Development of data encryption standard algorithm based on magic square

Suhad Muhajer Kareem[1] Abdul Monem S. Rahma[2]
[1]Department of Computer Science, College of Computer Science and Information Technology, University of Basra, Basrah, Iraq
[2]Department of Computer Science, Al-Maarif University College, Anbar, Iraq

Article Info

Article history:

Received Nov 28, 2021
Revised Jun 15, 2022
Accepted Jul 20, 2022

Keywords:

Data encryption standard
Entropy
Histogram
Magic square
Peak-signal-to-noise ratio

ABSTRACT

Data encryption standard is one of the famous algorithms that used in many fields for security purpose but it s susceptible for many attacks. This paper+ proposes a new modification of data encryption standard (DES) algorithm called magic square data encryption standard (MSDES) algorithm in order to include a high-level security by increase the mixing between the plaintext (96 bit) and the key (48 bit). This modification is done by using magic square 3×3 and an additional key is created using linear first shift register (LFSR) in each round of the Feistel of DES. A colour image encryption is simulated and presented as the comparison between the original DES and MSDES algorithm. The proposed algorithm gets the best results in complexity, histogram, entropy, peak-signal-to-noise ratio-(PSNR) and coefficient-correlation.

Corresponding Author:

Suhad Muhajer Kareem
Department of Computer Science, College of Computer Science and Information Technology
University of Basra
Basrah, Iraq
Email: suhad.kareem@uobasrah.edu.iq

1. INTRODUCTION

Cryptography is the research of mathematical methods regarding to sides of information safety like confidentiality, authentication for entity, data completeness [1], [2]. Cryptography function of a substantial role in secure-communication and it supplies-an-premium solution in order to present the necessary-protection against the-data sponge. In the cryptography, the plaintext original is indicated as an original-information and cipher text is referred as encrypted-information using encryption algorithms [3], [4]. Depending on the number of keys, two types of encryption algorithms can be categorized into symmetric and asymmetric. The encryption and decryption in symmetric algorithms based on Feistel structure such as data encryption standard (DES) (data-standard-system) and non-Feistel such-as Advanced-encryption-standard (AES) [5], [6]. In this paper, the focus will be on the DES algorithm which is one of symmetric encryption algorithms that applied in many applications and relied on Feistel network [7], [8].

Magic square can be defined as the-art-and the-study of methods-that considers the logical and quantity forms and also ordering-them and it also can be applied in cryptography and it has large application in recreation mathematics like puzzles [9], [10]. The magic square is the-order-of-the-numbers-in-the-form-of a-square-matrix with-equal dimensions that the amount of each row numbers, column numbers, and diagonal-numbers-is a constant number that is known by the magic constant [11]. Since Feistel structure in DES is composed of 16 rounds and in each round a sub key is generated to mix with data inside the F-function. This can be increased the quality of encryption and decryption process in DES algorithm, but one of the weakness of Feistel structure is the lack of mixing between the data, as the mixing process is done once at the beginning and the end using of the permutation process. Since DES suffers from many problems such as less in mixing between data, so, this paper proposed a new modification on Feistel of DES using magic square for increasing the security of it. This is done by increasing the mixing between data and the key using magic square that used summation fashion in each-round of -DES.

The remainder of paper is arranged as: in Section 2 give short introduction for DES and magic square, then Section 3 proposed literature survey. Section 4 proposed the MSDES algorithm. Finally, section 5 and 6 show the results and conclusion of work.

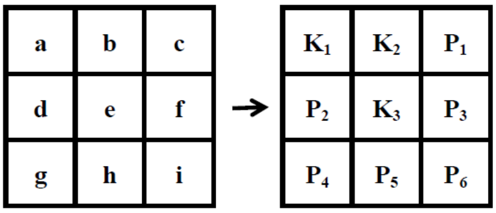
2. METHOD

2.1. DES (Data Encryption Standard) algorithm

In 1972, IBM developed DES as the earliest symmetric encryption algorithm. The algorithm depends on Feistel structure that accepts 64-bit keys, where only 56-bits have already been used because the residual 8-bits were applied for error discovery purposes [12], [13]. The encryption process relies on a 56-bit secret key maps input block (64 bit) into a output block (64 bit). The input block (64-bits) are pass through sixteen Feistel l iterations encompassed by two permutation process ((an initial permutation and inverse) [14], [15].

2.2. Magic square

Magic square matrix can be defined as a square matrix in which the sum of all elements in each column and in each row is same. The sum can be calculated from the formula $(n*(n^2+11))/22$, where n represents the size of square matrix. The core record for the magic square is how to organize a set of numbers into the matrix form where each row summation values or column summation values even the two diagonals values should give the same summation result [16]. To build magic square (3×3) offer in Figure 1 by employing (3-bytes) of randomly for choosing numbers in range (1-255) as a key and (6-bytes) of plain-text [11], [17].



a	b	c
d	e	f
g	h	i

→

K_1	K_2	P_1
P_2	K_3	P_3
P_4	P_5	P_6

Figure 1. Magic square (3×3) construction

For using magic square for encryption, there are many steps are used: firstly, create the magic square by employing the 3×3 magic square, secondly: multiply the mask with magic square. The final process is to compute the summation of specific row, column and diagonals of the last matrix (3×3) as a cipher text as it has shown in algorithm 1 [11]. For applying summations using:

$$K1 + K22 \ + P13 = sum11 \tag{1}$$

$$P44 \ + \ K33 \ + P33 = sum22 \tag{2}$$

$$K11 \ + \ P22 \ + P44 = sum33 \tag{3}$$

$$P11 \ + \ P33 + P66 = sum44 \tag{4}$$

$$K11 \ + \ K33 + P66 = sum55 \tag{5}$$

$$P11 \ + \ K33 + P44 = sum66 \tag{6}$$

In the side of decryption, the receiver has applied the following steps for complete the decryption process: at the first, construct augmented matrix (AA) of linear equation system of magic square (3×3) and the cipher text exchanged with the last column, the last column of the matrix (AA) by subtract the last known value of the key, then reducing the matrix (AA) by delete 1, 2, and 5 columns. Regenerate the magic square (3×3) using the result of Gaussian elimination instead of plain text and the last known value of the key, for number of rounds, multiplication of the magic square by the inverse of the encryption mask to gain the result in algorithm referenced in [11].

Example: the following example illustrates how to apply encryption and decryption using magic square 3×3:
Encryption step:
Input: Plain text=5-3-9-11-12-1, output: cipher -text, key=10 6 4
Mask=

252	1	252
3	1	253
2	254	2

Step 1: Create magic square

K_1	K_2	P_1
P_2	K_3	P_3
P_4	P_5	P_6

→

10	6	5
3	4	9
11	12	1

Step2: Multiplication magic square with mask one by one as the following:
(110 × 2252) over GF (2^8)=1622.

10	6	5
3	4	9
1	12	1

*

252	1	252
3	1	253
2	254	2

=

162	6	154
5	4	57
22	174	2

Step3: Cipher-text is the summation over GfF(2^8) by employing equations (1 through 6)=1266, 1666, 226, 1616, 1646 and 1366 respectively.

Decryption Step
Input: cipher text last known value of the key
Output: plain-text
Step1: construct enhanced matrix (AA) for the magic square as the next:

K11	K22	P11	P22	K32	P32	P42	P52	P62	
11	111	111	00	00	00	00	00	00	1126
00	00	00	00	00	00	111	11	11	1166
11	00	00	11	00	00	11	00	00	122
00	00	11	00	00	11	00	00	11	1161
11	00	00	00	11	00	00	00	11	1164
00	00	11	00	11	00	11	00	00	1136

Step2: Update the last column of the matrix (AA) based on the last known as:

$$\text{Last column} = \begin{cases} AA_{\text{i last column}} - k_1 & \text{if } AA_{i,1} = 1 \text{ or} \\ AA_{\text{i last column}} - k_2 & \text{if } AA_{i,2} = 1 \text{ or} \\ AA_{\text{i last column}} - k_3 & \text{if } AA_{i,5} = 1 \end{cases}$$

Step3: Delete the (first, second and the fifth) of key-column) of the array and resort the array:

P11	PP2	PP33	PP44	PP55	PP66	S-sum
111	00	00	00	00	00	154
00	11	00	11	00	00	180
11	00	11	0	00	11	161
11	00	00	11	00	00	140
00	00	00	11	11	11	166
00	00	00	00	00	11	2

Step4: Gaussian elimination over GF (2^8):

P11	P22	P33	P44	P55	P66	
11	00	00	00	00	00	154
00	11	00	00	00	00	5
00	00	11	00	00	00	45
00	00	00	11	00	00	22
00	00	00	00	11	00	188
00	00	00	00	00	11	14

Step5: Multiplication with reverse mask

1162	16	1154		17	1	17		10	6	5
15	14	145	*	208	1	22	=	3	4	9
122	1188	114		1	70	1		11	12	1

3. LITRATURE SURVEY

This section presents some of literature works that related to the magic square. Jabbar and Rahma [11] presented protocol depends on magic square of size 4*4, linear equation system and finite field. It is information encryption based on the key, a block of numerical data content of eight numbers of plain text in order to create a magic square, and cipher text which represents the summation of each row, each column, diagonal, and co-diagonal of the magic square. The algorithm is implemented for encryption and decryption, this algorithm depends on split data into blocks and mix with key in order to build a magic square to specify a cipher text performed as magic summations, and using gaussian elimination as method to recovery the plain text. Dawood *et al*. [18] have proposed a new method for the constructing of magic cube together with the utilize of the method of folded magic-square. This method treated a new step in the direction of constructing the magic-cube which used a clear insight and provided a straightforward popularized approach. Dharini *et al*. [19] presented a new approach for safely transporting data by the cloud and sharing networks through the secure-socket-layer (SSL) also, via combination between Rivest-Shamir-Adleman (RSA) algorithm and magic square for increasing an additional level of security to the cryptographic system. Thus, it has clarified and incorporated the RSA with magic square for executing it on data security in cloud computing. the work mension in Shibiraj *et al*. [3] purpuses to propose the appliction of Latin squares and the magic squares of odd order *n* in the encryption and decryption of Hill cipher. The pair of orthogonal diagonal Latin-square (ODLS) of odd order and the magic square so derived are used for double encryption and double decryption in the modified Hill cipher to make the cryptosystem more reliable. Different cipher text can be produced from a single diagonal Latin square (DLS) and diagraph letters are introduced in addition to the existing 26 letters of English alphabet to make the encryption and decryption possible for the modified Hill cipher. In this study, we apply magic square with additional key in 10 round of the DES algorithm, as it has showed in the following sections.

4. PROPOSED METHOD

The DES is one of encryption algorithms that used in many applications but it considered unsecure for many causes. It firstly relies on only a single bit (0 or 1). Similarly, it utilizes only-one binary function (XOR), as it does not contain enough combination between the plain text and the key and is exposed to aggressions. Hence, to outdo these problems, in this section, a new modification on DES algorithm called MSDES (Magic square DES) based on using magic square 3×3 is proposed to ameliorate the encryption performance and make the algorithm more complicated against aggressions. A perfect encryption algorithm has particular characteristic, like its strength to combat cryptanalysis of aggressions. This is accomplished by a perfect combination of bits that produce to hardness in guessing the key. Thus, in this paper, a new manipulation of bits by used more mathematical operation such as multiplication, using magic square (3×3) and used additional key that generated randomly based on linear feedback shift register (LFSR) technique for each round. The principal work of MSDES can be summarized as follows: MSDES accepts plaintext (96 bits or 12 byte), key1 (64 bit) and key2 (48 bits) as input, then ten rounds of the keys and plaintext are utilized. The structure of MSDES algorithm is based on the Feistel structure, which splits the input plaintext (96 bit)-block into two-halves: left (48-bits or 6 byte) and-right (48-bits or 6 byte), then make combination between the key1and plaintext and key2 and plaintext to construct two arrays 3×3 to apply multiplication for these two arrays. After that, apply selection process (horizontally, vertically or diagonal) on the resulted array (9 byte) for selection only 3 bytes. At the last step, apply magic square between the 3 bytes that selected from previous step and left (6 bytes) and the result from this step is provided in the next right block. This process is iterated ten rounds for produce the cipher text. Figure 2 show the modification on DES algorithm in one round. The overall work of MSDES can be summarized in the following Algorithm 1.

Algorithm 1. proposed MSDES algorithm

```
Input: Plaintext (96-bit) and key (64-bit) K1.
Output: Cipher text (96-bit)
Begin
 Step1: Divide plaintext into L (48-bit) and R (48-bit).
 Step2: Schedule input key (64-bit) for 16 round, let Ki.
 Step3: for ten round
 Step3.1: Select only 3 byte from K1i, let K¯1i.
 Step3.2: Add 6 byte of Ri to 3 byte of K¯1i (R¯i=Ri+K¯1i)
 to construct array1 3×3.
 Step3.3: Add 6 byte of Ri to 3 byte of K¯2i (R¯i=Ri+K¯2i)
 to construct array2 3×3.
 Step3.4: Multiplication array1 and array2 based on
 GF (28).
 Step3.5:Select only 3 byte from the result step3.4 (columns, rows or diagonals).
 Step3.6: Apply magic square as
 Step3.6.1: Construct magic square (3×3) using the 3
 bytes of step 3.3 and 6 byte of Li.
 Step3.4.2: Generate randomly mask array with
 numbers arrange (1-255).
 Step3.4.3: Multiply mask array with magic square
 based on the rules of finite field.
 Step3.4.4: Summations of magic square results from
 step 3.4.3 using the equations (1-6).
 End for
 Step4: Swap final blocks L10, R10.
End.
```

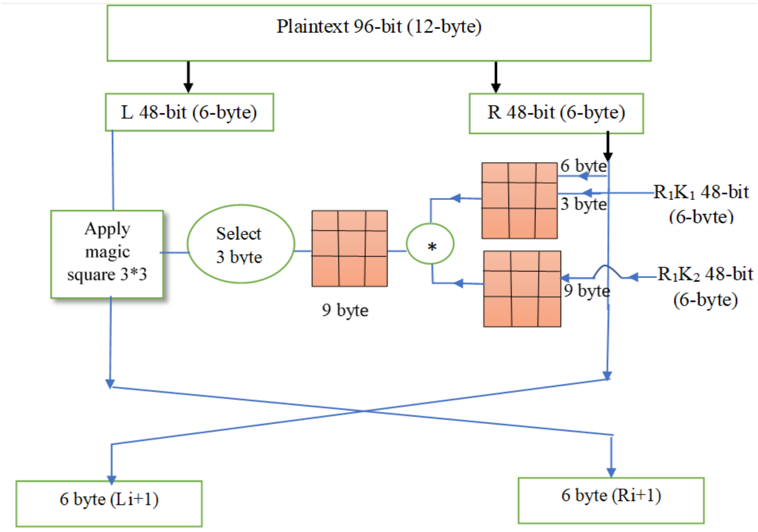


Figure 2. Modified one round in MSDES

5. EVALUATION

This section compares the proposed MSDES algorithm to the well-known DES algorithm for encrypting images. In this section, five evaluation methods are used for measuring the safety of test of encrypted images. Such as complexity, histogram, PSNR, information entropy, correlation -coefficient.

5.1. Complexity

Encryption algorithms are exposed to many attacks such as the brute force attack which the attacker tries all possible keys to violate the algorithm for retrieving the original text. So, the design of the suggested algorithm makes it stronger against this type of attack. The complexity of the proposed MSDES algorithm is computed as shown in:

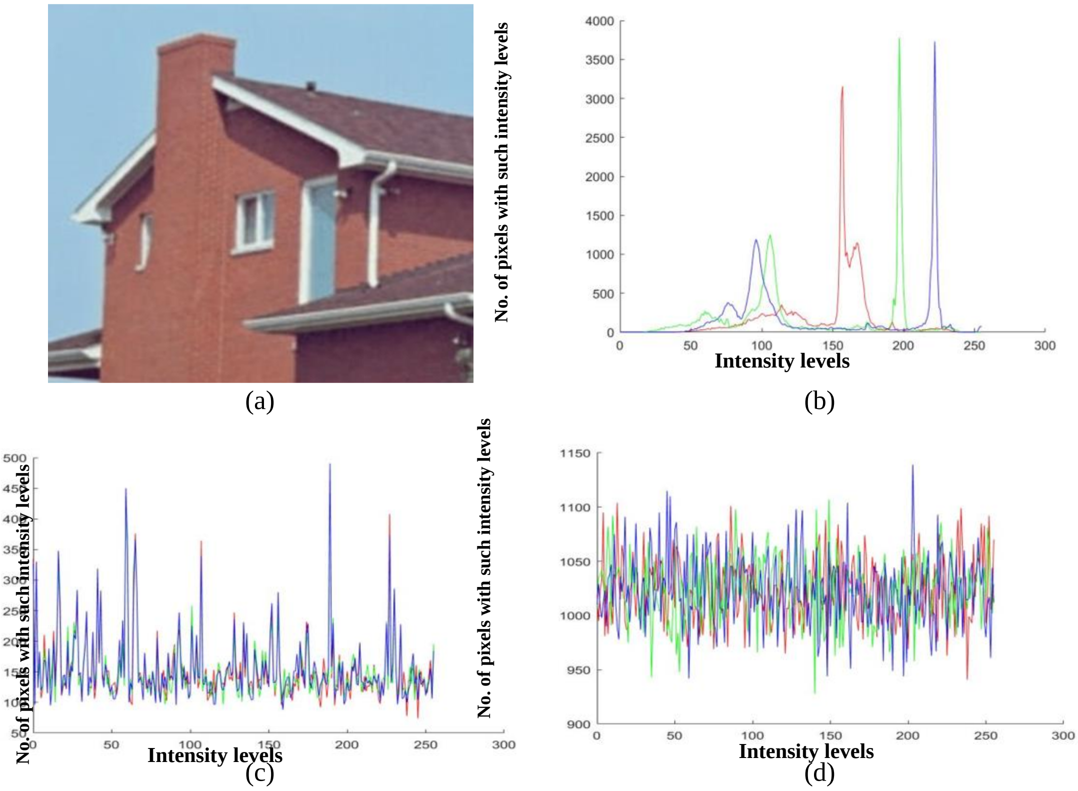
$$2 \times (2)8 \times 32 \times 2 = 2 \times (2)8 \times 25 \times 2 = 215$$

Then, the security of MSDES algorithm is computed based on using additional key and magic square:

$$2 \times (2)8 \times 48 \times 2 \times (256)5 \times (2)8 \times 3 \times 9 = 227 \times (256)5 \times 3 \times 9$$

5.2. Histogram

A histogram is one of security metrics is utilized to gauge the safety of the original and encrypted-mages by displying the: distribution between the pixels. The aim of utilizing histogram analysis to explain the graph that represent the intensity values between the pixels in images. A histogram analysis was computed for both the well-known and proposed MSDES algorithms. Two standard colour images with JPEGG formats are used in this metric and results are shown in Figures 3 and 4. Figures 3(a)-(d) shows the results of the histogram of image1 while Figures 4(a)-(d) shows the results of the histogram of image 2. Figure 3 consist of (a) orginal image1, (b) histogram of image1, (c) and (d) histogram of image 1 for DES and MSDES algorthims respectively. Figure 4 consist of (a) orginal image2, (b) histogram of image 2, (c) and (d) histogram of image 2 for DES and MSDES algorthims respectively.



(a) (b)

(c) (d)

Figure 3. Comparing histogram results for orginal image1 with histogram using original DES and MSDES, (a) orginal image 1, (b) histogram of image 1, (c) histogram of image1 using original DES, and (d) histogram of image1 using MSDES

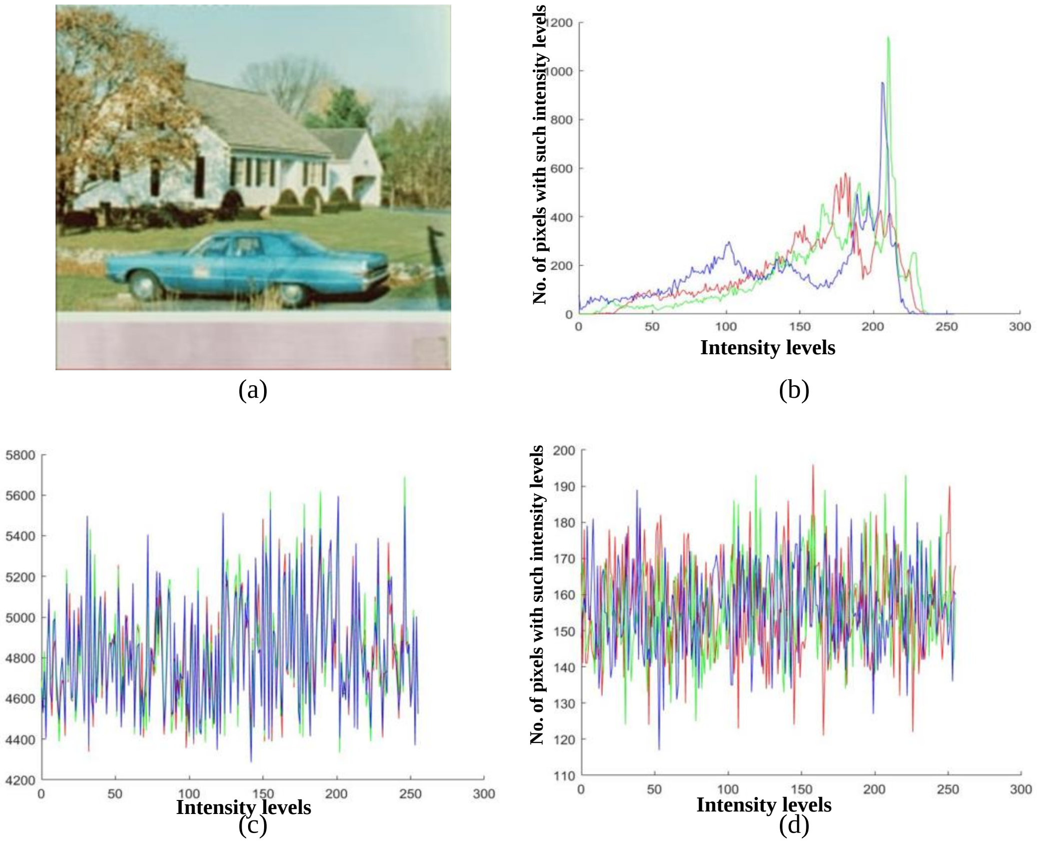
5.3. Peak-signal-to-noise-ratio (PSNR)

PSNR is metric used to compute the ratio between the original image and encrypted image which referred to signal and noise respectively. Basically, MSE represents a collective squared error between encrypted and original image. There is proportional relation between MSE and error in which lower value of MSE is the lower error [20]. PSNR is employed to estimate the resistance of the encryption algorithm which it can be calculated as shown in [21].

$$PSNR = 10\ log10\ \left(\frac{255^2}{MSE}\right) \tag{7}$$

$$MSE = \frac{1}{M \times N} \sum_{I,J} |I1(i,j) - I2(i,j)|^2 \tag{8}$$

Where MSEE is the key square error between the recovered image $I2$ $(i,)$ and the original image $I1(i, j)$, and M and N are the rows and columns which exemplify the width-and-height of the image. PSNR evaluates an encryption designer that mirrors the encryption goodness. The-lower-value of PSNR displays better encryption quality. The values of PSNRR and MSEE are computed-and listed in Table 1.



(a) (b)

(c) (d)

Figure 4. Comparing histogram results for orginal image1 with histogram using original DES and MSDES, (a) orginal image 2, (b) histogram of image 2, (c) histogram of image 2 using original DES, and (d) histogram of image 2 using MSDES

Table 1. MSEE and PSNRR between original and recovered images for the proposed MSDESS

Images	Guassian mean==0 and variance==0.001	
	MSE	PNSRR
Image1	0.285	50.62
Image2	0.200	54.99

5.4. Correlation coefficient

Many aggressions can be designed to attack the encrypted images established on a statistical analysis of the association between the pixels. To assessment the goodness of the encrypted image, the correlation-coefficient that references in equation 9 [22], [23]. Table 2 displays the-correlation distribution in all directions of two adjacent pixels in the plaintext images and cipher text image.

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{[\sum(X_i - \bar{X})^2 \, (Y_i - \bar{Y})^2\,]^{1/2}} \tag{9}$$

Where X and Y are the pixels and neighboring pixels of the original and encrypted-image. Table 2 listed the values of the correlation coefficient. Figure 5 show two encrypted images (a) and (b) that are used for calculate the results of correlation coffiecient and histogram distribution (Figure 3 & Figure 4).

Table 2. the values of correlation coefficient for DES and MSDES algorithms

Images	Correlation for DES algorithm		Correlation for MSDES algorithm	
Image a	Vertical	-0.0016614	Vertical	-0.00049242
	Horizontal	0.01933	Horizontal	-0.014488
	Diagonal	0.006674	Diagonal	0.015791
Image b	Vertical	0.010353	Vertical	0.0018144
	Horizontal	0.0029166	Horizontal	0.0073463
	Diagonal	-0.005454	Diagonal	-0.015235

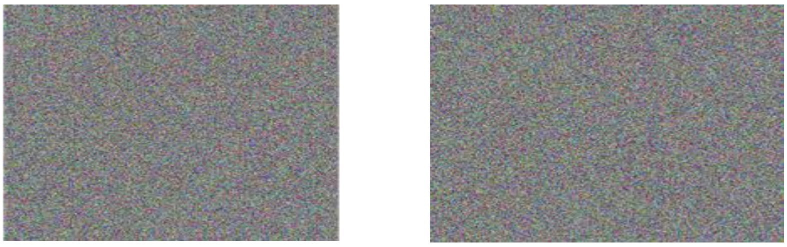
5.5. Entropy

In digital image, the information entropy can be an indicator of the distribution of pixel values and defined as is simply the average (expected) amount of the information from the data. For computing the randomness. For a typical random image, the information entropy is calculated by [24], [25]:

$$H(V) \quad = \sum_{i=0}^{255} P(v_i) log_2 \mathrm{P}(v_i) \tag{10}$$

where, vi is the i-the gray level of the image and P(v_i) is the probability of v_i .

Table 3 shows the values of information entropy of two encrypted images in Figures 5(a) and (b). The information entropy of the encrypted images of the MSDES algorithm is better than the information entropy of the encrypted images of the well-known DES algorithm. Thus, the rendering and security of the proposed-algorithm is obvious. The results in Table 3 display that the values of entropy of the-encrypted-image created by the MSDES algorithm are closed-to-08, and that denotes the encrypted images are closed to a random.



(a) (b)

Figure 5. Results of two encrypted images of the proposed MSDES algorithm, (a) encrypted image a and (b) encryted image b

Table 3. Information entropy for the encrypted-images-in-Figure 5

Image a		Image b	
Plain image	Encrypted image	Plain image	Encrypted image
7.521	7.863	7.244	7.912

6. CONCLUSION

This paper proposes a development to the common DES algorithm is known as MSDES algorithm. The MSDES algorithm is more secure against a differential analysis attack. This proposed is carried by using additional key and magic square 3×3 in each-round of the Feistel of DES. The plaintext of MSDES algorithm could be digits, images or text. The experimental results show the effectiveness of the encryption algorithm in many security metrics. The proposed MSDES provides more complexity in the key space against a brute force attack in 10 round of MSDES algorithm. According to the histogram, PSN, entropy and correlation coefficient analysis, proposed MSDES algorithm also supplies good results in encrypting and decrypting samples of images.

REFERENCES

[1] K. Limniotis, “Cryptography as the means to protect fundamental human rights,” *International Journal of Computer Applications*, vol. 5, no. 34, pp. 1–33, November. 2021, doi: 10.3390.

[2] A. S. Malalla and M. R. Shareef, “Improving hiding security of arabic text steganography by hybrid AES cryptography and text steganography,” *Journal of Engineering Research and Application www.ijera.com ISSN*, vol. 6, no. 65, pp. 2248–962260, 2016.

[3] N. Shibiraj and I. Tomba, “Modified hill cipher secure technique using Latin square and magic square,” *International Journal of Computer Sciences and Engineering*, vol. 6, no. 12, pp. 315–320, Dec. 2018, doi: 10.26438/ijcse/v6i12.315320.

[4] S. Chandra, B. Mandal, S. S. Alam, and S. Bhattacharyya, “Content based double encryption algorithm using symmetric key cryptography,” *Procedia Computer Science*, vol. 57, pp. 1228–1234, 2015, doi: 10.1016/j.procs.2015.07.420.

[5] E. A. Al-Bahrani and R. N. J. Kadhum, “A new cipher based on Feistel structure and chaotic maps,” *Baghdad Science Journal*, vol. 16, no. 1, pp. 270–280, Mar. 2019, doi: 10.21123/bsj.2019.16.1(Suppl.).0270.

[6] S. M. Kareem and A. M. S. Rahma, “A novel approach for the development of the Twofish algorithm based on multi-level key space,” *Journal of Information Security and Applications*, vol. 50, p. 102410, Feb. 2020, doi: 10.1016/j.jisa.2019.102410.

[7] H. Harahsheh and M. Qatawneh, “Performance evaluation of twofish algorithm on IMAN1 supercomputer,” *International Journal of Computer Applications*, vol. 179, no. 50, pp. 1–7, Jun. 2018, doi: 10.5120/ijca2018916654.

[8] M. Ebrahim, S. Khan, and U. Bin Khalid, “Symmetric algorithm survey: a comparative analysis,” *arXiv preprint*, 2014, [Online]. Available: http://arxiv.org/abs/1405.0398.

[9] M. Maity, “A modified version of polybius cipher using magic square and western music notes,” *International Journal for Technological Research in Engineering*, vol. 1, no. 10, pp. 1117–1119, 2014.

[10] P. J. Eccles, *An Introduction to Mathematical Reasoning*. Cambridge University Press, 1997.

[11] D. A. Jabbar and A. M. S. Rahma, “Proposed cryptography protocol based on magic square, linear algebra system and finite field,” *Journal of Advanced Research in Dynamical and Control Systems*, vol. 10, no. 10, pp. 101–105, 2018.

[12] S. M. Kareem and A. M. S. Rahma, “New modification on feistel DES algorithm based on multi-level keys,” *International Journal of Electrical and Computer Engineering*, vol. 10, no. 3, pp. 3125–3135, Jun. 2020, doi: 10.11591/ijece.v10i3.pp3125-3135.

[13] M. A. Al-Shabi, “A survey on symmetric and asymmetric cryptography algorithms in information security,” *International Journal of Scientific and Research Publications (IJSRP)*, vol. 9, no. 3, p. p8779, 2019, doi: 10.29322/ijsrp.9.03.2019.p8779.

[14] G. Prashanti, S. Deepthi, and S. R. K, “A novel approach for data encryption standard algorithm,” *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 2, no. 5, pp. 264–267, 2013.

[15] J. Thakur and N. Kumar, “DES, AES and Blowfish: Symmetric key cryptography algorithms simulation based performance analysis,” *International Journal of Emerging Technology and Advanced Engineering*, vol. 1, no. 2, pp. 6–12, 2011.

[16] D. I. G. Amalarethinam, J. Sai Geetha, and K. Mani, “Add-on security level for public key cryptosystem using magic rectangle with column/row shifting,” *International Journal of Computer Applications*, vol. 96, no. 14, pp. 38–43, Jun. 2014, doi: 10.5120/16866-6755.

[17] S. D. Mohammed and T. M. Hasan, “Cryptosystems using an improving hiding technique based on latin square and magic square,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 1, pp. 510–520, Oct. 2020, doi: 10.11591/ijeecs.v20.i1.pp510-520.

[18] O. A. Dawood, A. M. S. Rahma, and A. M. J. Abdul Hossen, “Generalized method for constructing magic cube by folded magic squares,” *International Journal of Intelligent Systems and Applications*, vol. 8, no. 1, pp. 1–8, Jan. 2016, doi: 10.5815/ijisa.2016.01.01.

[19] A. Dharini, R. M. S. Devi, and I. Chandrasekar, “Data security for cloud computing using RSA with magic square algorithm,” *International Journal of Innovation and Scientific Research*, vol. 11, no. 2, pp. 439–444, 2014.

[20] O. F. Mohammad, M. Shafry, M. Rahim, S. Rafeeq, M. Zeebaree, and F. Y. H. Ahmed, “A survey and analysis of the image encryption methods,” *International Journal of Applied Engineering Research*, vol. 12, no. 23, pp. 13265–13280, 2017, [Online]. Available: http://www.ripublication.com.

[21] B. Arpacı, E. Kurt, and K. Çelik, “A new algorithm for the colored image encryption via the modified Chua’s circuit,” *Engineering Science and Technology, an International Journal*, vol. 23, no. 3, pp. 595–604, Jun. 2020, doi: 10.1016/j.jestch.2019.09.001.

[22] P. Ramasamy, V. Ranganathan, S. Kadry, R. Damaševičius, and T. Blažauskas, “An image encryption scheme based on block scrambling, modified zigzag transformation and key generation using Enhanced Logistic-Tent Map,” *Entropy*, vol. 21, no. 7, p. 656, Jul. 2019, doi: 10.3390/e21070656.

[23] J. L. Rodgers and W. A. Nicewander, “Thirteen ways to look at the correlation coefficient,” *The American Statistician*, vol. 42, no. 1, p. 59, Feb. 1988, doi: 10.2307/2685263.

[24] A. Karawia, “Encryption algorithm of multiple-image using mixed image elements and two dimensional chaotic economic map,” *Entropy*, vol. 20, no. 10, p. 801, Oct. 2018, doi: 10.3390/e20100801.

[25] M. A. and D. Mohammed, “Image encryption technique based on the entropy value of a random block,” *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 7, 2017, doi: 10.14569/ijacsa.2017.080735.

BIOGRAPHIES OF AUTHORS

Suhad Muhajer Kareem she is lecture at college of computer science and information technology, university of basrah, iraq. She Holds a PhD degree in Computer science with data security. Her research areas are image/signal processing, security, data mining and text mining. She can be contacted at email: suhad.kareem@uobasrah.edu.iq.

Prof. Abdul Monem S. Rahma have an extensive background in the field of Cryptography and Information Security. In 1984, he received his PhD in Computer Science from the Loughborough University of Technology in the United Kingdom, and become a professor in Computer Science since 2008. He was the Deputy Dean of the Department of Computer Science, University of Technology, Baghdad, Iraq from 2005 to 2013; and then from 2013 to 2015 become the Dean of the department. Now Prof. Rahma the head of the Department of Computer Science, Al-Maarif University College, Anbar Iraq. He can be contacted at email: monem.rahma@uoa.edu.iq.