key sensitivity and randomness play a vital role in security schemes. In this paper, the Lü chaotic dynamical system defined in Equation (14) is used in the encryption process,

$$\left. \begin{aligned} \frac{dx}{dt} &= a(y-x) \\ \frac{dy}{dt} &= -xz + cy \\ \frac{dz}{dt} &= xy - bz \end{aligned} \right\}, \tag{13}$$

where a, b and c are parameters x, y and z are state variables. The Lü system shows chaos behavior at the parameter values $a=36, b=3$ and $c=20$ [10-11]. Figure 2 shows the chaotic behavior of the Lü system. The two important aspects, stability and Lyapunov exponent, related to the chaotic behavior of Lü system are presented here,

Figure 2

The chaotic behavior of the Lü system



Stability of Equilibria

Stability means that nearby trajectories converges to the stable point. The three equilibrium points of Lü system are

$$\begin{aligned} F_0 &= (0, 0, 0) \\ F_+ &= (+\sqrt{bc}, +\sqrt{bc}, c) \\ F_- &= (-\sqrt{bc}, -\sqrt{bc}, c). \end{aligned}$$

The Jacobian matrix is used to analyze the stability of these equilibrium points. The Jacobian matrix to the system (13) is defined in Equation (14):

$$J = \begin{pmatrix} -a & a & 0 \\ -z & c & -x \\ y & x & -b \end{pmatrix}. \tag{14}$$

The Jacobian matrix at equilibrium F_0 is given by

$$J_{(0,0,0)} = \begin{pmatrix} -a & a & 0 \\ 0 & c & 0 \\ 0 & 0 & -b \end{pmatrix}.$$

The above matrix is an upper-triangular matrix and it's eigen values are $\lambda_1 = -a, \lambda_2 = c$ and $\lambda_3 = -b$. Since $a=36, b=3$ and $c=20$ then $\lambda_1 < 0, \lambda_2 > 0$ and $\lambda_3 < 0$.

The Jacobian matrix at equilibrium F_+ is given by

$$J_{(\sqrt{bc}, \sqrt{bc}, c)} = \begin{pmatrix} -a & a & 0 \\ -c & c & -\sqrt{bc} \\ \sqrt{bc} & \sqrt{bc} & -b \end{pmatrix}$$

Its eigen values are $f(\lambda) = \lambda^3 + (a + b - c)\lambda^2 + ab\lambda + 2abc = 0$, the coefficients of this cubic polynomial are all positive, since $(a + b - c > 0)$, so that $f(\lambda) > 0$ for all $\lambda > 0$.

The Jacobian matrix at equilibrium F_- is given by

$$J_{(-\sqrt{bc}, -\sqrt{bc}, c)} = \begin{pmatrix} -a & a & 0 \\ -c & c & \sqrt{bc} \\ -\sqrt{bc} & -\sqrt{bc} & -b \end{pmatrix}$$

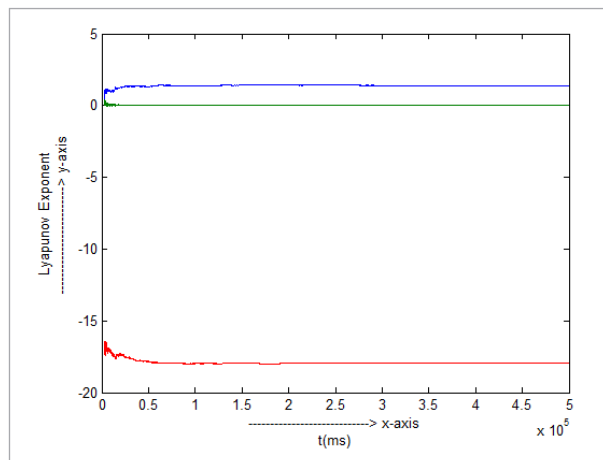
Its eigen values are $f(\lambda) = \lambda^3 + (a + b - c)\lambda^2 + ab\lambda + 2abc = 0$, the coefficients of this cubic polynomial are all positive, since $(a + b - c > 0)$, so that $f(\lambda) > 0$ for all $\lambda > 0$. From the above brief investigations, we found that the three equilibrium points are saddle focus-nodes. Hence, it is concluded that the Lü exhibits chaotic behaviour.

Lyapunov Exponent

The level of sensitivity to initial seeds in a dynamical system is defined by the Lyapunov Exponent (LE). A positive maximal Lyapunov exponent is the indica-

Figure 3

The Lyapunov exponents



tion of deterministic chaos [11]. The three Lyapunov exponents of the Lü system for the initial seeds ($x_0=-3, y_0=2, z_0=20$) are ($LE_1= 1.383848, LE_2= 0.000740$ and $LE_3= -17.934765$), as shown in Figure 3, since two Lyapunov exponents are positive for the Lü system, it is chaotic. These two key aspects make it clear that the Lü system is chaotic, and hence suitable for image cipher.

2.3. Secure Hash Algorithm-512

In this work, the SHA-512 is used to obtain the hash value of the plain image from which the keys are calculated. The input image-based keys greatly enhance the strength of the presented algorithm against known and chosen plaintext attacks. The SHA-512 is a hash algorithm that generates 512 bits of fixed-size values which are independent of the keys [23]. The SHA-512 processes the input in terms of block-sized 1024 bits and performs 80 round operations. The input is padded before applying the round functions in order to make the input size equal to multiples of 1024 bits. The key generation process is described in Section 3.1.

3. The Proposed Secured Compression Scheme

3.1. Secret Key Generation

The secret keys used in the presented scheme are shown in Figure 4, where the SHA-512 is the hash value and $[x_1, y_1, z_1, x_2, y_2, z_2, x_3, y_3, z_3]$ are the starting values of the Lü system.

Figure 4

Secret Keys

SHA-512	x_1	y_1	z_1	x_2	y_2	z_2	x_3	y_3	z_3
---------	-------	-------	-------	-------	-------	-------	-------	-------	-------

The 512-bit hash value is calculated by applying the secure hash algorithm on the input image. The starting values of the Lü system are calculated from the 512-bit hash, as given below.

Step1: The 512-bit hash is divided into 64 8-bit-sized blocks and converted into decimal numbers, d_1, d_2, \dots, d_{64} .

Step2: The procedure for obtaining the starting values $[x_p, y_p, z_p]$ used in the construction of the sensing matrix is given in Equation (15),

$$\begin{aligned}
 x_1 &= \frac{1}{2}(\text{mod}((d_1 \oplus d_{10} \oplus d_{19} \oplus d_{28} \oplus d_{37} \oplus d_{46} \oplus \\
 & d_{55} \oplus d_{64}), 256) + x_0) \\
 y_1 &= \frac{1}{2}(\text{mod}((d_2 \oplus d_{11} \oplus d_{20} \oplus d_{29} \oplus d_{38} \oplus d_{47} \oplus \\
 & d_{56} \oplus d_{64}), 256) + y_0) \\
 z_1 &= \frac{1}{2}(\text{mod}((d_3 \oplus d_{12} \oplus d_{21} \oplus d_{30} \oplus d_{39} \oplus d_{48} \oplus \\
 & d_{57} \oplus d_{64}), 256) + z_0)
 \end{aligned} \tag{15}$$

where \oplus is the XOR operation, and x_0, y_0 and z_0 are the initial seeds.

Step3: The procedure for obtaining the starting values $[x_2, y_2, z_2]$ used in the permutation is given in Equation (16):

$$\begin{aligned}
 x_2 &= \frac{1}{2}(\text{mod}((d_4 \oplus d_{13} \oplus d_{22} \oplus d_{31} \oplus d_{40} \oplus d_{49} \oplus \\
 & d_{58} \oplus d_{64}), 256) + x_0) \\
 y_2 &= \frac{1}{2}(\text{mod}((d_5 \oplus d_{14} \oplus d_{23} \oplus d_{32} \oplus d_{41} \oplus d_{50} \\
 & \oplus d_{59} \oplus d_{64}), 256) + y_0) \\
 z_2 &= \frac{1}{2}(\text{mod}((d_6 \oplus d_{15} \oplus d_{24} \oplus d_{33} \oplus d_{42} \oplus d_{51} \oplus \\
 & d_{60} \oplus d_{64}), 256) + z_0)
 \end{aligned} \tag{16}$$

Step4: The procedure for obtaining the starting values $[x_3, y_3, z_3]$ used in the substitution is given in Equation (17):

$$\begin{aligned}
 x_3 &= \frac{1}{2}(\text{mod}((d_7 \oplus d_{16} \oplus d_{25} \oplus d_{34} \oplus d_{43} \oplus d_{52} \\
 & \oplus d_{61} \oplus d_{64}), 256) + x_0) \\
 y_3 &= \frac{1}{2}(\text{mod}((d_8 \oplus d_{17} \oplus d_{26} \oplus d_{35} \oplus d_{44} \oplus d_{53} \\
 & \oplus d_{62} \oplus d_{64}), 256) + y_0) \\
 z_3 &= \frac{1}{2}(\text{mod}((d_9 \oplus d_{18} \oplus d_{27} \oplus d_{36} \oplus d_{45} \oplus d_{54} \oplus \\
 & d_{63} \oplus d_{64}), 256) + z_0)
 \end{aligned} \tag{17}$$

The starting values $[x_1, y_1, z_1], [x_2, y_2, z_2]$ and $[x_3, y_3, z_3]$ are calculated for every new plain image, and new keys used.

3.2. True Random Sequence Generation

The random sequences used in the construction of the sensing matrix, as well as in the permutation and substitution processes, are obtained using the Lü system, as follows:

Step1: Set the initial values (x_i, y_i and z_i) and parameters (a, b and c) of the system.

Step2: Solve the system using the Runge-Kutta scheme with a step interval of 0.0001.

Step3: Iterate the system according to the required length.

3.3. Sensing Matrix Construction

To recover the sparse signal with high precision, the sensing matrix must possess the restricted isometric property (RIP), such that it preserves significant information about the original signal. We construct a sensing matrix, based on the chaotic system, for two reasons. First, it is proved that chaos-based random sensing matrices satisfy the restricted isometric property with overwhelming probability [27]. Second, it circumvents the problem of sending the full sensing matrix to the reconstruction side. Typically, in compressive sensing, the sensing matrix must be sent to the reconstruction end, which needs a large memory space and bandwidth. In our approach, however, only the initial values and parameters of the Lü system need to be sent to efficiently reduce memory space and bandwidth. Hence, a sensing matrix constructed using the Lü system is efficient, and we have done so in this work. A chaotic incoherent sensing matrix is constructed as a circular matrix in which the first row of the matrix $\{R_1 = a_1, a_2, a_3 \dots a_n\}$ is obtained by iterating the Lü system 2500+N times. The former 2500 values of the obtained chaotic sequences (X, Y and Z) are then discarded to enhance the randomness in the sequence, and the remaining N values are processed according to Equation (18), where I_x, I_y and I_z are the starting values of the Lü system. The remaining rows are produced by the previous row moving one element to the right, and multiplied by the incoherence coefficient (i), to enhance the nonlinear correlations between the columns. The more the nonlinear correlations between the columns, the higher the precision in signal reconstruction. The matrix is normalized by the coefficient $\sqrt{\frac{1}{2m}}$, where m is the number of columns. Hence, the chaotic sensing matrix (19) has zero mean and zero symmetry.

$$\begin{aligned}
 X_{new} &= \text{mod}((\text{abs}(x_i) - \text{floor}(x_i)) \times 10^8, I_x) \\
 Y_{new} &= \text{mod}((\text{abs}(y_i) - \text{floor}(y_i)) \times 10^8, I_y) \\
 Z_{new} &= \text{mod}((\text{abs}(z_i) - \text{floor}(z_i)) \times 10^8, I_z) \\
 R_1(i) &= [X_{new}(i) \oplus Y_{new}(i) \oplus Z_{new}(i)]
 \end{aligned} \tag{18}$$

$$\sqrt{\frac{1}{2m}} \begin{bmatrix} a_1 & a_2 & \dots & a_{n-1} & a_n \\ ia_n & a_1 & a_2 & \dots & a_{n-1} \\ ia_{n-1} & i^3 a_n & a_1 & \dots & a_{n-2} \\ ia_{n-2} & i^3 a_{n-1} & i^5 a_n & \dots & a_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ ia_{n-m+2} & i^3 a_{n-m+3} & i^5 a_{n-m+3} & \dots & a_{n-m+1} \end{bmatrix} \cdot \Phi_{m \times n} = \tag{19}$$

3.4. Pixel Permutations

Permutation minimizes any regular patterns present in the cipher text, and includes the diffusion characteristics in the proposed cipher. Pixel permutation changes the position of pixels in compressively-sensed measurements, as shown below:

Step1: Set the initial values as $[x_2, y_2$ and $z_2]$ and iterate the Lü system to generate random sequences ($X2=\{x_p, x_2, \dots, x_M\}$, $Y2=\{y_p, y_2, \dots, y_M\}$ and $Z2=\{z_p, z_2, \dots, z_M\}$)).

Step2: Sort the random sequences in ascending order ($S_X2 = \{sx_p, sx_2, \dots, sx_M\}$, $S_Y2 = \{sy_p, sy_2, \dots, sy_M\}$ and $S_Z2 = \{sz_p, sz_2, \dots, sz_M\}$)).

Step3: Find the index permutation vector ($P_X2 = \{px_p, px_2, \dots, px_M\}$, $P_Y2 = \{py_p, py_2, \dots, py_M\}$ and $P_Z2 = \{pz_p, pz_2, \dots, pz_M\}$)).

Step4: Rearrange all the pixels in each color channel of the sensed measurements, according to the index permutation vector.

3.5. Pixel Substitutions

A combination of permutation, followed by substitution, creates good mixing properties in the cipher. Substitution includes the confusion characteristics in the proposed cipher. Pixel substitution changes the value of pixels in the measurement vector, as shown below:

Step1: Set the initial values as $[x_3, y_3$ and $z_3]$ and iterate the Lü system to generate random sequences ($X3=\{x_p, x_3, \dots, x_M\}$, $Y3=\{y_p, y_3, \dots, y_M\}$ and $Z3=\{z_p, z_3, \dots, z_M\}$)).

Step2: Preprocess the random sequences using Equation (20):

$$\begin{aligned} X3 &= X_i \times 10^{14} \text{ mod } 256 \text{ where } i = 1, 2, \dots, M \\ Y3 &= Y_i \times 10^{14} \text{ mod } 256 \text{ where } i = 1, 2, \dots, M \\ Z3 &= Z_i \times 10^{14} \text{ mod } 256 \text{ where } i = 1, 2, \dots, M \end{aligned} \tag{20}$$

Step3: Perform pixel-by-pixel substitution in each color channel using Equation (21),

$$\begin{aligned} Y_R &= de2bi(CS_{R_i}) \oplus de2bi(X3_i) \\ Y_G &= de2bi(CS_{G_i}) \oplus de2bi(Y3_i) \\ Y_B &= de2bi(CS_{B_i}) \oplus de2bi(Z3_i), \end{aligned} \tag{21}$$

where CS_{R_i} , CS_{G_i} and CS_{B_i} are the color components of the sensed measurements (CS).

3.6. Compression and Encryption (CE)

A diagram of the proposed CE method is shown in Figure 1. The joint CE process is presented here:

Step1: Apply the SHA-512 hash method on the input color image, I, and obtain a 512-bit hash from which the initial values of the Lü system are calculated as described in Section 3.1.

Step2: Generate true random chaotic sequences by setting the initial values and parameters to the Lü system and iterating them as detailed in Section 3.2.

Step3: Build the circular incoherence sensing matrix, Φ , with the restricted isometric property using the chaotic sequence given in Section 3.3.

Step4: Represent the input image sparsely in the discrete wavelet transform basis (ψ), and obtain sparse transform coefficients using

$$I = \psi \alpha.$$

Step5: Compress the DWT sparse coefficients using the circular incoherence sensing matrix and obtain measurements (CS) using

$$CS = \Phi \psi \alpha.$$

Step6: Perform uniform quantization on the measurements in the range 0 to 255 using Equation (22),

$$CS_q = \text{round} \left[255 \times \frac{CS - CS_{min}}{CS_{max} - CS_{min}} \right], \tag{22}$$

where CS_{max} and CS_{min} are the minimum and maximum values of the CS measurements, and round (k) rounds the value of k to the nearest integer.

Step7: Perform the pixel permutation operation on the sensed measurement given in Section 3.4.

Step8: Perform the pixel substitution operation on the permuted measurement presented in Section 3.5.

Step9: Reshape the substituted measurements to obtain the compressed-encrypted image, C.

In the reconstruction process, inverse substitution is performed, followed by inverse pixel scrambling. The OMP is then used to recover the sparse representation of the image and finally, the inverse discrete wavelet transform is applied to get the reconstructed image (\hat{I}).

4. Experimental Results

To evaluate the compression and encryption performance of the proposed scheme, extensive experimental tests were conducted in MATLAB R2014a. The natural color images shown in Figure 5, with pixels sized 512×512 , are used as test plain images. The param-

eters $a = 35$, $b = 3$ and $c = 20$, and the starting seeds $x_0 = -3$, $y_0 = 2$ and $z_0 = 20$, are used in the experiments. The input images are sparsely represented by employing the biorthogonal wavelet transform as the orthogonal transform basis (ψ), along with single-level decomposition. The sparse transform coefficients are compressed thereafter, using the chaotic circular sensing matrix. Finally, the measurements are permuted and substituted, based on the true random chaotic sequences, to produce a compressed-encrypted image. The orthogonal matching pursuit sparse reconstruction algorithm is used for image reconstruction. The input plain image, single-level decomposed image, compressed-encrypted image and corresponding reconstructed image of test image 11 are shown in Figure 6.

Figure 5
The Input Images

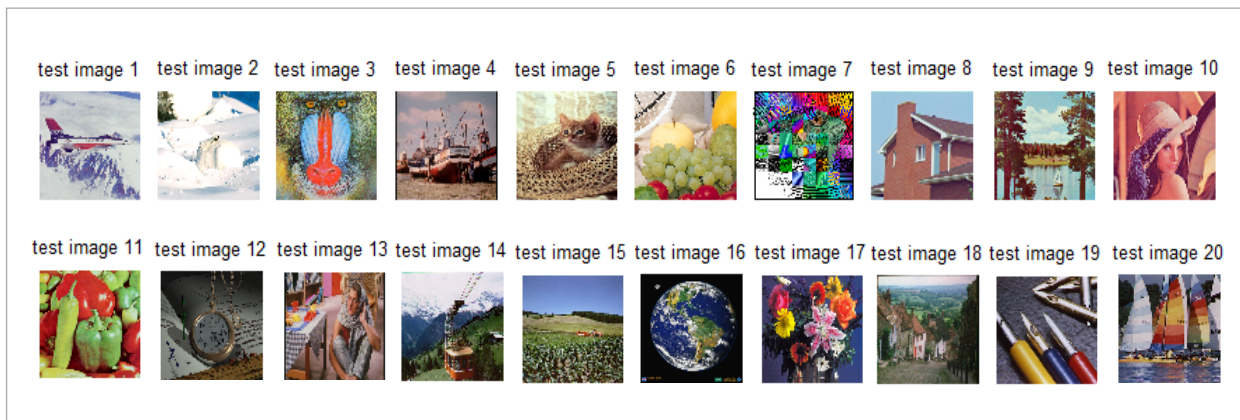
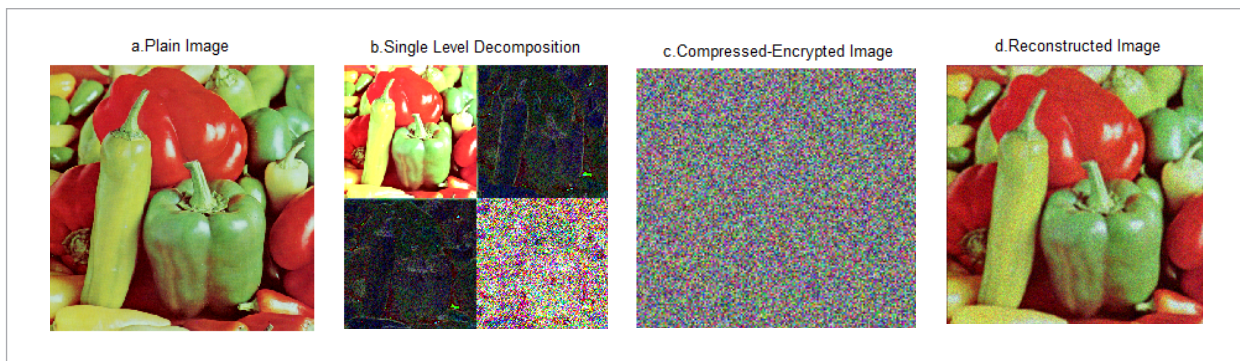


Figure 6
Plain test image 11, single-level decomposed image, compressed-encrypted image and corresponding reconstructed image



5. Results Analysis

A performance evaluation of the proposed scheme is carried out in two categories. In the first, image quality metrics such as the peak signal-to-noise ratio, structural similarity index, average difference, structural content, normalized cross-correlation, normalized absolute error, edge strength similarity and maximum difference are used to test the reconstruction ability of the proposed system. In the second category, the strength of encryption of the proposed scheme is tested, based on the correlation coefficient, UACI, key sensitivity, NPCR, key space and histogram metrics. Experimental results demonstrate that the proposed scheme produced highly satisfactory results in terms of security.

Peak Signal-to-Noise Ratio (PSNR)

The PSNR and SSIM quality metrics are used to assess the standard of the reconstructed image. The PSNR is a measure of image fidelity that computes the proportion between the peak image pixel value and corrupting noise power [26]. The PSNR is defined by Equation (23),

$$PSNR = 10 \log_{10} \left[\frac{255^2}{MSE} \right], \tag{23}$$

where MSE is the mean squared error. The blocks are obtained by splitting the wavelet coefficient image into non-overlapping blocks (B) sized $(\frac{M}{2^{DL}} \times \frac{N}{2^{DL}})$ as shown in Figure 7, where DL denotes the decomposition level. Table 1 lists the total number of sparse vectors and their length for an image. Table 2 lists

Figure 7

Wavelet coefficient image segmentation into blocks and sparse vector

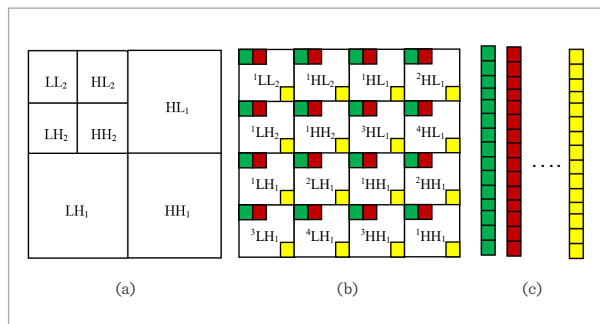


Table 1

Total number of sparse vectors and their lengths for an image (512 × 512)

Decomposition-Level (DL)	Number of Sparse Vector	Sparse Vector Length
2	16384	16
3	4096	64
4	1024	256
5	256	1024
6	64	4096
7	16	16384

Table 2

PSNR test results of the proposed system

Input Image	Reconstructed Images on Different Block Sizes		
	8 × 8	16 × 16	32 × 32
PSNR Values in db			
Input Image 1	34.1587	34.7492	35.8723
Input Image 2	36.5571	37.1908	38.0317
Input Image 3	34.9326	35.4641	36.5452
Input Image 4	34.2225	34.8672	35.9675
Input Image 5	36.0875	36.6820	37.7443
Input Image 6	36.8196	37.6179	38.7457
Input Image 7	34.9459	35.4932	36.6806
Input Image 8	37.7538	38.3086	39.1460
Input Image 9	35.8589	36.4751	37.7533
Input Image 10	38.3305	38.0462	39.2286
Input Image 11	37.0632	37.6342	38.8737
Input Image 12	36.0199	36.6491	37.7490
Input Image 13	37.6381	38.2429	39.1485
Input Image 14	34.9170	34.4367	35.7914
Input Image 15	36.9003	37.5477	38.4470
Input Image 16	35.8047	35.4998	36.6618
Input Image 17	36.7375	37.3173	38.2987
Input Image 18	37.7809	38.2540	39.2547
Input Image 19	36.1160	36.6729	37.5092
Input Image 20	35.3275	36.0049	36.9416

the PSNR values obtained by using the proposed approach with respect to different block sizes and the Table 3 shows the PSNR test result comparisons.

Table 3
PSNR test results comparison

Test Images (512 × 512)	PSNR Values in db (block size is 32 × 32)			
	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]
test image 1	35.8723	34.0112	35.5183	31.7734
test image 2	38.0317	36.4751	34.9296	32.0519
test image 3	36.5452	36.5092	37.7236	31.8564
test image 4	35.9675	34.7314	35.7290	31.5722
test image 5	37.7443	35.3542	33.9897	30.9259

Structural Similarity Index (SSIM)

The SSIM measures the resemblance between the plain image (P) and recovered image (R) in terms of three features: luminance, contrast and structure. The SSIM value ranges between 0 and 1 [36]. Image distortion is smaller when the index value approaches 1. The SSIM is measured by Equation (24),

$$SSIM_{(x,y)} = \frac{(2\mu_P\mu_R+c_1)(2\sigma_{PR}+c_2)}{(\mu_P^2+\mu_R^2+c_1)(\sigma_P^2+\sigma_R^2+c_2)}, \tag{24}$$

where $\mu_P, \mu_R, \sigma_{PR}, \mu_P^2, \sigma_P^2, c_1$ and c_2 are the mean of P, the mean of R, the covariance of P and R, the variance of P, the variance of R, and the constant of c_1 and c_2 , respectively.

Average Difference (AD)

The AD provides the average of change concerning the reconstructed and original plain images which, ideally, should be zero. The AD is defined by Equation (25),

$$AD = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N [P_{i,j} - R_{i,j}], \tag{25}$$

where $P_{i,j}$ and $R_{i,j}$ are the pixel values at location (i,j) of the plain and reconstructed images, respectively, M is the height and N the width of the image.

Structural Content (SC)

The SC is a kind of correlation-based metric that depends on the similarity between the original and

reconstructed images, with a higher SC value reflecting a poor quality image. The SC is expressed by Equation (26):

$$SC = \frac{\sum_{i=1}^M \sum_{j=1}^N (P_{i,j})^2}{\sum_{i=1}^M \sum_{j=1}^N (R_{i,j})^2}. \tag{26}$$

Normalized Cross-Correlation (NCC)

The NCC metric compares the reconstructed and reference images, and is invariant to local changes in intensity and brightness. In the NCC, the correlation is normalized by dividing the cross-correlation by the summation of the squares of the pixel values of the plain image. The NCC is calculated by Equation (27):

$$NCC = \sum_{i=1}^M \sum_{j=1}^N \frac{P_{i,j} \times R_{i,j}}{P_{i,j}^2}. \tag{27}$$

Normalized Absolute Error (NAE)

The NAE is a proportion of the summation of differences between the plain and reconstructed pixel values to the summation of the pixel values of the plain image. The NAE, with an ideal value of 0, is defined by Equation (28):

Edge Strength Similarity (ESSIM)

This image quality metric measures the similarity between the edge strength of the plain and reconstructed images. The ESSIM is calculated by Equation (29),

$$ESSIM = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N \frac{2E(P_{i,j})E(R_{i,j})+c}{(E(P_{i,j}))^2+(E(R_{i,j}))^2+c}, \tag{29}$$

where $E(P)$ and $E(R)$ are the edge strengths in the vertical direction of the plain and reconstructed images, respectively.

Maximum Difference (MD)

The MD quantitatively provides the maximum difference between the reconstructed and referenced images. A small MD value means that the reconstructed image is of good quality [23], while a large value implies poor image quality. The MD is measured by Equation (30):

$$MD = MAX|P(i,j) - R(i,j)|. \tag{30}$$

The results of all the tests above, presented in Table 4, reveal that the sparse reconstruction process has produced images of acceptable quality.

Table 4

Experimental result of SSIM, SC, AD, MD, NCC, NAE, ESSIM, NPCR and UACI tests

Input Image	SSIM	SC	AD	MD	NCC	NAE	ESSIM	NPCR	UACI
test image 1	0.6571	0.7292	0.0236	127.5809	0.7142	0.0375	0.8595	99.6280	33.4208
test image 2	0.7035	0.7324	0.0173	143.1014	0.7210	0.0184	0.8392	99.6191	33.3896
test image 3	0.5992	0.6504	0.0516	138.9701	0.6505	0.0231	0.8586	99.6204	33.4884
test image 4	0.6414	0.6378	0.0239	112.2179	0.6319	0.0617	0.8194	99.6188	33.4075
test image 5	0.6694	0.7446	0.0345	109.1685	0.7137	0.0742	0.8491	99.6185	33.3928
test image 6	0.7152	0.6500	0.0271	157.9079	0.7208	0.0377	0.8385	99.6089	33.4058
test image 7	0.6206	0.6483	0.0195	213.6174	0.6706	0.0810	0.8178	99.6227	33.3913
test image 8	0.7205	0.7513	0.0170	124.3669	0.6800	0.0296	0.8596	99.6115	33.4131
test image 9	0.7007	0.5862	0.0301	134.8350	0.6935	0.0691	0.8494	99.6131	33.3501
test image 10	0.6984	0.6902	0.0198	110.3204	0.6926	0.0480	0.8296	99.6164	33.3351
test image 11	0.6594	0.7430	0.0195	143.0355	0.6961	0.0494	0.8526	99.6126	33.2577
test image 12	0.6596	0.7153	0.0225	137.0326	0.6772	0.0543	0.8353	99.6112	34.1015
test image 13	0.7021	0.7069	0.0341	126.7063	0.6892	0.0379	0.8174	99.6156	33.3342
test image 14	0.6637	0.6599	0.0213	119.9554	0.7186	0.0605	0.8459	99.6027	33.4178
test image 15	0.6038	0.6402	0.0266	144.2960	0.6018	0.0443	0.8203	99.6099	33.3102
test image 16	0.6713	0.6357	0.0197	135.6501	0.6652	0.0332	0.8295	99.6252	33.4158
test image 17	0.6238	0.7267	0.0204	126.5483	0.6281	0.0542	0.8158	99.6092	33.3894
test image 18	0.7103	0.6090	0.0263	129.4447	0.7034	0.0612	0.8079	99.6106	33.3182
test image 19	0.5897	0.6213	0.0457	159.6584	0.5528	0.0476	0.8310	99.6038	33.2954
test image 20	0.6341	0.7010	0.0267	132.0315	0.5801	0.0327	0.8426	99.6046	33.3005

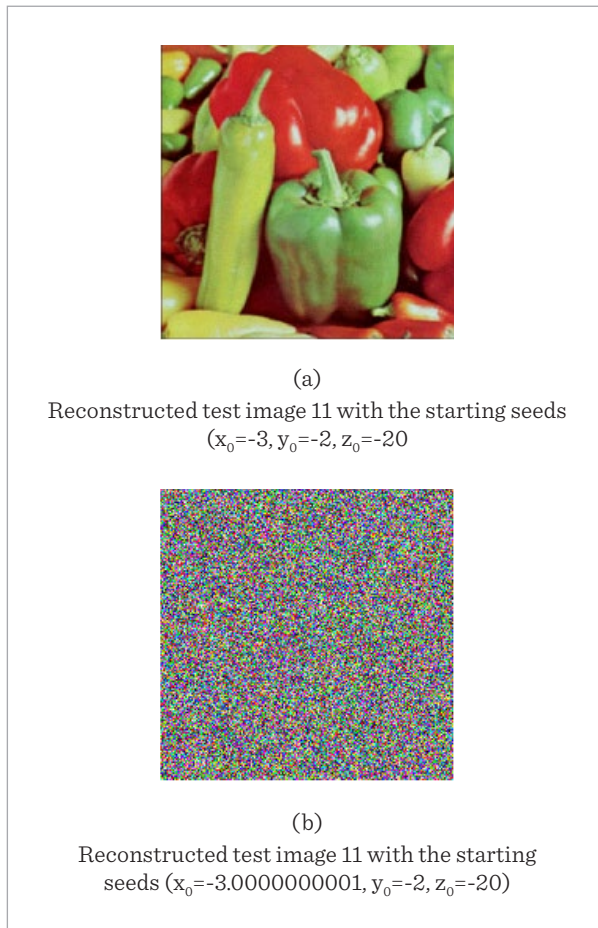
Key Space Analysis

It is widely accepted that secret keys play a critical role in the security of a cryptosystem. If everything about an algorithm is known to the public, then the secret key is everything in terms of security. Hence, to withstand bruteforce attacks, an effective cipher scheme should have a key space of at least 2^{250} [24]. The keys of this method are the 512-bit image hash, with starting values of $[x_p, y_p, z_p, x_{2p}, y_{2p}, z_{2p}, x_{3p}, y_{3p}$ and $z_{3p}]$ and a valid precision of 10^{-14} . Hence, the total key space approximately equals $2^{904} [2^{512} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14}]$, which is good enough to strongly resist key-based attacks.

Key Sensitivity Analysis

Key sensitivity is salient, which means that a small change in the secret key produces a completely different cipher image. The Lü system is highly sensitive to the starting values, which are used as secret keys in the proposed system. The key sensitivity test of the proposed system is carried out as follows. Test image 11 is compressed and encrypted with the starting seed values of $x_0=-3, y_0=2$ and $z_0=20$. The ciphered test image 11 is then reconstructed using the same starting seeds, and the slightly-changed starting seed values of $x_0=-3.0000000001, y_0=2$ and $z_0=20$. Figure 8 shows the two reconstructed images. From the results, it is clearly seen that the proposed system has great key sensitivity.

Figure 8
Key Sensitivity analysis



Histogram Analysis

A histogram is distinctive for every image. Given that this statistical feature can be perceived and exploited by an attacker, a strong cryptosystem must provide a fairly uniform distribution of the pixel intensities of the ciphered image [27, 31]. Figs. 9 (b), (d) and (f) show the histograms of the original images, compressed-encrypted images and reconstructed images, respectively. It is clear that since the histogram of the compressed-encrypted image is flat, no useful information can be obtained by interpreting the histograms of the compressed-encrypted images.

Correlation Coefficient Analysis (CC)

A strong correlation exists between adjacent pixels of digital images. The CC closeness measure can be

used to assess the performance of an algorithm’s resistance to statistical attacks. Since the CC of plain images is usually near 1 [2], the CC of the cipher image should, ideally, be close to 0. To measure the CC, 3000 pixels were randomly selected along three directions (vertical, horizontal and diagonal) and the CC obtained using Equation (31),

$$NAE = \frac{\sum_{i=1}^M \sum_{j=1}^N (|P_{i,j} - R_{i,j}|)}{\sum_{i=1}^M \sum_{j=1}^N P_{i,j}} \tag{31}$$

where $covar(P,R)$, $D(P)$, $D(R)$ is the covariance between P and R , variance of P , and variance of R , respectively. A correlation can range in value from -1.00 to +1.00, where a zero value correlation coefficient represents no relationship between the adjacent pixels being compared, while a greater-than-zero value represents a positive relationship, and a less-than-zero (negative) value represents a negative (or indirect) relationship.

Table 5 lists the correlation coefficients of all the test images. Figure 10 shows the correlation of the compressed-encrypted image for test image 6. It is seen from Table 5 and Figure 10 that the proposed scheme produced a very small degree of correlation between adjacent pixels, thus proving that it withstands statistical attacks.

NPCR and UACI Analysis

Commonly, the two diffusion performance criteria, NPCR and UACI, are employed to analyze the performance of the algorithm against differential attacks [26]. The NPCR measures the percentage of dissimilitude in pixel numbers between two ciphered images such that the corresponding plain images differ in only one pixel. The NPCR is calculated by Equation (32),

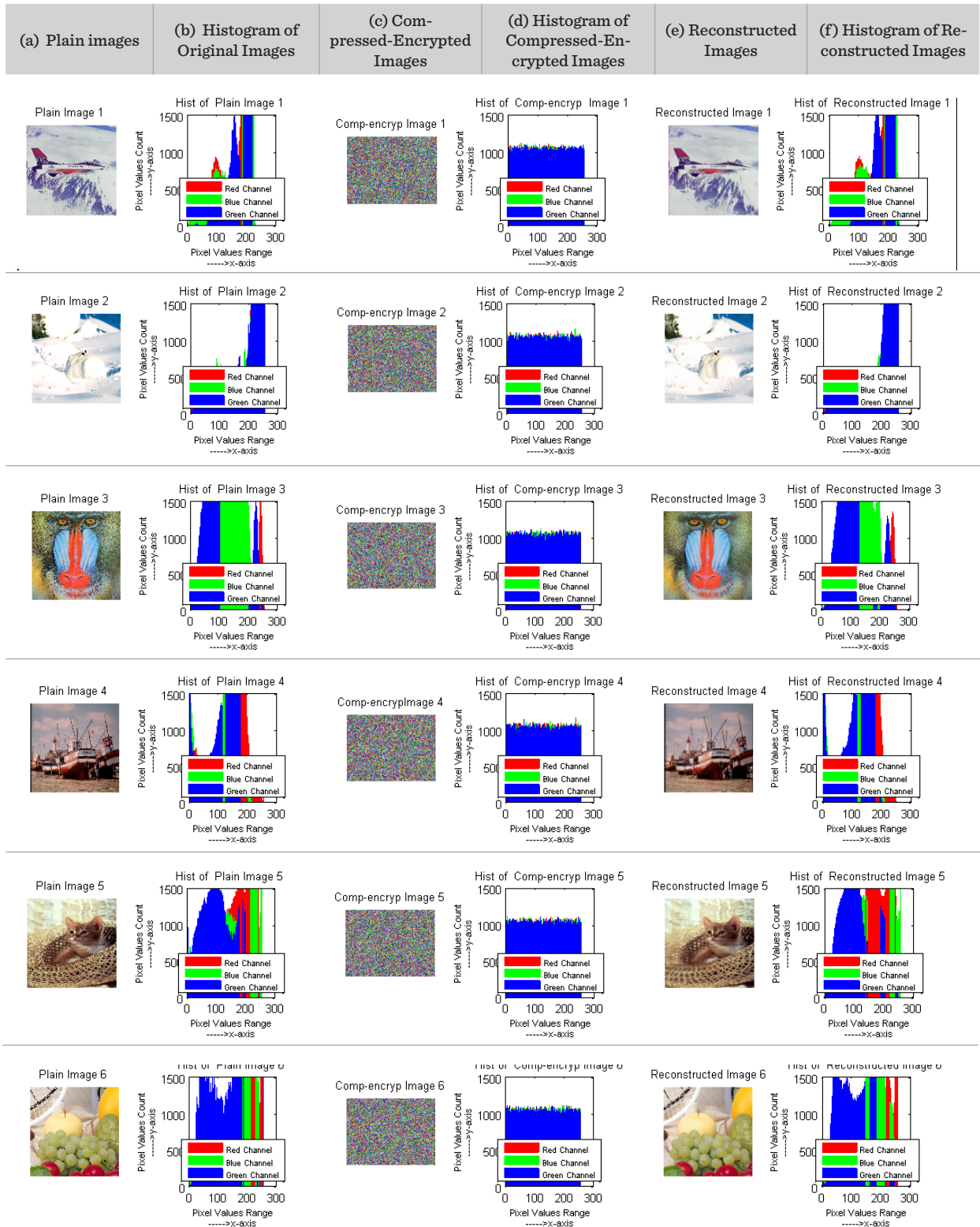
$$NPCR = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N D(i, j) 100\% , \tag{32}$$

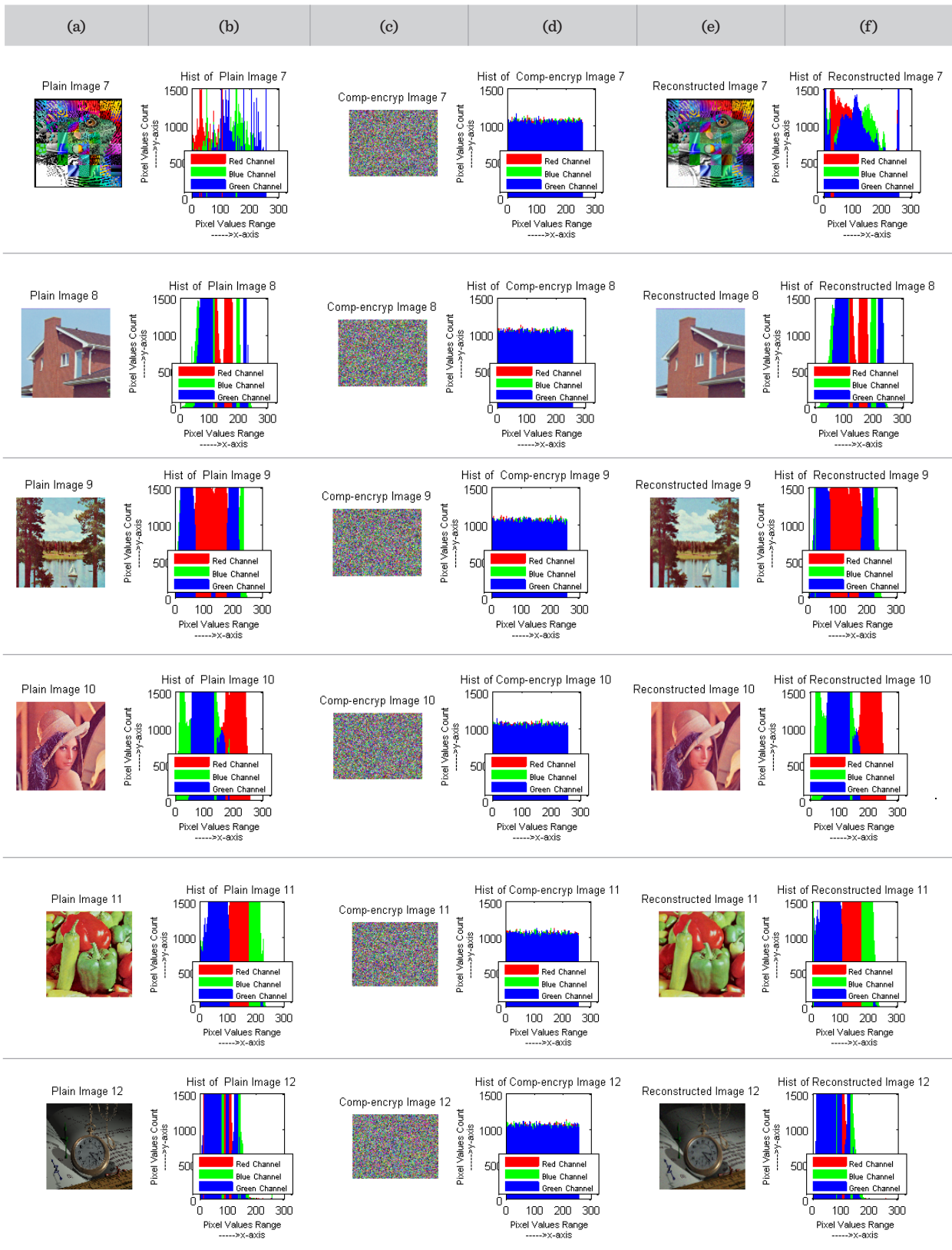
where $D(I, j)$ denotes the difference value of the corresponding pixel of two images.

The UACI measures changes in visual effects, and is calculated using Equation (33):

$$UACI = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N \frac{|P(i,j) - R(i,j)|}{255} \times 100\% . \tag{33}$$

Figure 9
Histogram analysis





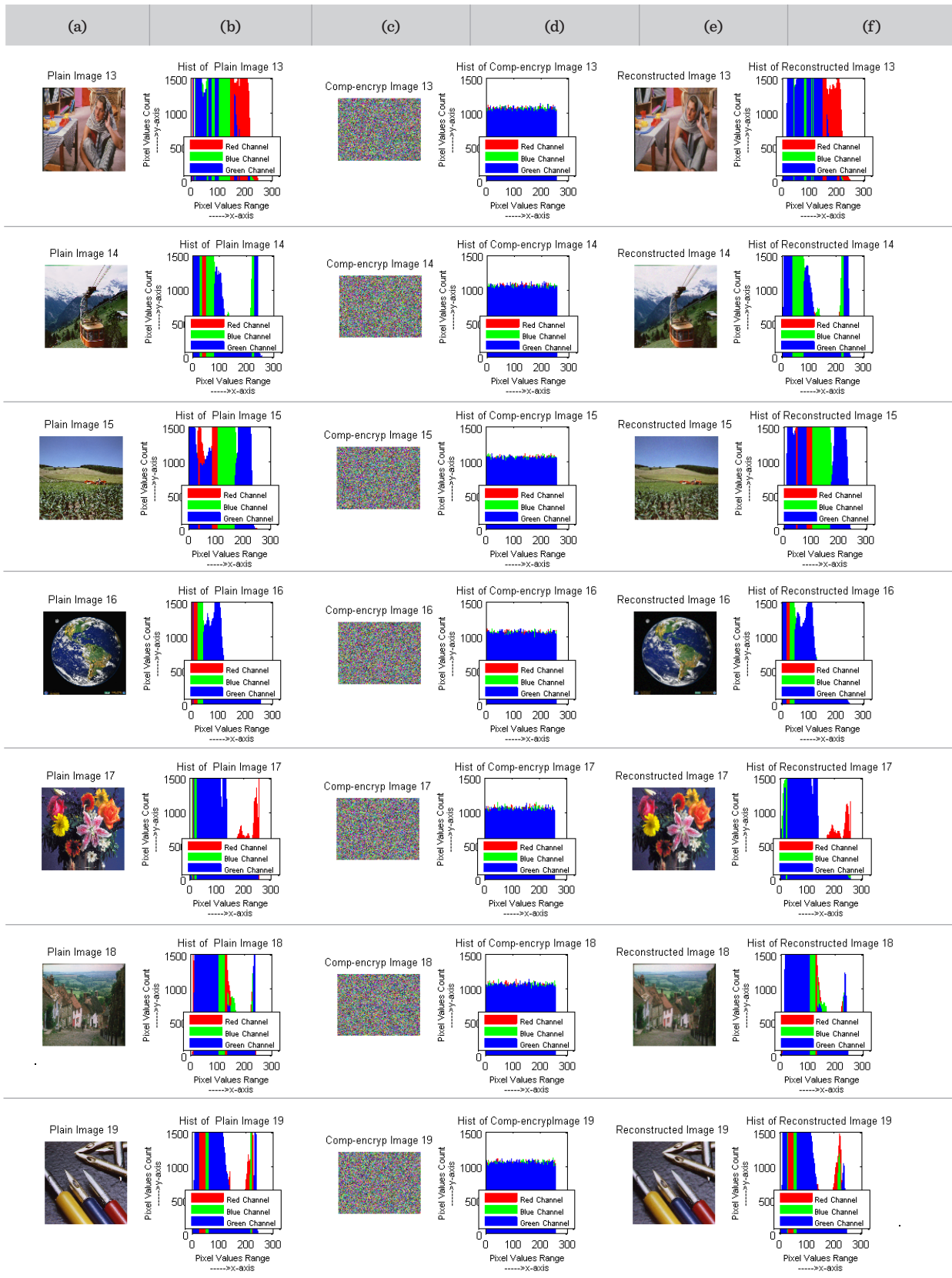


Table 5

Experimental result of horizontal, vertical and diagonal correlation coefficient tests

Image	Direction	Plain image	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]
test image 1	H	0.9575	0.0004	0.0061	0.0053	0.0037
	V	0.9587	0.0011	0.0019	0.0013	0.0016
	D	0.9253	0.0017	-0.0013	0.0088	0.0021
test image 2	H	0.9910	0.0010	0.0087	0.0047	0.0072
	V	0.9930	0.0014	0.0064	0.0077	0.0050
	D	0.9842	0.0007	0.0080	-0.0009	-0.0027
test image 3	H	0.9051	0.0029	-0.0005	0.0028	0.0035
	V	0.8527	0.0018	0.0083	0.0044	0.0003
	D	0.8232	-0.0005	0.0041	0.0028	-0.0072
test image 4	H	0.9723	-0.0024	0.0200	0.0084	-0.1443
	V	0.9765	0.0016	0.0029	0.0022	0.0034
	D	0.9506	0.0038	0.0043	0.0026	-0.0584
test image 5	H	0.9840	0.0031	0.0089	0.0070	0.0085
	V	0.9547	0.0005	0.0024	0.0074	0.0060
	D	0.9410	0.0028	0.0026	0.0063	0.0047
test image 6	H	0.9810	0.0001	0.0076	0.0058	0.0117
	V	0.9821	0.0025	0.0031	-0.0009	0.0026
	D	0.9678	0.0013	0.0009	0.0049	0.0021
test image 7	H	0.8944	0.0029	0.0061	0.0019	0.0046
	V	0.8844	0.0031	0.0055	0.0012	0.0024
	D	0.8196	0.0026	0.0067	0.0056	0.0054
test image 8	H	0.9930	0.0023	0.0060	0.0089	0.0033
	V	0.9901	-0.0002	0.0031	0.0055	0.0070
	D	0.9834	0.0024	0.0064	0.0075	0.0029
test image 9	H	0.9580	-0.0008	0.0057	0.0025	0.0082
	V	0.9593	0.0020	0.0053	0.0038	0.0054
	D	0.9251	0.0009	-0.0011	0.0040	0.0013
test image 10	H	0.9778	0.0007	0.0026	-0.0007	0.0001
	V	0.9881	-0.0004	0.0099	0.0079	0.0081
	D	0.9695	0.0021	0.0021	0.0068	0.0011
test image 11	H	0.9790	-0.0013	0.0059	0.0076	0.0055
	V	0.9820	0.0020	0.0054	0.0049	0.0078
	D	0.9691	0.0009	0.0079	0.0078	0.0014
test image 12	H	0.9539	0.0026	0.0019	0.0026	0.0025
	V	0.9548	0.0029	0.0074	0.0055	0.0019
	D	0.9403	0.0078	0.0086	0.0027	0.0081
test image 13	H	0.9292	-0.0193	0.0037	0.0049	0.0012
	V	0.9733	0.0008	0.0066	0.0078	0.0037
	D	0.9094	0.0012	0.0015	0.0004	0.0026

Image	Direction	Plain image	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]
test image 14	H	0.9869	0.0026	0.0099	0.0077	0.0060
	V	0.9836	-0.0275	0.0071	-0.0021	-0.0217
	D	0.9660	0.0006	-0.0307	0.0004	0.0024
test image 15	H	0.9520	-0.0117	0.0082	0.0028	0.0023
	V	0.9460	0.0005	0.0064	0.0014	0.0091
	D	0.9071	0.0024	0.0051	0.0059	0.0032
test image 16	H	0.9526	0.0036	0.0028	0.0193	0.0013
	V	0.9416	-0.1446	0.0093	0.0078	0.0086
	D	0.9189	0.0001	0.0015	0.0032	0.0033
test image 17	H	0.9704	0.0018	0.0042	0.0057	0.0014
	V	0.9842	0.0007	0.0026	0.0043	0.0074
	D	0.9608	0.0024	0.0022	-0.0031	0.0055
test image 18	H	0.9803	0.0015	-0.0182	-0.0030	-0.0085
	V	0.9847	-0.0069	0.0085	0.0021	0.0063
	D	0.9660	0.0005	0.0047	0.0032	0.0029
test image 19	H	0.9883	0.0007	0.0050	0.0064	0.0077
	V	0.9866	0.0019	0.0062	0.0019	0.0026
	D	0.9811	-0.0043	-0.0037	-0.0033	-0.0105
test image 20	H	0.9676	0.0023	0.0084	0.0095	0.0008
	V	0.9727	0.0009	0.0006	0.0052	0.0063
	D	0.9370	-0.0173	0.0010	0.0017	0.0049

Figure 10

The adjacent pixel distribution of test image 6 in horizontal, vertical and diagonal direction before and after encryption

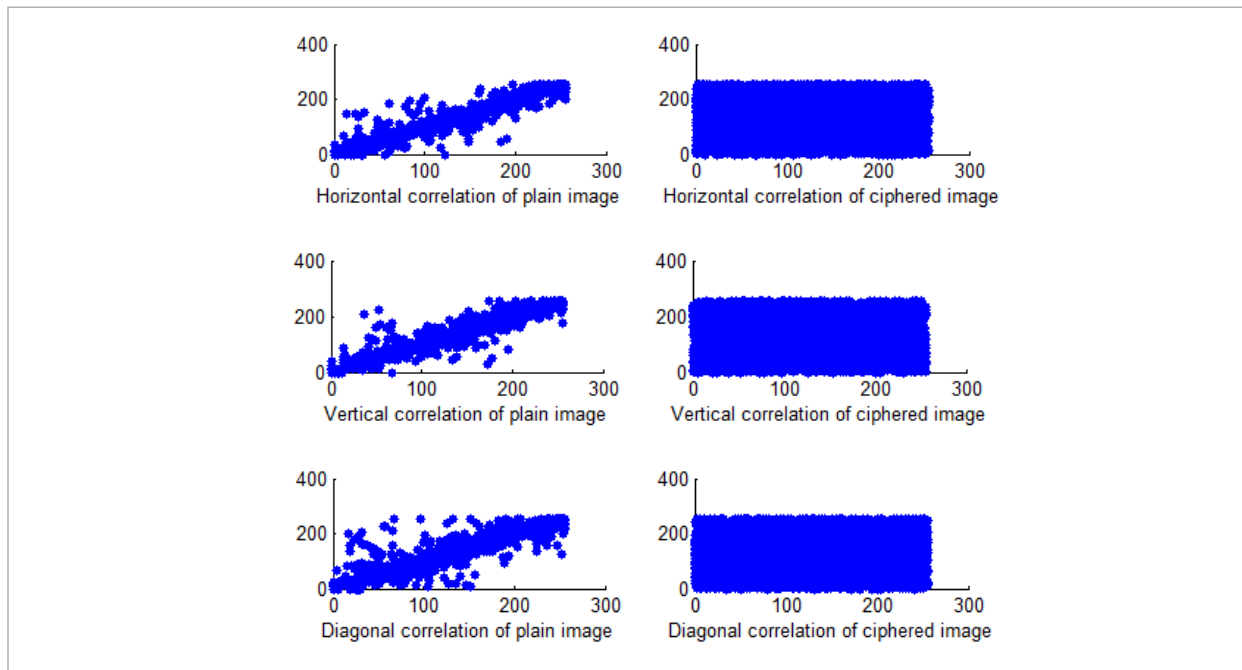


Table 6
Comparison of efficiency of the proposed approach with recent methods

Test Images (512 × 512)	SSIM values				SC values			
	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]
test image 1	0.6571	0.5756	0.6803	0.7023	0.7292	0.7133	0.7206	0.6810
test image 2	0.7035	0.6295	0.5398	0.6809	0.7324	0.7208	0.7286	0.6731
test image 3	0.5992	0.5406	0.6351	0.6013	0.6504	0.7466	0.7160	0.6345
test image 4	0.6414	0.6919	0.6835	0.7142	0.6378	0.6871	0.7031	0.6392
test image 5	0.6694	0.5184	0.5965	0.6274	0.7446	0.7354	0.7462	0.7026
AD values				MD values				
	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]
test image 1	0.0236	0.0210	0.0240	0.0308	127.5809	138.6075	125.0411	133.4250
test image 2	0.0173	0.0286	0.0269	0.0286	143.1014	156.3751	161.3715	157.7005
test image 3	0.0516	0.0609	0.0439	0.0599	138.9701	157.4271	145.1682	141.1397
test image 4	0.0239	0.0584	0.0310	0.0406	112.2179	124.3782	119.7198	105.8746
test image 5	0.0345	0.0232	0.0296	0.0367	109.1685	102.6950	106.8396	119.2542
NCC values				NAE values				
	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]
test image 1	0.7142	0.6889	0.7008	0.6463	0.0375	0.0447	0.0379	0.0428
test image 2	0.7210	0.6955	0.6937	0.6578	0.0184	0.0208	0.0352	0.0376
test image 3	0.6505	0.7002	0.7206	0.6104	0.1231	0.0672	0.0603	0.0759
test image 4	0.6319	0.6922	0.7092	0.6097	0.0617	0.0634	0.0620	0.0718
test image 5	0.7137	0.6690	0.6851	0.6390	0.0742	0.0616	0.0558	0.0637
NPCR				UACI				
	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]
test image 1	99.6280	99.6083	99.6059	99.5819	33.4208	33.3288	33.3797	33.0471
test image 2	99.6191	99.6109	99.6105	99.5832	33.3896	33.3769	33.3836	32.9548
test image 3	99.6204	99.6112	99.6124	99.6058	33.4884	33.3733	33.3722	32.9609
test image 4	99.6188	99.6209	99.6131	99.5963	33.4075	33.4148	33.4026	32.8900
test image 5	99.6185	99.6069	99.6147	99.5795	33.3928	33.3859	33.3592	33.0110
ESSIM values				Compression-Encryption Time (sec)				
	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]	Proposed	Shuqin et al. [28]	Liya et al. [16]	Xinsheng et al. [32]
test image 1	0.8595	0.8427	0.8795	0.8179	3.5393	2.7505	6.1688	11.7105
test image 2	0.8392	0.8016	0.8196	0.7587	4.3531	2.9339	6.1036	10.5035
test image 3	0.8586	0.8556	0.8489	0.8063	3.9194	3.2943	5.8331	10.3831
test image 4	0.8194	0.8127	0.8092	0.7939	3.4765	2.9800	5.6303	12.1097
test image 5	0.8491	0.8472	0.8596	0.8175	3.6629	3.4311	5.7215	10.6314

Table 7

The NIST suite test results of compressed-encrypted testimage-10

Statistical Test	Block/Temp (size)	p-value	D-R level	Result	Conclusion
Frequency (Monobit)	-	0.469022	3%	pass	random
Frequency (within a block)	10	0.847349	3%	Pass	random
Runs	-	0.791520	3%	pass	random
Longest run (once in a block)	10 ⁴	0.663015	3%	pass	random
Rank (Binary matrix)	-	0.430209	3%	pass	random
Discrete Fourier Transform (Spectral)	-	0.514816	3%	pass	random
Non-overlapping template matching	10	0.655314	3%	pass	random
Overlapping template matching	9	0.812206	3%	pass	random
Universal (Maurer)	12	0.445638	3%	pass	random
Linear complexity	10 ³	0.591405	3%	pass	random
Serial	-	0.832110	3%	pass	random
Approximate entropy	-	0.542388	3%	pass	random
Cumulative sums	-	0.498896	3%	pass	random
Random excursions	-	0.536804	3%	pass	random
Random excursions variant	-	0.516897	3%	pass	random

Table 8

The NIST suite test results of compressed-encrypted testimage-20

Statistical Test	Block/Temp (size)	p-value	D-R level	Result	Conclusion
Frequency (Monobit)	-	0.718365	3%	pass	random
Frequency (within a block)	10	0.497408	3%	Pass	random
Runs	-	0.560134	3%	pass	random
Longest- run (once in a block)	10 ⁴	0.408517	3%	pass	random
Rank (Binary matrix)	-	0.798925	3%	pass	random
Discrete- Fourier-Transform (Spectral)	-	0.690325	3%	pass	random
Non-overlapping template matching	10	0.533291	3%	pass	random
Overlapping template matching	9	0.658403	3%	pass	random
Universal (Maurer)	12	0.860152	3%	pass	random
Linear complexity	10 ³	0.626527	3%	pass	random
Serial	-	0.558243	3%	pass	random
Approximate entropy	-	0.572843	3%	pass	random
Cumulative sums	-	0.603549	3%	pass	random
Random excursions	-	0.483721	3%	pass	random
Random excursions variant	-	0.631473	3%	pass	random

Computational Complexity

The computation complexity of the proposed technique is evaluated in terms of time complexity. This method utilizes the faster logical exclusive-or operation and a single round of encryption to minimize, as far as feasible, the time consumed. The running time of the proposed method is listed in Table 9. As can be seen from the results, the speed of the compression-encryption process is acceptable, though the time taken by the reconstruction algorithm in this approach is slightly longer. The time-consuming operation in the proposed scheme, which is sparse signal recovery, is the outcome of arriving at the optimal solution during the reconstruction process. The compression-encryption speed of the proposed scheme is compared to that of the schemes presented in refer-

Table 9

Experimental outcomes of Time-Complexity

Test Images (512 × 512)	Compression-Encryption Time (sec)	Reconstruction Time (sec)
test image 1	3.5393	27.8928
test image 2	4.3531	29.1220
test image 3	3.9194	27.9049
test image 4	3.4765	28.0990
test image 5	3.6629	27.4607
test image 6	4.5502	28.9812
test image 7	3.4300	27.3059
test image 8	4.2483	28.4579
test image 9	3.7211	27.9951
test image 10	4.1470	29.2530
test image 11	4.1197	28.5117
test image 12	3.7389	27.6971
test image 13	4.3850	28.6045
test image 14	3.6276	27.4114
test image 15	3.5781	28.1245
test image 16	3.6034	27.4976
test image 17	4.1482	29.3319
test image 18	3.6020	28.3683
test image 19	3.9887	27.9506
test image 20	3.7114	27.6330

ences [16, 28, 32], with the results depicted in Table 6. The results show that the proposed technique is faster than the methods in references [16, 32], and a little slower than the method in reference [28].

NIST Statistical Test for Cipher Image

The NIST statistical package (SP 800-22) is a test suite comprising 15 tests that are used to evaluate the different types of randomness in the cipher images obtained using the proposed technique [36, 38]. To perform this test, we have used 20 encrypted images of dimensions 512×512. The decision rule level is set to 3% (i.e., a p-value ≥ 0.03 indicates success in the test). Every ciphered image produced by the proposed algorithm has successfully passed the entire suite of 15 tests. The NIST test outcomes of testimage-10 and testimage-20 are presented in Tables 7 and 8, it is hence concluded that the encrypted images show strong randomness.

5.1. Discussion

The proposed scheme is compared here with three peer encryption-compression methods. The image quality test results listed in Table 4 show that the average SSIM, NCC and ESSIM values of the reconstructed images in the proposed scheme are 0.6621, 0.6700 and 0.8459, respectively, proving that the quality of the reconstructed image is acceptable and outclasses that offered by the other two methods. The total key space of the proposed scheme approximately equals 2^{904} , which helps it resist key-based attacks strongly. The key sensitivity test result reveals that the proposed system has great key sensitivity. Figure 9 shows that since the histogram of the compressed-encrypted image is flat, no fruitful information can be obtained from such an image. As observed from Table 5 and Figure 10, the correlation between adjacent pixels of the compressed-encrypted image is almost 0 in all the three directions, demonstrating that the proposed scheme performs well against statistical attacks. Table 4 shows that the unified average changing intensity and number of pixel change rate values of the compressed-encrypted image in the proposed scheme, at 33.44% and 99.61%, respectively, are close to ideal values and thus help effectively defeat differential attacks. Additionally, in our approach, only the starting values of the Lü system need to be sent to the reconstruction side (rather than the whole sensing matrix) to efficiently reduce memory space and bandwidth. Moreover, the chaos-based random sensing matrices used in the proposed scheme

restrict the isometric property with an overwhelming probability that greatly enhances reconstruction quality, and the keys derived from the input image cause the scheme to be input image-sensitive. The analysis above makes it plain that the secured compression method presented has produced good results.

6. Conclusion

This paper has proposed secured color image compression based on compressive sampling and the Lü system.

Reference

- Alfalou, A., Brosseau, C., Abdallah, N. Simultaneous Compression and Encryption of Color Video Images. *Optics Communications*, 2015, 338, 371-379. <https://doi.org/10.1016/j.optcom.2014.10.020>
- Brindha, M., Ammasai, G. N. A Chaos Based Image Encryption and Lossless Compression Algorithm Using Hash Table and Chinese Remainder Theorem. *Applied Soft Computing*, 2016, 40, 379-390. <https://doi.org/10.1016/j.asoc.2015.09.055>
- Chai, X., Zheng, X., GanZhihua, Daojun, H., Chen, Y. An Image Encryption Algorithm Based On Chaotic System and Compressive Sensing. *Signal Processing*, 2018, 148, 124-144. <https://doi.org/10.1016/j.sigpro.2018.02.007>
- Chen, P., Guodong, Y., Xiaoling, H., Junwei, Z. Novel Meaningful Image Encryption Based on Block Compressive Sensing. *Security and Communication Networks*, 2019. <https://doi.org/10.1155/2019/6572105>
- Deng, J., Zhao, S., Wang, Y. Image Compression-Encryption Scheme Combining 2D Compressive Sensing with Discrete Fractional Random Transform. *Multimedia Tools Application*, 2017, 76, 10097-10117. <https://doi.org/10.1007/s11042-016-3600-2>
- Donoho, D. L. Compressed Sensing. *IEEE Transactions on Information Theory*, 2006, 52(4), 1289-1306. <https://doi.org/10.1109/TIT.2006.871582>
- Elaine, C. M., Nilson, M., Lirida, N., Hao, C., Jun, Y. A Review of Sparse Recovery Algorithms. *IEEE Access*, 2018, 7, 1300-1322. <https://doi.org/10.1109/ACCESS.2018.2886471>
- Emmanuel, J. C. The Restricted Isometry Property and Its Implications for Compressed Sensing. *Comptes Rendus Mathématique*, 2008, 346(9-10), 589-592. <https://doi.org/10.1016/j.crma.2008.03.014>
- Hongping, G., Song, X. A Novel Secure Data Transmission Scheme Using Chaotic Compressed Sensing. *IEEE Access*, 2017, 6, 4587-4598. <https://doi.org/10.1109/ACCESS.2017.2780323>
- Jinhu, L., Guanrong C., Suochun, Z. Dynamical Analysis of a New Chaotic Attractor. *International Journal of Bifurcation and Chaos*, 2002, 12(5), 1001-1015. <https://doi.org/10.1142/S0218127402004851>
- Jinhu, L., Guanrong, C., Suochun, Z. The Compound Structure of a New Chaotic Attractor. *Chaos, Solitons & Fractals*, 2002, 14(5), 669-672. [https://doi.org/10.1016/S0960-0779\(02\)00007-3](https://doi.org/10.1016/S0960-0779(02)00007-3)
- Junxin, C., Yu, Z., Lin, Q., Chong, F., Lisheng, X. Exploiting Chaos-Based Compressed Sensing and Cryptographic Algorithm for Image Encryption and Compression. *Optics & Laser Technology*, 2018, 99, 238-248. <https://doi.org/10.1016/j.optlastec.2017.09.008>
- Lihua, G., Chengzhi, D., Shumin, Pan., Nanrun, Z. Image Compression-Encryption Algorithms by Combining Hyper-Chaotic System With Discrete Fractional Random Transform. *Optics & Laser Technology*, 2018, 48-58. <https://doi.org/10.1016/j.optlastec.2018.01.007>
- Lihua, G., Kaide, Q., Chengzhi, D., Nanrun, Z. An Image Compression and Encryption Algorithm Based On Chaotic System And Compressive Sensing. *Optics & Laser Technology*, 2019, 115, 257-267. <https://doi.org/10.1016/j.optlastec.2019.01.039>
- Lihua, G., Kaide, Q., Chengzhi, D., Nanrun, Z. An Optical Image Compression and Encryption Scheme Based On Compressive Sensing and RSA Algorithm. *Optics and Lasers in Engineering*, 2019, 121, 169-180. <https://doi.org/10.1016/j.optlaseng.2019.03.006>
- Liya, Z., Huansheng, S., Xi, Z., Maode, Y., Liang, Z., Tao, Y. A Novel Image Encryption Scheme Based on Non-uniform Sampling in Block Compressive Sensing. *IEEE Access*, 2019, 22161-22174. <https://doi.org/10.1109/ACCESS.2019.2897721>
- Manoj, K., Ankita, V. An Efficient Encryption-Then-Compression Technique For Encrypted Images Using SVD. *Digital Signal Processing*, 2017, 60, 81-89. <https://doi.org/10.1016/j.dsp.2016.08.011>
- Miao, Z., Xiao-Jun, Tong., Jie, L., Zhu, W, Jinlong, L., Baolong, L., Jing, Ma. Image Compression and Encryp-

- tion Scheme Based on Compressive Sensing and Fourier Transform. *IEEE Access*, 2020, 8, 40838-40849. <https://doi.org/10.1109/ACCESS.2020.2976798>
19. Miao, Z., Xiaojun, T. Joint Image Encryption and Compression Scheme based on IWT And SPIHT. *Optics and Lasers in Engineering*, 2017, 90, 254-274. <https://doi.org/10.1016/j.optlaseng.2016.10.025>
 20. Nanrun, Z., Aidi, Z., Fen, Z., Lihua, G. Novel Image Compression-Encryption Hybrid Algorithm based on Key-Controlled Measurement Matrix In Compressive Sensing. *Optics & Laser Technology*, 2014, 62, 152-160. <https://doi.org/10.1016/j.optlastec.2014.02.015>
 21. Nanrun, Z., Haolin, L., Wang, D., Shumin, P., Zhihong, Z. Image Compression and Encryption Scheme based on 2D Compressive Sensing and Fractional Mellin Transform, *Optics Communications*, 2015 343, 15 10-21. <https://doi.org/10.1016/j.optcom.2014.12.084>
 22. Nanrun, Z., Shumin, P., Shan, C., Zhihong, Z. Image Compression-Encryption Scheme based on Hyper-Chaotic System and 2D Compressive Sensing. *Optics & Laser Technology*, 2016, 82, 121-133. <https://doi.org/10.1016/j.optlastec.2016.02.018>
 23. Ponuma, R., Amutha, R. Encryption of Image Data Using Compressive Sensing and Chaotic System. *Multimedia Tools and Applications*, 2019, 78, 11857-11881. <https://doi.org/10.1007/s11042-018-6745-3>
 24. Ponuma, R., Amutha, R. Compressive Sensing Based Image Compression-Encryption Using Novel 1D-Chaotic Map. *Multimedia Tools Applications*, 2018, 77, 19209-19234. <https://doi.org/10.1007/s11042-017-5378-2>
 25. Priya, R., Vidhyapriya, R., Seifedine, K., Robertas, D., Tomas, B. An Image Encryption Scheme Based on Block Scrambling, Modified Zigzag Transformation and Key Generation Using Enhanced Logistic-Tent Map. *Entropy*, 2019, 21(7). <https://doi.org/10.3390/e21070656>
 26. Qiaoyun, X., Kehui, S., Chun, C., Congxu, Z. A Fast Image Encryption Algorithm based on Compressive Sensing And Hyperchaotic Map, *Optics and Lasers in Engineering*, 2019 121, 203-214. <https://doi.org/10.1016/j.optlaseng.2019.04.011>
 27. Samar, M. I., Lobna, A.S., Ahmed, G.R., Ahmed, H.M., Mohamed, F. A., A Novel Image Encryption System Merging Fractional-Order Edge Detection and Generalized Chaotic Maps. *Signal Processing*, 2020, 167. <https://doi.org/10.1016/j.sigpro.2019.107280>
 28. Shuqin, Z., Congxu, Z., Wenhong, W. A Novel Image Compression-Encryption Scheme Based on Chaos and Compression Sensing. *IEEE Access*, 2018, 6. <https://doi.org/10.1109/ACCESS.2018.2874336>
 29. Sneha, P.S., Sankar, S., Kumar, A.S. A Chaotic Colour Image Encryption Scheme Combining Walsh-Hadamard Transform and Arnold-Tent Maps. *Journal of Ambient Intelligence and Human Computing*, 2020, 11, 1289-1308. <https://doi.org/10.1007/s12652-019-01385-0>
 30. Tongfeng, Z., Shouliang, L., Rongjun, G., Min, Y., Yide, M. A Novel 1D Hybrid Chaotic Map-Based Image Compression and Encryption using Compressed Sensing and Fibonacci-Lucas Transform. *Mathematical Problems in Engineering*, 2016. <https://doi.org/10.1155/2016/7683687>
 31. Tong, X., Zhang, M., Wang, Z. A Joint Color Image Encryption and Compression Scheme based on Hyper-Chaotic System. *Nonlinear Dyn*, 2016, 84, 2333-2356. <https://doi.org/10.1007/s11071-016-2648-x>
 32. Xinsheng, L., Taiyong, L., Jiang, W., Zhilong, X., Jiayi, S. Joint Image Compression and Encryption Based on Sparse Bayesian Learning and Bit-Level 3D Arnold Cat Maps. *PLOS ONE*, 2019. <https://doi.org/10.1371/journal.pone.0224382>
 33. Xiuli, C., Xianglong, F., Zhihua, G., Yushu, Z., Yang, L., Yiran, C. An Efficient Chaos-Based Image Compression and Encryption Scheme Using Block Compressive Sensing and Elementary Cellular Automata. *Neural Computing and Applications*, 2018. <https://doi.org/10.1007/s00521-018-3913-3>
 34. Xiuli, C., Haiyang, W., Zhihua, G., Yushu, Z., Yiran, C., Nixon, K. W. An Efficient Visually Meaningful Image Compression and Encryption Scheme based on Compressive Sensing and Dynamic LSB Embedding. *Optics and Lasers in Engineering*, 2020, 124. <https://doi.org/10.1016/j.optlaseng.2019.105837>
 35. Yao, S., Wang, T., Shen, W. Research of Incoherence Rotated Chaotic Measurement Matrix in Compressed Sensing. *Multimedia Tools Applications*, 2017, 76, 17699-17717. <https://doi.org/10.1007/s11042-015-2953-2>
 36. Yaqin, X., Jiayin, Yu., Shiyu, G., Qun, D., Erfu, W. Image Encryption Scheme with Compressed Sensing Based on New Three-Dimensional Chaotic System, *Entropy* 2019, 21(9). <https://doi.org/10.3390/e21090819>
 37. Zhang, Y., Biao, X., Nanrun, Z. A Novel Image Compression-Encryption Hybrid Algorithm based on the Analysis Sparse Representation. *Optics Communications*, 2017, 392, Pages 223-233. <https://doi.org/10.1016/j.optcom.2017.01.061>
 38. Zhang, H., Xiao-qing, W., Yu-jie, S., Xing-yuan W. A Novel Method For Lossless Image Compression and Encryption based on LWT, SPIHT and Cellular Automata. *Signal Processing: Image Communication*, 2020, 84. <https://doi.org/10.1016/j.image.2020.115829>

