



A new design of cryptosystem based on S-box and chaotic permutation

M. A. Ben Farah¹  · R. Guesmi¹ · A. Kachouri¹ · M. Samet¹

Received: 9 April 2019 / Revised: 6 December 2019 / Accepted: 31 January 2020 /
Published online: 18 March 2020

© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

In this paper, we present a new design of cryptosystem characterized by an optimized substitution box (S-box) and random permutation. Our proposed S-box is generated using a modified genetic algorithm. The crossover process is performed with sophisticated research using the best previous population. We use randomness and ergodicity of the logistic map to add complexity and robustness to our proposed method. Many tests proving the nonlinearity of our S-box have been carried out to demonstrate the efficiency of our algorithm. In the second part, we offer a new permutation algorithm based on a chaotic sequence generated from the logistic map. To show the performance of our proposition, we compare our results with previous algorithms. The results of its statistical analysis, like entropy value and correlation between adjacent pixels, show that the proposed image encryption scheme provides security for image encryption. The time speed of the proposed algorithm confirms the possibility of real-time implementation.

Keywords S-box · Logistic map · Randomness · Nonlinearity · Optimization

✉ M. A. Ben Farah
aminefrh5@gmail.com

R. Guesmi
ramzi.guesmi@gmail.com

A. Kachouri
abdennaceur.kachouri@enis.rnu.tn

M. Samet
mounir.samet@enis.rnu.tn

¹ LETI Laboratory, National Engineering School of Sfax, Sfax University, B.P.W. 3038, Sfax, Tunisia

1 Introduction

1.1 Research background

In the last decade, new technologies have known considerable development and thus, have had a significant impact on society. Technology has changed the communication world. Different means of communication have been developed, such as facebook, twitter, Instagram... These social media have facilitated communication and the share of information at high speed. One can say that technology and communication are interdependent. In this way, the more communication is established, the more we care about creating new solutions to privacy threats. Secure communication is essential to protect one's privacy. In general, two entities need to communicate without an interception and in a safe way. Several methods were used to ensure the security and confidentiality of transmitted information through the internet. Encryption is one of the essential measures to provide security. It is true that it is a crucial tool but it cannot provide a high level of protection on its own. Security researchers are thus busy developing new technologies and fixing out the flaws in existing ones. In this context, the chaotic system properties such as randomness, sensitivity to the initial condition, and ergodicity give promising results in building robust cryptosystems.

1.2 Review of the previous chaotic cryptosystems

Due to the importance of images and video transmission in our life, many encryption algorithms have been proposed, we can classify these algorithms in two classes. The first one includes those classical encryption schemes like RSA, AES, DES... [8, 32], while the second one contains algorithms exploiting the chaotic properties of nonlinear functions [6, 7, 37]. These maps have been used in the masking and confusion/diffusion steps. The most important point of this category is the exploitation of the ergodicity and randomness properties of chaotic functions. Indeed, many recent studies have focused on demonstrating the randomness effect of chaotic functions (NIST test) [23]. Currently, the challenge in encryption scheme is to find a secure algorithm with an excellent complexity performance. The last encryption schemes proposed in the previous few years were based on the twin study of performance/complexity in order to demonstrate the possibility of using chaotic functions in secured networks and web services. The techniques that we have used to demonstrate the robustness and complexity of the algorithms can be classified as follows: firstly, we should use an excellent dataset to generalize the use of our solution. Secondly, we study the histogram behavior of encrypted images to demonstrate the unpredictability process with statistical study like entropy values and correlation between two adjacent pixels. Finally, a speed study and a complexity discussion were conducted to evaluate the real-time implementation.

Most of the chaotic encryption schemes are based on confusion/diffusion Shannon properties [39, 40]. Recently, many algorithms used S-box (Substitution-box) to enhance the confusion phase [15, 25]. The advantages of using this technique are the simple implementation and incorporation in the cryptosystem and the nonlinearity effect on the encryption algorithms. AES and DES algorithms were the first to use S-boxes. Eventhough S-boxes have contributed to strengthening these algorithms, their major inconvenience remains their statistic behaviors. In our solution, we used chaotic systems to draw S-boxes in order to produce dynamicity. Jakimoski and Kocarev used tow chaotic logistic maps [18]: exponential and logistic. They demonstrated that both can be implemented in various processors and be robust

to several attacks. In [33], they proposed a solution using 2D chaotic map to obtain random and bijective S-boxes. Among the advantages of this method are low complexity, obvious parametrization, and large key space. Chen et al. [12] improved this method by using Backer map. Fatih et al. [30] proposed S-boxes using chaotic Lorenz system as a source of randomness. Authors in [26] proposed S-boxes based on complex chaotic system and random noise. Three S-Boxes are generated by a complex Chen system, and each S-box is used to encrypt one of the color components.

Now, researchers utilize meta-heuristic algorithms in different software applications such as image processing, cloud computing, big data and encryption mask phase. They use the following optimization techniques: the so-called PSO (Particle Swarm Optimization) or ACO (Ant Colony Optimization) [4, 5, 16, 20, 21, 27, 28].

In our previous work presented in [9], we proposed novel S-boxes based on crossover and mutation; we generated seven S-boxes with acceptable nonlinearity parameter. We have profited from the chaotic Properties and genetic algorithms. Development of the algorithm steps showed some weaknesses due to the randomness caused by the chaotic function. This resulted in a weak solution domain.

In the last five years, many researchers tried to include natural phenomena in the encryption algorithms to add more unpredictability in the confusion phase. Tests of these attempts have given encouraging results. For example, in [17], authors use DNA concept in the encryption algorithm and they generate a mask using 1D and 2D chaotic maps to obtain ciphered image. In [10], we used DNA sequences and hash algorithm to conceptualize a new cryptosystem. In [11], we proposed a new cryptosystem based on hash function key generator, in order to increase the robustness of our cryptosystem. Now, the main objective of security researches is to find a robust symmetric chaotic encryption algorithm for specific applications. The challenge is to demonstrate that chaotic encryption can be applied for real-time communication. For example, the author in [24] proposed a chaotic dynamic key cryptosystem based on a neural network so as to secure wireless communication. Another solution was proposed in [31]. Authors used a chaotic encryption algorithm to secure medical images. Authors in [13] proposed an application of chaotic cryptosystem in IoT E-healthcare. They used a 2D chaotic function to generate pseudo-random numbers used as keys of their scheme. Therefore, any proposed encryption algorithm must meet the robustness and speed criteria, so that it could be applied in several applications.

1.3 Our contribution

In this paper, we present a new design of a chaotic S-box based on modified genetic algorithm techniques. The selection of population is a fundamental part to optimize the algorithm. So, we develop a new intelligent crossover that explores individuals found in our work [9]. We also divided the research domain into sub-regions to facilitate population construction. The enhanced algorithm selects individuals according to the quality of the last operation in our algorithm in order to find the best population. The second originality of our work lies in creating an interdependency between the input image and the round number utilized by the encryption algorithm. This relation enhances security and gives more unpredictability to the proposed algorithm. In our work, we present an algorithm based on a chaotic sequence generated by the logistic map. This sequence allows us to change the order of sub-blocks randomly and add more security to the encryption algorithm. The results of statistical analysis and security analysis show that the proposed image encryption scheme guarantees efficiency and robustness for image encryption.

1.4 Structure of the paper

The remainder of the paper is organized as follows. In section 2, we proposed a new method to draw an S-box using intelligent crossover and the best exploitation of the chaotic range function; a statistical test was elaborated and showed the algorithm's performances. In section 3 we developed a new cryptosystem based on Shannon confusion/diffusion properties. In section 4, we carried out statistical tests and security analysis to demonstrate the robustness and the efficiency of the proposed encryption algorithm. Finally, a conclusion is presented in section 5.

2 The proposed S-boxes approach

An S-box (Substitution box) is used in an encryption algorithm to satisfy the Shannon confusion property. To evaluate an S-box, we determine the nonlinearity parameter (Eq. (1)), which measures the degree of confusion. Generally, this parameter contributes to the robustness of the encryption algorithm (An important step when we propose a secure cryptosystem). So, it's taken as an objective function in our optimization algorithm.

The nonlinearity criteria are defined by

$$N_f = 2^{n-1} \left(1 - 2^{-n} \max_{w \in GF(2^n)} |S_{\langle f \rangle}(w)| \right). \quad (1)$$

Mathematically, the nonlinearity of a Boolean function $f(x)$ can be represented by the Walsh Spectrum.

The Walsh Spectrum of $f(x)$ is given by

$$S_{\langle f \rangle}(w) = \sum_{x \in GF(2^n)} (-1)^{f(x) \oplus x \cdot w} \quad w \in GF(2^n) \quad (2)$$

and

$$x \cdot w = x_1 \cdot w_1 \oplus \dots \oplus x_n \cdot w_n. \quad (3)$$

To generate a highly nonlinear S-box, we studied the steps of the algorithm proposed in [9], and we make improvements after revealing some weaknesses. The main weakness in the algorithm is that all the process depends on the initial S-box (Step 1) (Fig. 1) If the initial S-box does not show excellent properties, neither will the final one. To overcome this weakness, we propose to generate a dynamic initial S-box (Fig. 2). Each time we obtain weak results, we regenerate another one until we get an S-box with excellent cryptographic properties. Analyzing the mentioned algorithm, we remark that the initial S-box depends heavily on the initial value of x_0 . On the other hand, we enhance the construction of population size of the optimization algorithm to explore all the research domain of the solution. We start by dividing the range of chaotic map initial conditions; this segmentation is essential and allows the excellent exploration of the solution domain (Table 1).

Thereby, our contribution is summarized as follows:

1. Segmentation of the initial condition range
2. Generate a pseudo-random sequence of initial values using the first range n_0
3. Generate the first initial S-Box using the initial value taken from the chaotic generator.

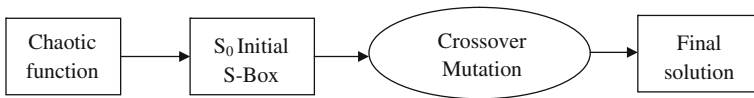


Fig. 1 Flowchart of the classical scheme

4. Mutation process.
5. Intelligent crossover of S-box elements: The selection of population is according to the last objective values.
6. If nonlinearity of the obtained S-Box is less than 107.5, regenerate another initial S-box by using another value of the initial condition.
7. Stop condition: Nonlinearity >107.5

The proposed algorithm is performed in three steps and presented briefly in the following flowchart (Fig. 3).

2.1 Description

Step 1 (Generating the initial dynamic S-box) The selection of an initial S-box is an essential step in our algorithm, and it’s the step that leads us to an excellent S-box with better performances. In our proposal, we choose to modify the initial S-box to maximize the exploration of the solution domain. In the flowing, we explain the necessary steps to determine an initial S-box.

1. Define S as a sequence, which is empty at the beginning.
2. Given the initial value x_0 , iterate Eq. (4) for 100 times to get rid of the transient effect.

$$x_{i+1} = \mu x_i(1-x_i) \tag{4}$$

3. Continue to iterate, one time, the Eq. (4), and denote the current state value as x' . Then an integer value X is obtained as below:

$$X = \text{floor}(256 \times x') \tag{5}$$

where floor(X) rounds the elements of X to the nearest integers towards minus infinity.

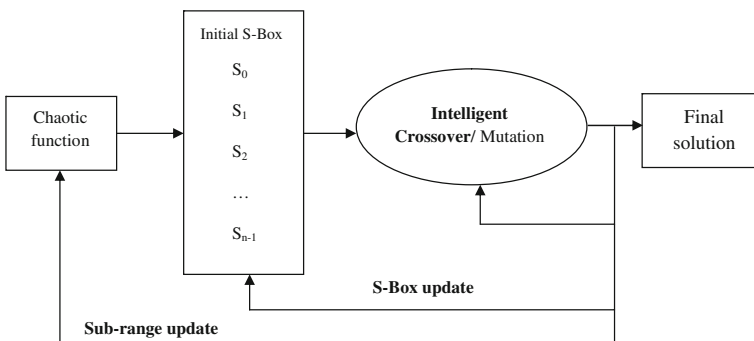


Fig. 2 Flowchart of the proposed scheme

Table 1 Mutation and crossover

Operation	Matrix arrangement	Equation
Crossover	Shuffling rows	$proW1_i = \text{mod}(\text{floor}(S1_i * 10^{14}), 4) + 2$ $proW2_i = \text{mod}(\text{floor}(S1_i * 10^{14}), 4) + 8$
Crossover	Shuffling columns	$pcol1_i = \text{mod}(\text{floor}(S2_i * 10^{14}), 4) + 2$ $pcol2_i = \text{mod}(\text{floor}(S2_i * 10^{14}), 4) + 8$
Mutation	Permutation	$pmut1_i = \text{mod}(\text{floor}(S3_i * 10^{14}), 4) + 2$ $pmut2_i = \text{mod}(\text{floor}(S3_i * 10^{14}), 4) + 8$

4. If X is not in sequence S , append it to S . Otherwise, go to item 3.
5. If the number of elements in S is not bigger than 256, go to item 3. Otherwise, output S .
6. Construct the initial (8×8) S-box from the sequence S using the algorithm 1. This S-box is used as the initial population.
7. After K iterations, if the final condition is not satisfied, take another value $\in n_i$ as initial condition and repeat previous steps to determine another initial S-box (Fig. 3) (Alg. 2, Alg. 3)

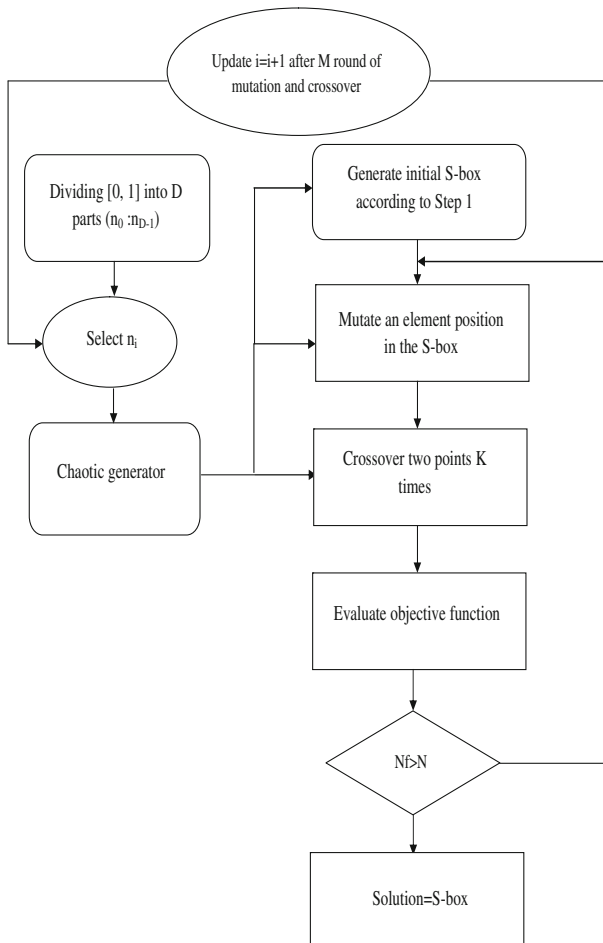


Fig. 3 Flowchart of the S-box generation

Algorithm 1 Optimization Process.

Algorithm 1 *Optimization Process***Input:** $S, S1, S2, S3$ **Output:** $SBox$ Divide $[0,1]$ into n sub-rangesChoose range n_0, x_0 **while** $i \leq 100$ **do**

Iterate logistic map

end whileGenerate $S1_i(1, 16), S2_i(1, 16), S3_i(1, 16)$

Execute Algorithm1

Execute Algorithm2

 $pmut1_i \leftarrow \text{mod}(\text{floor}(S3_i * 10^{14}), 4) + 2$ $pmut2_i \leftarrow \text{mod}(\text{floor}(S3_i * 10^{14}), 4) + 8$ $SBox \leftarrow SBoxp$ $prow1_i \leftarrow \text{mod}(\text{floor}(S1_i * 10^{14}), 4) + 2$ $prow2_i \leftarrow \text{mod}(\text{floor}(S1_i * 10^{14}), 4) + 8$ $SBox \leftarrow SBoxl$ $pcol1_i \leftarrow \text{mod}(\text{floor}(S2_i * 10^{14}), 4) + 2$ $pcol2_i \leftarrow \text{mod}(\text{floor}(S2_i * 10^{14}), 4) + 8$ $SBox \leftarrow SBoxc$ Calculate N_f **if** $N_f < 107.5$ **then** **if** $N_f \in [105.5, 107]$ **then** Recalculate $prow1_i, prow2_i, pcol1_i, pcol2_i$ Recalculate N_f **else**

Reject

end if**else**

Accept solution

 $S \leftarrow SBox$ **end if**Algorithm 2 *Gen_init_SBox*(x_0).**Algorithm 2** *Gen_init_SBox*(x_0)**Input:** x_0 **Output:** $SBox$ $i \leftarrow 1$ **while** $i \leq 100$ **do**

iterate_logistic_map

end while $nb \leftarrow 0$ **while** $nb < 256$ **do** $x' \leftarrow \text{iterate_logistic_map}$ $X \leftarrow \text{floor}(256 * x')$ **if** $X \notin S$ **then** $S(nb) \leftarrow X$ $nb \leftarrow nb + 1$ **end if****end while** $SBox \leftarrow S2Sbox(S)$

Step 2 (Generating the control parameters of mutation and crossover) The logistic map system defined in Eq. (4) is used to generate the control parameters of the genetic algorithm. This step is presented briefly.

Algorithm 3 $S2Sbox(S)$.

Algorithm 3 $S2Sbox(S)$

Input: S

Output: $SBox$

$i \leftarrow 1$

$k \leftarrow 1$

while $i \leq 16$ **do**

$j \leftarrow 1$

while $j \leq 16$ **do**

$SBox(i, j) \leftarrow S(k)$

$k \leftarrow k + 1$

$j \leftarrow j + 1$

end while

$i \leftarrow i + 1$

end while

1. Given the initial values x_0 , iterate Eq. (4) for 100 times to get rid of the transient effect.
2. Iterate it 16×3 times to generate three sequences, $S1_i = x_i$, $S2_i = y_i$ and $S3_i = z_i$, $i = 1, 2, \dots, 16$. The flowing table explain the exploitation process of the genetic algorithm in order to generate the S-box.

Step 3 (Intelligent crossover) The intelligent crossover part of the algorithm is described in Fig. 4. If the output of the algorithm is $\in [Nf_1, Nf_2]$, we change the initial condition slightly to find the best one, and else we repeat all the process. In our simulation we choose $Nf_1 = 105.5$ and $Nf_2 = 107$.

Finally, to select the best S-box solution to the problem, we adopt the following points:

1. Determining the nonlinearity of the first S-box calculated
2. Applying the crossover/mutation process (Fig. 4): Intelligent crossover
3. Calculating the new nonlinearity score
4. Evaluating the nonlinearty value
5. Taking a decision
6. Repeating previous steps with new initial S-box if condition is not satisfied
7. Evaluating the final solution

2.2 S-box evaluation

We take the simulation parameters when we execute our algorithm:

- $x_0 = 0.178888888888887$; $\mu = 3.577777777777777$;
- D is number of sub-regions = 12;
- Population number for each sub-region = 50;
- $Nf_1 = 105$; $Nf_2 = 107$;

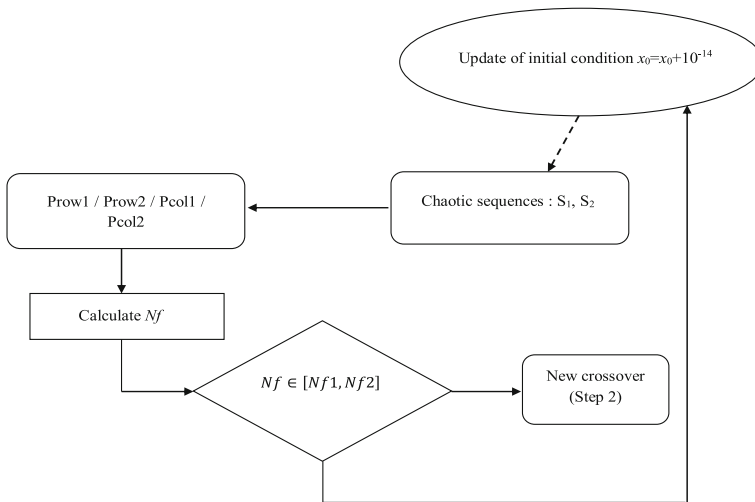


Fig. 4 Intelligent crossover process

The Tables 2 and 3 represent the proposed S-Box. we remark these points:

1. The bijective propriety is verified
2. The nonlinearity score of the optimized S-box ($Nonlinearity = 108$) is better than those proposed in previous work (Including our recent proposition in [9])
3. The mean value of the S-box dependence matrix (SAC), and we founded as average value 0.4988
4. The output bits independence criterion (BIC) means that all the avalanche variables must be pair-wise independent, in our case $BIC - SAC = 0.4969$ and $BIC - nonlinearity = 102.8571$.
5. The maximum input/output Xor is 12

3 The proposed cryptosystem

The process of encryption is presented in the following steps (Fig. 6):

Step 1. Inputting the original image $P(S,M)$, S and M represent the dimension of row and column respectively.

Step 2. Using the proposed SBox, we start the substitution phase, after that we divide the matrix into 4×4 blocks

Step 3. We iterate N times the logistic map and we construct a vector according to

$$V_{permute} = \text{mod}(\text{floor}(x_i * 10^3), 255). \tag{6}$$

Step 4. We permuted the blocks, this operation is made randomly according to the final vector presented in step 3. We repeated this operation N times. In our case we took $N = 10,000$. Algorithm 2 described the permutation process, we started by iterating logistic

map N times, after that we completed the matrix permutation (Matrixp) by the following instruction. We present in (Fig. 5) an example to explain the operation:

Step 5. In this step we used Xor operation between the permuted Matrix and chaotic sequence generated from logistic map. Finally, we obtain the encrypted image (first round).

Step 6. By updating the key of the algorithm according to the following equation, we repeat all steps described (Fig. 6).

$$\mu_{i+1} = 3.5 + \text{mod}(\mu_i + 10^{-14}, 0.4) \tag{7}$$

Algorithm 4 *Permutation Process*

Algorithm 4 *PermutationProcess*

Input: $i, j, k, S * M, N, Vpermute$

Output: *Matrixp*

```

i ← 1
j ← 1
k ← 1
while i ≤ N do
    iterate_logistic_map
end while
if S * M < N then
    Matrixp(i) ← Matrix(Vpermute(j))
    i ← i + 1
    j ← j + 1
else
    for i = 1; S * M < N; i = i + 1 do
        Matrixp(i) ← Matrixp(Vpermute(j))
    end for
    for i = S * M + 1; S * M < N; i = i + 1 do
        Matrixp(k) ← Matrix(Vpermute(i))
        k ← k + 1
    end for
end if

```

$$x0_{i+1} = 0.25 + \text{mod}(x0_i + 10^{-14}, 0.5) \tag{8}$$

The number of the rounds in our proposed algorithm is defined by

$$K = 3 + 0.5 \text{floor} \left(10 \frac{\sum \sum I m_{round_1}}{M * N} \right). \tag{9}$$

We notice a relation between the algorithm output in the first encryption round and the number of rounds executed. This operation enhances the security of the proposed encryption scheme by increasing the possibility of encryption rounds (the maximum value is 8). We have chosen eight encryption rounds as maximum value to keep an acceptable algorithm speed.

4 Simulation results

In our simulation, we used the optimized S-box detailed in section 2 and these simulations parameters:

Table 2 Optimized S-Box

31	125	185	143	145	66	101	110	189	113	92	246	174	181	79	6
27	0	46	52	21	81	213	32	76	154	99	77	161	123	255	186
14	209	1	157	38	222	88	228	89	191	36	73	45	90	210	179
60	165	26	156	86	176	223	56	82	193	203	51	11	114	190	238
17	194	216	235	28	140	247	177	47	168	155	128	242	115	207	230
132	169	19	172	78	42	218	147	5	164	192	167	135	7	2	126
243	33	127	195	239	111	93	61	199	72	137	53	48	160	163	22
10	67	201	233	87	133	139	229	54	104	226	237	102	225	40	196
130	105	200	219	141	251	144	131	205	58	217	138	4	91	212	236
112	142	29	64	106	62	188	24	136	183	97	30	221	44	118	25
248	158	20	43	84	253	149	75	162	59	245	70	120	234	187	71
241	240	208	232	37	98	129	55	65	153	244	68	204	8	121	16
34	197	198	63	94	83	41	170	109	134	166	117	74	224	85	119
206	148	214	100	15	108	96	35	103	50	95	175	182	254	171	180
13	220	80	57	122	150	107	23	3	146	250	215	49	152	69	211
159	39	116	12	252	178	173	184	124	249	18	151	202	227	231	9

1. For permutation vector: $x_0 = 0.2788888888888879$; $\mu = 3.677777777777779$;
2. For mask vector(xor): $x_0 = 0.378888888888887$; $\mu = 3.777777777777779$;

In the case of the IEEE floating-point standard, the double precession number is 2^{-52} . Therefore, the proposed encryption algorithm has a keyspace calculated as follow:

$$Keyspace = 2^{52.4} = 2^{208}. \tag{10}$$

4.1 Histogram analysis

The histogram of images is a statistical analysis that determines the distribution of pixels values. Histogram of the uniform distribution formed by the encrypted image indicates the unpredictability of the encryption algorithm. To validate this test, we take different images from USC-SIPI ‘Miscellaneous’ image data set with many textures. We remark that the histogram of encrypted images is different from the original images. Simulation results indicates that the proposed algorithm passes with success the histogram test. (Table 4)

Table 3 The performance comparison of chaotic S-boxes

S-box	Nonlinearity			SAC			BIC-SAC	BIC-N	I/O Xor
	Min	Max	Avg.	Min	Max	Avg.			
Proposed SBOX	106	110	108	0.4453	0.5313	0.4988	0.4969	102.8571	12
Ref. [9]	106	110	107.5	0.4375	0.57813	0.4971	0.5034	103.8571	10
Ref. [14]	–	–	106	0.4375	0.5938	0.5156	0.5048	–	–
Ref. [19]	98	108	103.2	0.3671	0.5975	0.5058	0.5031	104.2	12
Ref. [33]	99	106	103.3	0.4140	0.6015	0.4987	0.4995	103.3	10
Ref. [12]	100	106	103	0.4218	0.6093	0.5000	0.5024	103.1	14
Ref. [22]	96	106	103	0.3906	0.6250	0.5039	0.5010	100.3	12

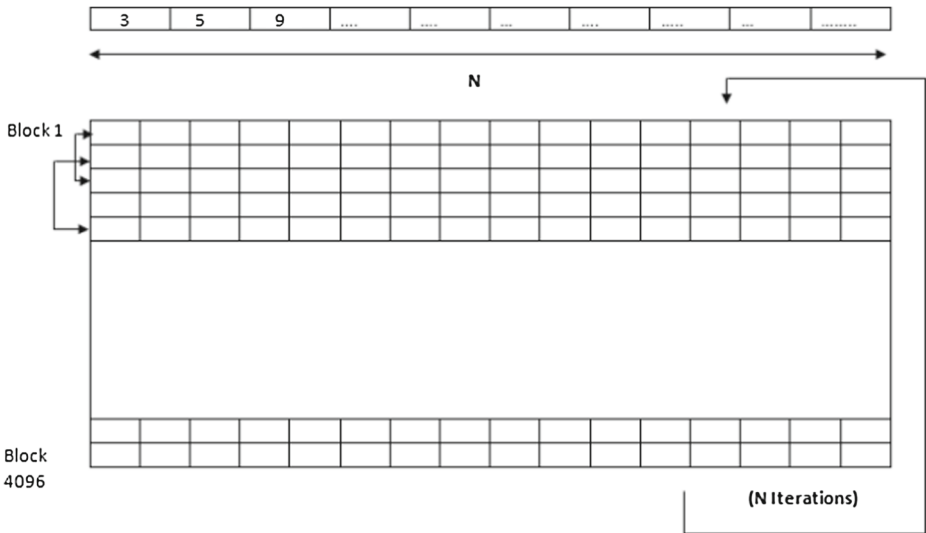


Fig. 5 Blocks scrambling

4.2 Correlation analysis of two adjacent pixels

Let, x and y the gray values of two adjacent pixels, and N is the number of pixels selected from the plain image. We take $E(x)$ and $E(y)$ as the mean values of x_i and y_i . The correlation coefficients of two adjacent pixels is given by

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{11}$$

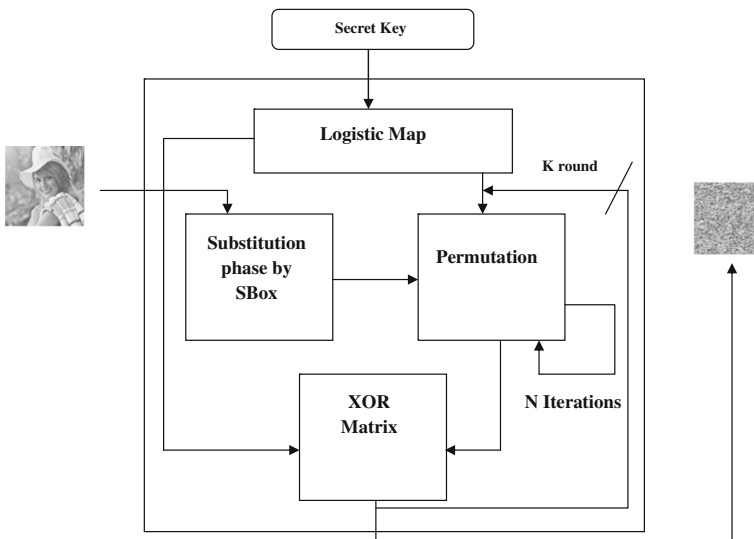
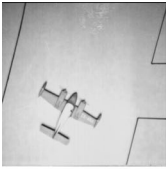
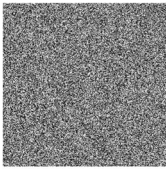
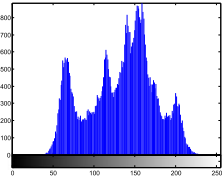
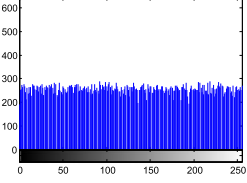
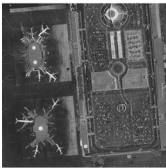
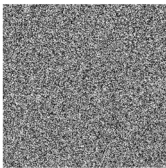
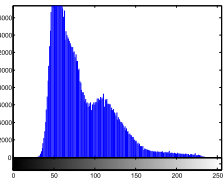
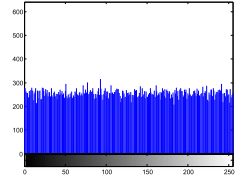

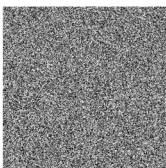
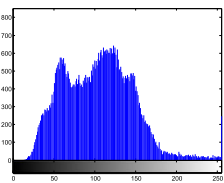
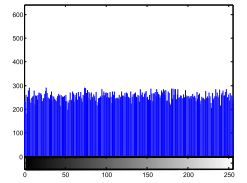
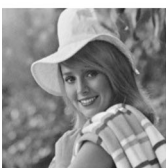
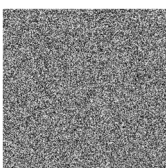
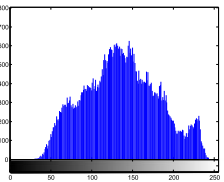
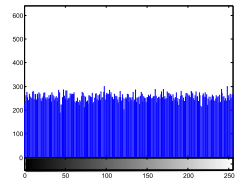

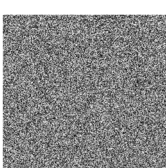
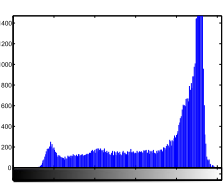
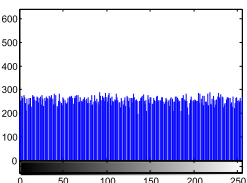


Fig. 6 The flowchart of the cryptosystem

Table 4 Histogram analysis

Original image	Encrypted image	Histogram	
		Original image	Encrypted image
			
			
			
			
			

where

$$cov(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(y)), \tag{12}$$

with

$$E(x) = \frac{1}{N} \sum_{i=1}^N x_i, \tag{13}$$

Table 5 Correlation coefficients

Image	Direction	Plain image	Encrypted image			
			Our scheme	[3]	[34]	[35]
Airplane	Horizontal	0.9559	.013205	-0.0002	0.0011	-0.0164
	Vertical	0.9310	.019524	-0.0090	-0.0035	-0.0181
	Diagonal	0.8706	-0.001245	-0.0066	-0.0029	0.0004
Barbara	Horizontal	0.8292	-0.005545	-0.0033	-0.0052	-0.0212
	Vertical	0.9580	-0.003215	-0.0269	-0.0067	-0.0161
	Diagonal	0.7860	.008845	-0.0121	0.0068	-0.0110
Boat512	Horizontal	0.9369	.008845	-0.0100	-0.0054	-0.0189
	Vertical	0.9735	.007549	-0.0124	-0.0009	0.0003
	Diagonal	0.9223	.010245	-0.0185	0.0026	-0.0204
Camera man	Horizontal	0.9242	-0.008758	-0.0095	-0.0211	0.0063
	Vertical	0.9571	-0.017256	-0.0170	-0.0103	-0.0142
	Diagonal	0.9006	-0.002545	-0.0119	0.0054	0.0168
Chemical plant	Horizontal	0.9445	-0.001254	-0.0134	0.0073	-0.0069
	Vertical	0.9042	.009858	-0.0005	-0.0073	-0.0100
	Diagonal	0.8566	.009878	-0.0033	-0.0115	-0.0078
Clock	Horizontal	0.9596	.01020	0.0024	-0.0140	-0.0248
	Vertical	0.9716	-0.00257	-0.0246	-0.0139	-0.0172
	Diagonal	0.9379	-0.003698	-0.0081	-0.0175	-0.0025
Couple512	Horizontal	0.9358	-0.002458	-0.0251	-0.0178	-0.0122
	Vertical	0.9214	.007858	-0.0213	-0.0025	0.0262
	Diagonal	0.8392	-0.002789	-0.0078	0.0001	-0.0257
Elaine	Horizontal	0.9771	.007856	-0.0232	-0.0065	-0.0191
	Vertical	0.9831	-0.008785	-0.0420	-0.0096	-0.0130
	Diagonal	0.9563	-0.036502	-0.0030	-0.0148	-0.0096
Lena	Horizontal	0.9208	.001254	-0.0048	-0.0086	-0.0066
	Vertical	0.9487	-0.001475	-0.0112	-0.0102	-0.0089
	Diagonal	0.9138	.009785	-0.0045	-0.0125	0.0424
Average	Horizontal	-	.002894	-0.0113	-0.0086	-0.0091
	Vertical	-	-0.003155	-0.01473	-0.0090	-0.0064
	Diagonal	-	-0.002298	0.0099	-0.0067	-0.0024

and

$$D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2. \tag{14}$$

In Table 5, the correlation values of the encrypted image are close to 0, and the uniform distribution of adjacent pixels indicate an excellent evaluation of the algorithm.

The resistance to statistical attacks is confirmed in our proposition.

We selected 3000 adjacent pixels (vertical, horizontal, and diagonal directions) from the original image and encrypted image.

Simulation results presented in Table 6 indicate the correlation of two adjacent pixels for original images taken from USC-SIPI ‘Miscellaneous’ data-set and encrypted images (Airplane, Airport and chemical plant).

4.3 Information entropy analysis

Let ‘m’ is the source of information and ‘p’ is the probability of symbol ‘m’.

Table 6 Correlation between two adjacent pixels

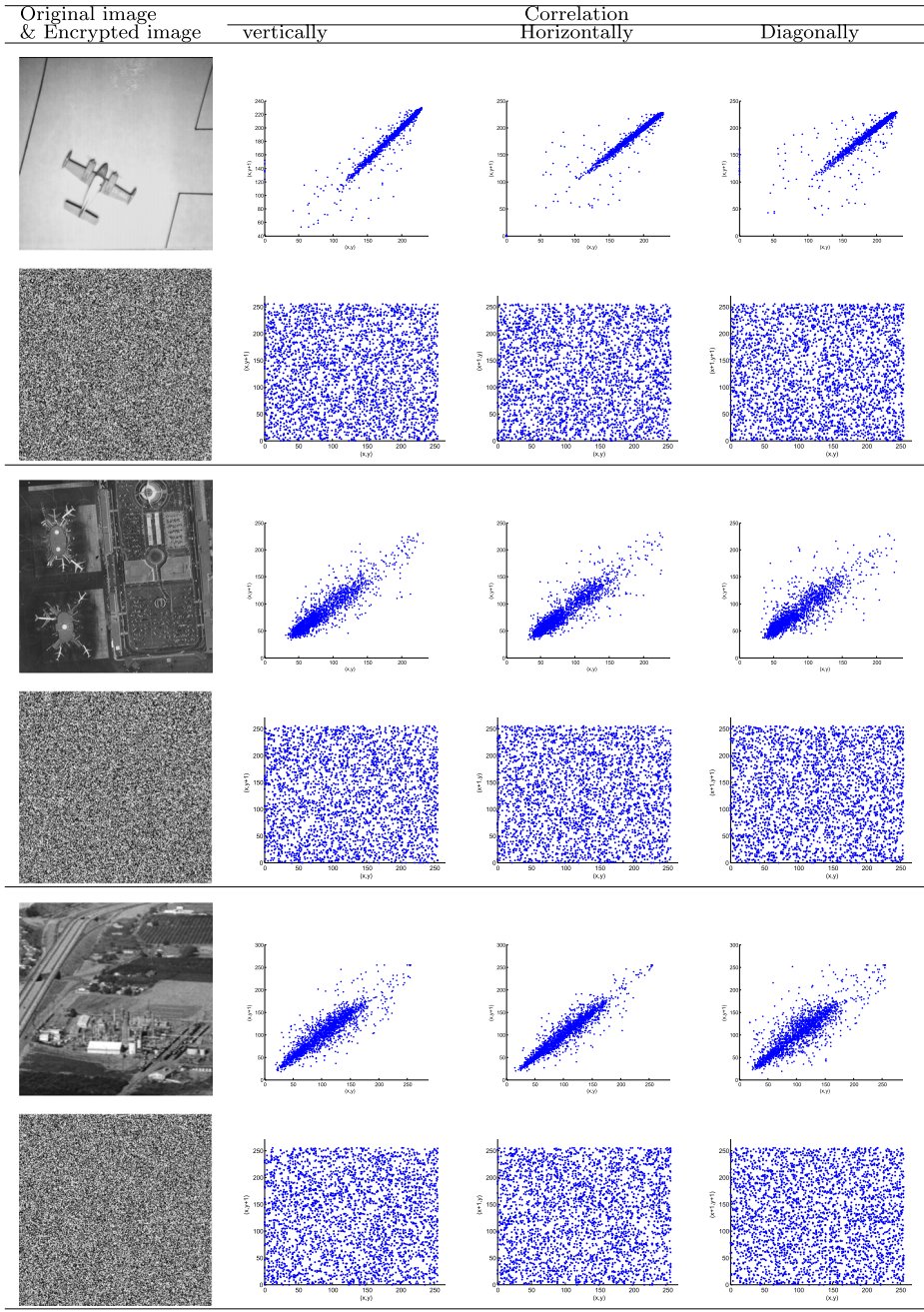


Table 7 Entropy value for multiple encrypted images

Image	Our scheme	[3]	[34]	[35]
Airplane	7.996213	7.9974	7.9965	7.9926
Barbara	7.995456	7.9978	7.9964	7.9937
Boat512	7.998585	7.9980	7.9979	7.9960
Camera man	7.998845	7.9985	7.9966	7.9955
Chemical plant	7.995569	7.9964	7.9986	7.9947
Clock	7.998654	7.9992	7.9974	7.9984
Couple512	7.999845	7.9976	7.9987	7.9951
Elaine	7.997985	7.9985	7.9972	7.9939
Lena	7.997845	7.9963	7.9991	7.9951
Average	7.997666	7.9978	7.9977	7.9953

The entropy is a parameter that measure the degree of randomness, calculated by

$$H(m) = \sum_{i=0}^{2^n-1} p(m_i) \log_2 \frac{1}{p(m_i)}. \tag{15}$$

Table 7 calculate the information entropy of the ciphered different images. Compared to other previous algorithms, we can confirm that the proposed method give us an acceptable average value close to 8.

4.4 Differential attack

To test and examine the performance of an encryption algorithm we calculate NPCR (Number of pixel change rate) and UACI (Unified average changing intensity, the following formulas indicate how to determine these parameters. The ideal image encryption algorithm is when UACI is in the range [33.38 33.70] and NPCR >99.50. Simulation results presented in Table 8 indicate the robustness of our proposition.

Table 8 Value of UACI and NPCR

Image		Our scheme (Avg. value)	[3]	[34]	[35]
Airplane	NPCR	99.605256	99.6083	99.6077	99.5565
	UACI	33.485625	33.5359	33.4143	33.4063
Barbara	NPCR	99.601223	99.6092	99.6162	99.5227
	UACI	33.460223	33.7431	33.5776	33.3890
Boat512	NPCR	99.600223	99.6102	99.6281	99.5609
	UACI	33.390212	33.5367	33.6145	33.4176
Cameraman	NPCR	99.611256	99.6205	99.6292	99.5749
	UACI	33.402356	33.7786	33.7050	33.3691
Chemical Plant	NPCR	99.610225	99.6121	99.6131	99.5325
	UACI	33.690230	33.6068	33.8543	33.3872
Clock	NPCR	99.590223	99.6102	99.6218	99.5910
	UACI	33.420223	33.5820	33.4836	33.4147
Couple512	NPCR	99.590223	99.6399	99.6292	99.5606
	UACI	33.410223	33.8085	33.6446	33.3723
Elaine	NPCR	99.620223	99.6143	99.6107	99.5431
	UACI	33.425256	33.5160	33.6163	33.3853
Lena	NPCR	99.609856	99.6228	99.6146	99.5511
	UACI	33.402568	33.7041	33.5561	33.3461

Table 9 Randomness test for the cipher-images

Statistic tests	P value for cipher-images		
	Chemical Plant	Airplane	Elaine
Frequency (Monobits) test	0.585258	0.385456	0.122557
Frequency test within a block	0.254778	0.817525	0.912545
Runs test	0.125588	0.352556	0.095458
Test for the longest run of ones in a block	0.477889	0.688952	0.958565
Binary matrix rank test	0.365489	0.788945	0.344569
Discrete Fourier transform (Spectral) test	0.092565	0.847552	0.458978
Non-overlapping template matching test	0.902565	0.058956	0.675889
Overlapping template matching test	0.095856	0.395625	0.145689
Maurer’s universal statistical” test	0.895623	0.937456	0.456523
Linear complexity test	0.658956	0.147836	0.063598
Serial test (1)	0.915263	0.452556	0.012258
Serial test (2)	0.945256	0.365258	0.253658
Approximate entropy test	0.123678	0.092563	0.782569
Cumulative sums	0.845256	0.582563	0.236589
Random excursions test	Success	Success	Success
Random excursions variant test	Success	Success	Success
Results	Pass	Pass	Pass

$$NPCR = \frac{\sum_{i,j} D(i,j)}{L} \times 100\% \tag{16}$$

and

$$UACI = \frac{1}{L} \left[\sum_{i,j} \frac{|C(i,j) - C'(i,j)|}{255} \right] \times 100\% \tag{17}$$

where L is the total number of pixels in the image. C and C' are respectively the ciphered images before and after one pixel of the plain image is changed. D(i, j) can be defined by

$$D(i,j) = \begin{cases} 1, & \text{if } C(i,j) \neq C'(i,j) \\ 0, & \text{if } C(i,j) = C'(i,j) \end{cases} \tag{18}$$

Table 10 Time speed

scheme	Encryption time (s)	Microprocessor/RAM/O.S
Our scheme	1.120248	2.53GHz, i3/3GB/Windows 7
[36]	1.32, <i>One round</i>	2.5GHz, AMD/4GB/-
[38]	0.059280	-
[1]	10.42(512x512)	3.70GHz E5-1620/64GB/Linux
[2]	1.7(256x256)	3.8GHz, i7/16GB/Linux
[29]	6.446117(1000iterations)	3.4GHz, i5/4GB/Windows 10

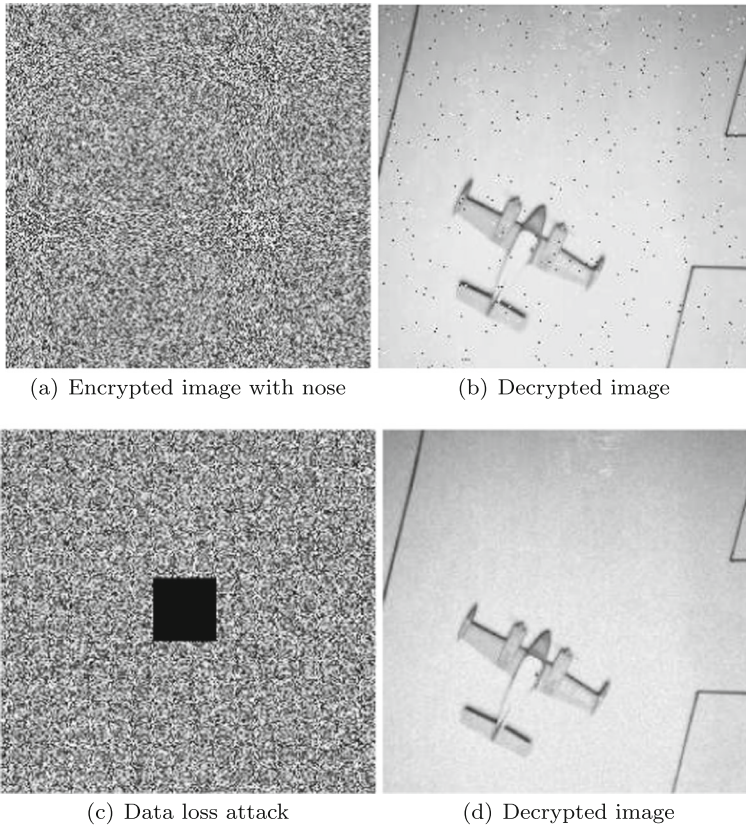


Fig. 7 Nose and data loss attacks

4.5 Randomness Nist 800–22 tests

Nist 800–22 tests, presented in [23] evaluate the randomness property for encryption algorithms. For test, we select different encrypted images, after that we calculate its P values (The formula of each P value for different tests is indicated in [23]), which should be in the interval $[0, 1]$. To validate the randomness test we choose three different images (Airport, Airplane and Chemical plant). (Table 9)

4.6 The speed performance

The speed of encryption algorithm is an important parameter to evaluate the possibility of real-time implementation. Table 10 we present a comparison between our proposition and previous work. We can confirm that our proposition has excellent performance.

Our experiments were realized using Lena image with a 256×256 size, microprocessor 2.53 GHz, and ram 3GB. We took the average value of 20 encryption test.

4.7 Noise and data loss attacks

Firstly, in the encrypted image, we added 1% salt and pepper noise; after that, we decrypt the resulted image. Secondly, we replaced a part of the encrypted image (40×40 , in our case) by

black pixels, and we study the decrypted image. The simulation results presented in Fig. 7 indicate that we have received the image even we apply noise or information loss attacks.

5 Conclusion

In this paper, we present a new proposition of cryptosystem based on S-box in the substitution process and a permutation algorithm based on a chaotic map. An optimization algorithm contains intelligent crossover, generates optimal S-box. The height nonlinearity score founded, demonstrate the unpredictability of the encryption algorithm. The presented S-box shows excellent randomness characteristics compared with others presented in previous papers. The permutation process is based on the chaotic sequence results of the logistic map, many statistical and security tests are carried out to show the performance of our proposition.

Our algorithm can be improved by studying the impact of changing the chaotic function in robustness and security. Also, we can enhance our scheme by adding an intelligent selective algorithm to become suitable for video sequences and multimedia information.

Acknowledgements The authors would like to thank the anonymous referees for their valuable comments and suggestions that enhanced the content and the form of this paper.

Compliance with ethical standards

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

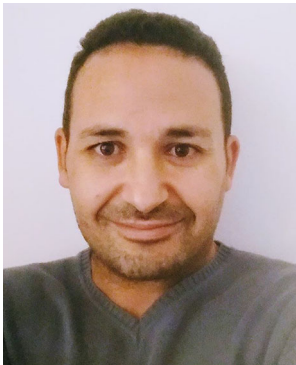
References

1. Artiles JAP, Chaves DPB, Pimentel C (2019) Image encryption using block cipher and chaotic sequences. *Signal Process Image Commun* 79:24–31
2. Asgari-Chenaghlu M, Balafar M-A, Feizi-Derakhshi M-R (2019) A novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation. *Signal Process* 157:1–13
3. Belazi A, Abd El-Latif AA, Belghith S (2016) A novel image encryption scheme based on substitution-permutation network and chaos. *Signal Process* 128:155–170
4. Chidambaram N, Raj P, Thenmozhi K, Rajagopalan S, Amirtharajan R (2019) A cloud compatible dna coded security solution for multimedia file sharing & storage. *Multimedia Tools and Applications*, pages 1–27
5. Farah A, Belazi A (2018) A novel chaotic jaya algorithm for unconstrained numerical optimization. *Nonlinear Dynamics* 93(3):1451–1480
6. Farah MAB, Guesmi R, Kachouri A, Samet M (2020) A novel chaos based optical image encryption using fractional fourier transform and dna sequence operation. *Opt Laser Technol* 121:105777
7. Fridrich J (1998) Symmetric ciphers based on two-dimensional chaotic maps. *International Journal of Bifurcation and chaos* 8(06):1259–1284
8. Gueron S, Feghali WK, Gopal V, Makaram R, Dixon MG, Chennupaty S, and Kounavis ME (2019) Flexible architecture and instruction for advanced encryption standard (aes), January 1 2019. US Patent 10,171,232
9. Guesmi R, Farah MAB, Kachouri A, and Samet M (2014) A novel design of chaos based s-boxes using genetic algorithm techniques. In *2014 IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA)*, pages 678–684. IEEE
10. Guesmi R, Farah MAB, Kachouri A, Samet M (2016) A novel chaos-based image encryption using dna sequence operation and secure hash algorithm sha-2. *Nonlinear Dynamics* 83(3):1123–1136
11. Guesmi R, Farah MAB, Kachouri A, Samet M (2016) Hash key-based image encryption using crossover operator and chaos. *Multimed Tools Appl* 75(8):4753–4769

12. Guo C, Chen Y, Liao X (2007) An extended method for obtaining s-boxes based on three-dimensional chaotic baker maps. *Chaos, Solitons Fractals* 31(3):571–579
13. Hamza R, Yan Z, Muhammad K, Bellavista P, and Titouna F (2019) A privacy-preserving cryptosystem for iot e-healthcare. *Information Sciences*
14. Hayat U, Azam NA (2019) A novel image encryption scheme based on an elliptic curve. *Signal Process* 155:391–402
15. Hussain I, Shah T, Gondal MA (2013) Application of s-box and chaotic map for image encryption. *Math Comput Model* 57(9–10):2576–2579
16. Hussien AG, Hassanien AE, Houssein EH, Bhattacharyya S, and Amin M (2019) S-shaped binary whale optimization algorithm for feature selection. In *Recent trends in signal and image processing*, pages 79–87. Springer
17. Jain A, Rajpal N (2016) A robust image encryption algorithm resistant to attacks using dna and chaotic logistic maps. *Multimed Tools Appl* 75(10):5455–5472
18. Jakimoski G, Kocarev L (2001) Chaos and cryptography: block encryption ciphers based on chaotic maps. *Ieee transactions on circuits and systems i: fundamental theory and applications* 48(2):163–169
19. Jakimoski G, Kocarev L (2002) Block encryption ciphers based on chaotic maps. *Ieee Transaction on Circuits System-I* 48:163–169
20. Kanmani M, Narasimhan V (2018) Swarm intelligent based contrast enhancement algorithm with improved visual perception for color images. *Multimed Tools Appl* 77(10):12701–12724
21. Kanmani M, Narasimhan V (2019) An optimal weighted averaging fusion strategy for remotely sensed images. *Multidimensional Systems and Signal Processing*, pages 1–25
22. Khan M, Shah T, Mahmood H, Gondal MA, Hussain I (2012) A novel technique for the construction of strong s-boxes based on chaotic lorenz systems. *Nonlinear Dyn* 70(3):2303–2311
23. Lawrence E Bassham III, Andrew L Rukhin JS, Smid ME, Barker EB, Leigh SD, Levenson M, Vangel M, Banks DL et al. (2010) Sp 800–22 rev. 1a. a statistical test suite for random and pseudorandom number generators for cryptographic applications
24. Liang C, Zhang Q, Ma J, Li K (2019) Research on neural network chaotic encryption algorithm in wireless network security communication. *EURASIP J Wirel Commun Netw* 2019(1):151
25. Liu H, Kadir A, Niu Y (2014) Chaos-based color image block encryption scheme using s-box. *AEU-international Journal of Electronics and Communications* 68(7):676–686
26. Liu H, Kadir A, Gong P (2015) A fast color image encryption scheme using one-time s-boxes based on complex chaotic system and random noise. *Opt Commun* 338:340–347
27. Madheswari K, Venkateswaran N (2017) Swarm intelligence based optimisation in thermal image fusion using dual tree discrete wavelet transform. *Quantitative Infrared Thermography Journal* 14(1):24–43
28. Manoj RJ, Praveena MDA, Vijayakumar K (2019) An aco-ann based feature selection algorithm for big data. *Clust Comput* 22(2):3953–3960
29. Nesa N, Ghosh T, Banerjee I (2019) Design of a chaos-based encryption scheme for sensor data using a novel logarithmic chaotic map. *Journal of Information Security and Applications* 47:320–328
30. Özkaynak F, Özer AB (2010) A method for designing strong s-boxes based on chaotic lorenz system. *Phys Lett A* 374(36):3733–3738
31. Parvees MYM, Samath JA, Bose BP (2018) Providing confidentiality for medical image—an enhanced chaotic encryption approach. In: *Advances in big data and cloud computing*. Springer, Singapore, p 309–317
32. Singh G (2013) A study of encryption algorithms (rsa, des, 3des and aes) for information security. *International Journal of Computer Applications*, 67(19)
33. Tang G, Liao X, Chen Y (2005) A novel method for designing s-boxes based on chaotic maps. *Chaos, Solitons Fractals* 23(2):413–419
34. Wang X, Lin T, Xue Q (2012) A novel colour image encryption algorithm based on chaos. *Signal Process* 92(4):1101–1108
35. Wang X, Liu L, Zhang Y (2015) A novel chaotic block image encryption algorithm based on dynamic random growth technique. *Opt Lasers Eng* 66:10–18
36. Xu L, Li Z, Li J, Hua W (2016) A novel bit-level image encryption algorithm based on chaotic maps. *Opt Lasers Eng* 78:17–25
37. Yang J, He S, Lin Y, Lv Z (2017) Multimedia cloud transmission and storage system based on internet of things. *Multimed Tools Appl* 76(17):17735–17750
38. Ye G, Pan C, Huang X, Mei Q (2018) An efficient pixel-level chaotic image encryption algorithm. *Nonlinear Dynamics* 94(1):745–756
39. Zhang Y, Xiao D (2014) An image encryption scheme based on rotation matrix bit-level permutation and block diffusion. *Commun Nonlinear Sci Numer Simul* 19(1):74–82

40. Zhang W, Wong K-w, Yu H, Zhu Z-l (2013) An image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion. *Commun Nonlinear Sci Numer Simul* 18(8):2066–2080

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Mohamed Amine Farah received the Eng. degree in Electrical Engineering (2003), the M.Sc. degree in electronics and communication (2004) and Ph.D. degree in Electrical Engineering from National Engineering School of Sfax, Tunisia in 2011. He joined the electrical engineering department of National Engineering School of Gabes (Tunisia) in 2009 as an Assistant Professor. His areas of research are: Signal Processing, Cryptography, Security, Chaos theory... Now, he is member of LETI Laboratory (Sfax, Tunisia) and preparing his habilitation degree in Signal Processing.



Ramzi Guesmi has received his MS degree in “Automatic and Signal Processing” in 2005 from the National School of Engineering Tunis ENIT, Tunisia and Ph.D. degree in Electrical Engineering from National Engineering School of Sfax, Tunisia in 2016. He joined the computer sciences department of ISSAT-kasserine (Tunisia) in 2017 as an Assistant Professor. His areas of research are: Cryptography, Lightweight Cryptography, Security, Chaos theory and IOT. Now, he is member of LETI Laboratory (Sfax, Tunisia) and preparing his habilitation degree.



Abdennaceur Kachouri was born in Sfax, Tunisia, in 1954. He received the engineering diploma from National school of Engineering of Sfax in 1981, a Master degree in Measurement and Instrumentation from National school of Bordeaux (ENSERB) of France in 1981, a Doctorate in Measurement and Instrumentation from ENSERB, in 1983. He “works” on several cooperation with communication research groups in Tunisia and France. Currently, he is Permanent Professor at ENIS School of Engineering and member in the “LETT” Laboratory ENIS Sfax.



Mounir Samet was born in Sfax, Tunisia in 1955. He obtained an Engineering Diploma from National school of Engineering of Sfax in 1981, a Master degree in Measurement and Instrumentation from National school of Bordeaux (ENSERB) of France in 1981, a Doctorate in Measurement and Instrumentation from ENSERB, in 1985 and the Habilitation Degree (Post Doctorate degree) in 1998. He “works” on several cooperation with medical research groups in Tunisia and France. Currently, he is Permanent Professor at ENIS School of Engineering and member in the “LETT” Laboratory ENIS Sfax.