

Development of playfair cryptosystem based on generation a multi-dimensional key matrix

Mustafa Dhiaa Al-Hassani, Methaq Talib Gaata

Department of Computer, College of Science, Mustansiriyah University, Baghdad, Iraq

Article Info

Article history:

Received Oct 21, 2022

Revised Dec 16, 2022

Accepted Jan 9, 2023

Keywords:

Chaotic map

Cryptography

Key matrix generation

Playfair encryption

Polygram cipher

ABSTRACT

Playfair is considered as one of the classical encryption symmetric methods, it has a limitation of using just 5×5 matrix, which means only 25 English letters could be represented. In this work, a 2D and 3D method is adopted as an expanded matrix that encompass the overall American standard code for information interchange (ASCII) codes in a permuted manner for all symbols of any language. Any sort of the multi-dimensional matrix will enhance the security by increasing the complexity on the attacker to try $256!$ patterns of keys probabilities instead of $25!$. The key-matrix is generated from the chaotic maps for some control parameters as patterns of non-repeating random numbers from 0 to 255 equivalent to their ASCII code values. The security of the proposed method not rely only on the number of key probabilities, but exceed that to: matrix dimensionality, encryption/decryption algorithms, initial chaotic parameters, and key-matrix values permutation. The efficiency of the proposed cryptosystem has been investigated when tested on 784 samples according to security measurements in which the obtained number of pixels change rate (NPCR) (99.609) is very close to the ideal value, while the correlation plotting close to zero (0.00058) and entropy near from 8 (7.9998).

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Mustafa Dhiaa Al-Hassani

Department of Computer, College of Science, Mustansiriyah University

Baghdad, Iraq

Email: dr_mdhiaa77@uomustansiriyah.edu.iq

1. INTRODUCTION

Information security has become a major issue that affects everyone who uses the Internet to ensure the integrity of data related to different sectors of life including health, scientific, commercials, financials, and security aspects that are being shared between any entities. Institutions, corporations, and governments all over the world are facing increasing numbers of cyber-attacks. Therefore, the data must be secure enough prior to transmission using one of the efficient encryption techniques [1]–[3]. Cryptographic operations are very important to invalidate the role of interception threats exercised by illegal third parties since encryption converts secret data into indistinguishable patterns. Although the traditional data encryption standard (DES), Rivest-Shamir-Adleman (RSA), and advanced encryption standard (AES) have achieved good results in texts encryption, it is not the practical ideal choice for encrypting image, audio, and video files due to their containing a large amount of data that needs high computation burdens [1], [3]–[5]. For more than a decade, image encryption including scrambling-diffusion structures based on chaotic maps has attracted the interest of researchers and professionals since it addresses the slowness of most encryption techniques. Thus, diffusion processes will enhance the robustness of the algorithms and increase its resistance against different attacks [2]–[4], [6]–[10].

Previous related works to the cryptosystem targets for this paper are demonstrated as: according to Arroyo *et al.* [8], a 16×16 dynamic matrix with multidimensional element-in-grid sequencer (MEGS)-based modified playfair technique is proposed including: rotation, shifting, rolling, and crossover matrix operations to be applied for password security or image steganography. The results indicate the ability of the suggested method to surpass avalanche effect when tested on various length plaintexts (up to 1,000 characters) and brute force attack analysis. According to Wang *et al.* [2], an encryption scheme for gray and color images is presented based on a cascaded modulated chaotic system (CMCS) for generating key and block scrambling-diffusion mechanism. To reduce the correlation between contiguous pixels, chaotic sequences is used to achieve intra-block permutation. According to Alghamdi *et al.* [4], a secure lightweight image encryption system is implemented based on logistic map and pre-shared SHA-2 key image, which is appropriate for real-time applications and devices. According to Chai [7], the plain image is decomposed into multi-planes of 8 layers, then the encrypted image is obtained after performing in sequence diffusion and scrambling planes operations. According to Laiphrakpam and Khumanthem [11], a robust image encryption method using elliptic curves and chaotic systems is introduced as a result of mixing the random sequence with the pixels value of the scrambled image using Arnold transform. According to Ye *et al.* [12], a hyper-chaotic system is presented that have the ability to encrypt two images simultaneously, and thus reduce the execution time of encryption. Image compression using compressive sensing is performed prior to encryption based on public key elliptic curve method. According to Luo *et al.* [13], an image encryption technique inspired by deoxyribonucleic nucleic acid (DNA) sequence with chaotic systems is presented. The parameters of a 2D logistic sine map and 1D chaotic system are determined by hashing the original image with a 256-bit key. Chaotic sequences are used to derive the DNA cryptosystem rule matrix. Then, rearrangement of rows and columns are performed with inter/intra DNA-plane permutation. The cipher image is formed by performing XOR processes between the DNA key matrix and the permuted original image.

This paper aims to develop the traditional playfair cryptosystem, by eliminating its limitations that deals with just 25-English letters in 2D matrix except 'j' character and in non-case sensitive mode, to a more secure system using an extended multi-dimensional key matrix that encompass all the American standard code for information interchange (ASCII) codes, which will allow to deal with all possible languages whether in 2D or 3D matrix spaces. The key matrix is generated from the chaotic maps for some control parameters processed in a permuted manner to produce patterns of non-repeating random numbers from 0 to 255 which equivalent to ASCII codes values. Consequently, it will increase the immunity of security against system attackers to try $256!$ patterns instead of testing only $25!$ possible keys.

2. METHOD

2.1. The traditional playfair cipher

The British war office utilized the polygram substitution playfair cipher, discovered by Charles Wheatstone and playfair baron in 1854, to encrypt secret messages in pairs of letters until the beginning of the 20th century. Users tend to use an easy-to-remember phrase to obtain the key matrix, from which the repeating characters in the key phrase are removed and append to them the unallocated letters in alphabetical order. The key is formulated in 2D 5×5 matrix (25 entries from English letters except the character 'J' is removed) and thus it offer up to $25! = 15,511,210,043,330,985,984,000,000$ possible keys. The secret message must be rearranged somewhat (i.e., J's must be replaced with I's, divide the message into groups as pairs of characters, don't allow duplicate letters in the same pair-if they happen insert the letter Z between them, add Z at the end of message if an odd pair is occur) prior of using the playfair cipher encryption rules. On the other hand, the decryption rules is merely the reverse process of encryption, such that, each character is altered by the one on its left in the extended key if the pairs of characters are in the same row of the key, else replace each of them by the upper one of it if the two characters located in the same column, otherwise follow exactly the same intersection rule in encryption [6], [14]–[17].

2.2. Chaotic-map

Chaos theory exhibits several suitable features (i.e., generate aperiodic and unpredictable random sequences, fast calculations, very sensitive to initial conditions) that make it to be used efficiently in real-time modern encryption systems and it ensure secure communications among any entities. Chaos random sequences are formed via handling iterative equations as shown in (1). Such that, a few minor changes in the initial parameters (r and x) will produce far and wide changes in outcome sequences [2]–[4], [9], [18].

$$x_{n+1} = r * x_n(1 - x_n), \text{ where } r \in [0, 4], x \in [0, 1], n = 0, 1, 2, \dots \quad (1)$$

Logistic map features will be enabled when the value of initial chaotic parameter (r) falls between 3.57 to 4 and $x_0 \geq 0.5$, moreover x_1, x_2, \dots is obtained when replacing the values of n into (1) [15], [18], [19].

2.3. The proposed modified playfair cryptosystem

The proposed modified cryptosystem must be settled into sender and recipient PCs for the sake of conducting encryption and decryption processes in both sides. The block diagram of encryption sub-system from the sender point of view is depicted in Figure 1 and implemented as in Algorithm 1, which consist of two main processing components (processing the key which lead to the generation of a full 2D KeyMatrix[,] or 3D KeyMatrix[, ,] of permuted ASCII values; and processing the secret data that produce the composition of SecretPairs[] sequence) prior to encryption processes stated in Algorithm 2. The sequence of steps to follow when encrypting a secret file of any multimedia type and extension are illustrated in Algorithm 1.

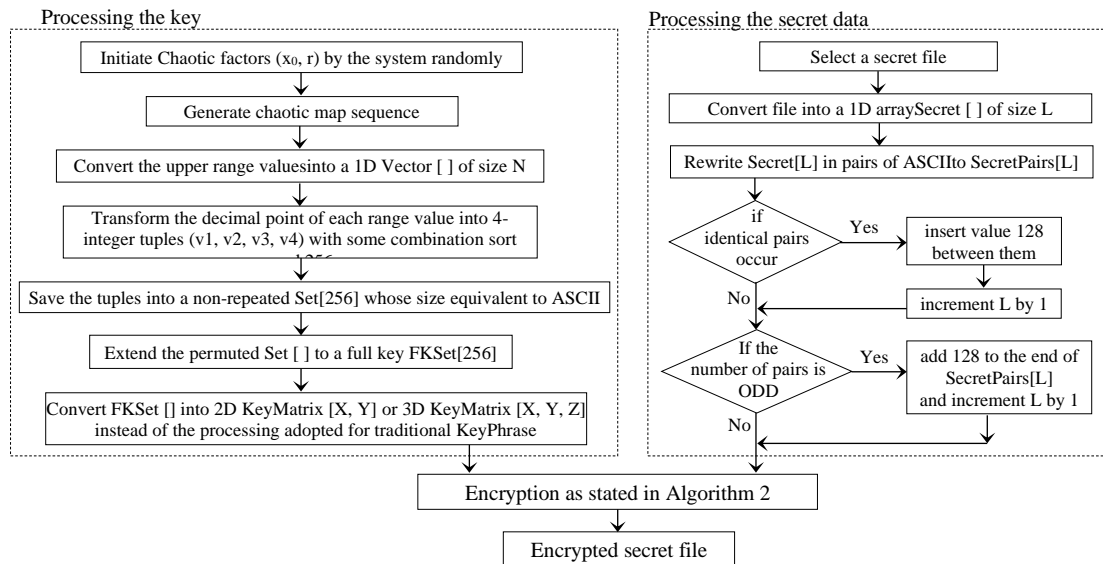


Figure 1. The flowchart of the proposed playfair encryption subsystem

Algorithm 1. The proposed modified playfair sender encryption subsystem

Input: secret file.

Output: encrypted secret file.

- a. The system initiate chaotic parameters (x_0, r) randomly, $0.5 \leq x_0 \leq 1.0$ and $3.57 \leq r \leq 4.0$.
- b. Generate chaotic sequence values and ranges.
- c. Convert the upper range values into a 1D vector[] of size N.
- d. Initialize an empty array of non-repeated values called set[] of size 256 equivalent to ASCII.
- e. Initialize $i \leftarrow 0$, $k \leftarrow 0$.
- f. While ($i < N$), otherwise goto step j).
- g. Transform the decimal point of vector[i] into 4-integer tuples (v_1, v_2, v_3, v_4) whose values range from 0 to 255, where v_1 represent the first decimal point, v_2 the first 2-decimal point, v_3 the first 3-decimal point mod 256, v_4 the first 4-decimal point mod 256.
- h. foreach value (v_1, v_2, v_3, v_4) not occurred in set[], save it into set[k] and increase k by 1;
- i. increment i by 1 and goto step f).
- j. if ($k=256$) goto step o).
- k. $j \leftarrow 0$
- l. While ($j < 256$)
- m. Extend the permuted set[] to a full key FKSet[] by appending the residual ASCII numbers when j not occurred in the set[] in alphabetical order, such that $FKSet[k] \leftarrow j$ and increase k by 1;
- n. increment j by 1 and goto step l).
- o. Convert FKSet[] of size 256 into multidimensional 2D KeyMatrix[,] whose dimensions are X, Y or 3D KeyMatrix[, ,] whose dimensions are X, Y and Z, as shown in Figure 2.
- p. The sender select a secret file of any type and convert it into a 1D array called secret[] of size L.
- q. Rewrite secret[] elements in pairs of ASCII numbers into SecretPairs[] by inserting the value 128 between each identical pairs and increment L by 1.
- r. If the number of pairs is ODD then also insert the value 128 to the end of SecretPairs[] and increment L by 1.
- s. The system perform encryption processes between the KeyMatrix and SecretPairs as stated in Algorithm 2.
- t. The encrypted secret file is generated including the added header information (16-bytes) about encryption parameters $(x_0, r, L, \text{KeyMatrix dimensions } (X, Y, Z), M)$ and the cipher data.

Algorithm 2. The proposed encryption rules for different keymatrix dimensions

Input: KeyMatrix of X, Y, Z dimensions, 1-D full key FKSet[] of size 256, SecretPairs[] of size L.

Output: Cipher data of size L.

- a. Initialize $p \leftarrow 0$.
- b. Get the current pair of ASCII values from the SecretPairs[] array from locations p and $p+1$.
- c. If the pair located in the identical row of the same KeyMatrix then each ASCII value position is substituted by the one on its right in the KeyMatrix shifted by its pair index mod 5, and goto step l).
- d. If $((Z=0 \text{ and } M>1) \text{ or } (Z>1))$ then each ASCII value position is replaced by the one on its right in that KeyMatrix shifted by its pair index mod 5, and goto step l).
- e. If the pair lie in the EXACT position but in different 2D KeyMatrix (i.e. $M>1$), then follow just the replacement according to the rules stated in Table 1, and goto step l).
- f. If the pair lie in the EXACT position but in different Z-axis in 3D KeyMatrix (i.e. $Z>1$), then replace each ASCII value by the value on the same position in the next dimension, and goto step l).
- g. If they lie in the same column of the same KeyMatrix then each ASCII value position is substituted by the one below it in the KeyMatrix shifted by its pair index mod 5, and goto step l).
- h. If $((Z=0 \text{ and } M>1) \text{ or } (Z>1))$ then each ASCII value position is replaced by the one below it in that KeyMatrix shifted by its pair index mod 5, and goto step l).
- i. If the pair lie in the same KeyMatrix but not in the same row nor in the same column then the first value is substituted by the value of intersecting the row of the 1st value with the column of the 2nd value, and the 2nd value is altered by the value of intersecting the row of the second value with the column of the 1st value, and goto step l).
- j. If $(M>1)$ and the pair lie in different KeyMatrix then it is replaced according to the intersection rules stated in Table 1, and goto step l).
- k. If $(Z>1)$ and the pair lie in different Z-axis of KeyMatrix then perform replacements according to the intersection rules keeping the ends of the pair in the same Z-dimension.
- l. Increment p by 2.
- m. While $(p<L)$ goto step b).
- n. Rewrite the cipher data as a 1D array called CipherData of size L.
- o. Perform the XOR operations: $\text{CipherData}[i] \leftarrow \text{CipherData}[i] \oplus \text{FKSet}[i \bmod 256]$, where $i=0$ to $L-1$.

It's worth mentioning that the multidimensional KeyMatrix is calculated in the same way by both sender and recipient according to the initial chaotic parameters (x_0, r) as indicated in Algorithm 1, where the chaotic ranges are manipulated somehow to create an array of non-repeated numbers in a permuted manner that are equivalent to ASCII code values (from 0 to 255), and therefore it can be used in various languages (like: English, French, and Arabic). The added header information consist of (16-bytes) represented as follows: 4-bytes for each initial float number (x_0) and (r) , 4-bytes for the SecretPairs length (L) , 1-byte is dedicated for each dimension value (X) , (Y) , and (Z) , 1-byte for status M that implies multiple 2D matrix. When the dimension value Z equal 0 and M equal 0, it means that the key is a 2D matrix as shown in Figure 2(a); once M is greater than 1, it refer to the number of cascaded 2D matrices with dimensions (X, Y) such that $(Z=0 \text{ and } M>1)$ as depicted in Figure 2(b). Otherwise, it refers to a 3D matrix $(Z>1)$. A 3D array is a multiple 2D matrices, that were described by using three subscripts: row, column and depth axes as illustrated in Figure 2(c). Such that the variety of 2D or 3D KeyMatrix could be represented as shown in Figures 2(a)-(c) that was extracted from the system interface according to the programming code results.

Any sort of the multi-dimensional key matrix (16×16 2D matrix, four 8×8 2D matrix, four 16×4 2D matrix, four 4×16 2D matrix, 8×8×4 | 8×4×8 | 4×8×8, 16×4×4 | 4×16×4 | 4×4×16, 32×4×2 | 32×2×4 | 4×32×2 | 4×2×32 | 2×32×4 | 2×4×32 3D matrices) will enhance the security by increasing the complexity on the attacker to try 256! patterns of keys probabilities according to different multidimensional spaces 2D or 3D matrices instead of testing only 25! possible keys in the traditional 2D 5×5 playfair mode. The system perform encryption processes between the KeyMatrix and SecretPairs as stated in Algorithm 2, taking into consideration the "variable" replacement (shift by index) of pair ends if they occur in the same row or column in encryption or decryption instead of the "static" shift by 1 to the right (during encryption) or left (during decryption) in the traditional playfair method. In addition, a new rule appears when both ends of the pair are occur in the exact coordinate position (x, y) only if the key is a 3-dimensional or several 2D matrices.

Suggested rules in Table 1 represented from R1 to R12 are of dual-purpose by both sender and recipient because it is reversible and thus it ensures the reconstruction of the exact pair in the decryption processes by the recipient. For example, the source pair in R1 which is $P(A, B)$ will be replaced by the resulted pair of intersection inside $P(C, D)$, and vice versa as shown by the rule R9. The first row represent the probability of the source KeyMatrix (A) with B, C , and D respectively in order to produce the rules R1, R2 and R3. Whereas, the next rows are dedicated in sequence for the KeyMatrix probabilities of (B) , (C) , and (D) . The security of the developed playfair method relies on a number of elements; i) the initial chaotic parameters (x_0, r)

that are selected randomly by the system; ii) the generation of chaotic sequence values and ranges; iii) range floating values manipulations into chain of integer tuples (v1, v2, v3, and v4); iv) KeyMatrix dimensions; v) KeyMatrix values generation and permutation; vi) the number of probabilities to generate the full KeyMatrix by the cryptanalyst=(256)!≈8.57×10+506 instead of trying just (25)!≈15511210043330985984000000 possible keys in the traditional mode; vii) encryption/decryption processes.

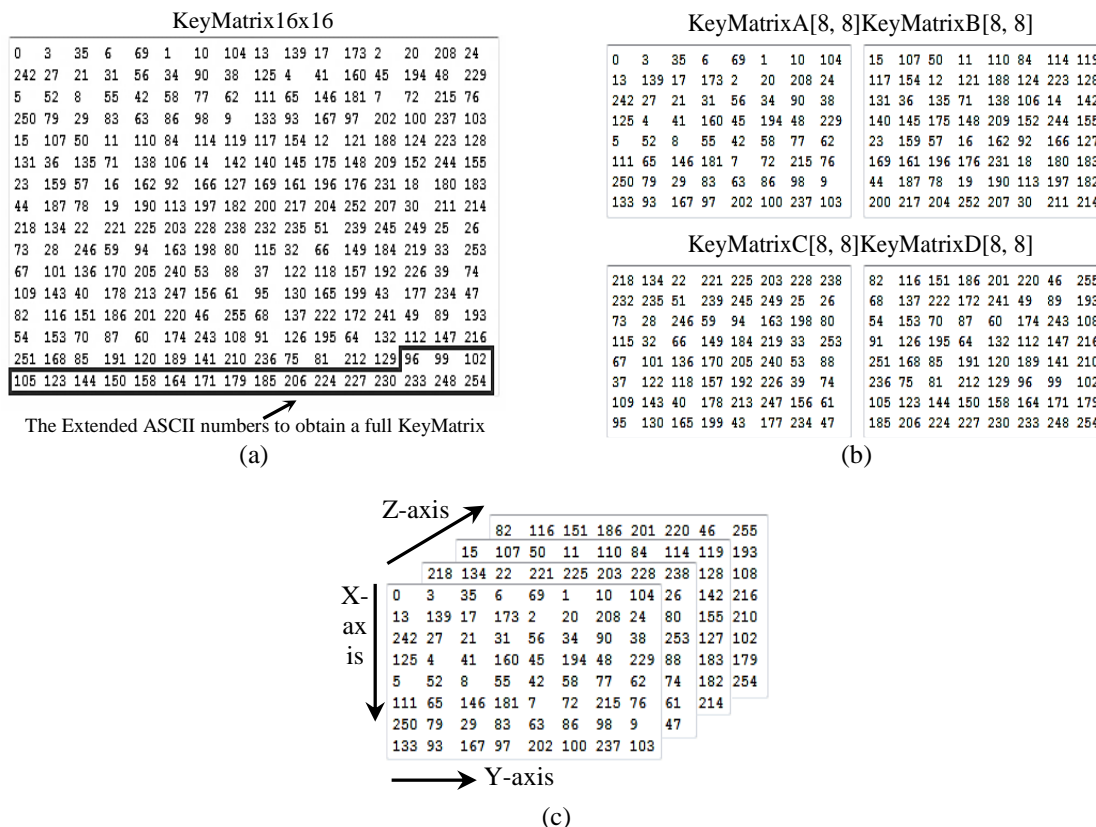


Figure 2. Samples of the full permuted key generated as a multidimensional matrix (a) 2D KeyMatrix of size [16, 16], (b) four cascaded 2D KeyMatrix of size [8, 8], and (c) 3D KeyMatrix of size [8, 8, 4]

Table 1. The proposed pair replacement rules when the intersection occur in different 2D KeyMatrix (a refer to the 1st KeyMatrix, B to the 2nd, C to the 3rd, and D to the 4th)

Source to destination	Source to destination	Source to destination	Source to destination
R1: P(A, B) → P(C, D)	R2: P(A, C) → P(B, D)	R3: P(A, D) → P(B, C)	R4: P(B, A) → P(D, C)
R5: P(B, C) → P(A, D)	R6: P(B, D) → P(A, C)	R7: P(C, A) → P(D, B)	R8: P(C, B) → P(D, A)
R9: P(C, D) → P(A, B)	R10: P(D, A) → P(C, B)	R11: P(D, B) → P(C, A)	R12: P(D, C) → P(B, A)

On the other hand, the recipient re-forms the exact secret data by reversing the decryption procedures through generating the KeyMatrix according to the initial chaotic factors (x0,r) extracted from the header of encrypted secret file and then processing the encrypted secret data as a sequence of pairs taking into consideration that decryption processes (shifting to the left rather than right in encryption if the pair ends occur in the same row, while the shift is directed to up rather than below if they occur in the same column, the intersection are the same for both encryption and decryption processes except the direction of replacement from source to destination matrix and vice versa, and follow the pair replacement rules when the intersection occur in different 2D KeyMatrix as indicated in Table 1).

3. RESULTS AND DISCUSSION

This section is concerned with the experimental results for the proposed system algorithms. A variety of metrics are used to test and analyse the proposed system's security and efficiency (such as: number of pixel change rate (NPCR), mean square error (MSE), entropy, correlation coefficient) on 784 samples of various multimedia file types and formats. Consequently, a discussion is presented according to the obtained results for further enhancements that could be made to increase the performance of proposed system [1], [4], [9], [14], [15].

3.1. Number of pixel change rate

The change rate between plain (P) and cipher (C) images corresponding pixels value which is utilized to evaluate the durability of the encryption algorithm against differential attack.

$$NPCR = \frac{\sum_{i=1}^H \sum_{j=1}^W D(i,j)}{H \times W} \times 100\% \text{ and } D(i,j) = \begin{cases} 0, P(i,j) = C(i,j) \\ 1, P(i,j) \neq C(i,j) \end{cases} \quad (2)$$

where D(i, j) refer to the bipolar network, H and W denotes image height and width respectively. The ideal NPCR value is 99.61 [3], [20]–[24].

3.2. Mean square error

Measures the amount of error or deviation in statistical models between any two samples either if they are 1D signals or 2D images. It evaluates the average squared difference between the observed and predicted values [10], [18], [23].

$$MSE = \frac{1}{N} \times \sum_{i=1}^N (x_i - y_i)^2 \quad (3)$$

where x_i is the i^{th} observed value, y_i is the corresponding predicted value, and N is the number of observations.

$$MSE = \frac{1}{H \times W} \times \sum_{i=1}^H \sum_{j=1}^W |P[i, j] - C[i, j]|^2 \quad (4)$$

where H and W denote the height and width of image respectively of the 2D matrix P and C, and P[i, j] or C[i, j] are pixel values of corresponding plain and cipher images at coordinates (i, j).

3.3. Entropy of information

Is a concept used for computing the amount of information uncertainty' and measures the randomness of encryption process.

$$H(X) = - \sum_{i=1}^n p(x_i) \times \log_2(p(x_i)) \quad (5)$$

where n refer to the number of potential events in the source and $p(x_i)$ refer to the probability distribution of each event x_i appearing in the source X. To obtain the ideal random mode (i.e., the information entropy is 8), the probability of each event must be 1/256. Accordingly, the randomness degree is very strong when the entropy of the cipher data is close to the ideal value [2], [10], [18], [19].

3.4. Correlation coefficients

Typically pixels have strong correlations and redundancy in a plain image or video files. Therefore, the quality of encryption algorithm depends on its ability to break these correlations towards it becomes close to 0 [18], [19].

$$Corr.(X, Y) = \frac{\sum_i \sum_j (x_{i,j} - \bar{x})(y_{i,j} - \bar{y})}{\sqrt{\sum_i \sum_j (x_{i,j} - \bar{x})^2 \sum_i \sum_j (y_{i,j} - \bar{y})^2}} \quad (6)$$

where Corr.(X, Y) refer to the correlation between plain image X with that of cipher image Y, while \bar{X} and \bar{Y} stand for their mean values of X and Y. The resulted correlation is determined between -1.0 to +1.0. Accordingly, the encryption and decryption processes are performed on different data samples as depicted in Figure 3, where the original image as stated in Figure 3(a) is encrypted to produce Figure 3(b). On the other side, the recipient performs decryption processes on the received data file to exactly retrieve the original secret image as shown in Figure 3(c). Similarly, the secret text denoted in Figure 3(d) is encrypted to obtain Figure 3(e), and conversely, the complete secret text shown in Figure 3(f) is restored as a result of decryption operations. It is worthwhile that Figure 3 is extracted from the interface of the system program.

From the security point of view, it is clearly indicated from the results shown in Table 2 that the proposed algorithm surpass all other systems according to its enormous secret key space, which implies infeasibility of brute-force attack. The randomness of cipher images and algorithm' performance is compared with that in previous related works are tested along with identical image samples according to NPCR, information entropy, and correlation coefficients, as indicated in Table 3. When analysing the results of the security tests shown in Table 3, it was found that the efficiency of the proposed algorithm when compared to other contemporary encryption methods which gives premium NPCR rates close to the optimal value and entropy closer to 8. Likewise, the correlation coefficients among adjoining pixels in the original image is very strong because of the high redundancy in digital images, while in the ciphered images it becomes very close to zero. The closer encrypted image correlation to 0, is the better algorithm that conduct to random optimal case. This indicates the positive effect of applying the proposed encryption algorithm in breaking the correlation coefficients between the image pixels, and thus more resistance against the attacks of statistical analysis.

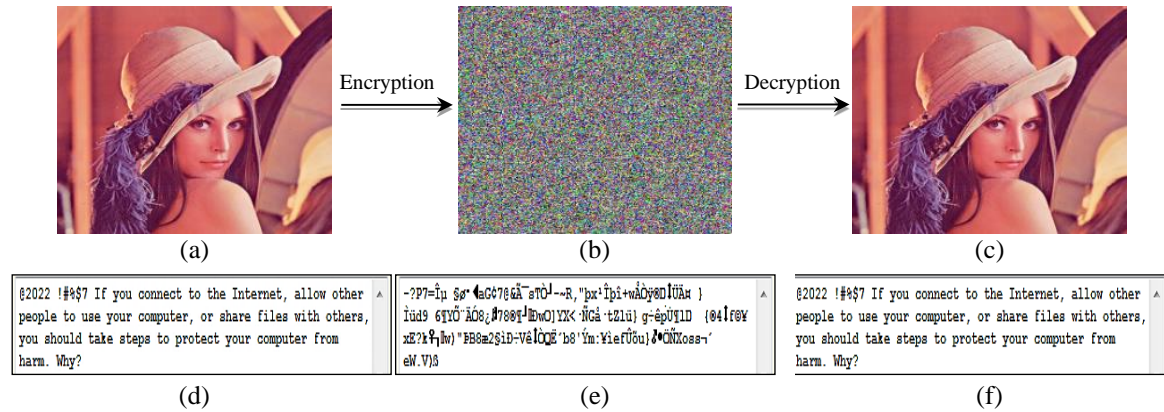


Figure 3. Simulation tests of the proposed cryptosystem for some samples (a) original Lena 256×256, (b) encrypted image (MSE=8976.45), (c) decrypted image, (d) message.txt, (e) encrypted message, and (f) reconstructed message

Table 2. Comparison with other encryption algorithms according to secret key space

Security	Ref. [2]	Ref. [3]	Ref. [14]	Ref. [18]	Ref. [25]	Ref. [26]	Our proposed encryption method
Secret key space	$1.15 \times 10^{+77}$	$9.76 \times 10^{+60}$	$9.71 \times 10^{+83}$	10^{+141}	$8.39 \times 10^{+54}$	$9.35 \times 10^{+49}$	$8.57 \times 10^{+506}$

Table 3. Performance comparison with other encryption algorithms according to Lena image of size 256×256

Ref.	NPCR (%)	Information entropy	Correlation coefficients	Ref.	NPCR (%)	Information entropy	Correlation coefficients
[1]	99.626	7.9972	+0.0029	[18]	99.62	7.9993	0.00067
[2]	99.628	7.9998	-0.0062	[27]	99.61	7.9969	-0.0032
[3]	99.603	7.9992	+0.0040	[28]	99.50	7.9992	0.0012
[5]	99.65	7.9970	-0.0014	[29]	99.59	7.9972	+0.0116
[7]	99.59	7.9993	-0.0225	[30]	99.60	7.9031	-0.0057
Our proposed encryption method	99.609	7.9998	+0.00058				

4. CONCLUSION

Due to the limited capabilities of the traditional playfair method, it becomes necessary to develop this method in order to take advantage of its capabilities to encompass all languages by including the whole ASCII codes. This paper propose to create an extended key matrix consisting of 256 cells in the form of a 2D, 3D, or multi-2D matrix whose values were generated from the chaotic map in a random order towards increasing the complexity for the attackers to test 256! possible keys instead of the 25! that was followed in the traditional method, in addition to that new algorithms and replacement/intersection rules in encryption and decryption processes were suggested. Experiments confirmed the efficacy of the proposed newly developed approach when compared with previous related works according to security measures for the same standard samples in addition to that it offers a very high secret keys space.

ACKNOWLEDGEMENTS

The authors would like to thank Mustansiriyah University (<https://www.uomustansiriyah.edu.iq/>), Baghdad-Iraq for its support in the present work.




REFERENCES

- [1] Y. Niu and X. Zhang, "A novel plaintext-related image encryption scheme based on chaotic system and pixel permutation," *IEEE Access*, vol. 8, pp. 22082–22093, 2020, doi: 10.1109/ACCESS.2020.2970103.
- [2] T. Wang, B. Ge, C. Xia, and G. Dai, "Multi-image encryption algorithm based on cascaded modulation chaotic system and block-scrambling-diffusion," *Entropy*, vol. 24, no. 8, pp. 1–23, 2022, doi: 10.3390/e24081053.
- [3] R. Ge, G. Yang, J. Wu, Y. Chen, G. Coatrieux, and L. Luo, "A novel chaos-based symmetric image encryption using bit-pair level process," *IEEE Access*, vol. 7, pp. 99470–99480, 2019, doi: 10.1109/ACCESS.2019.2927415.
- [4] Y. Alghamdi, A. Munir, and J. Ahmad, "A lightweight image encryption algorithm based on chaotic map and random substitution," *Entropy*, vol. 24, no. 10, pp. 1–25, 2022, doi: 10.3390/e24101344.
- [5] H. Zhu, L. Dai, Y. Liu, and L. Wu, "A three-dimensional bit-level image encryption algorithm with Rubik's cube method," *Mathematics and Computers in Simulation*, vol. 185, pp. 754–770, 2021, doi: 10.1016/j.matcom.2021.02.009.
- [6] W. Stallings, *Cryptography and network security: principles and practice*, 7th ed. India: Pearson Education, 2017.
- [7] X. Chai, "An image encryption algorithm based on bit level Brownian motion and new chaotic systems," *Multimedia Tools and Applications*, vol. 76, no. 1, pp. 1159–1175, 2017, doi: 10.1007/s11042-015-3088-1.
- [8] J. C. T. Arroyo, A. M. Sison, R. P. Medina, and A. J. P. Delima, "An enhanced playfair algorithm with dynamic matrix using the novel multidimensional element-in-grid sequencer (MEGS)," *International Journal of Engineering Trends and Technology*, vol. 70, no. 3, pp. 132–139, 2022, doi: 10.14445/22315381/IJETT-V70I3P215.




- [9] H. K. Zghair, H. A. Ismael, and A. A. -H. A. -Shamery, "Image scrambler based on novel 4-D hyperchaotic system and magic square with fast Walsh–Hadamard transform," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 6, pp. 3530–3538, 2022, doi: 10.11591/eei.v11i6.4339.
- [10] E. R. Arboleda, J. L. Balaba, and J. C. L. Espineli, "Chaotic rivest-shamir-adlerman algorithm with data encryption standard scheduling," *Bulletin of Electrical Engineering and Informatics*, vol. 6, no. 3, pp. 219–227, 2017, doi: 10.11591/eei.v6i3.627.
- [11] D. S. Laiphrakpam and M. S. Khumanthem, "A robust image encryption scheme based on chaotic system and elliptic curve over finite field," *Multimedia Tools and Applications*, vol. 77, no. 7, pp. 8629–8652, 2018, doi: 10.1007/s11042-017-4755-1.
- [12] G. Ye, M. Liu, and M. Wu, "Double image encryption algorithm based on compressive sensing and elliptic curve," *Alexandria Engineering Journal*, vol. 61, no. 9, pp. 6785–6795, 2022, doi: 10.1016/j.aej.2021.12.023.
- [13] Y. Luo, X. Ouyang, J. Liu, L. Cao, and Y. Zou, "An image encryption scheme based on particle swarm optimization algorithm and hyperchaotic system," *Soft Computing*, vol. 26, no. 11, pp. 5409–5435, 2022, doi: 10.1007/s00500-021-06554-y.
- [14] A. A. Maryoosh, Z. S. Dhaif, and R. A. Mustafa, "Image confusion and diffusion based on multi-chaotic system and mix-column," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 4, pp. 2100–2109, 2021, doi: 10.11591/eei.v10i4.2942.
- [15] M. D. A. -Hassani, "A novel technique for secure data cryptosystem based on chaotic key image generation," *Baghdad Science Journal*, vol. 19, no. 4, pp. 905–913, 2022, doi: 10.21123/bsj.2022.19.4.0905.
- [16] M. R. -Ousley, *Information security: the complete reference*, 2nd ed. New York, USA: McGraw Hill Professional, 2013.
- [17] M. E. Whitman and H. J. Mattord, *Principles of information security*, 7th ed. Boston, MA: Cengage Learning, 2021.
- [18] X. Chai, Z. Gan, K. Yuan, Y. Chen, and X. Liu, "A novel image encryption scheme based on DNA sequence operations and chaotic systems," *Neural Computing and Applications*, vol. 31, no. 1, pp. 219–237, 2019, doi: 10.1007/s00521-017-2993-9.
- [19] B. Yousif, F. Khalifa, A. Makram, and A. Takieldein, "A novel image encryption/decryption scheme based on integrating multiple chaotic maps," *AIP Advances*, vol. 10, no. 7, pp. 1–9, 2020, doi: 10.1063/5.0009225.
- [20] M. Zarebnia, H. Pakmanesh, and R. Parvaz, "A fast multiple-image encryption algorithm based on hybrid chaotic systems for gray scale images," *Optik*, vol. 179, pp. 761–773, 2019, doi: 10.1016/j.ijleo.2018.10.025.
- [21] T. u. Haq and T. Shah, "Algebra-chaos amalgam and DNA transform based multiple digital image encryption," *Journal of Information Security and Applications*, vol. 54, pp. 1–17, 2020, doi: 10.1016/j.jisa.2020.102592.
- [22] H. -S. Ye, N. -R. Zhou, and L. -H. Gong, "Multi-image compression-encryption scheme based on quaternion discrete fractional Hartley transform and improved pixel adaptive diffusion," *Signal Processing*, vol. 175, pp. 1–14, 2020, doi: 10.1016/j.sigpro.2020.107652.
- [23] R. Saidi, N. Cherrid, T. Bentahar, H. Mayache, and A. Bentahar, "Number of pixel change rate and unified average changing intensity for sensitivity analysis of encrypted insar interferogram," *Ingénierie des systèmes d'information*, vol. 25, no. 5, pp. 601–607, 2020, doi: 10.18280/isi.250507.
- [24] Y. Sang, J. Sang, and M. S. Alam, "Image encryption based on logistic chaotic systems and deep autoencoder," *Pattern Recognition Letters*, vol. 153, pp. 59–66, 2022, doi: 10.1016/j.patrec.2021.11.025.
- [25] X. -Y. Wang, Y. -Q. Zhang, and X. -M. Bao, "A novel chaotic image encryption scheme using DNA sequence operations," *Optics and Lasers in Engineering*, vol. 73, pp. 53–61, 2015, doi: 10.1016/j.optlaseng.2015.03.022.
- [26] L. Liu, Q. Zhang, and X. Wei, "A RGB image encryption algorithm based on DNA encoding and chaos map," *Computers & Electrical Engineering*, vol. 38, no. 5, pp. 1240–1248, 2012, doi: 10.1016/j.compeleceng.2012.02.007.
- [27] J. S. A. E. Fouda, J. Y. Effa, S. L. Sabat, and M. Ali, "A fast chaotic block cipher for image encryption," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 3, pp. 578–588, 2014, doi: 10.1016/j.cnsns.2013.07.016.
- [28] X. Zhang and Z. Zhao, "Chaos-based image encryption with total shuffling and bidirectional diffusion," *Nonlinear Dynamics*, vol. 75, no. 1–2, pp. 319–330, 2014, doi: 10.1007/s11071-013-1068-4.
- [29] X. Chai, K. Yang, and Z. Gan, "A new chaos-based image encryption algorithm with dynamic key selection mechanisms," *Multimedia Tools and Applications*, vol. 76, no. 7, pp. 9907–9927, 2017, doi: 10.1007/s11042-016-3585-x.
- [30] Z. Hua, B. Xu, F. Jin, and H. Huang, "Image encryption using josephus problem and filtering diffusion," *IEEE Access*, vol. 7, pp. 8660–8674, 2019, doi: 10.1109/ACCESS.2018.2890116.

BIOGRAPHIES OF AUTHORS



Mustafa Dhiaa Al-Hassani    is assistant professor at College of Science, Mustansiriyah University, Iraq. He holds a Ph.D degree in Computer Science with specialization in information security. His interest in the area: biometrics, information security, multimedia processing, data compression, pattern recognition, and e-learning. He occupied several administrative positions: Director of Computer Center at Mustansiriyah University from 2010 to 2014. Currently, he is the Vice-President of Mustansiriyah university for Scientific Affairs (since 2019). He received many international/local certificates and scientific awards. He has supervised and co-supervised more than 10 M.Sc and 4 Ph.D students' thesis. He has authored or coauthored a lot of papers: more than 20 publications in proceedings/journals, and 4 textbooks in computer science fields were printed by the ministry of higher education and scientific research. He can be contacted at email: dr_mdhiaa77@uomustansiriyah.edu.iq.



Methaq Talib Gaata    is assistant professor at College of Science, Mustansiriyah University, Iraq. He holds a Ph.D degree in Computer Science with a specialization in information security. His research areas are information hiding, multimedia processing, biometrics, pattern recognition, and computer networks. He worked as the Head of the Department of Computer Science from 2016 to 2020. Currently, he is the Deputy Dean for Scientific Affairs and Postgraduate Studies at the College of Science. He has supervised and co-supervised more than 25 master's and 5 Ph.D, students. He has authored or co-authored more than 30 publications in proceedings and journals, with a 5 H-index. He can be contacted at email: dr.methaq@uomustansiriyah.edu.iq.