



# Image encryption using sparse coding and compressive sensing

R. Ponuma<sup>1</sup> · R. Amutha<sup>1</sup>

Received: 14 July 2018 / Revised: 19 November 2018 / Accepted: 18 February 2019 /  
Published online: 25 February 2019  
© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

An encryption algorithm based on sparse coding and compressive sensing is proposed. Sparse coding is used to find the sparse representation of images as a linear combination of atoms from an overcomplete learned dictionary. The overcomplete dictionary is learned using K-SVD, utilizing non-overlapping patches obtained from a set of images. Compressed sensing is used to sample data at a rate below the Nyquist rate. A Gaussian measurement matrix compressively samples the plain image. As these measurements are linear, chaos based permutation and substitution operations are performed to obtain the cipher image. Bit-level scrambling and block substitution is done to confuse and diffuse the measurements. Simulation results verify the performance of the proposed technique against various statistical attacks.

**Keywords** Chaotic map · Compressive sensing · Dictionary learning · Encryption · Measurement matrix · Sparse coding

## 1 Introduction

Multimedia data communication plays a vital role in ecommerce, navigation and information systems, entertainment, education, military communication and industries. All these applications require fast and secure transmission of the data, which in turn necessitates the use of encryption as well as compression. With the incessant use of multimedia in mobility communication and the ease of accessibility, the security of data is of paramount importance. Security can be achieved by the encryption of data. Chaos has been widely used in the encryption of data because of its inherent properties. Chaotic systems are periodic and highly sensitive.

Several chaos based image encryption schemes have been proposed. Zahmoul et al. (2017) proposed a new chaotic map based on Beta function. The new map was used to generate pseudo random sequences utilized to perform permutation and substitution operations. The experimental results showed that the proposed method is highly secure

---

✉ R. Ponuma  
ponumar@ssn.edu.in

<sup>1</sup> Department of ECE, Sri Sivasubramaniya Nadar College of Engineering, Chennai, India

compared to other chaos based encryption schemes. Xu et al. (2016) proposed a bit-level image encryption algorithm based on piecewise linear chaotic maps. Plain image is converted into two equal sized binary sequences. Then permutation is done at bit-level by interchanging the binary elements of the sequences. In Hanis and Amutha (2019) a modified logistic map based authenticated image encryption scheme is proposed. The modified logistic map is used to construct key matrices, which are then used to encrypt the images.

Encryption provides data security whereas compression of data assist in easy storage and transmission. Compression and encryption can be done separately, or they can be performed simultaneously reducing the computational overhead. Compressive Sensing (CS) (Donoho 2006) is a technique for compressively sampling sparse data. In CS, a sparse signal can be recovered from few linear measurements (Candes and Wakin 2008). Several CS based image compression-encryption schemes are available in literature.

Zhang et al. (2016) showed the relation between CS and symmetric key encryption. The random projection using sensing matrix is associated with encryption and the signal recovery to decryption. The parameters used to generate the measurement matrix acts as the shared secret key between the communicating entities. The robustness of the CS based encryption schemes against brute-force attack, noise attacks, packet loss and shear attacks are demonstrated in Cambareri et al. (2015), Huang and Sakurai (2011) and Orsdemir et al. (2008). Huang et al. (2014) employed a block cipher structure to break the linearity in the CS measurements. They also used a parallel computing environment, which greatly increased the encryption speed. In papers (Chen et al. 2016; Ponuma and Amutha 2018a, b, c; Zhou et al. 2016, 2014a) compressive sensing is employed to obtain joint image compression-encryption. Zhou et al. (2014b) proposed a novel hybrid image compression-encryption scheme using circulant measurement matrix. The seed for the circulant matrix is generated using a logistic map and the chaotic map parameters acts as the shared secret key.

In this paper, a sparse coding and compressive sensing based image encryption technique is proposed. Digital image possess some intrinsic features like bulk data capacity, redundancy of data and strong correlation among pixels, which can be exploited to achieve superior compression. Natural images can be sparsely represented using over complete dictionary (Olshausen and Field 1997). The dictionary can be a fixed dictionary or a learned dictionary. The fixed basis like DCT, wavelets, curvelets are not adaptive. A learned dictionary trained from a set of input images, provides better sparse representation of the image. The dictionary learning can be performed using algorithms like K-SVD (Aharon et al. 2006), MOD (Engan et al. 2007), RLS-DLA (Skretting and Engan 2010) etc. K-SVD is widely used for dictionary learning. In the proposed scheme, the inherent sparsity of the image is represented using a learned basis obtained using non-overlapping patches from a set of images. The sparse coefficients are then subjected to the compressive sensing process. The image recovery is done using Orthogonal Matching Pursuit (OMP) algorithm (Tropp and Gilbert 2007). The remainder of the paper is organized as follows. Section 2 reviews the concepts of CS theory and dictionary learning. The proposed framework for image encryption scheme is provided in Sect. 3. The experimental results and analysis are given in Sect. 4. Section 5 concludes the paper.

## 2 Methodology

### 2.1 Compressive sensing

Compressive sensing (Baraniuk 2007) states that a sparse signal can be reconstructed from a set of undersampled data with high probability. For a signal  $x \in R^N$ , the sparse representation  $\alpha$  is given by

$$\alpha = \Psi^T x. \quad (1)$$

where  $\Psi$  is the basis. The basis can be an orthogonal matrix (discrete cosine transform, discrete wavelet transform etc.) or a learned dictionary. If the signal can be represented as a linear combination of few vectors from the sparsifying basis, then the signal can be recovered successfully. The signal is recovered from linear measurements obtained by using the measurement matrix  $\Phi \in R^{M \times N}$ . The linear measurement  $y$  is represented as

$$y = \theta \alpha \quad (2)$$

where  $\theta = \Phi \Psi$  is the sensing matrix. A  $s$ -sparse signal can be reconstructed from  $M$  measurements ( $M \ll N$ ), if the sensing matrix  $\theta$  satisfies the Restricted Isometry Property (RIP) (Candes 2008). By assuming that  $x$  can be expressed in the basis  $\Psi$ ,  $\alpha$  can be estimated by solving the following  $L_1$  norm minimization problem.

$$\min \|\alpha\|_1 \text{ subject to } y = \theta \alpha \quad (3)$$

Iterative greedy algorithms can be used to solve the above convex optimization problem.

### 2.2 Dictionary learning

For a compressively sensed data to be recovered successfully the signal should be sparse in a domain, i.e. in a dictionary. The dictionary can be a fixed dictionary or a learned dictionary. The dictionary learning algorithm aims to find a dictionary, that sparsely represents the signal. In Olshausen and Field (1997), it is shown that an overcomplete dictionary  $D$  containing  $K$  prototype signal atoms for columns  $\{d_j\}$ ,  $j = 1, 2, \dots, K$  can be used to represent a signal  $x$  as a sparse linear combination of these atoms. The representation of  $x$  may be either exact  $x = D\alpha$  or approximate  $x \approx D\alpha$ , satisfying  $\|x - D\alpha\|_2 \leq \epsilon$ . The vector  $\alpha \in R^K$  contains the representation coefficients of the signal  $x$ .

In dictionary learning the goal is to find a dictionary  $D$  that yields a better sparse representation for a set of images. The K-SVD technique of dictionary learning is used in this paper. This iterative method alternates between a sparse coding stage and a dictionary update stage. The dictionary is learned using the patches  $x_i$  extracted from the grayscale image set.<sup>1</sup> Each image of size  $H \times W$  is first divided into non-overlapping blocks ( $B_i$ ) of size  $\sqrt{N} \times \sqrt{N}$ . The blocks are then vectorized to form the training patches of size  $N \times 1$ . The Dictionary  $D$  is first initialized with  $K$  randomly extracted patches from the training set based on a threshold ( $thr$ ) obtained using spatial frequency ( $SF$ ) of the image blocks. The spatial frequency of the blocks ( $B_i$ ) is taken as the threshold. In Huang and Jing (2007) the Spatial Frequency is defined as

$$SF = \sqrt{(RF)^2 + (CF)^2}$$

<sup>1</sup> <http://decsai.ugr.es/cvg/CG/base.htm>.

$$\begin{aligned}
 RF &= \sqrt{\frac{1}{\sqrt{N} \times \sqrt{N}} \sum_{j=1}^{\sqrt{N}} \sum_{k=2}^{\sqrt{N}} [B_i(j, k) - B_i(j, k+1)]^2} \\
 CF &= \sqrt{\frac{1}{\sqrt{N} \times \sqrt{N}} \sum_{j=2}^{\sqrt{N}} \sum_{k=1}^{\sqrt{N}} [B_i(j, k) - B_i(j-1, k)]^2} \quad (4)
 \end{aligned}$$

where  $RF$  and  $CF$  are the row and column frequency respectively. The patches for the initial dictionary are selected as given in Ashwini and Amutha (2018). The patches ( $p_i$ ) are sorted in ascending order based on spatial frequency and divided into five sets as follows,

$$\begin{cases}
 SF(p_i) < \frac{thr}{4}, & p_i \in \text{Set 1} \\
 \frac{thr}{4} \leq SF(p_i) < \frac{thr}{2}, & p_i \in \text{Set 2} \\
 \frac{thr}{2} \leq SF(p_i) < thr, & p_i \in \text{Set 3} \\
 thr < SF(p_i) < 2thr, & p_i \in \text{Set 4} \\
 SF(p_i) \geq 2thr, & p_i \in \text{Set 5}
 \end{cases} \quad (5)$$

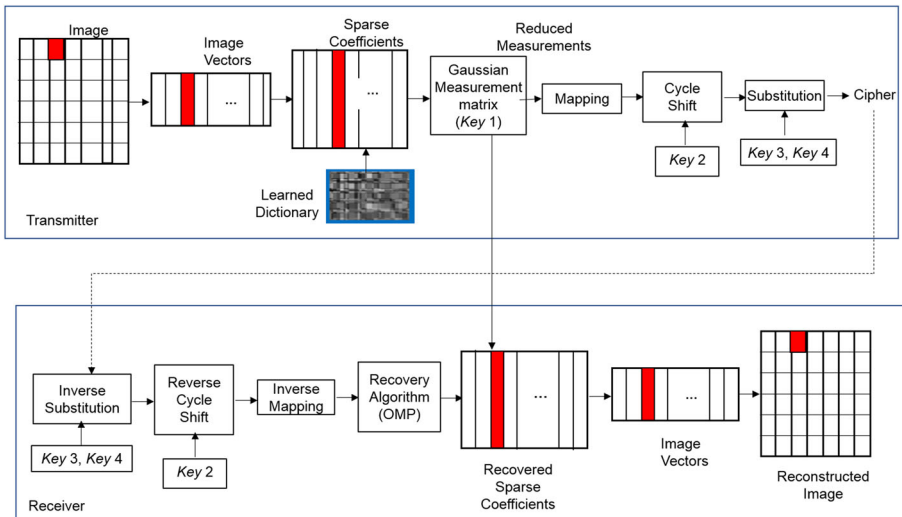
From the five sets,  $K$  patches are selected to initialize the dictionary. The first  $K/5$  and last  $K/5$  patches are selected from the first and second set respectively. From the sets 3, 4 and 5 the first  $K/10$  and last  $K/10$  patches are selected. Using these  $K$  selected patches as the atoms of the initial dictionary and K-SVD algorithm, the learned dictionary  $D$  is trained. The dictionary is learned using a sparsity parameter  $T_1 = 10$  i.e. the sparsity of each atom in the dictionary is 10. The size of the resulting dictionary is  $N \times K$ . As both smooth and textured patches are selected uniformly the learned dictionary provides better approximation of the image.

### 2.3 Measurement matrix

In CS, generally a random measurement matrix provides dimensionality reduction. The measurement matrix construction must ensure the reconstruction of the sparse signal from the reduced measurements. The commonly used measurement matrices are constructed from random variables that follow Gaussian, Bernoulli distribution etc. The measurement matrix can also be structurally random like Hadamard, Circulant etc. In this paper, a measurement matrix constructed using Gaussian random variables is used to obtain compressed measurements. The Gaussian matrix is of size  $M \times N$  and matrix elements are zero mean and variance  $1/M$  Gaussian random variables. The whole matrix is used as the key and it is shared between the sender and the receiver using a secured channel. By employing the measurement matrix as the secret key, security is embedded in compressive sensing technique.

## 3 The proposed encryption algorithm

The block diagram of the sparse coding and compressive sensing based image encryption scheme is shown in Fig. 1. The encryption is implemented using four keys. The measurement matrix acts as the first key. The other three keys are the parameters of the three logistic maps used for scrambling and substitution operations. The encryption algorithm is as follows:



**Fig. 1** Block diagram of the sparse coding and compressive sensing based encryption

1. The input image of size  $H \times W$  is divided into  $b$  blocks of size  $\sqrt{N} \times \sqrt{N}$ . The blocks are non-overlapping. Each block is then vectorized to form a sequence  $x$  of size  $N \times 1$ .
2. The sparse code ( $\alpha$ ) of  $x$  is computed using the learned dictionary  $D$  and OMP algorithm. The sparsity is set at  $T = 12$ . The sparse code ( $\alpha$ ) of dimension  $K \times 1$ , sparsely represents  $x$  such that

$$x = D\alpha \tag{6}$$

3. CS is used to compressively sample  $\alpha$  using the Gaussian measurement matrix  $\Phi$ . The measurements are of dimension  $M \times 1$  and the number of measurements  $M = \lfloor \text{SamplingRatio} \times N \rfloor$ .

$$y_m = \Phi\alpha \tag{7}$$

4. The measurements for all the  $b$  blocks of the image is computed and the measurement vectors are concatenated to form an encrypted image ( $C_{img}$ ) of size  $M \times b$ .
5. The encrypted image ( $C_{img}$ ) is mapped to the interval  $[0, 255]$  using the following mapping

$$C_{map}(i, j) = \left\lfloor \frac{C_{img}(i, j) - C_{imgmin}}{C_{imgmax} - C_{imgmin}} \right\rfloor \times 255 \tag{8}$$

6. The mapped image is then encrypted using a logistic map by scrambling and substitution operations. The logistic map is generated based on the following equation,

$$t_{l+1} = \mu t_l(1 - t_l), t_l \in [0, 1] \tag{9}$$

The logistic map is used to generate a sequence ( $t_l$ ) of length  $M \times b$  which is then converted into integer sequence in the interval  $[0, 7]$ .

$$t_l = \lfloor \text{mod}(t_l \times 10^3, 8) \rfloor \tag{10}$$

7. Each pixel of  $C_{map}$  is decomposed into an 8-bits binary number. The binary digits of each pixel are subjected to cyclic right shift operation by using  $t_l$  and then converted to

integer value. The encrypted image  $C_{scr}$  is of size  $M \times b$ .

$$C_{scr}(i, j) = \text{bitshift}(C_{img}(i, j), t_1(i, j)) \quad (11)$$

8. The bit scrambled encrypted image is then subjected to a substitution operation. A pseudorandom sequence ( $PR_1$ ) that is in the interval  $[0, 255]$  is generated using the logistic map. The chaotic sequence ( $t_2$ ) of size  $M \times 1$  is converted into an integer sequence as follows:

$$PR_1 = \text{mod}(t_2 \times 10^7, 256) \quad (12)$$

The substitution operation for each column of  $C_{scr}$  results in an encrypted image  $\hat{C}$ . The substitution operation is as follows,

$$\hat{C}(n) = \begin{cases} C_{scr}(n) \oplus PR_1, n = 1 \\ C_{scr}(n) \oplus PR_1 \oplus \hat{C}(n-1), \text{ otherwise} \end{cases} \quad (13)$$

where  $n = 1, 2, \dots, b$ .

9. To further increase the security the encrypted image  $\hat{C}$ , it is first reshaped to form a cipher image of size  $H_1 \times W$ . Then it is divided into four equal sized blocks, followed by another round of substitution operation as shown in (13) using another random sequence ( $PR_2$ ) generated using logistic map.

The decryption procedure involves the inverse of the techniques performed in the encryption stage. The ciphered image is decrypted using the parameters of the three logistic maps,  $C_{imgmax}$  and  $C_{imgmin}$ . The original image is recovered using the OMP algorithm using the measurement matrix  $\Phi$ .

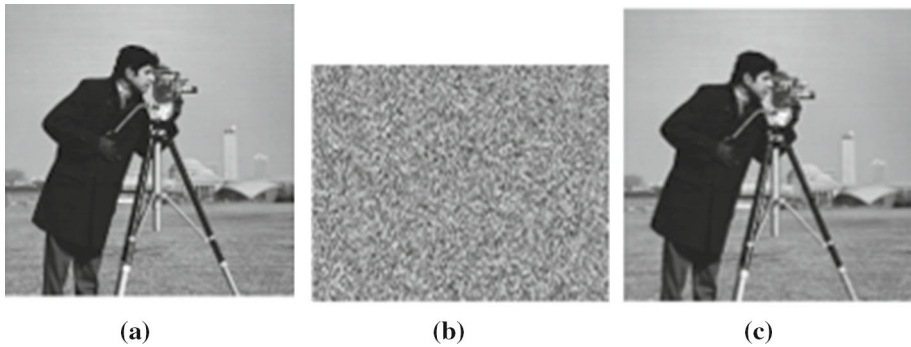
## 4 Experimental results and analysis

The proposed scheme is verified using the test images from the USC-SIPI database. The test image cameraman of size  $256 \times 256$  is shown in Fig. 2a. The learned dictionary is trained as discussed in Sect. 2. The K-SVD algorithm<sup>2</sup> is used for training. The size of the learned dictionary is  $64 \times 100$ . The measurement matrix is created as given in Sect. 2. The learned dictionary  $D$  is used to sparsely represent the test images. The test image is divided into non-overlapping blocks of size  $8 \times 8$ . Each block is rasterized to form a sequence of length  $64 \times 1$ . The block sparsity is set at  $T = 12$  and the OMP algorithm is used to sparsely represent the image. The sparse coefficients are of size  $100 \times 1$ . The matrix  $\Phi$  then measures the sparse coefficients. The measurements are then scrambled and subjected to a pseudorandom encryption to obtain the cipher image shown in Fig. 2b. The two chaotic sequences are generated with initial conditions  $\mu_{10} = 3.99$ ,  $\mu_{20} = 3.96$ ,  $\mu_{30} = 3.65$ ,  $t_{10} = 0.23$  and  $t_{20} = 0.18$ ,  $t_{30} = 0.5301$ . The encryption scheme was tested with several test images. The sparse coefficients were recovered using the OMP algorithm at the receiver.

### 4.1 PSNR and SSIM analysis

The Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM) is used to evaluate the reconstruction performance of the proposed scheme. For an image of size  $N \times N$  the PSNR is computed using the formula,

<sup>2</sup> <http://www.cs.technion.ac.il/~ronrubin/software.html>.



**Fig. 2** **a** Original image, **b** ciphered image, **c** recovered image

**Table 1** PSNR and SSIM analysis

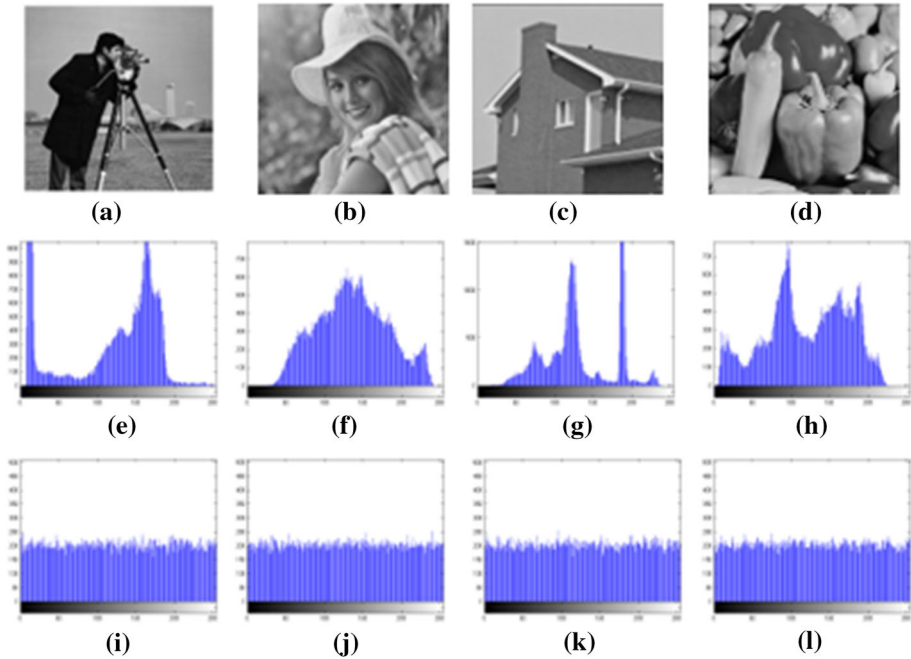
Test image	PSNR (dB)			SSIM		
	Proposed	Hu et al. (2017)	Zhou et al. (2014a)	Proposed	Hu et al. (2017)	Zhou et al. (2014a)
Baboon	20.1637	20.5268	22.1599	0.6657	0.5420	0.4616
Boat	22.3672	22.2174	22.6369	0.7570	0.5753	0.4767
Lake	21.7285	19.9962	21.5376	0.7671	0.5783	0.4496
Man	22.3528	20.8783	23.0375	0.7247	0.6049	0.4654
Peppers	25.6583	23.0676	24.6136	0.8179	0.6427	0.5242
Average	22.4541	21.3373	22.7971	0.7465	0.5886	0.4755

$$PSNR = 10 \log \frac{255^2}{(1/N^2) \sum_{i=1}^N \sum_{j=1}^N [PI(i, j) - PI'(i, j)]^2} \tag{14}$$

where  $PI(i, j)$  is the plain image and  $PI'(i, j)$  is the reconstructed image. The SSIM between the original and the recovered image is computed by using the mean ( $\mu$ ), standard deviation ( $\sigma$ ) and cross-covariance ( $\sigma_{pp'}$ ) of both images.

$$SSIM = \frac{(2\mu_p\mu_{p'} + C_1)(2\sigma_{pp'} + C_2)}{(\mu_p^2 + \mu_{p'}^2 + C_1)(\sigma_p^2 + \sigma_{p'}^2 + C_2)} \tag{15}$$

where  $C_1 = 0.01 \times (2^8 - 1)$ ,  $C_2 = 0.01 \times (2^8 - 1)$ . For a sampling ratio of 0.5 the PSNR and SSIM of the test images are computed. The test images are of size  $256 \times 256$ . The metrics are compared with the schemes proposed in Hu et al. (2017) and Zhou et al. (2014a). Table 1 indicates that the proposed scheme achieves better reconstruction quality for the test images Lakes and Peppers in comparison with other schemes. A 1 dB increase in PSNR is obtained than that obtained in Hu et al. (2017). The PSNR is approximately same as that of the scheme in Zhou et al. (2014a). The average SSIM is higher than (Hu et al. 2017; Zhou et al. 2014a) by 26.83%, 56.99% respectively.



**Fig. 3** **a** Cameraman, **b** Elaine, **c** house, **d** peppers, Histogram of **e** cameraman, **f** Elaine, **g** house, **h** peppers, **i** encrypted cameraman, **j** encrypted elaine, **k** encrypted house, **l** encrypted peppers

## 4.2 Histogram analysis

Image statistics can be used by an intruder to determine useful information from a transmitted image. Histogram represents the distribution of the gray levels in an image. Hence the cipher image histogram should be dissimilar from the histogram of the plain image. Also, the cipher image histogram must be uniform. Figure 3, shows the histogram of the test images and their corresponding cipher. The histogram of the ciphers is relatively uniform, which indicates that the proposed encryption algorithm has the ability to resist statistical attacks. Also, the histogram of the cipher image does not resemble the histogram of the plain image.

## 4.3 Correlation coefficient analysis

The adjacent pixels in digital images are redundant and have high degree of correlation. In the proposed scheme the plain image is encrypted as a noise like random image. Therefore, the adjacent pixel correlation in the cipher image must be very less. The correlation coefficient is measured by randomly selecting 2000 adjacent pixels in the horizontal, vertical and diagonal directions. The correlation coefficient is computed as follows,

$$r_{xy} = \frac{\text{cov}(p, q)}{\sqrt{S(p)S(q)}} \quad (16)$$



**Table 2** Correlation analysis

Test image	Direction	Correlation coefficient of plain image	Correlation coefficient of cipher image		
			Hu et al. (2017)	Zhou et al. (2014a)	Proposed
Baboon	Horizontal	0.8857	0.0074	− 0.0235	0.0108
	Vertical	0.8309	0.0164	0.0234	0.0211
	Diagonal	0.7956	0.01080	− 0.0193	− 0.0143
Boat	Horizontal	0.9223	0.02003	0.0264	− 0.0100
	Vertical	0.9440	0.0030	0.0154	− 0.0144
	Diagonal	0.8742	− 0.0443	0.0308	0.0187
Lake	Horizontal	0.9587	0.0146	− 0.0118	− 0.0100
	Vertical	0.9570	0.0086	− 0.0199	− 0.0144
	Diagonal	0.9321	− 0.0101	0.0005	0.0187
Man	Horizontal	0.9455	− 0.0038	0.0324	0.0146
	Vertical	0.9574	− 0.0006	0.0137	− 0.0276
	Diagonal	0.9071	− 0.0045	− 0.0152	− 0.0011
Peppers	Horizontal	0.9636	0.0233	0.0325	0.0023
	Vertical	0.9757	− 0.0077	− 0.0192	0.0014
	Diagonal	0.9414	0.0078	0.0133	− 0.0212
Average	Horizontal	0.9352	0.0124	0.0112	0.0062
	Vertical	0.9330	0.0039	0.0027	− 0.0347
	Diagonal	0.8901	− 0.0081	0.0020	0.0107

where

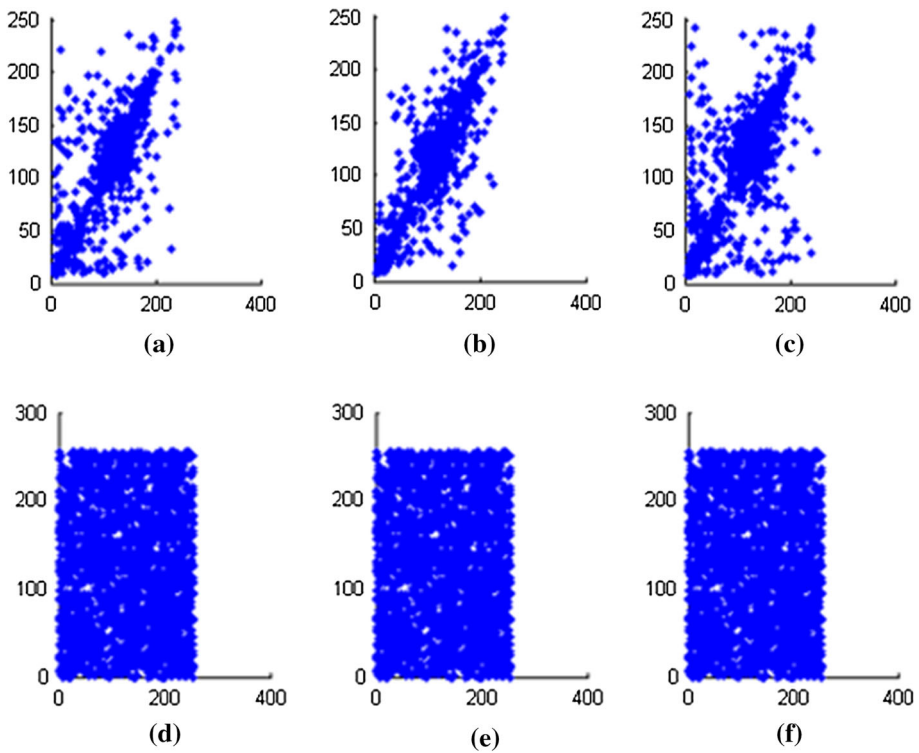
$$cov(p, q) = \frac{1}{2000} \sum_{i=1}^{2000} [p_i - E(p)][q_i - E(q)]$$

$$S(p) = \frac{1}{2000} \sum_{i=1}^{2000} [p_i - E(p)]^2, E(p) = \frac{1}{2000} \sum_{i=1}^{2000} p_i$$

The correlation coefficients of the ciphered image are tabulated in Table 2, from which we know that the correlation between the adjacent pixels is irrelevant. In comparison to the correlation coefficient achieved by the algorithms (Hu et al. 2017; Zhou et al. 2014a) the proposed method achieves minimum correlation for the test image Boat in the horizontal and diagonal directions. The correlation coefficients of other test images are comparable to other schemes. The Fig. 4 shows the comparison of adjacent pixel correlation for the plain image cameraman and its corresponding cipher image.

### 4.4 Entropy analysis

Information entropy can be used to determine the randomness of the encrypted image. For a grayscale image to be random the gray levels have to be uniformly distributed. For a noise-like ciphered image the information entropy must be nearer to 8. The information entropy is computed using



**Fig. 4** Scatter plot for cameraman image. Adjacent pixels in Plain Image **a** Horizontal, **b** vertical, **c** diagonal direction; Adjacent pixels in Cipher Image **d** horizontal, **e** vertical, **f** diagonal direction

**Table 3** Entropy of plain image and cipher image

Test image	Entropy of plain image	Entropy of cipher image		
		Proposed	Hu et al. (2017)	Zhou et al. (2014a)
Baboon	7.5770	7.9972	7.9938	7.9890
Boat	7.5360	7.9967	7.9939	7.9885
Lake	7.2283	7.9969	7.9949	7.9890
Man	7.1583	7.9967	7.9942	7.9883
Peppers	7.4582	7.9971	7.9938	7.9889
Average	7.3916	7.9969	7.9941	7.9887

$$H(m) = - \sum_{i=0}^{255} Pr(m_i) \log_2 Pr(m_i) \tag{17}$$

where  $Pr(m_i)$  is the probability of occurrence of gray level  $m$ . The information entropy is presented in Table 3. The average entropy of the proposed scheme is 7.9969 which is better than (Hu et al. 2017; Zhou et al. 2014a) and also closer to the ideal value.

### 4.5 UACI and NPCR analysis

An image encryption algorithm must be highly sensitive. The algorithm should generate a completely different cipher image, even for a one-bit pixel change. The Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are the measures that verify the sensitivity of the encryption algorithm.

$$NPCR = \sum_{i,j} \frac{D(i, j)}{M \times N} \times 100\% \tag{18}$$

$$UACI = \frac{1}{M \times N} \frac{\sum_{i,j} |CI_1(i, j) - CI_2(i, j)|}{L} \times 100\% \tag{19}$$

$CI_1(i, j)$  and  $CI_2(i, j)$  are the two ciphered images and  $L$  is the number of gray levels.  $D(i, j)$  is determined based on the rule

$$D(i, j) = \begin{cases} 0, & CI_1(i, j) = CI_2(i, j) \\ 1, & \text{otherwise} \end{cases} \tag{20}$$

The NPCR and UACI values tabulated in Table 4 indicates that the algorithm can resist differential attacks as they are close to the theoretical values 99.61 and 33.46 respectively. As the scheme in Zhou et al. (2014a) does not employ permutation and substitution operations its NPCR and UACI values are less. The average NPCR and UACI of the proposed scheme and scheme in Zhou et al. (2014a) are closer to the theoretical value.

### 4.6 Key space analysis

The Gaussian measurement matrix is used as the  $Key_1$  in the encryption process. If the data precision is  $10^{15}$  and the matrix size is  $M \times N$ , the keyspace for  $Key_1$  is  $(10^{15})^{M \times N}$ . The  $Key_2, Key_3$  and  $Key_4$  are the parameters of the three logistic maps i.e.  $\mu_1, \mu_2, \mu_3, t_{10}, t_{20}$  and  $t_{30}$ . The key space for  $Key_2, Key_3$  and  $Key_4$  is  $(10^{15})^6$ . A large keyspace ensures that the proposed scheme resists brute-force attack. The brute-force attack becomes infeasible as the effort required to decipher the key, grows exponentially with increasing key space.

$$KeySpace = (10^{15})^{M \times N} \times 10^{90} \tag{21}$$

The key space of the proposed scheme is compared with the schemes in Hu et al. (2017) and Zhou et al. (2014a) and it is shown in Table 5. The key space achieved using the proposed scheme is larger and hence greater the security.

### 4.7 Deviation from uniform histogram

The histogram of the cipher must be uniform i.e. the probability of occurrence of each pixel is uniform. For a cipher image ( $C$ ) of size  $M \times N$  the ideal histogram is mathematically represented as

$$Ideal\_Hist(C(i, j)) = \begin{cases} \frac{M \times N}{256}, & 0 \leq C(i, j) \leq 255 \\ 0, & \text{otherwise} \end{cases} \tag{22}$$

The strength of the algorithm can be verified using the deviation of the cipher image histogram from the ideal histogram. A lower value of  $D_H$  represents a better encryption quality. The

**Table 4** NPCR and UACI analysis

Test image	NPCR%		UACI%	
	Proposed	Hu et al. (2017)	Zhou et al. (2014a)	Proposed
Baboon	99.6321	99.6735	0.3876	33.4955
Boat	99.6128	99.6674	0.3906	33.3523
Lake	99.6128	99.6735	0.3906	33.4288
Man	99.6042	99.5972	0.3876	33.4733
Peppers	99.6008	99.5911	0.3906	33.3379
Average	99.6125	99.6405	0.3894	33.3981
				Hu et al. (2017)
				33.6889
				33.4745
				33.8226
				33.3759
				33.3975
				33.5519
				Zhou et al. (2014a)
				0.1201
				0.1180
				0.1205
				0.1307
				0.1469
				0.1272

**Table 5** Key space analysis

Proposed	Hu et al. (2017)	Zhou et al. (2014a)
$> 2^{299}$	$\approx 2^{200}$	$\approx 2^{113}$

**Table 6** Deviation from uniform histogram

Test image	Deviation from uniform histogram		
	Proposed	Hu et al. (2017)	Zhou et al. (2014a)
Baboon	0.0498	0.0718	0.0739
Boat	0.0535	0.0724	0.0722
Lake	0.0509	0.0673	0.0664
Man	0.0546	0.0705	0.0713
Peppers	0.0511	0.0698	0.0760
Average	0.0520	0.0704	0.0720

deviation from ideal histogram is given as:

$$D_H = \frac{\sum |Ideal\_Hist(C(i, j)) - Hist(C(i, j))|}{M \times N} \tag{23}$$

Table 6 compares the deviation of the histogram of the cipher from the ideal uniform histogram. We infer that the deviation is minimal in the proposed scheme, which indicates that the cryptanalysis of the histogram for statistical information by the intruder is difficult. The average DH of the proposed scheme is better than the compared algorithms (Hu et al. 2017; Zhou et al. 2014a) by 26.14% and 27.78% respectively.

## 5 Conclusion

In this paper, an encryption technique based on sparse coding and compressive sensing is proposed. The sparse coding using a learned dictionary provides a better representation of the image. Compressive sensing is used concurrently encrypt as well as compress data. The data is compressive sampled and randomly projected using the Gaussian measurement matrix. The security of the measurements is further enhanced by employing a chaos based bit scrambling and pseudorandom encryption. The experimental analysis performed shows that the proposed algorithm provides cryptographically strong cipher image.

## Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

Aharon, M., Elad, M., & Bruckstein, A. (2006). K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11), 4311–4322.

Ashwini, K., & Amutha, R. (2018). Compressive sensing based simultaneous fusion and compression of multi-focus images using learned dictionary. *Multimedia Tools and Applications*, 77(19), 25889–25904.

Baraniuk, R. (2007). Compressive sensing [Lecture Notes]. *IEEE Signal Processing Magazine*, 24, 118–121.

- Cambareri, V., Mangia, M., Pareschi, F., Rovatti, R., & Setti, G. (2015). On known-plaintext attacks to a compressed sensing-based encryption: A quantitative analysis. *IEEE Transactions on Information Forensics and Security*, *10*(10), 2182–2195.
- Candes, E., & Wakin, M. (2008). An introduction to compressive sampling. *IEEE Signal Processing Magazine*, *25*(2), 21–30.
- Candes, E. J. (2008). The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathématique*, *346*(9–10), 589–592.
- Chen, T., Zhang, M., Wu, J., Yuen, C., & Tong, Y. (2016). Image encryption and compression based on kronecker compressed sensing and elementary cellular automata scrambling. *Optics & Laser Technology*, *84*, 118–133.
- Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, *52*(4), 1289–1306.
- Engan, K., Skretting, K., & Husøy, J. H. (2007). Family of iterative LS-based dictionary learning algorithms, ILS-DLA, for sparse signal representation. *Digital Signal Processing*, *17*(1), 32–49.
- Hanis, S., & Amutha, R. (2019). A fast double-keyed authenticated image encryption scheme using an improved chaotic map and a butterfly-like structure. *Nonlinear Dynamics*, *95*(1), 421–432.
- Hu, G., Xiao, D., Wang, Y., & Xiang, T. (2017). An image coding scheme using parallel compressive sensing for simultaneous compression-encryption applications. *Journal of Visual Communication and Image Representation*, *44*, 116–127.
- Huang, R., Rhee, K. H., & Uchida, S. (2014). A parallel image encryption method based on compressive sensing. *Multimedia Tools and Applications*, *72*(1), 71–93.
- Huang, R., & Sakurai, K. (2011). A robust and compression-combined digital image encryption method based on compressive sensing. In *Proceedings: 7th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IHHMSP 2011*, pp. 105–108.
- Huang, W., & Jing, Z. (2007). Evaluation of focus measures in multi-focus image fusion. *Pattern Recognition Letters*, *28*(4), 493–500.
- Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, *37*(23), 3311–3325.
- Orsdemir, A., Altun, H. O., Sharma, G., & Bocko, M. F. (2008). On the security and robustness of encryption via compressed sensing. In *Proceedings: IEEE Military Communications Conference MILCOM*.
- Ponuma, R., & Amutha, R. (2018). Compressive sensing and chaos-based image compression encryption. In *Advances in Soft Computing and Machine Learning in Image Processing*, pp. 373–392. Springer.
- Ponuma, R., & Amutha, R. (2018b). Compressive sensing based image compression-encryption using novel 1d-chaotic map. *Multimedia Tools and Applications*, *77*(15), 19209–19234.
- Ponuma, R., & Amutha, R. (2018). Encryption of image data using compressive sensing and chaotic system. *Multimedia Tools and Applications*. <https://doi.org/10.1007/s11042-018-6745-3>.
- Skretting, K., & Engan, K. (2010). Recursive least squares dictionary learning algorithm. *IEEE Transactions on Signal Processing*, *58*(4), 2121–2130.
- Tropp, J. A., & Gilbert, A. C. (2007). Signal recovery from random measurements via orthogonal Matching Pursuit - Semantic Scholar. *IEEE Transactions on Information Theory*, *53*(12), 4655–4666.
- Xu, L., Li, Z., Li, J., & Hua, W. (2016). A novel bit-level image encryption algorithm based on chaotic maps. *Optics and Lasers in Engineering*, *78*, 17–25.
- Zahmoul, R., Ejbali, R., & Zaied, M. (2017). Image encryption based on new Beta chaotic maps. *Optics and Lasers in Engineering*, *96*, 39–49.
- Zhang, Y., Zhou, J., Chen, F., Zhang, L. Y., Wong, K.-W., He, X., et al. (2016). Embedding cryptographic features in compressive sensing. *Neurocomputing*, *205*, 472–480.
- Zhou, N., Pan, S., Cheng, S., & Zhou, Z. (2016). Image compression-encryption scheme based on hyper-chaotic system and 2D compressive sensing. *Optics & Laser Technology*, *82*, 121–133.
- Zhou, N., Zhang, A., Wu, J., Pei, D., & Yang, Y. (2014a). Novel hybrid image compression-encryption algorithm based on compressive sensing. *Optik*, *125*(18), 5075–5080.
- Zhou, N., Zhang, A., Zheng, F., & Gong, L. (2014b). Novel image compression-encryption hybrid algorithm based on key-controlled measurement matrix in compressive sensing. *Optics & Laser Technology*, *62*, 152–160.



**R. Ponuma** received her B.E. degree in Electronics and Communication Engineering from Manonmaniam Sundaranar University, India, in 2002 and M.E. degree in Communication Systems from Anna University, Chennai, India, in 2004. Her current research interests include Image processing and Cryptography. She is a Life member of Indian Society for Technical Education (ISTE).



**R. Amutha** graduated from Thiagarajar college of Engineering, India, in the year 1987. She received her M.E. degree from PSG college of Technology, Coimbatore, India and her Ph.D. degree from Anna University in 2006. She has got 29 years of teaching and 15 years of research experience. She is a Life member of Indian Society for Technical Education (ISTE) and Institution of electronics and Telecommunication Engineers (IETE). Her research area includes Wireless communication network and Image processing.