Mustansiriyah Journal of Pure and Applied Sciences, Print ISSN: 2957-9910. Online ISSN: 2957-9929, Vol. 3, No. 1, 2025, Pages 49-63
DOI: 10.47831/mjpas.v3i1.119

MJPAS

MUSTANSIRIYAH JOURNAL OF PURE AND APPLIED SCIENCES

Journal homepage*:*
https://mjpas.uomustansiriyah.edu.iq/index.php/mjpas

RESEARCH ARTICLE - Computer Science

Hybrid Algorithm (DES-Present) for encryption color image using 2D-Chaotic System

Suha Husam Jasim [*1] **, Haider Kadhim Hoomod** [2]**, Khalid Ali Hussein** [3]

[*123] Suha Husam Jasim Department of Computer Science, Education, University Mustansiriyah, Baghdad 10052, Baghdad, Iraq, (suha_hussam@uomustansiriyah.edu.iq).

[2] Haider Kadhim Homood, (drhjnew@gmail.com).[3] Khalid Ali Hussein, (dr.khalid.ali68@gmail.com).

[*] Correspondent contact: suha_hussam@uomustansiriyah.edu.iq

Article Info.

Article history:

Received
10 February 2024

Accepted
22 April 2024

Publishing
30 January 2025

Abstract

In recent years, image encryption algorithms have been developed to protect against unauthorized persons and maintain the privacy of recipients. Therefore, it is very important to choose the right algorithm and build it in a way that meets security and performance requirements. The DES algorithm is slow in encryption and decryption operations and vulnerable to attacks. There is a lightweight block cipher for devices with limited resources. May be more vulnerable than differential cryptanalysis. The study suggests that to encrypt and decrypt images as quickly as possible, to achieve security, the key of the two algorithms (DES-Present) was generated using a two-dimensional chaos equation system. Where the Twice DES algorithm executes four rounds within each round, the present algorithm executes only four rounds. The performance evaluation of the proposed algorithm was measured through several metrics, where a PSNR low value of (8.614653) indicates that the good encryption quality, Reverse it high value MSE (8945.715332) and MAE (77.625417) this indicates that the original image is completely different from the encrypted image. The NPCR high value of (99.4965%) indicates a high degree of accuracy in changing pixel values. In addition, the uniform average change intensity (UACI) less than30 % shows that the algorithm is good at making large changes in pixel intensity. Correlation coefficient test: horizontal 0.002574; diagonal -0.003017; vertical -0.035860 this indicates that all three correlation values are close to zero. Key space equal for the($2^{232.4}$) is It is quite enough space to resist brute force penetration. The results showed that the proposed algorithm is good for encrypting color images, which makes it suitable for use in various applications that need security and speed.

The official journal published by the College of Education at Mustansiriya University

Keywords: DES, Hybrid Algorithm, Image, Two-dimensional Chaotic equation, Present Lightweight.

1. INTRODUCTION

Digital technologies have made computer security crucial. Computer science uses cryptography to protect data transmission and storage by converting plain text into cipher text[1]. Most image storage uses pixel arrays with different color values. The popular RGB color model uses these three primary colors to show the lighting in an image. Pixel color values determine how an image appears on screen or in print[2]. Symmetric, asymmetric and hybrid algorithms are the main methods for image encryption. Symmetric key algorithms use the same key for fast encryption and decryption, but require secure key transfers between sender and receiver. Asymmetric key algorithms use separate encryption and decryption keys but are computationally expensive compared to symmetric key methods. Hybrid algorithms combine symmetric and asymmetric encryption for safety and performance[3].

Computational complexity, attack resistance, scalability, key management needs, etc. determine the advantages of the algorithm type[4]. Key generation is crucial to image encryption, which is the creation of a random and unpredictable sequence of bits using chaos systems as it is better than complex mathematical operations[5][6]. Potential vulnerabilities include statistical methods, brute force, and cryptanalysis. In a brute force attack, you try each key combination until one works. Statistical attacks look for biases or patterns to decode data. Cryptanalysis can compromise the security of encryption algorithms[7][8]. Implementing strong image encryption has many benefits. Privacy although intercepted during transmission or stored on vulnerable devices, encrypted images remain secret because only those with the decryption keys can decrypt them. Safety, image encryption prevents unauthorized modification and ensures that any change renders the decryption useless. Upon login, advanced encryption can include image validation to ensure image integrity[9]. The main contributions in this in the paper it is DES algorithm with the Present lightweight algorithm for encryption color Image using Two-dimensional Chaotic equation for the generate key gave the dynamic .

2. RELATED WORK

Fernando et al. (2019) suggested testing AES and DES on Raspberry Pi mini PCs. The study sought time- and memory-efficient algorithms for encrypting and decrypting data with the same key. The study found that AES was faster than DES. DES-encrypted messages use less memory than AES-encrypted ones. AES' 128-bit keys were more secure than DES' 64-bit keys. Developers can improve application and device encryption by comparing the two algorithms[10]. Putra et al. (2019) suggested studying BC3, AES, and DES power analysis attacks. Power analysis revealed cryptographic device keys to scientists. Five hundred traces helped researchers find the AES secret key. Researchers recovered 75% of the Data Encryption Standard secret key using 320 traces. The study found that Pearson correlation coefficient (PCC) power analysis outperformed the difference of means[11]. Gamier et al. (2019) suggested DES and RSA analysis and implementation using image and audio steganography[12]. Patel (2019) suggested compared Blowfish, AES and DES, shows multi-layer security comparison results. DES outperforms AES and Blowfish on text files under 1000 bytes. When processing 10,000-byte text files, AES, DES, and Blowfish perform similarly. The Blowfish algorithm outperforms AES and DES in execution memory[13]. Yang and Xiuqing (2020) suggested studying computer-based DES accounting data encryption. A quantum genetic algorithm improves the DES S-box design in the study. Studies show that 64 DES cypher texts and changed bits differ by 32 bits when encrypting accounting data. Key character validity simplifies data encryption and prevents key loss and leakage[14]. Laia, Zamzummim, 2021. The proposed algorithm implements DES and Blum-Blum-Shub (BBS) to secure message encryption and decryption. The NetBeans IDE application uses DES and BBS as an external key generator for security. Heterogeneous, many-core processors optimized for performance can encrypt and decrypt large files using DES and AES[15]. Xing et al. (2021) tested serial AES and DES codes on their experimental platform. Evaluations used task-specific criteria. The improved algorithm outperformed popular open-source AES and DES implementations in extensive testing. Serial DES and AES performed 40 and 72 times slower than parallelized ones[16]. Barhoush et al. (2022) suggested increasing the key size while maintaining costs to improve DES security. The paper suggests using random permutation and distributing initial and final permutation tables between encryption and decryption algorithms to improve DES encryption. The enhanced DES22 algorithm supports 128-, 256-, and 512-bit keys. Experiments show that DES22 is faster and more secure than AES[17]. A new CBC symmetric cypher and matrix power function were proposed by Mihalkovich et al. (2022). Comparison of AES-128 and triple DES. It evaluates three 64-bit arithmetic cyphers. The proposed MPF implementation cypher needs more memory than its competitors. The CPU evaluates cryptographic algorithms to determine the number of clock cycles needed to encrypt fixed-size plaintexts. This paper compares plaintext encryption operations and memory requirements for each cypher using pre-set parameters[18]. Alsuwaiedi and Rahma (2023) suggested improved DES security with the Magic Square Data Encryption Standard (MSDES) algorithm. The MSDES algorithm combines plaintext and keys for security. MSDES outperformed DES in complexity, histogram, entropy, PSNR, and coefficient-correlation in simulated color image

encryption[19]. Wiemers and Mittmann (2023) suggested studying DES key schedule side-channel attacks. In their test setup, template attacks on several microcontrollers reduced the average entropy of 56-bit DES keys to 48 bits [20]. More clarifications and highlighted about the research gaps as the show table (1).

Table1.mothode and the result related works

Ref	mothed	Result
[10]	AES and DES on Raspberry Pi	The study sought time- and memory-efficient algorithms for encrypting and decrypting data with the same key. The study found that AES was faster than DES. DES-encrypted messages use less memory than AES-encrypted ones.
[11]	BC3, AES, and DES	The result recovered 75% of the Data Encryption Standard secret key using 320 traces.
[12]	DES and RSA	analysis and implementation using image and audio steganography shows multi-layer security
[13]	Blowfish, DES, and AES	DES outperforms AES and Blowfish on text files under 1000 bytes. When processing 10,000-byte text files, AES, DES, and Blowfish perform similarly.
[14]	A quantum genetic algorithm improves the DES S-box design in the study	Show that 64 DES cypher texts and changed bits differ by 32 bits when encrypting accounting data.
[15]	DES and Blum-Blum-Shub (BBS)	The result The NetBeans IDE application uses DES and BBS as an external key generator for security
[16]	AES and DES	The improved algorithm outperformed popular open-source AES and DES implementations in extensive testing
[17]	Increasing the key size while maintaining costs to improve DES security.	Experiments show that DES22 is faster and more secure than AES
[18]	AES-128 and triple DES.	The cipher is faster 1.5 times faster compare AES-128 and faster 47 times faster than TDES.
[19]	improved DES security	MSDES outperformed DES in complexity, histogram, entropy, PSNR, and coefficient-correlation in simulated color image encryption
[20]	DES key schedule side-channel attacks	template attacks on several microcontrollers reduced the average entropy of 56-bit DES keys to 48 bits

3. METHODOLOGY

A hybrid algorithm (S-DES-Present) has been presented. Using the two-dimensional chaotic system to dynamically construct keys. Each execution phase makes it harder to predict the key, making key guessing harder. Unauthorized person. Many IoT algorithms are efficient yet need a lot of time, processor power, and memory, which are scarce. Thus, a balanced method between complexity, implementation time, resource use, and security is essential. Unlike DES, which is slow and vulnerable to assaults, the lightweight algorithm's minimal complexity makes it vulnerable to many attacks. This aims to provide an efficient color image encryption technique. DES will be paired with Present lightweight to achieve a balance between strong security and rapid encryption time amongst different encryption techniques.

3.1 The DES algorithm

DES key-symmetric block ciphers[21]. The DES encryption steps are, Key generation starts with 64-bit keys[17]. Permutation yields a 56-bit key (8th bit discarded).Key halved to 28 bits. Key Generation, 16 circular left shift rounds on both 28-bit halves[22]. Compress 56-bit halves for 48-bit round keys. IP alters 64-bit plaintext with an initial permutation table[23]. Split 64-bit blocks into 32-bit halves for the

Feistily Network (16 rounds).Per round, copy right-half-left. Permutations expand to 48 bits for the right half. Right-half Round-XOR grew. Use the S-box substitution on the results-box straight-permuters.Left-half XOR permuted output[24]. Replace left/right. To apply the Inverse Initial Permutation (IP^-1), swap the left and right halves after 16 rounds. 64-bit initial inverse permutation. Permutation, substitution, and XOR occur 16 times in DES ,and key schedule to generate unique round keys each round [25]. As shown in the following figure (1).

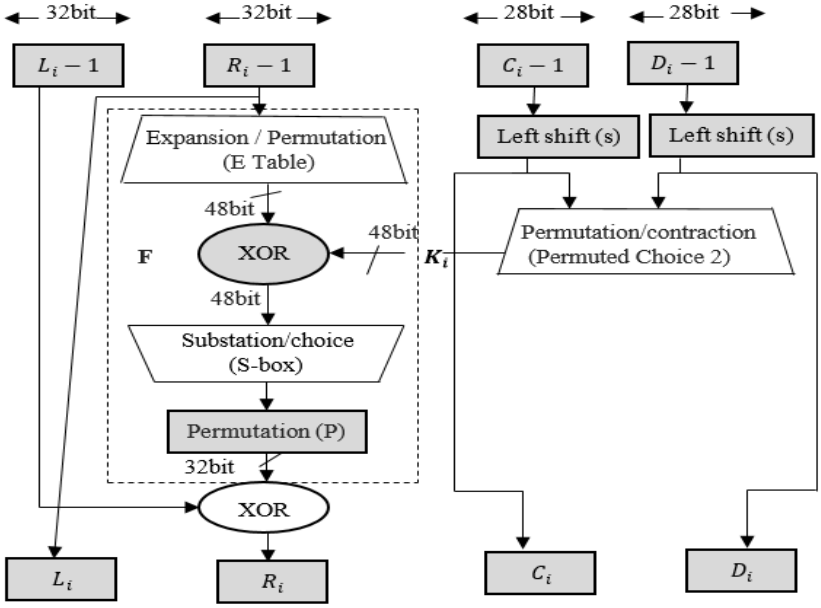


Fig 1. One round of DES algorithm[26].

3.2 Present Algorithm

Lightweight symmetric encryption algorithm used in cyber security to protect data. Since 2007, it has been used in AI, computer science, and insurance. It is also used as a problem-solving algorithm. Fast, resource-efficient, and secure[27]. The algorithm uses confusion and diffusion to encrypt. Secret keys are used for 80- or 128-bit encryption and decryption. The algorithm splits data into 64-bit chunks and performs 31 rounds of encryption or decryption[28]. Each round uses a "P-layer" for permutation and an "S-box" for substitution. This works best in low-resource environments with slow memory and processing. Unlike heavy ciphers, S-Box hardware optimization requires only 4 bits[29][30]. Layers of S-boxes are nonlinear. XOR the updated block k in the first 31 rounds and the round-key Ki for 1 to 32 if I is positive. Increased keys in round 32. The non-linear S-Box layer runs 4-bit S-Box 16 times per round[31][32]. Benefits of current encryption, it is much lighter than symmetric encryption algorithms and can be used on small devices with fewer resources. It is suitable for low-hardware environments because it is cheaper to implement than comparable algorithms. Its encryption key scheduling simplifies implementation. Internet of Things optical scanner fingerprint templates can be encrypted. Biometrics and user credentials can be encrypted. Due to its simple bending architecture, the current encryption algorithm is vulnerable to many attacks[33]. As shown in the following figure (2).

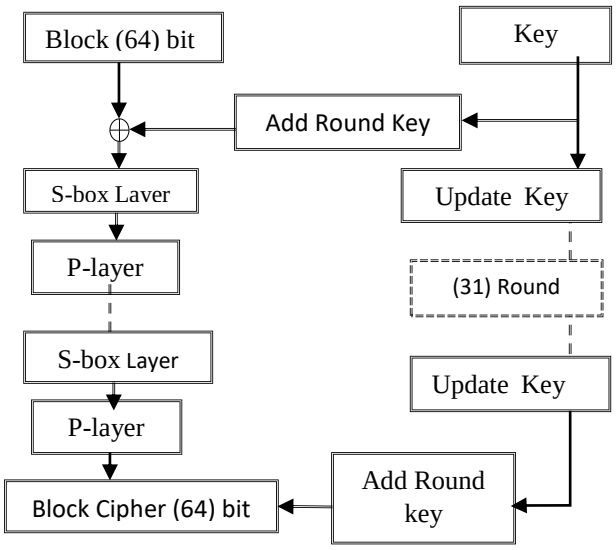


Fig 2. Block Diagram of Present Algorithm [31].

3.3 Chaotic Map Generation

The chaotic map generation is a computational procedure utilized to generate key streams within the suggested encryption algorithm. The nonlinear chaotic system is characterized by two quadratic nonlinear equations with two initial values and five control parameters. The composition consists of two equations, one for the variable x and another for the variable y. The outputs generated by this map generation process consist of three primary stream vectors, namely X, Y, and Z. Each vector has a size of (W x H x 3), where W and H represent the image's width and height in pixels. As in Eq (1) Two-dimensional equation system.

$$X_{i+1} = a\,Y_i^2 - b\,X_i^2 - c$$
$$Y_{i+1} = d\,X_i\,Y_i - e\,X_i \qquad (1)$$

Where, The values of a, b, c, d, and e have been assigned as follows: a = 4, b = 1.1, c = 4.4, d = 0.1, and e = 8. These values have been chosen in order to generate phase portraits that exhibit chaotic behavior[34].

Table 2. Two-dimensional Chaotic Key Generation [34]

Algorithm 1. Chaotic Keys Generation
Input: a, b, c, d, e, X_0, y_0, W, H Output: X, Y, Z // keys stream vectors of dimension (1,N)
start Algorithm Processing Algorithm: Step1: $X_1 = X_0$, : $y_1 = y_0$, : $z_{10} = X_1 \oplus y_1$ Step2: N = W×H × 3 Step3: Iterate (N-1) times $X_{i+1} = a\,Y_i^2 - b\,X_i^2 - c$ // i= (2, 3.....N). $Y_{i+1} = d\,X_i\,Y_i - e\,X_i$// i= (2, 3.....N). $z_{i+1} = X_{i+1_1} \oplus Y_{i+1}$. $X_i = X_{i+1}$ $Y_i = Y_{i+1}$ End Iteration End.

4. SUGGESTED ALGORITHMS

The procedures employed to generate and implement the hybrid algorithm (S-DES-Present), which was proposed as a means of securing image files via the two-dimensional chaotic system, are detailed in this document. Enhanced are security and resistance to brute force, statistical, and differential attacks. As shown in the block diagram in Figure 3. The DES algorithm is runs twice. Input the 128-bit block and block is split block into 64 bits to be an input to the one DES algorithm. And The 56-bit key is generated using the two-dimensional chaotic system equation and execute the algorithm four rounds instead of 16 rounds. This is considered an improvement to the DES algorithm, and within each one round for the DES algorithm there is income present algorithm it is implemented four rounds instead of 31 rounds. The present algorithm takes its input from the left side of the DES algorithm, where each side equals 32 bits. When the left sides of the algorithm are combined, it becomes 64 bits, and the key is 80 or 128 bits long, which is generated using the two-dimensional chaos system equation. Forget the block encryption 128 bit.

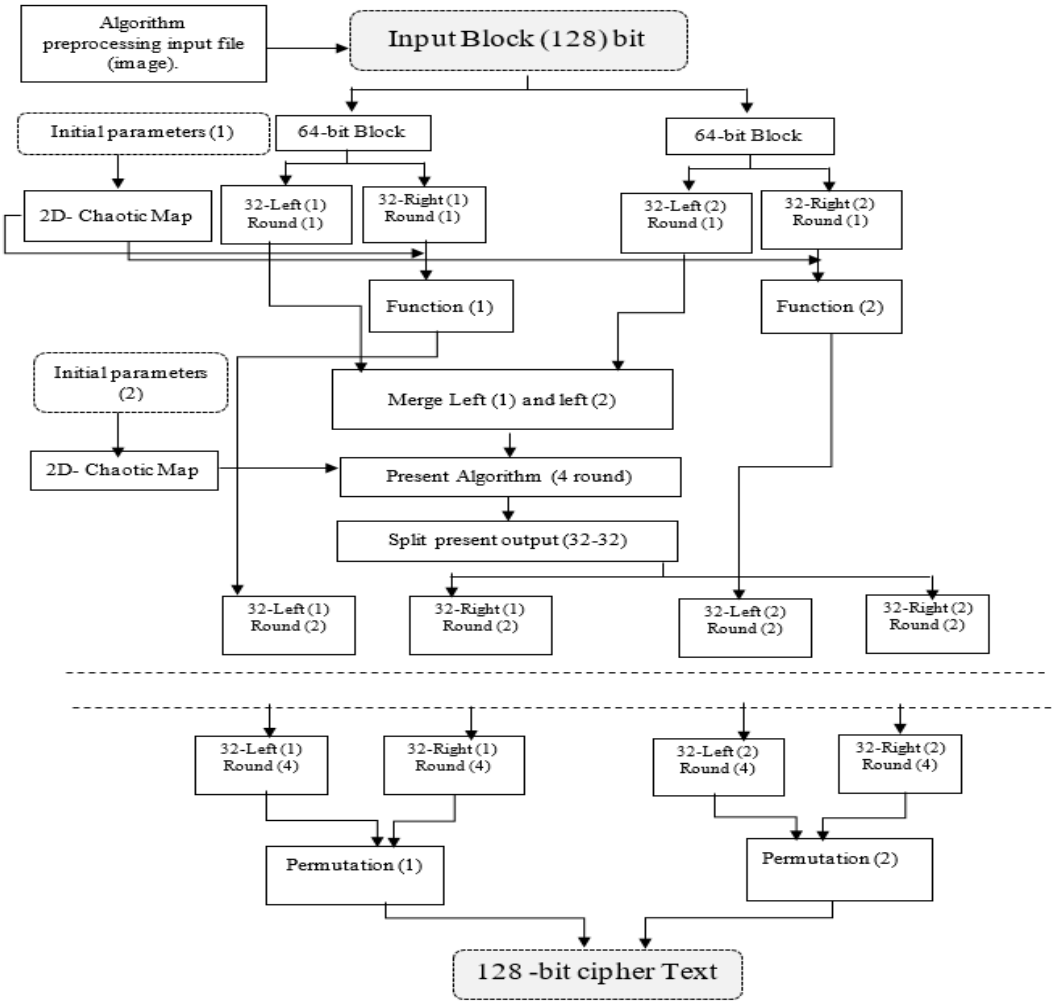


Fig 3. Hybrid algorithm (S-DES-Present) Sequence

Table 3. Encryption Hybrid Sequential

Algorithm (2) Encryption Hybrid algorithm (S-DES- present)

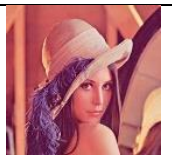
Input: the block is 128 bits, Key_1 , Key_2
Output : Cipher text is 128 bits
step1: initial parameter ,S-Box Table ,P-player for present ,
Table Initial Permutation (IP), the Table Expansion box (E), Table , Permutation Box (p) in f-function, Table Permutation boxIP^{-1}.
step2: Apply Algorithm preprocessing image.
step3: For i ← 1 to 4
3:1 Apply Algorithm Encryption (DES) Sequential // Apply Algorithm twice.
3:2 Block presnt ← to merge (left DES one , left DES two)
3:3 out put the presnt ← Apply the encryption Present Algorithm // Split block into two 32 bit.
3:4 Right one i ← $Right\ DES1 \oplus$ out put the presnt algorithm left one 32 bit.
3:5 Right two i ← $Right\ DES2 \oplus$ out put the presnt algorithm left two 32 bit .
3:6 End for i
step4: Cipher1 ← Swap (out put the presnt left one 32 bit ,Right one).//Apply Table IP^{-1}
4:1 Cipher 2 ← Swap (out put the presnt left two 32 bit ,Right two).// Apply Table IP^{-1}
step6: Cipher text ← Combine (Cipher 1, Cipher 2)
step7: End.

To decrypt the algorithm, we apply the reversal of the steps. We enter the 128-bit encrypted block and reverse the rounds.

5. SIMULATION RESULT

Hybrid algorithm (S-DES_Presnt) performance is evaluated by measuring stags-image quality. Histogram, UACI, NPCRT, MAE, MSE, Peak Signal to Noise Ratio (PSNR) and SSIM are used to assets the stage-image quality. Note that the Hybrid algorithm (S-DES_Presnt) has been performed using python programming language Visual Studio Code, Windows-11 pro operating system has been used to perform the experiments using the laptop computer processor: Intel(R), Core(TM) i7-1075H CPU @ 2.60GHz, 2.59 GHz, and (16.0 GB) RAM. In all the experiments, 256×256 color image is used as the host image. The data for the set on which the tests are being performed is, as the table (4).

Table 4. Data set image

number	1	2
Image		
Name	Baboon	Lena

5.1 Histogram Analysis

One important statistic to consider when evaluating the suggested system is the histogram analysis. It is shows image brightness and contrast, Peaks in the histogram indicate high-intensity areas, which can help explain the image. Figure (4) shows the histograms Analysis of the two images.

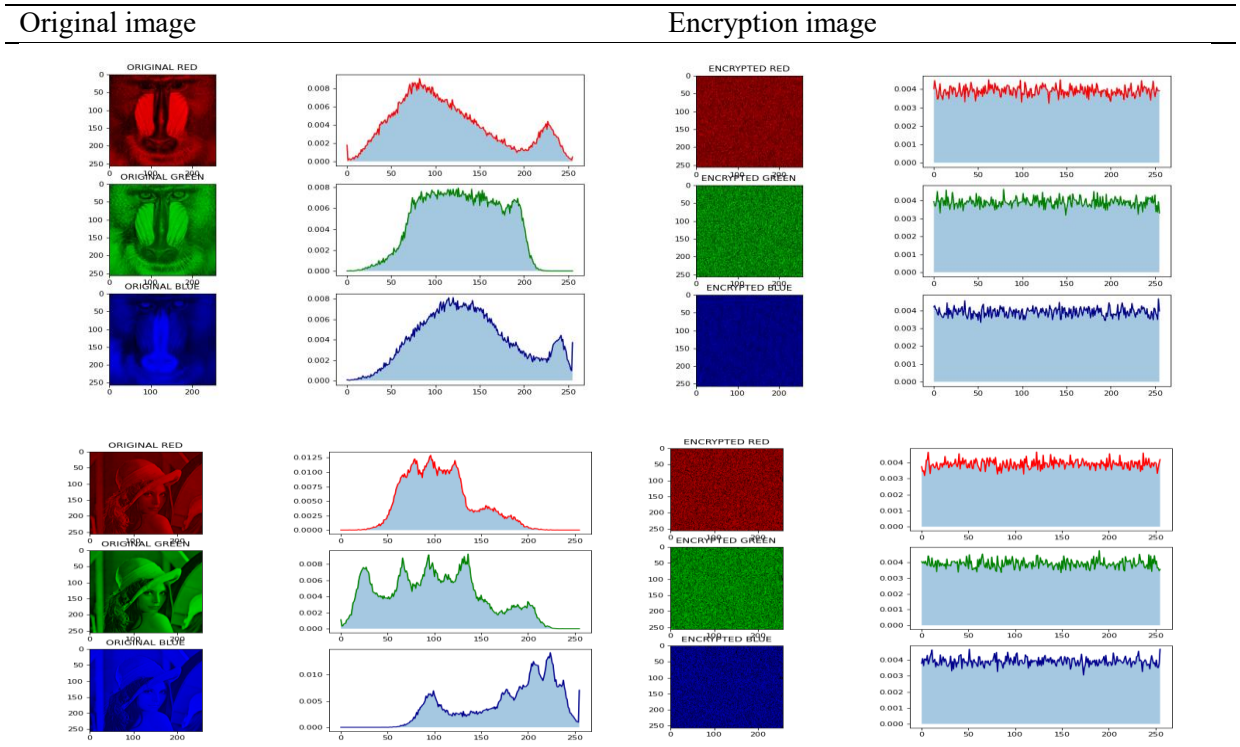


Fig 4. Histogram Analysis for (S-DES-Present)

The result the Figure 4 It contains a different distribution of RGB in the original image and a uniform distribution of RGB in the encrypted image. This indicates the strength of the encrypted image in the face of various statistical attacks.

5.2 The Correlation Coefficient Test

The correlation coefficient test measures the strength and direction of a linear relationship between two variables. The Pearson correlation coefficient (r) is the most common. The Pearson correlation coefficient measures the linear relationship between two variables and ranges from -1 to 1.the value (1) is a perfect positive linear relationship =, the value (-1) is a perfect negative linear relationship. And r = 0 means there is no linear relationship. As shown in the Figure 5 and table (5).

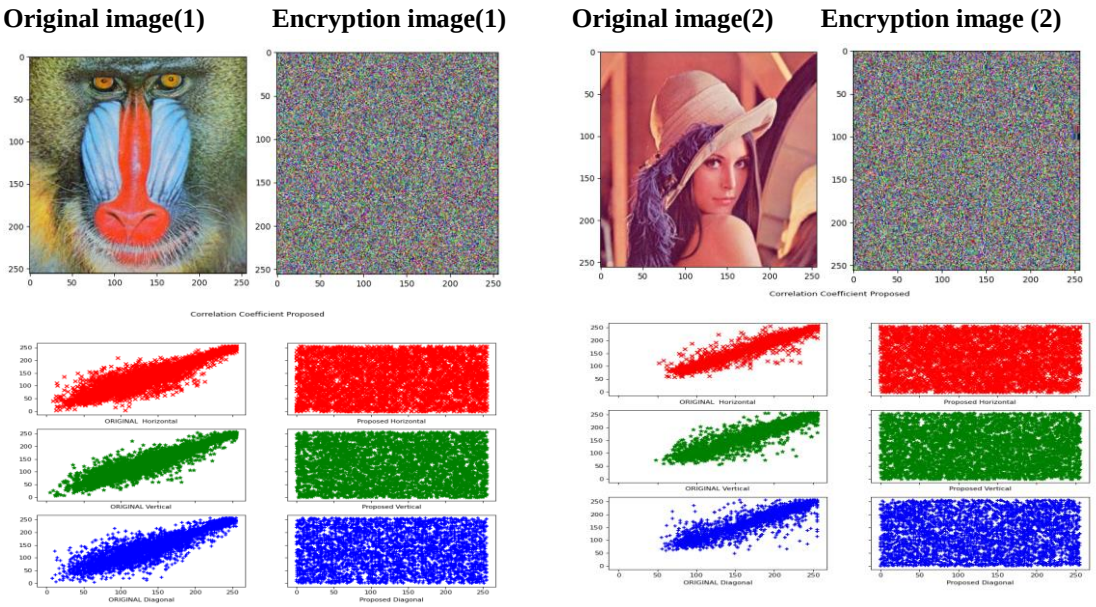


Fig 5. Correlation Coefficient Analysis for (S-DES-Present)

The result Figure 5 All three correlation coefficients (horizontal, vertical, diagonal) do not have any linear relationship between them.

Table 5. Correlation Coefficient Analysis

Image	Horizontal correlation	Vertical correlation	Diagonal correlation
1	0.002574	-0.035860	-0.003017
2	-0.00941	0.00579	0.014684

The result Table 3. All three correlation coefficients (horizontal, vertical, diagonal) close the near zero , This indicates that the encryption process distributed the pixels randomly and eliminated linear relationships to successfully complete the encryption process against statistical attacks.

5.3 The Analysis of Information Entropy

A higher entropy value suggests that the data is more uncertain or random. A lower entropy value suggests a higher degree of predictability or less uncertainty. As shown in the table (6).

Table 6. Information Entropy

Image	Entropy original	Entropy encryption
1	7.69907	7.99917
2	7.773088	7.998954

The result is that all the encryption images are close to the 8-bit value, and this indicates that the distribution of gray values has increased, which cannot reach the expectation level.

5.4 The NPCR (Number of Pixel Changes Rate) and UACI (Unified Average Changing Intensity).

NPCR calculates the percentage of pixel differences between the original and encrypted images.UACI calculates the average intensity change for pixels between the original and encrypted images. As shown in the table (7).

Table 7. Number of Pixel Changes Rate, Unified Average Changing Intensity

Image	1	2	Image	1	2
NPCRT	99.4156%	99.4965%	UACI	19.9347%	22.0046%

The result is that the NPCR test values are high in percentage of the rate of change in the number of pixels of the encoded image when there is only one pixel of the original image. The UACI test values are low in relation to the average change in pixel density between the original images and the encrypted images. It is generally a good idea to have higher NPCR values and lower UACI values to resist differential attacks.

5.5 The Mean Square Error (MSE) and the peak signal-to-noise ratio test (PNSR) And Mean Absolute Error (MAE).

MSE measures the average squared difference between the original and reconstructed pixels. A high MSE indicates better image quality because it shows high differences between original and reconstructed images. PSNR is the logarithmic ratio of the image's maximum intensity (255 for an 8-bit image) to the square root of the MSE. . (MAE) measures the average absolute difference between original and reconstructed pixels. As shown in the table (8).

Table 8. Test (MSE), Test (PNSR), and Test (MAE)

Image	1	2
MSE	8212.705699	8945.715332
PNSR	8.985941	8.614653
MAE	74.70636	77.625417

The result is that the MSE and MAE values are very high, which means that there is a difference between the original color image and the encryption color image, and the PSNR values are low, which means that the encryption quality is good.

5.6 Structural Similarity Index (SSIM)

A metric called the Structural Similarity Index (SSIM) is used to compare two images. Compared to basic pixel-related comparisons, it provides a more thorough evaluation by measuring both structural information and luminance comparison. The SSIM value is a number between -1 and 1, where 1 denotes total similarity, 0 no similarity, and -1 complete dissimilarity. As shown in the table (9).

Table 9. Structural Similarity Index

Image	1	2
SSIM	0.023701	0.016207

The result all values are close to zero, which means that there is no similarity between the original image and the encryption image.

5.7 Encryption and Decryption Time for the Proposed Algorithm

In order to determine how long it takes for algorithms to encrypt and decrypt data, there are a number of factors that the algorithms must take into consideration. The complexity of the algorithm, the length of the key, the capabilities of the hardware, and the amount of data that is being processed are all factors that are considered in this context. As shown in the table (10).

Table 10. Encryption and Decryption Time

Image	1	2
Encryption	14.32599663734436	14.391594886779785
Decryption	13.12599663734436	14.131337213516235

The result is that the proposed algorithm(S-DES-Present) takes a short time to encryption the color image and d encryption the color image.

5.8 Memory Usage

Memory usage refers to how much RAM the proposed algorithm uses. Software performance and efficiency depend on memory usage. Below is memory usage table 11.

Table 11. Memory Usage

Image	1	2
Memory	47.5	48.3

The result is that the proposed algorithm(S-DES-Present) takes a less the resource the using memory.

5.9 Randomness NIST Tests

NIST offers the "NIST Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications." These tests are widely used to evaluate sequence randomness and random or pseudorandom number generator quality-values in NIST randomness tests indicate how well data matches observed expected distribution. As shown in the table (12).

Table 12. Randomness NIST Tests

Type of Test	P-Value	State
Frequency (Monobit) Test	0.8512686236882057	Random
Frequency Test within a Block	0.4180195006078763	Random
Runs Test	0.15072850226381374	Random
Test for the Longest Run of Ones in a Block	0.7954494057112036	Random
Binary Matrix Rank Test	0.7914340912155237	Random
Discrete Fourier Transform (Spectral) Test	0.7307533347220032	Random
Non-overlapping Template Matching Test	0.6311687799358652	Random
Overlapping Template Matching Test	0.30447390124332896	Random
Linear Complexity Test	0.4813938576772515	Random

Serial Test	0.6453372179692912	Random
Approximate Entropy Test	0.5595071742079318	Random
Cumulative Sums Test (Forward)	0.9172054054882282	Random
Cumulative Sums Test (Backward)	0.7604343455645042	Random
Random Excursions Test (+1)	7.526315789473685	Random
Random Excursions Variant Test(+1.0)	0.06104468950598921	Random

the result , p-values >= 0.001 indicate High value string random 99.9% , while p-values< 0.001 indicate low value indicate that it deviates randomness 99.9% .

6. COMPARISON

The proposed algorithm is faster in encryption and decryption compared to the original des algorithm.as the show table (13).

Table 13. Time Encryption and Decryption Original DES Algorithm

Image	2	1
Encryption	24.99960899353027	24.375625133514404
Decryption	24.324954986572266	24.19103693962097

7. CONCLUSION

The algorithm (S-DES-Present) provides high speed and security for protect the image from unauthorized persons. The algorithm (S-DES-Present) uses a 2D chaotic system for the dynamism, unpredictable nature, and randomness of key creation. The key space($2^{232.4}$ it's so large and passes every test conducted by NIST that it cannot be broken by brute force. The color image quality is evaluated using the following standards: The NPCR values 99.4156%, 99.4965%. The indicated encryption algorithm changes the number of pixels between the original image and the encrypted image significantly. A UACI value 19.9347%, 22.0046% below 30.909% it indicates that the average pixel density is low between the original image and the encrypted image. This is evidence that the encrypted image is not subject to any modification. All test values result Correlation coefficient analysis Close to zero value for all three correlations horizontal,vertical,diagonal ; for example, image number 1: horizontal: 0.002574, vertical: -0.035860, diagonal: -0.003017. Encryption works well when the correlation coefficient is small, close to zero. In information entropy encryption algorithm has high data randomness and unpredictability, which is desirable. From the original 7.773088 to the encrypted image 7.998954 serves as evidence near the value 8 bit. The histogram of the encrypted image appears at one frequency, while the original image shows highs and lows. This is evidence of the complete difference between the original image and the encrypted image. The MES has high values 8212.705699, 8945.715332, which means the restored image is different from the original, and the PNSR has low values between 8.985941, 8.614653, indicating encryption color image good quality. The value test SSIM is 0.023701, 0.016207 It indicates that there is no similarity in structure between the encrypted image and the original image. With respect to the execution time, the result is that the encryption and decryption times of the proposed algorithm(S-DES-Present) decrease, compared to the decryption and encryption times of the original DES algorithm. The findings demonstrate the efficacy and feasibility of our encryption algorithm for real-world implementations. Essential for the purpose of insurance correspondence. The algorithm's encryption and decryption times are sufficiently rapid, thereby making it suitable for situations that require processing in real time or very close to real time.

8. ACKNOWLEDGEMENT

Al Mustansiriyah University's computer science department and college of education provided support for this research.

9. REFERENCE

[1] R. H. AL-Hashemy and S. A. Mehdi, "A new algorithm based on magic square and a novel chaotic system for image encryption," *J. Intell. Syst.*, vol. 29, no. 1, pp. 1202–1215, 2019.

[2] H. R. Shakir, S. A. A. Mehdi, and A. A. Hattab, "Chaotic-DNA system for efficient image encryption," *Bull. Electr. Eng. Informatics*, vol. 11, no. 5, pp. 2645–2656, 2022.

[3] A. A. Rashid and K. A. Hussein, "Image encryption algorithm based on the density and 6D logistic map.," *Int. J. Electr. Comput. Eng.*, vol. 13, no. 2, 2023.

[4] K. A. Hussein, S. A. Mahmood, and M. A. Abbass, "A New Permutation-Substitution Scheme Based on Henon Chaotic Map for Image Encryption," in *2019 2nd Scientific Conference of Computer Sciences (SCCS)*, IEEE, 2019, pp. 63–68.

[5] U. Erkan, A. Toktas, S. Enginoğlu, E. Akbacak, and D. N. H. Thanh, "An image encryption scheme based on chaotic logarithmic map and key generation using deep CNN," *Multimed. Tools Appl.*, vol. 81, no. 5, pp. 7365–7391, 2022.

[6] H. Najm, H. K. Hoomod, and R. Hassan, "A proposed hybrid cryptography algorithm based on GOST and salsa (20)," *Period. Eng. Nat. Sci.*, vol. 8, no. 3, pp. 1829–1835, 2020.

[7] Z. M. J. Kubba and H. K. Hoomod, "Modified PRESENT Encryption algorithm based on new 5D Chaotic system," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, 2020, p. 32023.

[8] S. S. Siddique and N. S. Fatima, "Digital File Rights Management System Using Blockchain," *Procedia Comput. Sci.*, vol. 215, pp. 309–320, 2022.

[9] U. Zia *et al.*, "Survey on image encryption techniques using chaotic maps in spatial, transform and spatiotemporal domains," *Int. J. Inf. Secur.*, vol. 21, no. 4, pp. 917–935, 2022.

[10] E. Fernando, D. Agustin, M. Irsan, D. F. Murad, H. Rohayani, and D. Sujana, "Performance comparison of symmetries encryption algorithm AES and DES with raspberry pi," in *2019 International Conference on Sustainable Information Engineering and Technology (SIET)*, IEEE, 2019, pp. 353–357.

[11] S. D. Putra, M. Yudhiprawira, S. Sutikno, Y. Kurniawan, and A. S. Ahmad, "Power analysis attack against encryption devices: a comprehensive analysis of AES, DES, and BC3," *TELKOMNIKA (Telecommunication Comput. Electron. Control.*, vol. 17, no. 3, pp. 1282–1289, 2019.

[12] A. Gambhir, Khushboo, and R. Arya, "Performance analysis and implementation of DES algorithm and RSA algorithm with image and audio steganography techniques," in *Computing, Communication and Signal Processing: Proceedings of ICCASP 2018*, Springer, 2019, pp. 1021–1028.

[13] K. Patel, "Performance analysis of AES, DES and Blowfish cryptographic algorithms on small and large data files," *Int. J. Inf. Technol.*, vol. 11, no. 4, pp. 813–819, 2019.

[14] Y. Wu and X. Dai, "Encryption of accounting data using DES algorithm in computing environment," *J. Intell. Fuzzy Syst.*, vol. 39, no. 4, pp. 5085–5095, 2020.

[15] O. Laia and E. M. Zamzami, "Analysis of combination algorithm data encryption standard (DES) and Blum-Blum-Shub (BBS)," in *Journal of Physics: Conference Series*, IOP Publishing, 2021, p. 12017.

[16] B. Xing, D. Wang, Y. Yang, Z. Wei, J. Wu, and C. He, "Accelerating DES and AES algorithms for a heterogeneous many-core processor," *Int. J. Parallel Program.*, vol. 49, no. 3, pp. 463–486, 2021.

[17] M. Barhoush, B. Abed-Alguni, R. Hammad, M. Al-Fawa, and R. N. Hassan, "DES22: DES-based algorithm with improved security," *Jordan J Comput Inf Technol*, vol. 8, no. 01, pp. 22–33, 2022.

[18] A. Mihalkovich, M. Levinskas, and P. Makauskas, "MPF based symmetric cipher performance comparison to AES and TDES," *Math. Model. Eng.*, vol. 8, no. 2, pp. 15–25, 2022.

[19] H. K. A. Alsuwaiedi and A. M. S. Rahma, "A new modified DES algorithm based on the development of binary encryption functions," *J. King Saud Univ. Inf. Sci.*, vol. 35, no. 8, p. 101716, 2023.

[20] A. Wiemers and J. Mittmann, "Improving recent side-channel attacks against the DES key schedule," *J. Cryptogr. Eng.*, vol. 13, no. 1, pp. 1–17, 2023.

[21] R. Banoth and R. Regar, *Classical and Modern Cryptography for Beginners*. Springer, 2023.

[22] M. Yunus, I. S. Sakkinah, U. E. Rahmawati, A. Deharja, and M. W. Santi, "File Security Design in Electronic Health Record (EHR) System with Triple DES Algorithm (3DES) at Jember Family Health Home Clinic," *Int. J. Heal. Inf. Syst.*, vol. 1, no. 1, pp. 1–8, 2023.

[23] P. K. Tiwari, V. Choudhary, and S. R. Aman, "Analysis and Comparison of DES, AES, RSA Encryption Algorithms," in *2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, IEEE, 2022, pp. 1913–1918.

[24] H. Alabdulrazzaq and M. N. Alenezi, "Performance evaluation of cryptographic algorithms: DES, 3DES, blowfish, twofish, and threefish," *Int. J. Commun. Networks Inf. Secur.*, vol. 14, no. 1, pp. 51–61, 2022.

[25] M. Akbar, I. Ahmad, and T. Regula, "Study and improved data storage in cloud computing using cryptography," *Int. Res. J. Adv. Sci. Hub*, vol. 3, no. Special Issue ICOST 2S, pp. 94–99, 2021.

[26] D. Chowdhury *et al.*, "DeCrypt: a 3DES inspired optimised cryptographic algorithm," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 5, pp. 4745–4755, 2023.

[27] X. Dong, S. Yan, and C. Duan, "A lightweight vehicles detection network model based on YOLOv5," *Eng. Appl. Artif. Intell.*, vol. 113, p. 104914, 2022.

[28] M. Hussam, "New lightweight hybrid encryption algorithm for cloud computing (LMGHA-128bit) by using new 5-D hyperchaos system," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 10, pp. 2531–2540, 2021.

[29] N. Katuk and I. R. Chiadighikaobi, "An Enhanced Block Pre-processing of PRESENT Algorithm for Fingerprint Template Encryption in the Internet of Things Environment," *Int. J. Commun. Networks Inf. Secur.*, vol. 13, no. 3, 2022.

[30] J. Bahrami, M. Ebrahimabadi, J.-L. Danger, S. Guilley, and N. Karimi, "Leakage power analysis in different S-box masking protection schemes," in *2022 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, IEEE, 2022, pp. 1263–1268.

[31] T. K. Goyal, V. Sahula, and D. Kumawat, "Energy efficient lightweight cryptography algorithms for IoT devices," *IETE J. Res.*, vol. 68, no. 3, pp. 1722–1735, 2022.

[32] Y. Yao, M. Yang, P. Kiaei, and P. Schaumont, "Dimming Down LED: An Open-source Threshold Implementation on Light Encryption Device (LED) Block Cipher," *arXiv Prepr. arXiv2108.12079*, 2021.

[33] M. Usman, "Lightweight encryption for the low powered iot devices," *arXiv Prepr. arXiv2012.00193*, 2020.

[34] S. A. Mahmood, K. A. Hussein, Y. N. Jurn, and E. A. Albahrani, "Parallelizable cipher of color image based on two-dimensional chaotic system," *Indones. J. Electr. Eng. Comput. Sci*, vol. 18, no. 1, pp. 101–111, 2019.