Received: September 17, 2021. Revised: October 22, 2021.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Designing a Novel Efficient Substitution-Box by Using a Flower Pollination Algorithm and Chaos System

Sameeh Abdulghafour Jassim[1]* **Alaa K. Farhan[1]**

[1]*Department of Computer Sciences, University of Technology, Baghdad, Iraq*
* Corresponding author's Email: cs.19.22@grad.uotechnology.edu.iq, prog85sameeh@gmail.com

Abstract: Substitution box (S-box) is a crucial component of symmetric cryptosystems and is mostly used in modern cryptographic ciphers to provide strong data sanctuary. The cryptographic strength of an S-box used in a cipher is directly proportional to the data security provided by the cipher. Thus, this study used a novel approach by combining the hybrid chaos (integrated three functions of chaos maps) to generate pseudo-random number generators and metaheuristic techniques based on the flower pollination algorithm (FPA) to develop an acceptable configuration (8 × 8) S-box. Chaotic maps enhance the initial population of FPA to be an appropriate starting point for the FPA. The objective function of the FPA was the nonlinearity score of the S-box. The FPA is used to balance exploration and exploitation operations. The suggested S-box was compared with several references and passed all the security requirements of S-box, including bijection, strict avalanche criteria (SAC), output bits independence criteria (BIC), nonlinearity, histogram, entropy, correlation coefficients, unified average changing intensity (UACI) and number of pixel change rates (NPCR), in addition to the algebraic cryptanalysis test. The proposed approach had an asymptotic computational complexity of O(n2). The nonlinearity property was 107.25. It is considered an effective and acceptable result. Whereas SAC and BIC were 0.498 and 104.7857, respectively. Simultaneously, all results of UACI and NPCR were close to the predicted normal levels 33.2255 and 99.5693, respectively. While the encryption histogram is more uniform than the simple histogram. Finally, the entropy and correlation coefficients scores result of various images were close to eight and zero, respectively. Therefore, the proposed S-box is proven to be strong and resistant to cryptography attacks.

Keywords: Flower pollination algorithm, S-box, Chaotic maps, Nonlinearity, Security.

1. Introduction

The encryption process is one of the most important techniques used to protect digital data. It is categorised into asymmetric and symmetric ciphers. Stream and block ciphers are the two types of symmetric-key cryptosystems. A stream cipher encrypts data bit by bit or byte by byte, whereas a block cipher uses the secret key to encrypt an entire block of plaintext at a time [1, 2]. Shannon's theory of diffusion and confusion is the basis of build block ciphers, which are executed in mathematical procedures (substitution-permutation networks) [3]. A substitution box (S-box) takes a block of n bits as input and transforms them into a new nonlinear block of m bits. Its mapping should be one-to-one and written as GF(2^n) → GF(2^m) in Galois Field theory. A permutation is a linear transformation that shuffles input bits. The results of the first round of the S-box is permutated before passing it on to the next round. The integration of the two methods produces cipher robust and secret [4].

The flower pollination algorithm (FPA) is simulated flower pollination behaviour. It is a population-based metaheuristic optimisation method. Wind and ballistics are most likely to be responsible for dispersion [5]. FPA is inspired by the self-pollination and cross-pollination of flowering plants. Therefore, it can efficiently integrate local and global searches. Furthermore, it employs Levy flights instead of using standard Gaussian random walks. FPA picks a population of pollens/flowers, and global pollination is upgraded based on the distance between its current location and the global best solution with a step size determined by the Levy distribution. Taking any two pollens from the population is used for local pollination. In each generation, the best pollens are preserved in the population [5, 6]. The applications of the FPA are engineering optimisation problems, manufacturing scheduling and train neural networks [7].

In this study, FPA is a novel, effective optimisation approach used in the S-box design method based on chaotic maps. FPA is controlled convergence speed by using a step size determined by the Levy distribution to avoid premature convergence issues. Simultaneously, it balances between exploration and exploitation operations. To find a better solution, FPA manipulates the initial population of S-boxes. Chaotic maps enhance the initial solutions (S-box) population. They produce nonlinear S-boxes, which are an appropriate starting point for the FPA to improve its nonlinearity. The following three chaotic systems are utilised to build the new module sinusoidal map, quadratic map and piecewise map. The three maps are sensitive to control parameters and initial conditions when producing pseudo-random number generators (PRNG). The results show that FPA could be utilised to generate cryptographically strong S-boxes. Our contributions were as follows:

a. A novel method based on the FPA approach is used to generate a strong S-box.
b. The initial population of FPA is enhanced using chaos maps to generate a good starting point for searching rather than starting from scratch, increasing the speed of obtaining the ideal solution and uses less memory.
c. Integrated three functions of chaos maps use hybrid chaos to generate PRNG.
d. The generated S-box is evaluated with security evaluation criteria, including bijection, strict avalanche criteria, output bits independence criteria, nonlinearity, histogram, entropy, correlation coefficients, unified average changing intensity (UACI) and the number of pixel change rates (NPCR), in addition to the algebraic cryptanalysis test.

The rest of this study is structured as follows. Section 2 explains the related works. Section 3 describes the S-box problem. Section 4 discusses the proposed algorithms. Section 5 presents the evaluation of the proposed algorithms. Section 6 explains the most important results and provides the conclusion.

2. Related works

The S-box could be built in various ways, including random search, algebraic techniques and metaheuristic methods. The random search approach typically produces S-boxes with poor cryptographic properties. The greatest cryptographic characteristics are found in S-boxes created using algebraic methods. However, obtaining an extensive collection of strong S-boxes utilising this method is impossible. On the other hand, metaheuristic algorithms are efficient in hardware and software implementation [8].

Since the 1990s, many researchers have reported on the efficiency of using metaheuristic techniques for S-box design. Some of these strategies included swarm and evolutionary algorithms: tree-seed algorithm [9], cuckoo search algorithm [10, 11], globalised firefly algorithm [8], simulated annealing [12], ant colony optimisation [13] and artificial bee colony [14]. These algorithms generate multi S-boxes based on RNA computing [15]. These methods can generate S-boxes with desirable properties. However, numerous issues in the design and analysis of S-boxes are unsolved [14]. A single metaheuristic-based S-box cannot be described as superior to its peers. As a result, researching novel metaheuristic-based S-box design strategies is a worthwhile undertaking [16].

In this study, we used fourteen studies to compare with the proposed technique. In [17-21] utilised continuous or discrete chaotic methods to produce the existing S-boxes. However, this technique does not produce excellent nonlinearity or other performance metrics. Whereas [22, 23] utilised algebraic techniques, and [24] utilised chaos and algebraic approach to construct S-boxes. Both approaches suffering from algebraic and statistical attacks. Moreover, [4, 25] utilised a hyperchaotic system (five-dimensional). So, it’s not efficient in hardware and software implementation. Finally, [8, 13, 26, 27] utilised metaheuristic algorithms with the chaos system. These methods can generate S-boxes with desirable properties. But they are not suitable for all applications. Simultaneously, there are numerous issues regarding the analysis and design of S-boxes that still exist. Several studies have proven that nature-inspired metaheuristics were an effective solution for machine learning and complex engineering case studies. Furthermore, they could be balancing between the local search and global search and avoid trapping in local optima [28-30]. Consequently, the FPA is proposed to solve these issues.

3. Problem description

This section will describe the S-box design problem. To prevent suspicion in DES S-box, the National Security Agency has published some guidelines for evaluating the cryptographic characteristics of S-boxes [31]. The criteria that are most accepted for assessing performance and considered as fundamental qualities for developing cryptographically resistant S-boxes are as follows [13, 25]:

i) Bijection: this property is a mapping in which each input bit corresponds to a single, unique output bit. In this study, (8×8) S-box is required to have various output values in a period [0, 255] [4, 23].

ii) Nonlinearity: The Walsh spectrum can be used to indicate nonlinearity for a Boolean function g(x). Walsh Transformation is defined as in Eq. (2) [10]:

$$S_f(w) = \sum_{x \in GF(2^n)} (-1)^{g(x) \oplus x.w} \quad (1)$$

$$N_f = 2^{n-1} - \frac{1}{2} max_{w \in GF(n^n)} |S_f(w)| \quad (2)$$

where $S_f(w)$ is the Walsh spectrum of *g(x)*, *x.w* is the dot product of *x* and *w*, and $x.w = x_1 \oplus w_1 + \ldots + x_n \oplus w_n$.

iii) Strict avalanche criteria (SAC): In 1985, AF Webster and SE Tavares proposed the SAC. The SAC essentially states that, if the input is only modified by one bit, the entire output bits should change by 50%. Consequently, an S-box with an SAC value close to 0.5 is considered to be robust[4].

iv) Output bits independence criteria (BIC): Established by AF Webster and SE Tavares, BIC is one of the most significant characteristics of an S-box. This criterion indicates that the independence degree between any two ciphertext bits should be high when one bit in the plaintext is complemented [10].

v) Differential approximation probability (DP): The S-box achieves a nonlinearity of an encryption process. It may ideally maintain differential uniformity. However, to guarantee uniform mapping, the differential uniformity is applied in this way, and the differential input should map uniquely to the differential output map. The DP is used to quantify differential uniformity. DP is expressed mathematically in Eq. (3) [4, 8] as follows:

$$DP = \frac{[\#\{g \in M \mid s(g) \oplus S(g \oplus \Delta_g) = \Delta_h]}{256} \quad (3)$$

Where Δ_g and Δ_h are the input and output differential, respectively, and *M*={0,1, …, 255}.

4. The proposed algorithms

This section explains FPA and its proposed approach. Chaotic maps utilised to obtain the initial randomness values for S-box in FPA are also defined.

4.1 Chaotic maps

Chaos theory is a field of physics and mathematics that studies the behaviour of the dynamical system. It is also extremely sensitive to their control parameters and initial value. Chaotic systems include constant feedback loops, self-organisation, fractals, repetition, underlying patterns and dependency on programming according to initial values [33]. In this study, three chaotic maps (hybrid) are used to provide a starting point to the S-box (sinusoidal, quadratic and piecewise maps), as shown in Fig. 1.

i) Sinusoidal map: The following Eq. (4) formally defines the sinusoidal iterator [9]:

$$y_{j+1} = k y_j^2 \, sin(\pi y_j), \\ k = 2.3 \; and \; y_0 = 0.8 \quad (4)$$

ii) Quadratic map: This well-known chaotic map has complex dynamic behaviours. This chaos map has been frequently utilised in cryptography. It may be written as a standard quadratic equation, as shown in Eq. (5) [34, 35]:

$$y_{j+1} = k - (y_j)^2 \quad (5)$$

k = is a controlling parameter [0, 2]=0.5, j is the number of iterations, and y_j [0, 1] =0.15 is the chaotic sequence created.

iii) Piecewise map: is a chaotic map expressed as Eq. (6) [9, 11]:

$$y_{j+1} = \begin{cases} \frac{y_j}{c} & 0 \le y_j < c \\ \frac{y_j - c}{0.5} & c \le y_j < 0.5 \\ \frac{1 - y_j - c}{0.5} & 0.5 \le y_j < 1 - c \\ \frac{1 - y_j}{c} & 1 - c \le y_j < 1 \end{cases} \quad (6)$$

where, c (0, 0.5] = 0.4 and y_0 = 0.7.

4.2 Integration of chaotic maps

First, the combination among sinusoidal, quadratic and piecewise maps uses Rand function to generate random integer numbers (1,2, or 3).

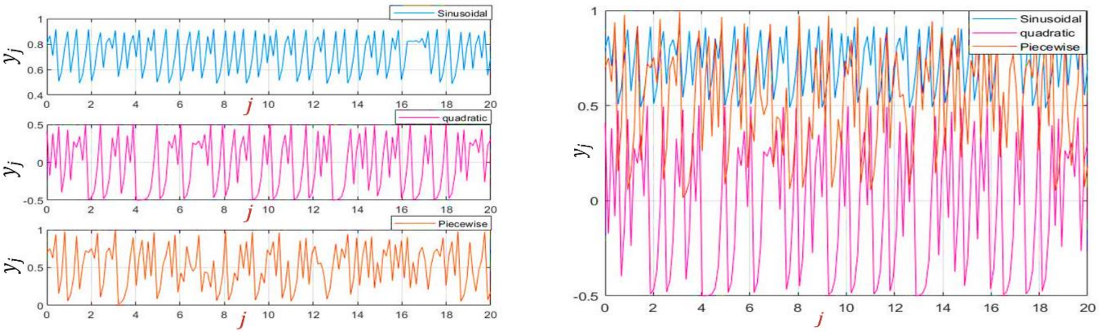

Figure. 1 Sinusoidal, quadratic and piecewise map before and after integration

Depending on that generated number, one block (each contains eight numbers) is chosen from one of these maps, as shown in Algorithm 1.

The proposed S-box contains 256 non-repeated discrete values. Therefore, Eq. (7) is used to convert the produced sequence of Algorithm 1 into integer values in the range [0, 255].

$$S\text{-}box = round\ (Yi \times 255), \tag{7}$$

where *Yi* is a float number resulted from algorithm 1. The repeated numbers are then replaced randomly by new numbers.

Algorithm 1. Initial S-box generation based on chaotic maps

Input: sinusoidal, quadratic, and piecewise maps float values.
Output: initial S-box with 256 float values.

```
For i=0: 31
    Generate random integer number Rand=[1,3];
        If Rand value = 1, then S-box= 8 values of the sinusoidal output.
        Elseif Rand value = 2, then S-box= 8 values of the quadratic output.
        Elseif Rand value = 3, then S-box= 8 values of the piecewise output.
        End if.
End for
```

The initial S-box result was obtained by using chaos maps representing a good starting point for searching using FPA than starting from scratch, increasing the ideal solution’s speed and using less memory. Table 1 shows the initial values S-Box of FPA generated by chaos maps. This S-box needs to be manipulated by FPA to obtain a robust and secure S-box used in cryptography operations.

Table 1. The initial values S-Box of FPA generated by chaos maps

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

4.3 Flower pollination algorithm (FPA)

FPA can easily manage various difficulties compared with other intelligent optimisation algorithms because it can automatically subdivide a population. Therefore, four rules govern this algorithm: i) Local pollination includes abiotic methods and self-pollination. ii) Global pollination includes biotic methods and cross-pollination. iii) The likelihood of duplication is related to the similarity of the two flowers involved. As a result, flower consistency is considered. iv) Global and local pollination are controlled by the switch probability p. Eqs. (7) and (9) can be used to transform these rules into equations [5, 6]:

$$x_i^{n+1} = x_i^n + L(x_i^n - g_*) \qquad (8)$$

$$x_i^{n+1} = x_i^n + \varepsilon(x_i^n - x_k^n) \qquad (9)$$

where x_i^n represented the solution vector, and g_* is the current best solution through the iterations. *L* is the step size from the Levy distribution, which indicates the pollination strength. Levy flying is a technique for simulating the long-distance migration of insects. The parameter *L* is calculated as follows:

$$\mathrm{L} \sim \frac{\lambda\Gamma(\lambda)\sin(\frac{\pi\lambda}{2})}{\pi}\,\frac{1}{\mathrm{S}^{1+\lambda}}\,,(S \gg \mathrm{S}_0 > 0) \qquad (10)$$

λ denotes the probability density function. *Γ(λ)* represents the standard gamma function, whereas S represents the pollination step size. For positive random values, this probability distribution is continuous.

4.4 Adapted FPA

The S-box elements must be unique, according to the bijectivity requirement. However, FPA created by the swarm operation does not always meet this requirement. As a result, each FPA S-box adjustment (as shown in Fig. 2.) is accomplished as follows:

1. The initial population of FPA is used based on Algorithm 1. Therefore, it is suitable for S-box requirements.
2. The results of Eq. 10 must be converted into values in the range (0-255) using Eq. (7) to guarantee the bijection condition because the S-box contains only 256 discrete values.
3. Eq. (2) is used as an objective function to evaluate the current best solution.
4. The numbers of replacement elements depend on initialised parameters.
5. In this way, we ensure that the values of the S-box are non-repeated integers and within the upper and lower bounds. Algorithm 2 shows the adapted FPA.

Algorithm 2. S-box generation based on adapted FPA

```
input: the result of Algorithm 1 as initial values of S-box, probability P[0,1], initialise parameters.
output: manipulated S-box with high nonlinearity.
  Evaluate the S-box by using Eq. 2 and find the g current best solution;
   while (stopping criterion is not satisfied) do
     for each flower
       If rand < P
          Used Eq. 8 for global pollination
       Else
         Select two random solutions and used Eq. 9
         for local pollination;
      Endif
      Evaluate new solutions;
      Update the results with the best new ones;
    End for
    Store the current best solution;
   End while.
```

5. Evaluation of the proposed S-box

A variety of experiments based on S-box generation were carried out to determine adapted FPA efficiency. All experiments were carried out on a computer with an Intel(R) Core™ i7- 8565U CPU running at 3.79 GHz and 8 GB of RAM, running Windows 10 (64-bit OS) and MATLAB Release 2021a. The proposed approach has an asymptotic computational complexity of $O(n^2)$. [24, 36, 37] have $O(n^3)$, $O(n^4)$ and $O(n^5)$, respectively.

5.1 Algebraic cryptanalysis

A variety of multivariate systems and stream ciphers have been successfully broken using algebraic approaches. However, their viability versus block ciphers is still a matter of debate. Algebraic cryptanalysis aims to crack cryptosystems utilising symbolic computation, mathematical tools and modern algebraic techniques. In more detail, an algebraic assault may be broken down into two parts: to compute the solutions of the resulting polynomial system, the cryptosystem and its specifics are transformed into a set of multivariate polynomial equations. The complexity of solving the related polynomial problem determines the security of a cryptographic primitive. A public key, symmetric cryptosystems and block and stream ciphers are vulnerable to these attacks [38, 39].

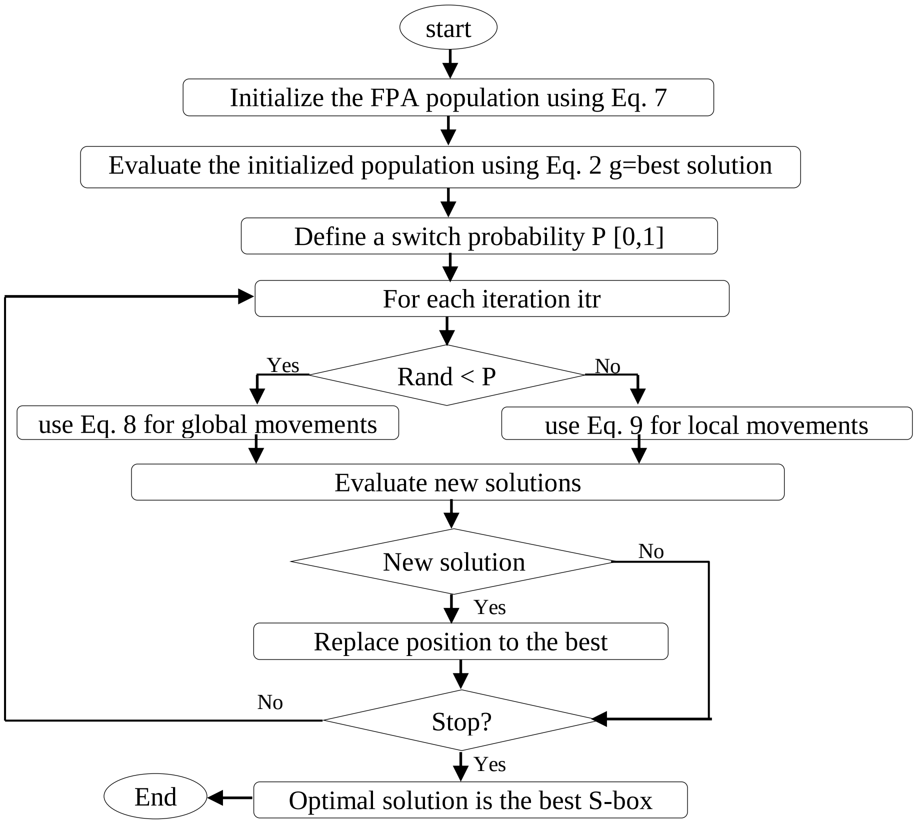
start
Initialize the FPA population using Eq. 7
Evaluate the initialized population using Eq. 2 g=best solution
Define a switch probability P [0,1]
For each iteration itr
Rand < P
Yes
No
use Eq. 8 for global movements
use Eq. 9 for local movements
Evaluate new solutions
New solution
No
Yes
Replace position to the best
No
Stop?
Yes
End
Optimal solution is the best S-box

Figure. 2 Flowchart of the adapted FPA for S-box design

Table 2. S-Box generated by the adapted FPA

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

Definition:

Resistance to algebraic attacks is defined by providing r equations of t monomials in $GF(2)^8$:

$$\Gamma = \frac{(t-r)^{\lceil (t-r)/n \rceil}}{n} \quad (11)$$

For S-box obtained by adapted FPA, $t = 510$, $r = 255$, and $n = 8$. Thus, we obtained $\Gamma \approx 2^{160}$. Given that Γ must be greater than 2^{32} to be secured, AES has $t = 81$, $r = 23$ and $n = 8$, $\Gamma \approx 2^{22.9}$ is considered a weak point for the S-box of AES design [38]. As a

Table 3. Nonlinearity of S-boxes analysis

Ref.	N1	N2	N3	N4	N5	N6	N7	N8	Min.	Max.	Avg.
Proposed	110	106	106	108	108	108	106	106	106	110	107.25
[17]	104	106	108	106	102	104	106	106	102	108	105.25
[18]	102	102	100	106	102	100	104	98	98	106	101.75
[19]	108	106	104	106	108	106	106	106	104	108	106.25
[20]	108	108	106	106	106	106	106	106	106	108	106.5
[21]	106	106	106	106	105	106	107	106	105	107	106
[13]	108	106	106	106	106	110	106	108	106	110	107
[24]	108	106	108	110	110	108	104	100	100	110	106.75
[22]	104	100	108	106	102	106	104	108	100	108	104.75
[26]	106	106	110	108	106	108	108	108	106	108	107.5
[25]	110	106	108	106	106	106	104	106	104	110	106.5
[27]	108	108	108	108	108	108	108	108	108	108	108
[8]	108	108	108	108	108	108	110	110	108	110	108.5
[23]	112	110	112	112	110	112	112	112	110	112	111.5
[4]	112	110	112	108	108	110	112	112	108	112	110.5

result, we conclude that the proposed S-box is quite strong and is impossible to attack.

5.2 Analysis of the generated S-box's performance

The robust S-box must meet the following criteria, some of which are listed in Section 2. The results of S-boxes produced using FPA techniques were compared with fourteen S-boxes generated using other approaches.

i) Bijective property: The obtained S-box is bijective because all values of it are in the range [0,255]. The Hamming weight of all Boolean functions is 128 128 128 128 128 128 128 128.

ii) Nonlinearity property: Nonlinearities of eight Boolean functions of the generated S-boxes compared with other S-boxes generated using various approaches are computed by Eq. (2) and shown in Table 3. S-box got by FPA has nonlinearity scores of 110, 106, 106, 108, 108, 108, 106 and 106, with an average of 107.25, which was the aim of the objective function. The FPA approach is considered adequate with acceptable results way to obtain high nonlinearity compared with other approaches.

iii) Strict avalanche criteria (SAC):

As shown in Table 4, the dependency matrix is used to describe the SAC of the proposed S-box. The generated S-box has an average SAC value of 0.4973. It's close to the optimal value (0.5). Table 6 shows a comparison of the SAC of the proposed S-boxes in this study. The FPA approach has been shown to produce results with acceptable SAC characteristics.

Table 4. Dependency matrix of proposed S-box

0.515	0.515	0.453	0.500	0.500	0.531	0.484	0.531
0.484	0.500	0.437	0.531	0.500	0.453	0.500	0.468
0.515	0.578	0.468	0.515	0.437	0.500	0.468	0.437
0.546	0.453	0.562	0.593	0.531	0.406	0.546	0.453
0.515	0.468	0.578	0.453	0.531	0.468	0.500	0.375
0.515	0.453	0.484	0.515	0.468	0.484	0.437	0.500
0.531	0.515	0.468	0.531	0.531	0.500	0.531	0.453
0.484	0.593	0.437	0.515	0.484	0.562	0.468	0.546

Table 5. BIC-nonlinearity criterion of proposed S-box

0	108	106	106	96	106	106	102
108	0	104	98	100	104	104	104
106	104	0	104	106	104	102	96
106	98	104	0	104	104	100	106
96	100	106	104	0	106	98	96
106	104	104	104	106	0	100	102
106	104	102	100	98	100	0	106
102	104	96	106	96	102	106	0

5.3 Resistance against differential analysis

The differential attack is another popular type of attack. Attackers pick a basic image and edit it with slight modifications, such as a one-bit change. They then encrypt the two images using the cryptosystem.

Consequently, they try to decipher the scheme by tracking the discrepancies [40]. In general, two measures are used to assess differential attack invulnerability: UACI and NPCR. The formulae for computing these two measures are listed below [41]:

$$UACI = \frac{1}{W \times H} \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} \frac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\% \tag{12}$$

where C_1 and C_2 are the cipher images resulted from the proposed approach.

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100\% \tag{13}$$

where $D(i,j) = \begin{cases} 1, & if\ (C_1(i,j) \neq C_2(i,j) \\ 0, & if (C_1(i,j) = C_2(i,j) \end{cases}$

The primary purpose of S-boxes is to create confusion. Therefore, the proposed approach is applied ShiftRows and MixColumns to obtain efficient confusion and diffusion property of images. In general, Table 7 shows the overall findings of the suggested method, which demonstrate that all UACI scores are close to the predicted normal levels (33.2255 percent). Furthermore, NPCR scores are nearly identical to theoretical normal levels (99.5693 percent) [41]. As a result, we concluded that the suggested technique passed the UACI and NPCR tests. Consequently, the proposed approach offers enough diffusion operations while also being robust to differential attack.

5.4 Histogram analysis:

The pixel distribution of an image is presented by the histogram [42]. The eavesdropper may attack the encryption image over a histogram analysis [43].

Table 6. Comparison of SAC and BIC results

Ref.	SAC Avg.	BIC Avg.
Proposed S-box	0.498	104.7857
[17]	0.4994	103.57
[18]	0.5017	102.64
[19]	0.4996	103
[20]	0.5001	104.07
[21]	0.5065	102.64
[13]	0.5014	104.21
[24]	0.5002	104
[22]	0.4938	105.07
[26]	0.5092	103.07
[25]	0.4995	104.57
[27]	0.5068	103.35
[8]	0.491	103.78
[23]	0.506	104.2
[4]	0.506	106.43

As a result, to avoid statistical histogram attacks, the histogram of the cipher image should be as uniform as possible. Fig. 3 depicts various original images and histograms of R, G and B channels of the original images before and after encryption. According to the histogram, the encryption histogram is more uniform than the simple histogram. As a result, the system is robust against histogram attacks and stable.

5.5 Correlation coefficient analysis

An image correlation coefficient is a statistical connection between neighbouring pixels. The strong correlation between nearby pixels of a basic image and cipher image should be broken by an ideal encryption method. Using Eq. (14) below, we compute the correlation coefficient of pixels from the plain image and the corresponding cipher image [44].

Table 7. UACI and NPCR scores of the tested images

The tested image	Size of Image (W.H)	Channel	The score of UACI (%)	The score of NPCR (%)
Penguins	16 × 16	R	34.536	98.978
		G	34.581	98.997
		B	34.588	98.988
Baboon	16 × 16	R	34.625	98.983
		G	34.658	98.943
		B	34.654	98.943
Peppers	16 × 16	R	34.698	99.009
		G	34.683	99.140
		B	34.689	99.059
Butterfly	16 × 16	R	34.598	98.959
		G	34.579	98.975
		B	34.587	98.958

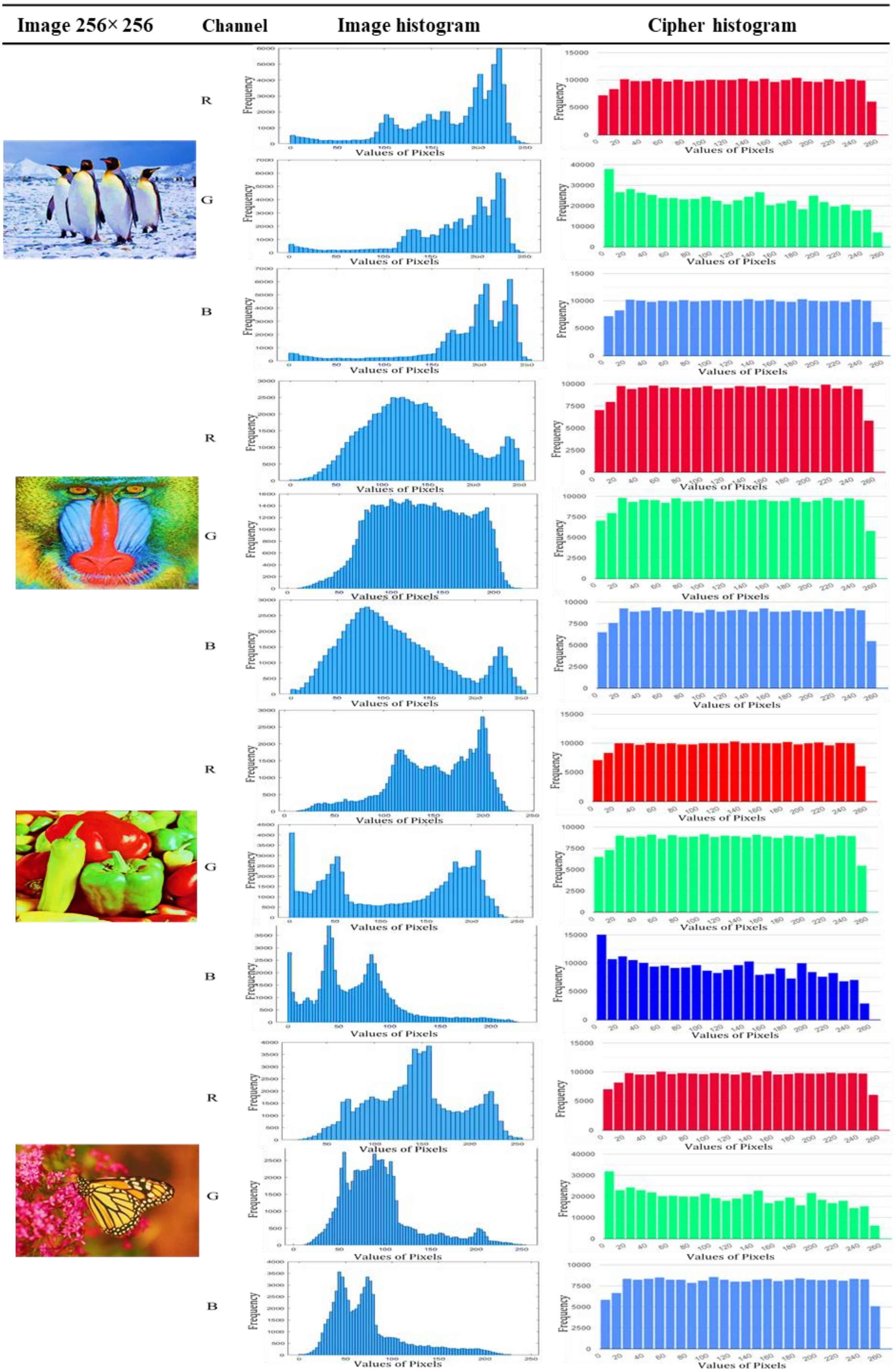

Figure. 3 Histogram of images

Received: September 17, 2021. Revised: October 22, 2021.

Table 8. Entropy and correlation coefficients scores of various images

Image	Channel	Entropy of plain image	Entropy of cipher image	Correlation
Penguins	R	7.3260	7.9919	-0.00089
	G	7.7278	7.9278	-0.00974
	B	7.0593	7.9918	-0.00437
Baboon	R	7.6841	7.9925	-0.00156
	G	7.3780	7.9924	0.00586
	B	7.6961	7.9926	0.00012
Peppers	R	7.3216	7.9925	0.00605
	G	7.6003	7.9924	0.00031
	B	7.1354	7.9270	0.00271
Butterfly	R	7.3260	7.9923	0.00118
	G	7.7278	7.9295	-0.00149
	B	7.0593	7.9919	0.00083

$$C_{xy}=\frac{\frac{1}{N}\sum_{i=1}^{N}(x_i-E(x))(y_i-E(y))}{\sqrt{D(x)}\sqrt{D(y)}} \quad (14)$$

$$d(x)=\frac{1}{N}\sum_{i=1}^{N}(x_i-E(x))^2 \ , \ E(x)=\frac{1}{N}\sum_{i=1}^{N}x_i$$

where x_i and y_i are the gray image values, N is the total pixels, and $E(x)$, $E(y)$ are x_i and y_i mean values, respectively. We may deduce that the correlation findings between plain image pixels and cipher image pixels are poor (around zero) (as shown in Table 7). As a result, the suggested strategy is successful in preventing statistical analysis assaults.

5.6 Shannon entropy analysis

Information entropy is a fundamental notion in computer science. It is necessary for data compression in security and information coding. Furthermore, the Shannon entropy is a valuable metric for calculating the degree of disorder or chaos, assessing the complexity of compounding processes and estimating the divergence of probability distributions [45]. Entropy is a measure of a cryptographic key's unpredictability that is frequently used in cryptanalysis. To crack a key using a brute force attack, the attacker needs 2^{k-1} (k=key length in bits). If the potential keys are not chosen randomly, entropy fails to capture the required number of guesses [46]. The entropy values are situated at an optimum interval, as shown in Table 8. As a result, the output image of the proposed technique is secured from various statistical assaults.

6. Conclusion

This work presented a novel approach for generating a robust 8×8 S-box. The obtained S-box is compared with several works. Accordingly, the most important results show that the resistance to algebraic attacks was $\Gamma \approx 2^{160}$. Thus, Γ must be greater than 2^{32} to be secured. The asymptotic computational complexity was O(n^2). The nonlinearity property was 107.25. It is considered an effective and acceptable result. SAC and BIC were 0.498 and 104.7857, respectively. Those criteria were not utilised as an objective function of the FPA. However, their results were considered accepted and better than other references. Furthermore, the resistance against differential analysis was proven using histogram, entropy, correlation coefficients, UACI and NPCR measurements. Therefore, the proposed S-box provides good confusion, and it is robust and resistant to cryptography attacks. In future work, the generated 8×8 S-box will be further improved into 16×16 S-box by developing enhanced another metaheuristic algorithm technique.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

The first author was responsible for the paper's idea, software, methodology, formal analysis, data curation, validation, resources, writing a review, editing, writing original draft preparation, and visualization. Whereas, the second author was responsible for project management and supervision.

References

[1] S. Q. A. A. Rahman, S. A. Jassim, and A. M. Sagheer, "Design a mobile application for vehicles managing of a transportation issue", *Bull. Electr. Eng. Informatics*, Vol. 10, No. 4, 2021.

[2] S. Jassim and W. Kareem, "Searching over

encrypted shared data via cloud data storage", *J. Theor. Appl. Inf. Technol.*, Vol. 96, 2018.

[3] F. Ishfaq, "A MATLAB Tool for the Analysis of Cryptographic Properties of S-boxes.", *Capital University*, 2018.

[4] A. Alhudhaif, M. Ahmad, A. Alkhayyat, N. Tsafack, A. K. Farhan, and R. Ahmed, "Block Cipher Nonlinear Confusion Components Based on New 5-D Hyperchaotic System", *IEEE Access*, 2021.

[5] M. Ramadas and A. Abraham, "Metaheuristics for data clustering and image segmentation", *Springer*, 2019.

[6] S. M. N. S. D. K. Lin, "Search and Optimization by Metaheuristics, Techniques and Algorithms Inspired by Nature", *Springer International Publishing*, 2016.

[7] X. S. Yang, "Flower pollination algorithm for global optimization", In: *Proc. of International Conference on Unconventional Computing and Natural Computation*, pp. 240-249, 2012.

[8] H. S. Alhadawi, D. Lambić, M. F. Zolkipli, and M. Ahmad, "Globalized firefly algorithm and chaos for designing substitution box", *J. Inf. Secur. Appl.*, Vol. 55, p. 102671, 2020.

[9] S. Barshandeh and M. Haghzadeh, "A new hybrid chaotic atom search optimization based on tree-seed algorithm and Levy flight for solving optimization problems", *Eng. Comput.*, pp. 1-44, 2020.

[10] T. Akhtar, N. Din, and J. Uddin, "Substitution box design based on chaotic maps and cuckoo search algorithm", In: *Proc. of International Conference on Advanced Communication Technologies and Networking*, pp. 1-7, 2019.

[11] L. Wang and Y. Zhong, "Cuckoo search algorithm with chaotic maps", *Math. Probl. Eng.*, Vol. 2015, 2015.

[12] J. A. Clark, J. L. Jacob, and S. Stepney, "The design of S-boxes by simulated annealing", *New Gener. Comput.*, Vol. 23, No. 3, pp. 219-231, 2005.

[13] M. Ahmad, D. Bhatia, and Y. Hassan, "A novel ant colony optimization based scheme for substitution box design", *Procedia Comput. Sci.*, Vol. 57, pp. 572-580, 2015.

[14] Y. Tian and Z. Lu, "S-box: Six-dimensional compound hyperchaotic map and artificial bee colony algorithm", *J. Syst. Eng. Electron.*, Vol. 27, No. 1, pp. 232-241, 2016.

[15] A. K. Farhan, R. S. Ali, H. R. Yassein, N. M. G. A. Saidi, and G. H. A. Majeed, "A new approach to generate multi S-boxes based on RNA computing", *Int. J. Innov. Comput. Inf. Control*, Vol. 16, pp. 331-348, 2020.

[16] Tao, Hai, S. Q. Salih, M. K. Saggi, E. Dodangeh, C. Voyant, N. A. Ansari, Z. M. Yaseen, and S. Shahid, "A newly developed integrative bio-inspired artificial intelligence model for wind speed prediction", *IEEE Access*, Vol. 8, pp. 83347-83358, 2020.

[17] F. Özkaynak, "On the effect of chaotic system in performance characteristics of chaos based s-box designs", *Phys. A Stat. Mech. its Appl.*, Vol. 550, p. 124072, 2020.

[18] Z. M. Z. Muhammad and F. Özkaynak, "A Cryptographic Confusion Primitive Based on Lotka-Volterra Chaotic System and Its Practical Applications in Image Encryption", In: *Proc. of 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering*, pp. 694-698, 2020.

[19] M. A. B. Farah, A. Farah, and T. Farah, "An image encryption scheme based on a new hybrid chaotic map and optimized substitution box", *Nonlinear Dyn.*, pp. 1-24, 2019.

[20] D. Lambić, "A new discrete-space chaotic map based on the multiplication of integer numbers and its application in S-box design", *Nonlinear Dyn.*, Vol. 100, No. 1, pp. 699-711, 2020.

[21] V. M. S. García, R. F. Carapia, C. R. Márquez, B. L. Benoso, and M. A. Pérez, "Substitution box generation using Chaos: An image encryption application", *Appl. Math. Comput.*, Vol. 332, pp. 123-135, 2018.

[22] I. Hussain, T. Shah, M. A. Gondal, W. A. Khan, and H. Mahmood, "A group theoretic approach to construct cryptographically strong substitution boxes", *Neural Comput. Appl.*, Vol. 23, No. 1, pp. 97-104, 2013.

[23] A. H. Zahid, M. Ahmad, A. Alkhayyat, M. T. Hassan, A. Manzoor, and A. K. Farhan, "Efficient Dynamic S-Box Generation Using Linear Trigonometric Transformation for Security Applications", *IEEE Access*, 2021.

[24] S. S. Jamal, A. Anees, M. Ahmad, M. F. Khan, and I. Hussain, "Construction of cryptographic S-boxes based on mobius transformation and chaotic tent-sine system", *IEEE Access*, Vol. 7, pp. 173273-173285, 2019.

[25] T. Farah, R. Rhouma, and S. Belghith, "A novel method for designing S-box based on chaotic map and teaching-learning-based optimization", *Nonlinear Dyn.*, Vol. 88, No. 2, pp. 1059-1074, 2017.

[26] Y. Tian and Z. Lu, "Chaotic S-box: Intertwining logistic map and bacterial foraging optimization", *Math. Probl. Eng.*, Vol. 2017, 2017.

[27] Y. Wang, K. W. Wong, C. Li, and Y. Li, "A novel method to design S-box based on chaotic map and genetic algorithm", *Phys. Lett. A*, Vol. 376, No. 6-7, pp. 827-833, 2012.

[28] H. A. Abdulwahab, A. Noraziah, A. A. Alsewari, and S. Q. Salih, "An enhanced version of black hole algorithm via levy flight for optimization and data clustering problems", *IEEE Access*, Vol. 7, pp. 142085-142096, 2019.

[29] Yaseen, Z. Mundher, A. M. A. Juboori, U. Beyaztas, N. A. Ansari, K. Chau, C. Qi, M. Ali, S. Q. Salih, and S. Shahid, "Prediction of evaporation in arid and semi-arid regions: A comparative study using different machine learning models", *Eng. Appl. Comput. Fluid Mech.*, Vol. 14, No. 1, pp. 70-89, 2020.

[30] Yaseen, Z. Mundher, M. F. Allawi, H. Karami, M. Ehteram, S. Farzin, A. N. Ahmed, S. B. Koting, N. S. Mohd, W. Z. B. Jaafar, and H. A. Afan, "A hybrid bat-swarm algorithm for optimizing dam and reservoir operation", *Neural Comput. Appl.*, Vol. 31, No. 12, pp. 8807-8821, 2019.

[31] D. Teodorović, "Bee colony optimization (BCO)", *Innovations in Swarm Intelligence*, Springer, pp. 39-60, 2009.

[32] N. Ifada and R. Nayak, "A New Weighted-learning Approach for Exploiting Data Sparsity in Tag-based Item Recommendation Systems", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 1, pp. 387-399, 2021.

[33] M. S. Fadhil, A. K. Farhan, and M. N. Fadhil, "Designing Substitution Box Based on the 1D Logistic Map Chaotic System", In: *Proc. of IOP Conference Series: Materials Science and Engineering*, Vol. 1076, No. 1, p. 12041, 2021.

[34] D. Herbadji, A. Belmeguenai, N. Derouiche, and H. Liu, "Colour image encryption scheme based on enhanced quadratic chaotic map", *IET Image Process.*, Vol. 14, No. 1, pp. 40-52, 2019.

[35] I. Fibriani, W. Widjonarko, A. Prasetyo, A. M. Raharjo, and D. E. Irawan, "Multi Deep Learning to Diagnose COVID-19 in Lung X-Ray Images with Majority Vote Technique", *International Journal of Intelligent Engineering and Systems*, Vol.13, No.6, pp. 560-568, 2020.

[36] M. F. Khan, A. Ahmed, K. Saleem, and T. Shah, "A novel design of cryptographic SP-network based on gold sequences and chaotic logistic tent system", *IEEE Access*, Vol. 7, pp. 84980-84991, 2019.

[37] X. Wu, D. Wang, J. Kurths, and H. Kan, "A novel lossless color image encryption scheme using 2D DWT and 6D hyperchaotic system", *Inf. Sci. (Ny).*, Vol. 349, pp. 137-153, 2016.

[38] A. A. Abdel-Hafez, Reda-Elbarkouky, and Wageda_Hafez, "Algebraic Cryptanalysis of AES using Gröbner Basis", *International Advanced Research Journal in Science, Engineering and Technology*, Vol. 3, No. 12, pp. 183-189, 2016.

[39] W. Alsobky, H. Saeed, and A. N. Elwakeil, "Different Types of Attacks on Block Ciphers".

[40] Y. Liu, Z. Qin, X. Liao, and J. Wu, "A Chaotic Image Encryption Scheme Based on Hénon-Chebyshev Modulation Map and Genetic Operations", *Int. J. Bifurc. Chaos*, Vol. 30, No. 06, p. 2050090, 2020.

[41] E. Yavuz, "A novel chaotic image encryption algorithm based on content-sensitive dynamic function switching scheme", *Opt. Laser Technol.*, Vol. 114, pp. 224-239, 2019.

[42] S. A. Mohseni, H. R. Wu, J. A. Thom, and A. B. Hadiashar, "Recognizing Induced Emotions With Only One Feature: A Novel Color Histogram-Based System", *IEEE Access*, Vol. 8, pp. 37173-37190, 2020.

[43] A. A. Shah, S. A. Parah, M. Rashid, and M. Elhoseny, "Efficient image encryption scheme based on generalized logistic map for real time image processing", *J. Real-Time Image Process.*, Vol. 17, No. 6, pp. 2139-2151, 2020.

[44] C. Xu, J. Sun, and C. Wang, "An image encryption algorithm based on random walk and hyperchaotic systems", *Int. J. Bifurc. Chaos*, Vol. 30, No. 04, p. 2050060, 2020.

[45] M. Cholewa and B. Płaczek, "Application of Positional Entropy to Fast Shannon Entropy Estimation for Samples of Digital Signals", *Entropy*, Vol. 22, No. 10, p. 1173, 2020.

[46] Nannipieri, Pietro, S. D. Matteo, L. Baldanzi, L. Crocetti, J. Belli, L. Fanucci, and S. Saponara, "True Random Number Generator Based on Fibonacci-Galois Ring Oscillators for FPGA", *Applied Sciences*, Vol. 11, No. 8. 2021.