Multimed Tools Appl (2018) 77:11669–11692
DOI 10.1007/s11042-017-4811-x

Chaos triggered image encryption - a reconfigurable security solution

Balakrishnan Ramalingam[1] · **Dhivya Ravichandran**[1] · **Arun Adhithiya Annadurai**[1] · **Amirtharajan Rengarajan**[1] · **John Bosco Balaguru Rayappan**[1]

Received: 24 September 2016 / Revised: 10 March 2017 / Accepted: 3 May 2017 /
Published online: 1 June 2017

Abstract Recently, diverse types of chaotic image encryption algorithms have been explored to meet the high demands in realizing secured real time image sharing applications. In this context, to achieve high sensitivity and superior key space, a multiple chaotic map based image encryption algorithm has been proposed. The proposed algorithm employs three-stage permutation and diffusion to withstand several attacks and the same is modelled in reconfigurable platform namely Field Programmable Gate Array (FPGA). The comprehensive analysis is done with various parameters to exhibit the robustness of the proposed algorithm and its ability to withstand brute-force, differential and statistical attacks. The synthesized result demonstrates that the reconfigurable hardware architecture takes approximately 0.098 ms for encrypting an image of size 256 × 256. Further the resource utilization and timing analyzer results are reported.

Keywords Security · FPGA · Chaos · Multiple chaotic maps · Image encryption

1 Introduction

A plethora of multimedia files (text, image, audio, video) is transmitted over the Internet with the rapid development in Information and communication technology. Internet is alleged to be easily intercepted by eavesdroppers and hence it becomes liable to high security threats. Thus, security has become a key issue during the storage and transmission of the digital data files over an unprotected public channel. In this scenario, cryptography plays a vital role in protecting the digital files against malicious attacks [6, 23]. The conventional encryption schemes such as Advanced Encryption Standard (AES), Rivest Cipher 6 (RC6), International

✉ John Bosco Balaguru Rayappan
rjbosco@ece.sastra.edu

[1] School of Electrical & Electronics Engineering (SEEE), SASTRA University, Thanjavur 613 401, India

Data Encryption Algorithm (IDEA) and other symmetric cryptographic algorithms are mostly focused on encrypting the textual data. But, image encryption schemes differ from text encryption as they are identified by the existence of huge data and high redundancy [5, 13, 17].

In recent years, chaotic based cryptosystem has achieved greater attention among researchers for encrypting bulk images. The key advantage of chaotic cryptic modules over traditional encryption algorithms are elevated ergodicity, high sensitivity to initial conditions and large key space [17–19, 30]. Chaotic map works in the aspect of achieving greater randomness, which is one of the key features for designing any successful cryptosystem. There are many image encryption schemes available employing one and two-dimensional chaotic maps. But some of these schemes showed security flaws due to limited discontinuous range of chaotic behaviours [8, 9, 21]. Recently many works have been reported with high dimensional chaotic maps to enhance key space and highly random behaviour in chaos based image crypto system [11, 31]. Further, by applying more than one chaotic map enlarges the key space, which makes the system highly resistant against various cryptanalysis attacks [25, 33].

A substitution-diffusion architecture based symmetric image cipher has been proposed by Patidar et al., [20]. The chaotic standard map and logistic map have generated the chaotic key stream to accomplish the robust substitution and diffusion plot. Xingyuan et al., have proposed a colour image encryption scheme using the combination of 1D and 2D logistic map structures [26]. The two chaotic maps are iterated alternatively to generate the chaotic matrix to permute and diffuse the image pixels. Zhang et al., [32] have given spatio-temporal chaotic based image encryption algorithm. This spatio-temporal system embraces of logistic map and piece-wise linear chaotic construct to generate pseudo-random number series to substitution and diffusion phase. DNA and improved 1D chaotic map based colour image encryption scheme was proposed by Xiangjun et al. The three improved one-dimensional chaotic map was used to generate the secret key to randomly encode the plain image into DNA matrices [29]. Pareek et al., [17] have implemented a scheme with interconnected constructs of two logistic maps to encrypt images. In this scheme, secret key with a size of 80 bits has been used for initial conditions in both logistic maps. In the iteration stage, random sequence from the first chaotic map was given as initial seed to second chaotic map. Chen and Lorenz chaotic map based colour image encryption was proposed by leyuan et al. The combined scheme have enriched chaotic range, enlarged key space and improved complex behavior [28]. Abolfazl et al., have developed robust keystream generator for colour image encryption scheme using hyper chaotic functions and non-uniform cellular automata [14]. The combination has achieved better entropy and strong resistance against noise attacks.

Nevertheless, combination of chaotic maps not only increases the security of the cryptosystem but also increases the difficulty of software implementation because of its computational complexity [15]. Application Specific Integrated Circuit (ASIC) based hardware implementations have more advantages over software. It provides inherent parallelism compared to software and achieves higher throughput. Its custom design of hardware significantly speeds up the execution. At the same time, ASIC based hardware design cycle is more expensive over software and less flexible to remodel the security algorithm. Graphic Processing Unit (GPU) is a hardware platform widely used in security application. It is a powerful device for computationally demanding applications. However, GPU has higher latencies and consumes huge power and has moderate flexibility which are not suitable for real-time security application and battery operated devices.

Field Programmable Gate Array (FPGA) is an impressive platform for implementing real time digital image encryption schemes [2, 3, 15]. It provides inherent parallelism to achieve higher throughput and also has software like programmable flexibility, which can be reconfigured at runtime in the field to meet the customer requirement, or add the new security

features through secure remote update. Moreover, System on-chip FPGA has in-build ARM based hard processor system consisting of multi-core ARM processor, which can be used to configure the FPGA fabric through U-boot configuration scheme. It eliminates the need of external circuitry and reduces the design cost. Hence this work focuses on the implementation of image encryption algorithm adapting multiple chaotic maps on reconfigurable platform.

2 Methodology

The proposed image encryption scheme has two phases namely permutation and diffusion. Each phase uses two one-dimensional chaotic maps. Permutation phase makes use of Combined Tent and Sine map (CTS) to shuffle the pixel values between the image planes. The diffusion phase takes up the Combined Logistic map and Sine map (CLS) to alter the value of pixels.

Figure 1 shows the functional block diagram of the combined chaotic scheme. This scheme unifies the nonlinear combination of two different one dimensional chaotic maps, which are considered as seed maps. The combined chaotic system is defined by the following the Eq. (1),

$$x_{i=1} = (F_1(x_i, p) + F_2(x_1, q)) \bmod 1 \tag{1}$$

where, $F_1(p,\ x_i\ )$ and $F_2(q, x_i\ )$ are the two one dimensional chaotic maps with control parameters p and q, mod represents the modulo operation for ensuring the range of output data is within the interval [0,1] and i is the iteration number. In each iteration, outputs of both the chaotic maps are integrated through XOR operation.

In order to implement the combined chaotic scheme, three nonlinear one-dimensional chaotic maps such as logistic, tent and sine are considered, which are mathematically defined by Eqs. (2)–(4) as given in Table 1. Using these chaotic seed maps, two new chaotic systems namely Combined Logistic – Sine (CLS), Combined Tent – Sine (CTS) have been modelled. It can be mathematically represented by Eqs. (5) and (6) as listed in Table 1.

The output of combined chaotic system shows the mixed chaotic property of two seed maps. Hence, it has wider chaotic range and larger key space than its seed map. Even if one seed map is out of the chaotic range, the proposed system still maintains its chaotic behaviour. This can be demonstrated by the results of bifurcation diagram and Lyapunov exponents as shown in Fig. 2 (a - e) and Fig. 3 (a - e).

The combined chaotic system has achieved large positive Lyapunov exponents values, which ensures the enhanced chaotic behavior. Further, the bifurcation diagram of combined

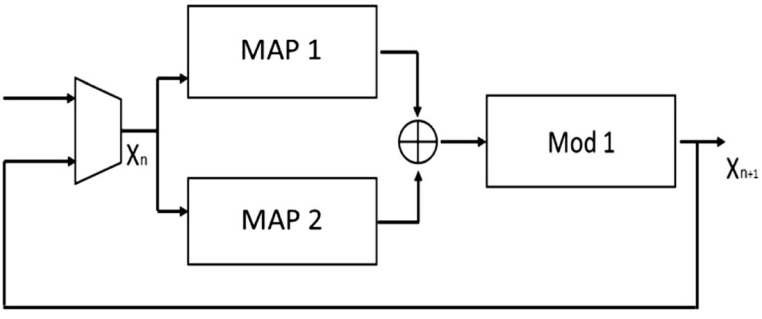

Fig. 1 Block diagram of the combined chaotic system

Table 1 Mathematical representation of the chaotic maps

Chaotic map	Function	Parameters	S. No
Logistic Map	$x_{N+1} = l(x_N, r) = r*x_N*(1-x_N)\ ; 0 < x_N < 1$	$r \rightarrow$ Control parameter $x_N \rightarrow$ Seed value for the N + 1th iteration $N \rightarrow$Number of iterations	(2)
Sine Map	$x_{N+1} = s(x_N, a) = a*\frac{\sin(\pi*x_N)}{4}\ ; 0 < x_N < 1$	$a \rightarrow$ Control parameter $x_N \rightarrow$ Seed value for the N + 1th iteration $N \rightarrow$Number of iterations	(3)
Tent Map	$x_{N+1} = t(x_N, u) = \begin{cases} u*\frac{x_N}{2} & ; 0 < x_N < 0.5 \\ u*\frac{x_N}{2} & ; 0.5 < x_N < 1 \end{cases}$	$u \rightarrow$ Control parameter $x_N \rightarrow$ Seed value for the N + 1th iteration $N \rightarrow$Number of iterations	(4)
Combined Logistic – Sine (CLS)	$x_{N+1} = ls(x_N, r, a) = mod\left(r*x_N*(1-x_N) \oplus a*\frac{\sin(\pi*x_N)}{4}, 1 \right)\ ;\ 0 < x_N < 1$	$a, r \rightarrow$ Control parameters $x_N \rightarrow$ Seed value for the N + 1th iteration $N \rightarrow$Number of iterations	(5)
Combined Tent – Sine (CTS)	$x_{N+1} = ts(x_N, u, a) = \begin{cases} mod\left(u*\frac{x_N}{2} \oplus a*\frac{x_N}{2}, 1\right); 0 < x_N < 0.5 \\ mod\left(u*\frac{x_N}{2} \oplus a*\frac{x_N}{2}, 1\right); 0.5 < x_N < 1 \end{cases}$	$u, a \rightarrow$ Control parameter $x_N \rightarrow$ Seed value for the N + 1th iteration $N \rightarrow$ number of iterations	(6)

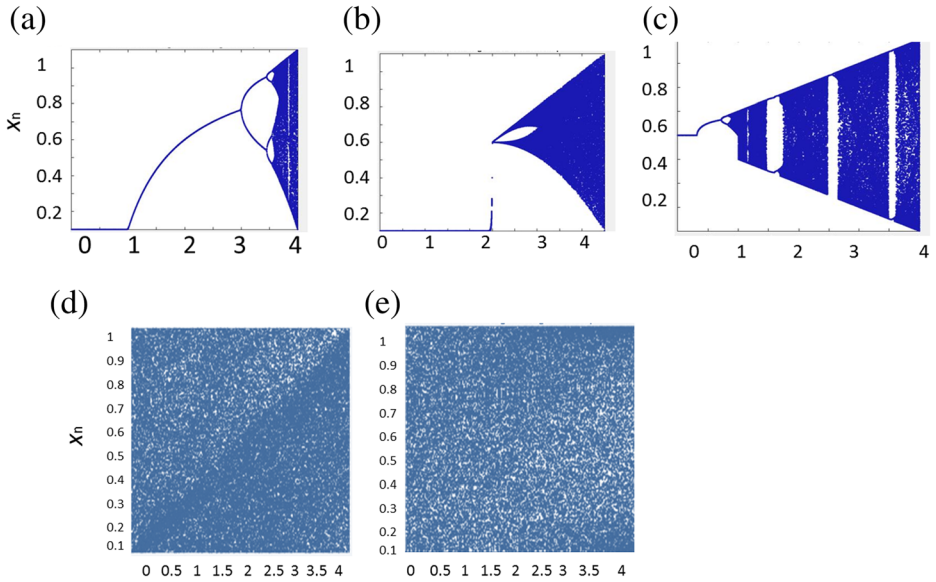

Fig. 2 Bifurcations diagram of (**a**) Logistic map, (**b**) Tent Map, (**c**) Sine map, (**d**) CLS and (**e**) CTS

chaotic system has spread out throughout the entire region of 0 and 1, which again proves its suitability for multimedia security applications.

2.1 Permutation phase

In Permutation phase, image pixels are rearranged through three routines namely inter-planar row permutation, inter-planar column permutation and inter-planar pixel shuffling using CTS chaotic random sequence. The functional parameter of three stage permutation has given in Table 2.

Fig. 4 shows the functionality of three stage permutation. Its described through 3 × 3 RGB color image where Ri (i = 1,2,..9), Gi (i = 1,2,..9) and Bi (i = 1,2,..9) represent the nine pixels in red, green and blue planes of the image respectively. Three sequences are generated from the CTS map. At first, sequence 1 of length P1 (= 3 M) is generated to perform inter-planar row shuffling. Next, sequence 2 of length P2 (=3 N) is generated to perform inter-planar column shuffling. Finally, sequence 3 of length P1 × P2 is generated to shuffle the inter-planar pixels to get the final shuffled image.

2.2 Diffusion phase

The diffusion process was achieved by three routines which is shown in Fig. 5. The functional parameter of three routine are listed in Table 3.

Step 1: Let the red, green and blue planes of the colour image I from permutation operation be R, G and B respectively. Let Q1 be the first stage diffused image and R', G', B′ be the red, green and blue planes of Q1 respectively. Q1(w, h) be the pixel position of image Q1 at (w,h) {where w = 1,2…M and h = 1,2,…,N} is estimated using equation ((Eq. 7))

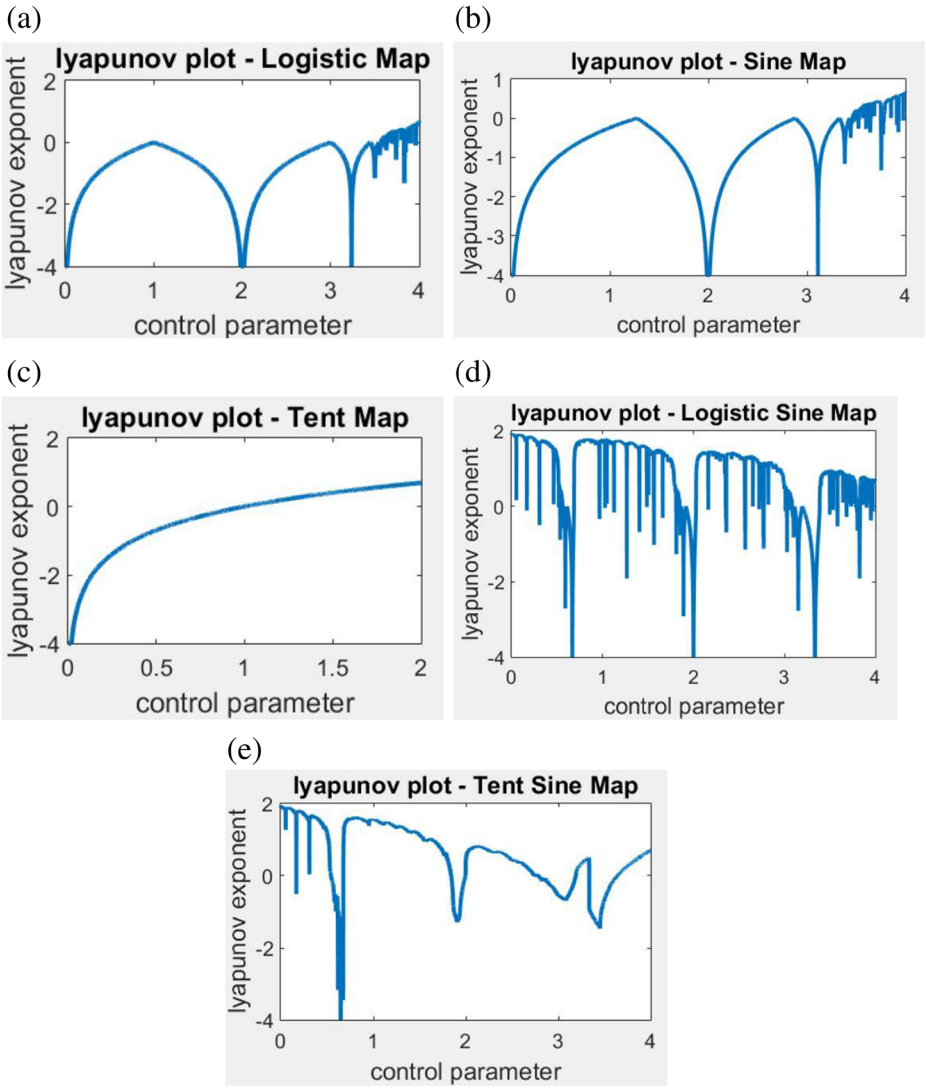

Fig. 3 Lyapunov exponents of (**a**) Logistic map, (**b**) Sine map, (**c**) Tent map, (**d**) CLS and (**e**) CTS

$$Q_1 = \begin{matrix} R'_{w,h} = R_{w,h} \oplus G_{w,h-1} \oplus B_{w,h-1} \\ G'_{w,h} = G_{w,h} \oplus R_{w,h-1} \oplus B_{w,h-1} \\ B'_{w,h} = B_{w,h} \oplus R_{w,h-1} \oplus G_{w,h-1} \end{matrix} \quad (7)$$

Here Q1(w,h) = I(w,h) at h = 1. In this process, the pixel values of every plane with that of the other two planes are diffused.

Step 2: The chaotic sequence of length 3 × M × N is generated. Let the chaotic series generated by CLS be Φ (x, y) and the individual element in the chaotic sequence is quantized in the range (0, 255) by the Eq. (8),

Table 2 Permutation phase parameter description

Parameters		Description
Permutation phase	M	Size of row in original image
	N	Size of column in original image
	P1	P1 = 3 M; thrice the size of row of original image
	P2	P1 = 3 N; thrice the size of column of original image

$$Q2 = \text{floor}\,(256, \Phi\,(x, y)) \tag{8}$$

Q2 represents the quantized chaotic sequence. Now, Q2 is XOR with Q1 to produce the diffused image,

$$Q3 = \text{XOR}\,(Q1, Q2) \tag{9}$$

Step 3: As a first part of this step, three planes of image Q3 is concatenated and converted into 1D array. Let D be the user defined random number (restricted to 0 to 255). The final diffused image Q is obtained by using (Eq. (10)),

$$Q(i) = \begin{cases} Q_3(i) \oplus D & ;for\ \ i = 1 \\ Q_3(i) \oplus D \oplus Q(i-1) & ;\text{otherwise} \end{cases} \tag{10}$$

2.3 Decryption process

Step 1: As a first part of this step, Q is converted into 1D array. The second part took each element of the 1D array and XOR the same with the predefined

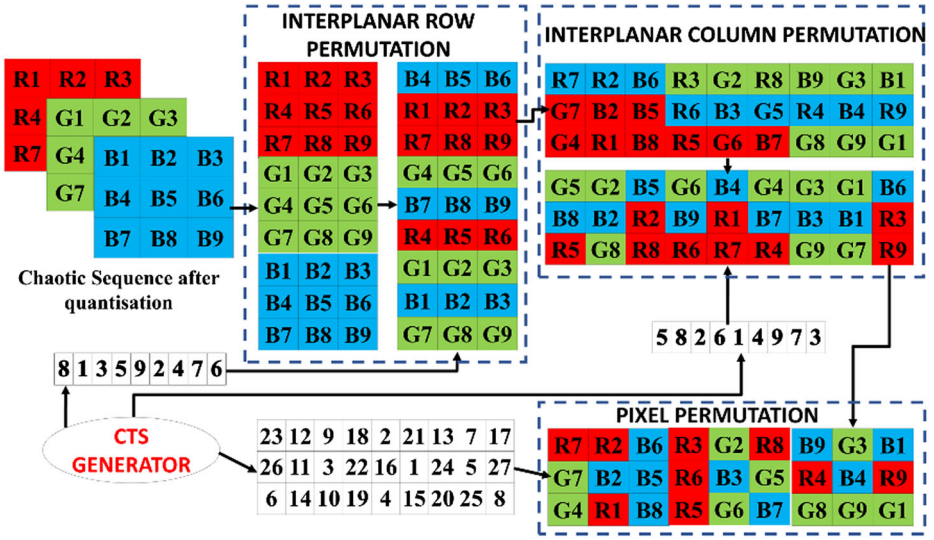

Fig. 4 Permutation phase using CTS map

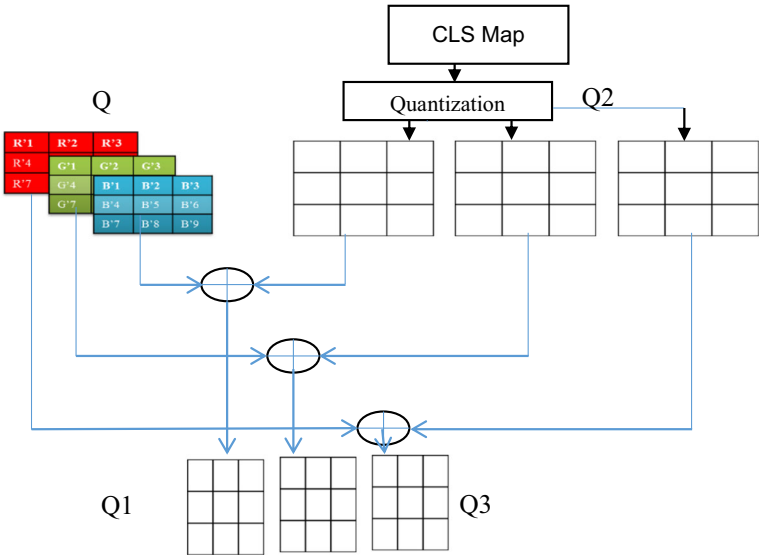

Fig. 5 Diffusion Phase

pseudo random number sequence D(i) (restricted between 0 to 255) to obtain Q_3 using (Eq. (11)),

$$Q_3(i) = \begin{cases} Q(i) \oplus D(i) & ;for\ \ i = 1 \\ Q(i) \oplus D(i) \oplus Q(i-1) & ;\text{otherwise} \end{cases} \tag{11}$$

Step 2: Let the chaotic series generated by CLS be Φ (x, y) and the individual element value is restricted 0 to 255 by following the Eqs. (12) and (13),

$$Q_2 = \text{floor}\ (256, \Phi\ (x, y)) \tag{12}$$

Then, Q_2 is XORed with Q_3 to produce the diffused image,

$$Q_1 = XOR\ (Q_3, Q_2) \tag{13}$$

Table 3 Diffusion phase parameter description

Parameter	Description
Q_1	First stage diffused image
$R_{w,h}$	Red plane pixel at position (w, h) of Q_1
$G_{w,h}$	Green plane pixel at position (w, h) of Q_1
$B_{w,h}$	Blue plane pixel at position (w, h) of Q_1
$R_{w,\ h\text{-}1}$	Red plane pixel at position (w, h-1) of Q_1
$G_{w,\ h\text{-}1}$	Green plane pixel at position (w, h-1) of Q_1
$B_{w,\ h\text{-}1}$	Blue plane pixel at position (w, h-1) of Q_1
$\Phi\ (x, y)$	Chaotic series generated by CLS
Q_2	Second stage diffused image
Q_3	Third stage diffused image
D	User defined random number
Q	Final diffused image

Step 3: All the elements are extracted in the opposite order of encryption without losing the generality of this step using (Eq. (14))

$$\text{Encrypted image} = \begin{matrix} R_{w,h} = R_{w,h} \oplus G_{w,h-1} \oplus B_{w,h-1} \\ G_{w,h} = G_{w,h} \oplus R_{w,h-1} \oplus B_{w,h-1} \\ B_{w,h} = B_{w,h} \oplus R_{w,h-1} \oplus G_{w,h-1} \end{matrix} \quad (14)$$

3 Hardware architecture

Hardware architecture of the proposed image crypto system is shown in Fig. 6. The core consists of FSM based control unit, Permutation Unit (PE), Diffusion Unit (DU), chaotic map based secret key generator (CSG), 52 Kb on-chip memory and 512 kb on board SRAM. The functional description of each core is described as follows:

3.1 FSM based control unit

The control unit is instigated with Finite State Machine (FSM), which manages the sequence of entire encryption and decryption processes. FSM generates required Start of Process (SoP) and End of Process (EoP) signals to all the integrated hardware blocks include CSG, Permutation Unit, Diffusion Unit, Memory Interface Circuit (MIC), 52 KB on-chip memory and 512 KB external SRAM. Fig. 7. describes the sequence of encryption and decryption process through FSM state diagram.

ST1: In ST1 state, FSM reads the initial seeds and control parameters in IEEE 754 format from external SRAM memory through MIC and load them into CSG unit to generate the permutation and diffusion phase secret keys.

ST2: In ST2 state, FSM control unit triggers the CSG unit through SoP signal to generate the random key sequence for three-stage permutation process and waits for EoP

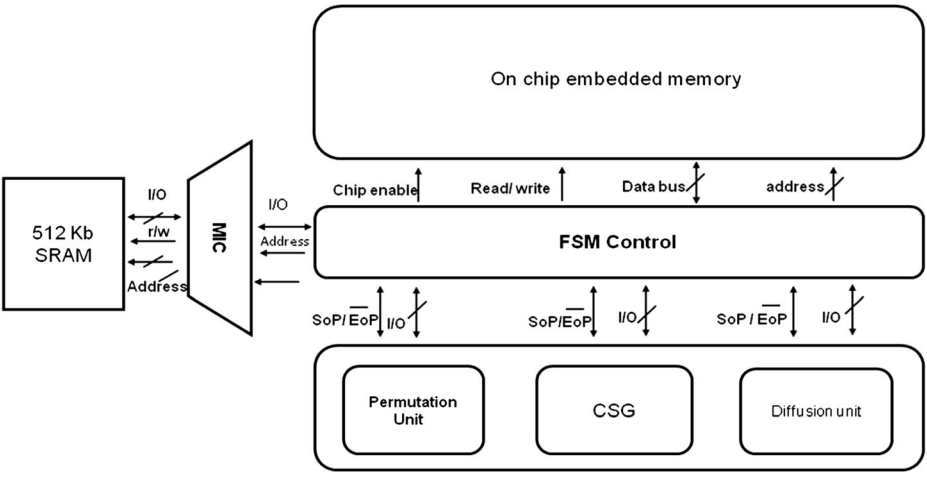

Fig. 6 Hardware architecture of the image encryption algorithm

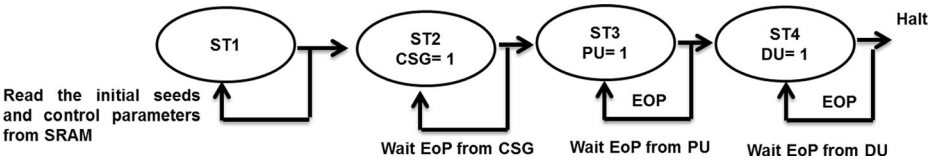

Fig. 7 Finite State Machine

acknowledgement signal from CSG to switch over to ST3. The generated random sequences in ST2 are stored in on-chip memory and used in ST3 to perform the permutation process.

ST3: In ST3, FSM issues the SoP to PU to perform the three-stage permutation process and waits in ST3 till it gets End of Process signal (EoP) from PU. Then the control is passed to ST4 state.

ST4: In ST4 state, FSM control unit enables both CSG and DU unit for execute the diffusion process. It remains in state ST4 till it gets End of Process (EoP) command from DU. The process skips to HALT state at the next clock cycle.

3.2 Chaotic sequence generation unit

Internal architecture of CSG unit is shown in Fig. 8. It consists of IEEE 754 standard single precision floating point arithmetic IP cores. This core comprises of subtractor, multiplier, divider and sine

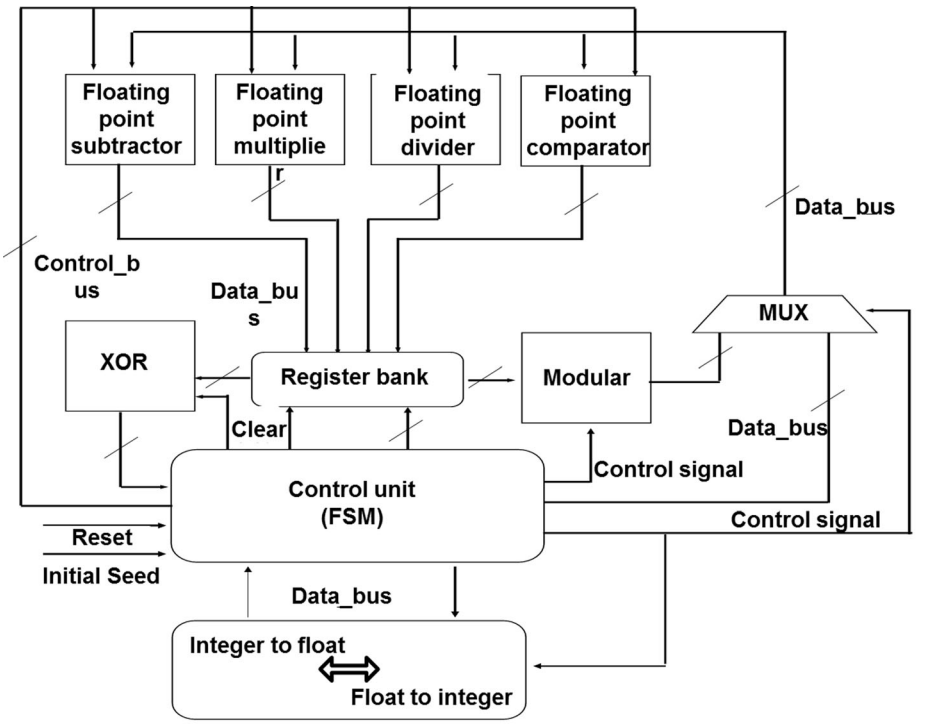

Fig. 8 Hardware architecture of CSG Unit

circuits, which are used to accomplish the floating point coupled chaotic systems such as CTS and CLS in FPGA. IEEE 754 standard 32 bit single precision floating point format is used to represent both initial condition and control parameters of chaotic maps. The 1-bit SoP and EoP signals are used to control all the arithmetic modules within the units and synchronization between modules and the systems. In addition, the two-input XOR gate has been used to bond two chaotic maps binary sequence and produces the final pseudo random sequence. The output of each iteration is stored in on-chip memory for permutation and diffusion processes.

3.3 Permutation architecture

Functional description of permutation unit is shown in Fig. 9, which is interlinked with on- chip memory and Memory Interface Circuit (MIC) to perform the three-stage permutation starting from row scrambling to inter-planar pixel shuffling. In each permutation stage, image pixels positions are rearranged randomly in external SRAM according to CLS random sequence, which is accomplished through Memory Interface Circuit (MIC).

The permutation unit computes the starting address of every row and column in SRAM according to Eqs. (15–17) and feed to the MIC.

$$\text{Relative address for row permutation} = \text{Base address} + (\text{row displacement} \times \text{row pointer}) \quad (15)$$

$$\text{Relative address for column permutation} = \text{Base address} + (\text{row index} \times \text{row pointer} + \text{Column displacement}) \quad (16)$$

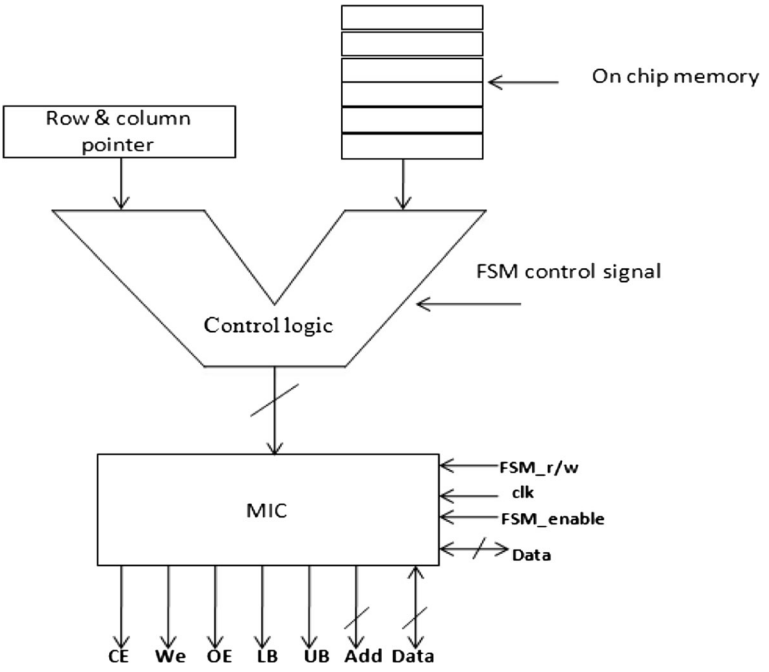

Fig. 9 Hardware architecture of Permutation Unit

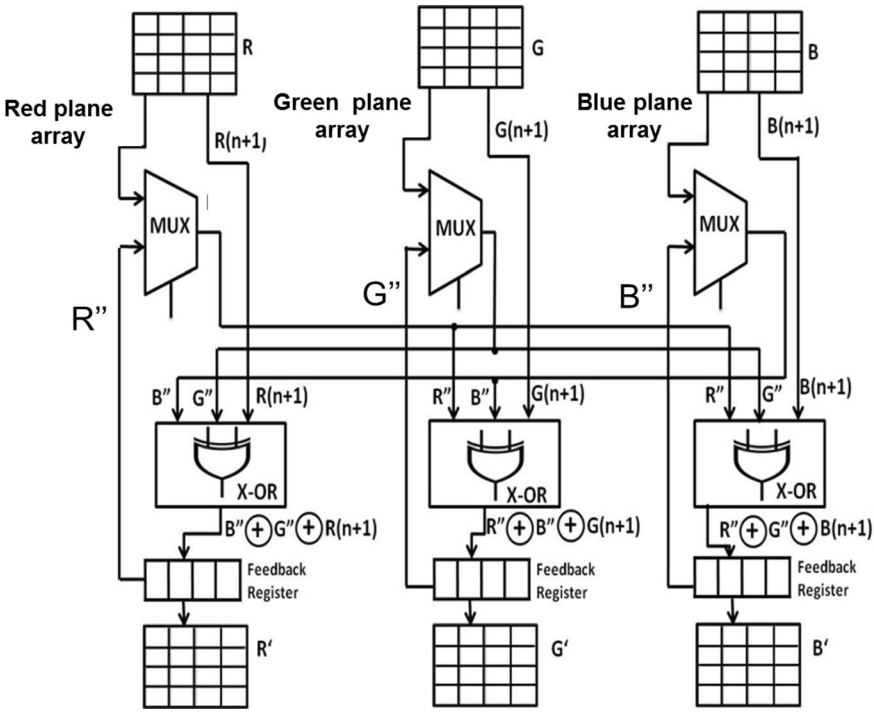

Fig. 10 Hardware architecture of diffusion unit

$$\text{Pixel permutation} = \text{Base address} + \text{pixel displacement} \tag{17}$$

where, base address represents the starting address of the permuted image in SRAM memory, row displacement represents shifting index of the pixel according to the chaotic random sequence, row index represents the starting position of the each row in SRAM, which varies from 0 to n in sequential manner (for 8 × 8 image, n = 8), row pointer represents the width of the image, column displacement represents shifting index of the pixel according to the chaotic random sequence and pixel displacement represents shifting index of the pixel according to the chaotic random sequence.

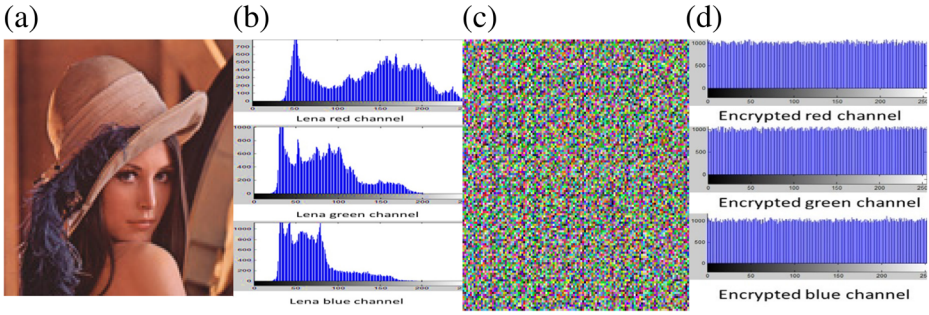

Fig. 11 (**a**) Lena, (**b**) histogram of Lena, (**c**) encrypted Lena and (**d**) histogram of encrypted Lena

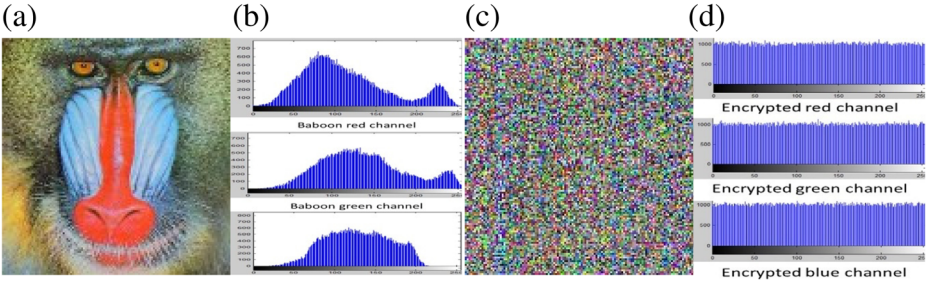

Fig. 12 (**a**) Baboon, (**b**) histogram of Baboon, (**c**) encrypted Baboon and (**d**) histogram of encrypted Baboon

3.4 Diffusion architecture

The schematic representation of diffusion unit is shown in Fig. 10. It comprises of three parallel XOR gates (each input is 8 bit width), sequential execution logic and intermediate register to achieve the three-stage XOR diffusion process sequentially.

In the first stage, pixel values of every plane (Red) are XORed with other two planes (Green and Blue) n + 1 array elements according to Eq. (7). In the next two stages, pixel values are further diffused by XOR gate using chaotic sequence generated by CLS scheme according to Eqs. (8 & 9). After three diffusion stages pixel values are stored in the output buffer R'G'B′ and finally restored into SRAM memory.

4 Performance and security analysis

Performance of the chao-image crypto system has been analysed to test whether it satisfies the trade-off between security and computational speed, cost and power. In order to analyse the same, statistical, key space, sensitivity, attack, hardware performance and complexity analyses were carried out following the standards.

4.1 Statistical analysis

In this analysis, the confusion and diffusion characteristics of the proposed chao-image crypto system have been tested by estimating histogram, correlation coefficient and information entropy.

Table 4 Variances of histograms compared among all secret keys in the proposed algorithm

Plain image	Plain image histogram variance	Key 1	Cipher image variance		
			u	r	x_N
Lena	521,592.321	19,543.421	16,326.872	15,031.514	16,127.583
Baboon	526,319.614	15,431.252	16,528.842	15,295.367	16,312.836
Pepper	338,086.591	16,527.779	16,251.174	16,132.603	16,267.244
Average		16,581.224	16,722.112	16,057.213	16,778.524

Table 5 Percentage of histogram variance with all the secret keys

Plain image	u (%)	r (%)	x_N (%)
Lena	1.321	1.134	1.221
Baboon	0.932	0.180	0.338
Pepper	0.371	0.530	1.543
Average	1.921	0.308	1.265

4.1.1 Histogram analysis

The histogram of the original and encrypted images has been tested for estimating the statistical resemblances. Histogram analysis of plain images and their respective encrypted images are shown in Fig. 11 (a - d) and Fig. 12 (a - d). The obtained results revealed the absence of biasing between the original and cipher images, which confirmed the failure of statistical attacks on the proposed algorithm.

Histogram of the encrypted image provides the visual proof of the sternness of the algorithm against statistical attack. However, it is necessary to estimate the quantitative value of uniform pixel distribution. This can be realized through calculating the variances of the cipher image histogram. The variance of histogram can be calculated by the following equation Eq. (18),

$$\mathrm{Var}\ (\mathrm{H}) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\left(H_i - H_j\right)^2}{2} \tag{18}$$

where, H_i, H_j are the number of pixels at various gray levels i, j respectively. The histogram variances for the various secret keys on plain image are calculated and tabulated in Table 4. Lower the value of obtained variance indicates the higher uniformity of pixel distribution. In addition, the percentage of histogram variance for all the secret keys are calculated and tabulated in Table 5.

From the above analysis, it is noticed that the variance value and their percentage difference for variation in secret keys are small. Thus the quantitative analysis also proves the resistance of the proposed algorithm against statistical attacks.

4.1.2 Correlation analysis

Typically, digital images have high correlation among the adjacent pixels in all the directions. To overcome the statistical attacks, it is mandatory to reduce the correlation among adjacent pixels in any cipher image. The following performance measures are attempted to examine the correlation among pixels. Correlation coefficient for those images is computed using the following Eqs. (19–22).

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{19}$$

Table 6 Correlation coefficients of adjacent pixels in original and encrypted image

Image	Horizontal			Vertical			Diagonal		
	R	G	B	R	G	B	R	G	B
Lena	0.9300	0.9163	0.8935	0.9606	0.9503	0.9313	0.8947	0.8817	0.8575
Lena encrypted	−0.0423	−0.0391	−0.0458	−0.0297	−0.0381	−0.0322	−0.0131	−0.0172	−0.0102
Baboon	0.8990	0.8317	0.8981	0.8965	0.8326	0.9064	0.8711	0.7857	0.8699
Baboon encrypted	−0.0004	−0.0003	−0.0004	0.0091	0.0098	0.0087	0.0093	0.0009	0.0009

Table 7 Correlation coefficients analysis of the proposed method with the available methods considering Lena image

Horizontal	Vertica	Diagonal	Ref no.
0.0965	−0.0318	0.0362	[10]
−0.0058	−0.0026	−0.0024	[4]
0.0005	0.0008	0.0011	[12]
0.0445	−0.0168	−0.0022	[7]
0.0017	0.0016	0.0011	[28]
−0.0084	0.0004	−0.0015	[26]
−0.0424	−0.0333	−0.0135	Proposed scheme

$$D(x) = \frac{1}{N}\sum_{i=0}^{N}[x_i - E(x)]^2 \tag{20}$$

$$\mathrm{Conv}(x,y) = \frac{1}{N}\sum_{i=1}^{N}[x_i - E(x)][y_i - E(y)] \tag{21}$$

$$r_{xy} = \frac{\mathrm{conv}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{22}$$

where, x and y are the grayscale values of the two adjacent pixels and N is the total pair of randomly selected pixels. The correlations between neighbouring pixel in the original and encrypted images are presented in Table 6.

From Table 6 one can observe the absence of correlation between the plain and its corresponding cipher image. Table 7 elucidates that the proposed scheme offers less correlation compared to other methods.

4.1.3 Information entropy

Information entropy is a quantitative measure of uncertainty of a random variable ‘m’ and it can be computed as follows (Eq. (23)),

$$H(m) = \sum_{i=0}^{M-1} P(m_i)log_2\frac{1}{P(m_i)} \tag{23}$$

where, P (mi) is the possibility of appearance of the symbol mi. Cipher image should possess entropy value nearing 8 in order to resist the statistical attack [27]. The entropy of ciphered images are listed in Table 8. These results confirmed that all entities are almost equal to speculative value of 8 and the proposed algorithm is more secure against entropy attack.

Table 8 Results of information entropy analysis

Image	Red	Green	Blue
Lena	7.9995	7.9994	7.9998
Baboon	7.9989	7.9864	7.9865
Pepper	7.9883	7.9910	7.9918
Airplane image	7.9977	7.9819	7.9899
Barbara	7.9897	7.9978	7.9984

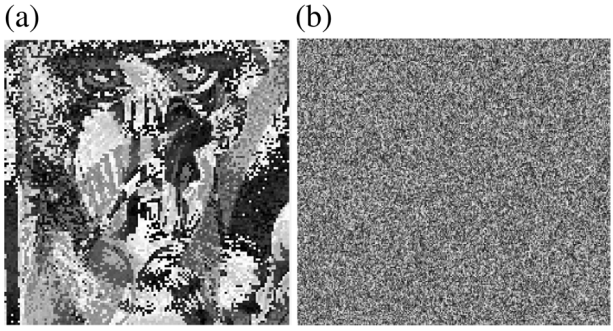

Fig. 13 Chosen plain text attack analysis (**a**) XOR (I1, I2) and (**b**) XOR (I1', I2')

4.2 Key space analysis

Key space analysis is the most important analysis to be carried out for evaluating the performance of any algorithm. Normally, high key space value reflects the elevated resistance towards brute force attack. Since multiple chaotic maps are employed in the proposed model, secret key generating variables are ultimately increased. In the case of CTS map, if the two initial conditions and parameters (T_0, S_0, u, a) have precision set to 14 decimal points, and hence the key space will be $10^{56} \approx 2^{224}$, which is huge to break through brute force attack.

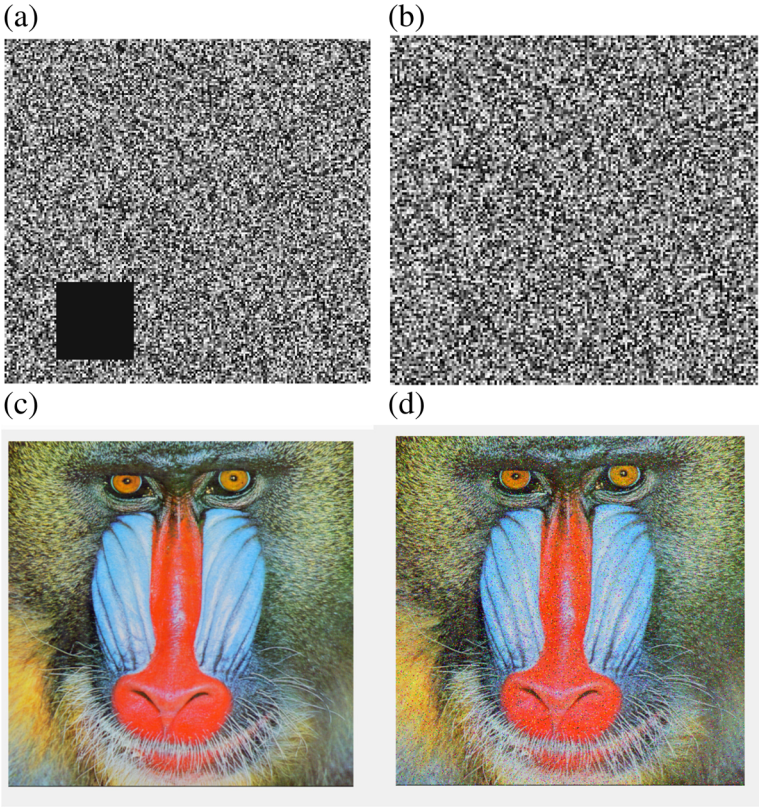

Fig. 14 Data loss and noise effects: (**a**) Encrypted Baboon with 60 × 60 data loss; (**b**) Encrypted Baboon with 1% salt and pepper noise; (**c**) Decrypted image of (a); (**d**) Decrypted image of (b)

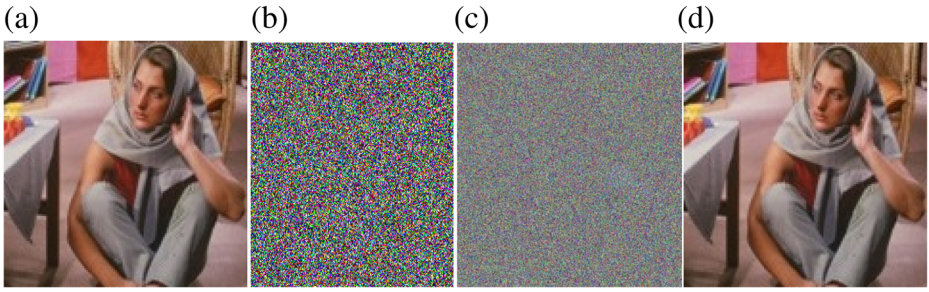

Fig. 15 (**a**) Original Barbara image, (**b**) Decrypted Barbara image with wrong initial condition T_0 = 0.44**5**76457898635, (**c**) Decrypted Barbara image with wrong parameter u = 1.88 and (d) Decrypted Barbara image with correct initial conditions and parameters (T_0 = 0.44**8**76457898635, S_0 = 0.87634594562348, u = 1.22, a = 3.98)

4.3 Chosen plain text attack analysis

Chosen plain text attack is one of the four traditional types of attacks that the cryptosystem may undergo. If the proposed cryptosystem is immune to resist this attack, then it has resistance towards other three attacks. The cryptanalyst considers two plain images P1, P2 and two cipher images C1, C2. In order to prove the systems' resistance towards chosen plain text attack, the proposed cryptosystem need to break the equation XOR (P1, P2) = XOR (C1, C2).

From the Fig.13, it is noticed that the equality has been broken which proves the systems' resistance towards chosen plain text attack.

4.4 Data loss and noise attacks

In real time data communication applications, the encrypted images will certainly experience few data losses as well as noise effects when transmitted. The cryptanalyst may intentionally do such attacks to alter the information contents while transmission. Hence, it is necessary for the encryption algorithm to resist such security attacks. To verify the performance of the proposed algorithm against these attacks, the intentional cropping and noise addition is done on the encrypted Lena image and the corresponding decrypted image is studied. The simulation results of these two security attacks (Fig. 14). It can be interpreted that the proposed scheme can get back most of the visual information even with the data loss and noise effects. Therefore the proposed method demonstrates its immunity against these security attacks.

Table 9 NPCR and UACI for the cipher images

NPCR				UACI				Ref.
R	G	B	Avg	R	G	B	Avg	
99.5983	99.6022	99.5392	99.57	33.6883	33.1105	33.5039	33.43	[10]
99.6013	99.6131	99.6226	99.61	33.4210	33.4485	33.4815	33.42	[4]
-	-	-	99.63	-	-	-	33.43	[28]
-	-	-	99.60	-	-	-	33.48	[26]
99.6155	99.6723	99.6432	99.64	33.8122	34.7362	32.7591	33.48	Proposed Scheme (Lena)

Table 10 Comparison analysis of correlation, NPCR and UACI values for different color image modalities

Image	Modalities		Correlation			NPCR	UACI
			Vertical	Horizontal	Diagonal		
Lena	RGB	Original	0.9300	0.9163	0.8935	99.64	33.48
		Encrypted	−0.0423	−0.0391	-0.0458		
	HSV	Original	0.900642	0.862577	0.835008	99.42	28.40
		Encrypted	0.158848	0.152099	0.152003		
	XYZ	Original	0.973459	0.947872	0.927407	99.37	26.31
		Encrypted	0.000694	0.004061	0.001779		

4.5 Key sensitivity analysis

The proposed model utilizes two different key sets to perform permutation and diffusion. Permutation uses two initial conditions and two control parameters to generate permutation key. If the permutation key set is P = {T_0 = 0.44876457898635, S_0 = 0.87634594562348, u = 1.22, a = 3.98}, even a slight change in the key set resulted in a undesired output. Fig. 15 depicts the effectiveness of the key sensitivity where the effect of just one bit change in key in the resultant image is highlighted.

4.6 NPCR and UACI analysis

Number of Pixel Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are the two widely used security analysis metrics to analyze the performance against differential attack. Number of pixel changes during differential attack is represented by NPCR and UACI focus on the difference between any two adjacent pixels. The NPCR and UACI are described by the Eqs. (24–26) respectively.

$$\mathrm{NPCR} = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100\% \tag{24}$$

$$\mathrm{UACI} = \frac{1}{M \times N}\left[\sum_{i,j} \frac{|C_1(i,j) - C_2\ (i,j)|}{255}\right] \times 100\% \tag{25}$$

$$D(i,j) = \left\{ \begin{array}{l} 0\ \ if C_1(i,j) = C_2(i,j) \\ 1\ \ if C_1(i,j) \neq C_2(i,j) \end{array} \right\} \tag{26}$$

where, C_1 and C_2 are the two encrypted images related to two original images with tiny change i.e., one pixel difference. M and N are the width and height of the image, D (i , j) is a bipolar array with the same size as images C_1 and C_2. D (i , j) is defined by Eq. (22). The values of NPCR (>99%) and UACI (≈33%) for Lena, Baboon, Pepper and Barbara images are listed in Table 9.

Table 11 Hardware resource utilization on Cyclone II FPGA

Description	Resource utilization	
Family	Cyclone II EP2C35662C6	Virtex 4 FPGA
Total logic elements	3851 (LE)	4273 (slices)
Total ram bits (M4 k blocks)	425,984	425,984
Embedded multiplier	11	24

Table 12 Computing complexity analysis

Hardware	Execution time (μs)
CTS	0.73
CLS	0.59
Memory read / write for one pixel	0.02

Observational results prove that the computed NPCR and UACI are near to the theoretical values and greater than existing image encryption schemes. Therefore, the projected encryption scheme is immune to differential attacks.

4.7 Comparison with other color image modalities

Comparison analysis of correlation, NPCR and UACI values for different color image modalities was listed in Table 10. From the above analysis (Table 10) the RGB color model achieves low correlation values, better NPCR and UACI than other two color image modalities. This is due to the independent of RGB color coefficient whereas other modalities have dependent pixel coefficient values.

5 Hardware performance analysis

The proposed image encryption scheme was modeled in Verilog Hardware Description Language (HD), synthesis in both Quartus II IDE and Xilinx ISE 14.0 and tested in Cyclone II and Virtex 4 FPGA. The synthesis results are listed in Table 11.

From the above analysis, it is realized that on chip memory has largely utilized to implement the proposed scheme which is mainly used to store the chaotic key sequence and pixel values in the execution stage.

In architectural point of view, it is very hard to compare the implementation of proposed algorithm in two different vendor FPGA-in terms of resource utilization (configurable logic block (CLB) or logic element (LE)). Because, every vendor have use different terminologies in their CAD tools to describe the logic capacity of their products where Altera uses the Logic Element (LE) methodology and Xilinx FPGAs uses slice to measure their logic capacity of a FPGA. Moreover, ALTERA and Xilinx logic array blocks are entirely different where Xilinx use 4 or 6-input LUT in Virtex-4 FPGA and ALTERA uses 4 -input or 3 input LUT in Cyclone II series FPGA.

5.1 Computing complexity analysis

Computational complexity of the algorithm has been estimated in real- time with zero plus logic analyzer and the results are given in Table 12.

Table 13 Hardware resource utilization of different schemes

Scheme	FPGA family	Resource utilization	Ref.
Lorenz hyper chaotic system	Spartan II	4721	[22]
Chaos-based random number generators	Virtex II	8968	[3]
Modified logistic map	Virtex-6	2016	[16]
Chao - cryptic image encryption (integrated)	Cyclone II	3851	Proposed Scheme

Table 14 Speed performance analysis

Implementation platform	Execution time (ms)	Ref.
1.3 GHz Pentium processor (Matlab)	54.7	[24]
3.0 GHz Pentium Core 2 Duo (Matlab)	864.5	[12]
2.7 GHZ Pentium processor (VC++)	93.0	[1]
Virtex-II-Pro FPGA	1.31	[15]
Cyclone II FPGA	0.098	Proposed scheme

From the above analysis, it is evident that the chaotic key stream generator is the most time-consuming part, which is due to the floating point computational process and combinational functions. It can be overcome by adding more parallel units to achieve better throughput but incur more logic resources in FPGA and consume more power.

5.2 Comparison with exiting scheme

The performance of hybrid image crypto system is compared with other existing schemes in terms of resource utilization and execution time. The comparison results are listed in Tables 13 & 14.

As can be seen, the proposed chaotic image crypto system perform faster than the existing software and hardware based schemes because of parallel execution of permutation and diffusion processes and also consume low resource utilization.

6 Limitation of hardware model

The proposed design has inbuilt BMP header reader. It can access the image pixel from external memory without any software assistance. However, for other colour image formats, the design should have header structure or software assistance to read the image pixels from the memory. The on board SRAM (512 KB) or SDRAM (8 MB) is used in the proposed design to store the image file. Hence, the design can adopt any larger image size. Further, the combined chaotic scheme can generate large number of secret key without any reputation for encrypting large size images. In addition, hardware blocks in the FPGA include CSG, PU and DU can operate in parallel at 400 MHz operating frequency to achieve higher throughput and real-time implementation.

7 Conclusion

Multiple chaotic map based image encryption algorithm is proposed and implemented in reconfigurable hardware platform. CTS and CLS maps are used to confuse and diffuse inter-planar image pixels respectively. The proposed blend has increased the key space and enhanced the robustness of cryptosystem. The performance is analyzed with several measures and proved that the proposed cryptic model is superior over the traditional encryption schemes and can withstand several attacks. The intended design is operated at 200 MHz and accumulates 3851 logic elements. Timing and power scrutiny confirmed that it is an attack resistant model for real-time hardware security applications. The proposed design can be extended for even larger image size and higher operating clock frequencies.

Acknowledgements The authors wish to express their sincere thanks to SASTRA University, Thanjavur for their financial support and extending infrastructural facilities to carry out this work.

References

1. Aneesh R, Jiju K (2012) Design of FPGA based 8-bit RISC controller IP core using VHDL. In: India Conf (INDICON), 2012 Annu IEEE, pp 427–32
2. Azzaz MS, Tanougast C, Sadoudi S, Dandache A (2013) Robust chaotic key stream generator for real-time images encryption. J Real-Time Image Process 8:297–306
3. Barakat ML, Mansingka AS, Radwan AG, Salama KN (2014) Hardware stream cipher with controllable chaos generator for colour image encryption. Image Process IET 8:33–43
4. Dong C (2014) Signal Processing : image communication color image encryption using one-time keys and coupled chaotic systems. Signal Process Image Commun 29:628–640
5. El-Samie FEA, Ahmed HEH, Elashry IF, Shahieen MH, Faragallah OS, El-Rabaie ESM, et al (2013) Image encryption: a communication perspective. CRC press, Taylor & Francis, pp 3–19
6. Ferguson N, Schneier B (2003) Practical cryptography, 1st edn. Wiley Publishing, Indiana
7. Kadir A, Hamdulla A, Guo WQ (2014) Color image encryption using skew tent map and hyper chaotic system of 6th-order CNN. Optik 125:1671–1675
8. Kocarev L (2001) Chaos-based cryptography: a brief overview. Circuits Syst Mag IEEE 1:6–21
9. Li C, Li S, Lo K-T (2011) Breaking a modified substitution–diffusion image cipher based on chaotic standard and logistic maps. Commun Nonlinear Sci Numer Simul 16:837–843
10. Liu H, Wang X (2010) Color image encryption based on one-time keys and robust chaotic maps. Comput Math with Appl 59:3320–3327
11. Liu L, Zhang Q, Wei X (2012) A RGB image encryption algorithm based on DNA encoding and chaos map. Comput Electr Eng 38:1240–1248
12. Mohammad Seyedzadeh S, Mirzakuchaki S (2012) A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map. Signal Process 92:1202–1215
13. Nemade VS (2012) Review of different image encryption techniques. World J 2:95–98
14. Abolfazl Yaghouti N, Mohammad Hossein M, Masood Niazi T (2017) Color image encryption based on hybrid hyper-chaotic system and cellular automata. Opt Lasers Eng 90:225–237
15. Ou SC, Chung HY, Sung WT (2006) Improving the compression and encryption of images using FPGA-based cryptosystems. Multimed Tools Appl 28:5–22
16. Pande A, Zambreno J (2013) A chaotic encryption scheme for real-time embedded systems : design and implementation. Telecommun Syst 52:551–561
17. Pareek NK, Patidar V, Sud KK (2010) Image encryption using chaotic logistic map. Image Vis Comput 24:926–934
18. Pareek NK, Patidar V, Sud KK. A random bit generator using chaotic maps. 10:32–8
19. Pareek NK, Patidar V, Sud KK (2011) Colour image encryption scheme based on permutation and substitution techniques. Adv Comput Sci Inf Technol Pt I 131:413–427
20. Patidar V, Pareek NK, Sud KK (2009) A new substitution–diffusion based image cipher using chaotic standard and logistic maps. Commun Nonlinear Sci Numer Simul 14:3056–3075
21. Ponomarenko VI, Prokhorov MD (2002) Extracting information masked by the chaotic signal of a time-delay system. Phys Rev E 66:262–215
22. Sadoudi S, Tanougast C, Azzaz M, Dandache A (2013) Design and FPGA implementation of a wireless hyperchaotic communication system for secure real-time image transmission. EURASIP J Image Video Process 2013:43
23. Schneier B (1996) Applied cryptography: protocols, algorithms, and source code in C. John Wiley & Sons, Inc., New York, pp 13-15
24. Wang Y, Wong K-W, Liao X, Xiang T, Chen G (2009) A chaos-based image encryption algorithm with variable control parameters. Chaos, Solitons Fractals 41:1773–1783
25. Wang Y, Lei P, Yang H, Cao H (2015) Security analysis on a color image encryption based on DNA encoding and chaos map. Comput Electr Eng 46:433–446
26. Wang X, Zhao Y, Zhang H, Guo K (2016) A novel color image encryption scheme using alternate chaotic mapping structure. Opt Lasers Eng 82:79–86
27. Wu Y, Zhou Y, Saveriades G, Agaian S, Noonan JP, Natarajan P (2013) Local Shannon entropy measure with statistical tests for image randomness. Inf Sci 222(10):323–342
28. Wu X, Kan H, Kurths J (2015) A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps. Appl Soft Comput J 37:24–39
29. Wu X, Kan H, Kurths J (2015) A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps. Appl Soft Comput 37:24–39

30. Yavuz E, Yazıcı R, Kasapbaşı MC, Yamaç E (2015) A chaos-based image encryption algorithm with simple logical functions. Comput Electr Eng 54:471–483
31. Zhang X, Shao L, Zhao Z, Liang Z (2014) An image encryption scheme based on constructing large permutation with chaotic sequence. Comput Electr Eng 40:931–941
32. Zhang X, Zhao Z, Wang J (2014) Chaotic image encryption based on circular substitution box and key stream buffer. Signal Process Image Commun 29:902–913
33. Zhou Y, Bao L, Chen CLP (2014) A new 1D chaotic system for image encryption. Signal Process 97:172–182

Dr. Balakrishnan Ramalingam received his M.Sc. degree in Electronics from Bharathidasan University, Tiruchirapalli, India in 2009. He received his Ph.D. degree from SASTRA University, Thanjavur, India in 2016. His research interests include embedded system design, reconfigurable hardware, information security and multiprocessor system on chip.

Dhivya Ravichandran received her B.E. (Electronics and communication engineering) in 2012, from St. Joseph's college of engineering, Chennai and M.Tech in Advanced communication systems from SASTRA University, Thanjavur in 2014. She is currently working as research scholar in school of electrical and electronics engineering, SASTRA University, India. Her research areas include information security, embedded systems and medical image security.

Arun Adhithiya Annadurai currently pursuing his B.Tech. final year (Electronics and Communication Engineering) at SASTRA University, Thanjavur, INDIA. His research areas include information security, embedded systems, medical image security and Internet of Things.

Dr. Amirtharajan Rengarajan was born in Thanjavur, Tamil Nadu province India, in 1975. He received B.E. degree in Electronics and Communication Engineering from P.S.G. College of Technology, Bharathiyar University, Coimbatore, India in 1997. M.Tech. and Ph. D. from SASTRA University Thanjavur, India in 2007 and 2012 respectively. He joined SASTRA University, Thanjavur, Tamil Nadu, India (Previously Shanmugha College of Engineering) as a Lecturer in the Department of Electronics and Communication Engineering since 1997 and is now Associate Professor, His research interests include Image Processing, Information Hiding, Computer Communication and Network Security. So far, he filed one international patent; he has published more than 125+ research articles in national and international journals and 22 I.E. conference papers with 4 Best Paper Awards. He also holds the Certificate of Appreciation from IBM in 2009 for Great Mind Challenge, Mentor IBM Academic Initiative Program. Recently, he received the Founder Chancellor Award for the best Ph.D. thesis for 2013 from SASTRA University and he received the SASTRA Anukul Puraskar for Higher Involvement in Research and Education Award for 2011–2012 and 2013–2014. He serves as a Life Member in CRSI, SSI, IAENG, and IACSIT. He also served as the TPC Member and Review Member for more than 30+ IEEE and Springer supported international conferences apart from more than 10 peer reviewed journals. He had been working on funded project in the field of steganography supported by DRDO, Government of India, New Delhi, India.

Dr. John Bosco Balaguru Rayappan was born in Trichy, Tamil Nadu province, India in 1974. He received the B.Sc., M.Sc. and M.Phil. Degree in Physics from St. Joseph College, Bharathidasan University, Trichy and Ph.D. in Physics from Bharathidasan University, Trichy, Tamil Nadu India in 1994, 1996, 1998 and 2003, respectively. He joined the faculty of SASTRA University, Thanjavur, India in Dec 2003 and is now working as Professor & Associate Dean Research School of Electrical and Electronics Engineering at SASTRA University, Thanjavur, Tamil Nadu, India. His research interests include Lattice Dynamics, Nanosensors, Embedded System and Steganography. So far he has published 197+ Research articles in National and International journals and 14 conference papers. He has Supervised 25 Master Students and Supervising 5 Ph.D. Scholars. Currently he is working on four funded projects in the fields of Nanosensors and Steganography supported by DST and DRDO, Government of India, New Delhi.