Bulletin of Electrical Engineering and Informatics
Vol. 15, No. 2, April 2026, pp. 1731~1742
ISSN: 2302-9285, DOI: 10.11591/eei.v15i2.10939

Optimized chaos based encryption-decryption of multimedia data

Rashad J. Rasras[1], Mutaz Rasmi Abu Sara[2], Jihad Nader[1], Ziad Alqadi[1]
[1]Department of Electrical Engineering, Faculty of Engineering Technology, Al-Balqa Applied University, Amman, Jordan
[2]Department of Information Technology, Faculty of Engineering and Information Technology, Palestine Ahliya University, Bethlehem, Palestine

Article Info

Article history:

Received Jun 29, 2025
Revised Jan 7, 2026
Accepted Mar 10, 2026

Keywords:

Chaotic logistic map model
Cryptography
Decision support system
Lorenz dynamical system
Public key
Secret image key
Short message

ABSTRACT

These days, a major concern is the security of multimedia data exchange over the internet. Compared to conventional methods, chaos-based encryption has proven increasing role and superiority in current multimedia cryptography. This research proposes an innovative multimedia encryption technique utilizing optimized chaotic systems and simple rearrangement operation. Data blocking will be used to optimize the chaotic model and to reduce the key generation time. To increase the private key length and to raise the level of data security the presented method will be implemented in multiple rounds, where each round will be treated as an independent task and the selected sequence of rounds to be executed depends on the user wish. The method can be efficiently used to cipher-decipher short messages, images, and digital speech signal, and it will use variable length of private key, variable block size and variable number of rounds. The output of simulation confirms effectiveness of proposed technique in giving strong security, and good speed up in ciphering and deciphering process comparing with other existing standard and chaotic based methods. The proposed technique shows high resistance to attacks while the quality of encrypted/decrypted digital data remains acceptable.

Corresponding Author:

Rashad J. Rasras
Department of Electrical Engineering, Faculty of Engineering Technology, Al-Balqa Applied University
Amman 11134, P.O. Box 15008, Jordan
Email: rashad.rasras@bau.edu.jo

1. INTRODUCTION

One of techniques used for securing multimedia data transmission via internet network is encryption technology based on chaos. Multimedia data exchange via open networks and the internet requires strong security measures to ensure confidentiality and stop unwanted access. The conventional data cryptography methods based on standard algorithms such as data encryption standard (DES), triple data encryption standard (3DES), advanced encryption standard (AES), Rivest cipher 6 (RC6), and blowfish (BF) are inefficient and are not suitable for multimedia applications, because of different factors such as massive redundancy, strong correlations, and very huge size [1]. The importance of chaotic systems in cryptography has increased because of their randomness, non-periodicity, and sensitivity to initial conditions [2]. Chaotic based encryption systems have proven to be highly effective in providing the enhanced security and privacy required by employing variable keys. These keys are very difficult to predict because of their intricate structures and chaotic behavior [3]-[6].

In literature chaos-based encryption methods cover a broad range of multimedia form text, audio, image, and video. Yousif [7] it was developed a speech cryptographic method-based on a combination of

permutation and substitution of speech samples using a chaotic map, the implemented method showed low correlation and strong security characteristics. Amina and Mohamed [8], it was proposed a new chaotic approach for medical images, the technique included two stages: pixel diffusion and chaotic confusion. Low correlation with high security properties were proved by the method. Valli and Ganesan [9] introduced video encryption technique. Their technique's primary drawback is a longer encryption time. Hu *et al*. [10] proposed a color image encryption algorithm using a cloud model Fibonacci chaotic system combined with matrix convolution enhancing security and randomness for robust encryption. The main drawback of suggested method was a considerable computing cost.

Chaotic and hybrid methods were introduced to enhance the security and speed of data cryptography, some of these methods increased the speed to 911 K bytes per second [11]-[13], and this was a good achievement provided by the chaotic based.

According to a summary of related research, chaotic systems have been used effectively for the encryption of multimedia data. While traditional chaos-based cryptography systems could work well for text data, they are unable to provide the same level of security for voice and visual data. The main reasons of this are media data's high redundancy and bulk data capacity [14]-[20].

It is essential to guarantee strong safety with minimal computational complexity in order to increase time efficiency, which is essential for multimedia data transmission and storage, strong security and efficient encryption speed remain hard to combine, especially for real-time applications. To reduce computing complexity while maintaining robust encryption, more optimization is required. Solving these shortcomings is essential for creating encryption methods that are more practical, secure, and efficient. The aim of this research is to create crypto method achieving the following objectives [21]-[29]:

- Flexibility, the method must be efficiently used to encrypt-decrypt messages, images and decision support system (DSSs), changing the data type must not require any changes in the method algorithm.
- Speed, the method must provide a high speed especially when the data size is big, the method must minimize the encryption-decryption time.
- Large public key (PK) space to give good security.
- Quality which evaluated by mean squared error (MSE) and peak signal-to-noise ratio (PSNR) parameters values, where MSE equal zero, and PSNR must be infinite.
- Simple and easy to implement by using minimum number of arithmetic and logic operations.

2. METHOD

In this section we first introduce chaos system model then we present the proposed multimedia encryption method.

2.1. Chaos system

Chaos systems appears random but are controlled by deterministic in (1) and it is hard to predict they are sensitive to initial value.

$$X_{n+1} = rx_n(1 - x_n) \quad (1)$$

To evaluate $X_{n+1,}$ the initial value of x_n and constant value r_{must} be given, the equation starts with a fixed value of r, and an initial value of x0. The equation runs recursively, obtaining *x1*, *x2*, and *xn*, each of these iterations can be save in an array with a selected length, this array will be called logistic data set (LDS). The values of r must be within the range 3 to 4, while values of x must be within the range 0 to 1, and these values are fractional decimals with double data type (8 bytes are used to represent each value).

Chaotic logistic map model (CLMM) model has the following features:

- It can be easily implemented, for a given values of r and x it is easy to create LDS with any selected length, Figure 1 shows a simple sequence of mat lab instructions which can be easily use to create LDS with L elements, while Figure 2 shows the generated LDS and indices key (IK).
- The parameters r and x can be used as a private key.
- The generated LDS can be used as a secret key.
- LDS can be produced to form an IK which could be considered as a secret key (see Figure 2).
- Generated LDS and IK are sensitive to initial values of r, x, and L, slight changes in initial values will produce a new data set LDS and a new IK. Figure 3 and 4 show how LDS and IK are sensitive to the values of r, x and L.

– One or more CLMMs, each of them will have its own values of r, x, and L can be used to form a long public key (PK), this length will be strong enough to prevent hacking attacks, and they can be used to generate different IKs to be used in different processes.

```
L = 12;
r = 3.71; x = 0.11;
for j = 1: L
        x = x × r × (1 − x);
        LDS (j) = x;
end
[rrl IK] = sort(LDS);
```

Figure 1. CLMM implementation process

```
LDS=
  0.3632   0.8581   0.4518   0.9189   0.2765   0.7422   0.7098   0.7642   0.6685   0.8221   0.5425   0.9208
     1        2        3        4        5        6        7        8        9       10       11       12      <-- Index
IK =
     5    1    3   11    9    7    6    8   10    2    4   12
```

Figure 2. Results of running CLMM

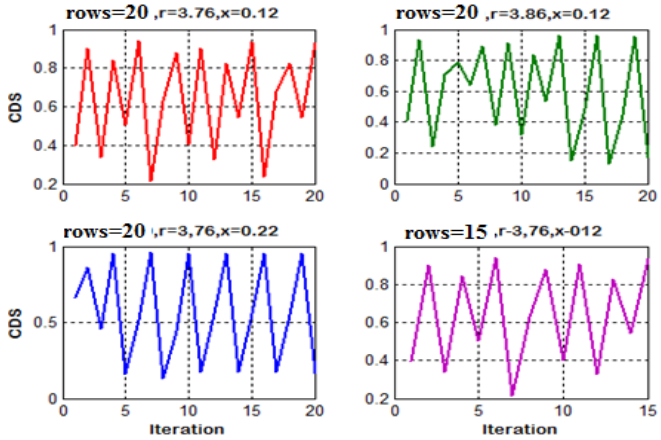


Figure 3. CLMM generated LDS sensitivity

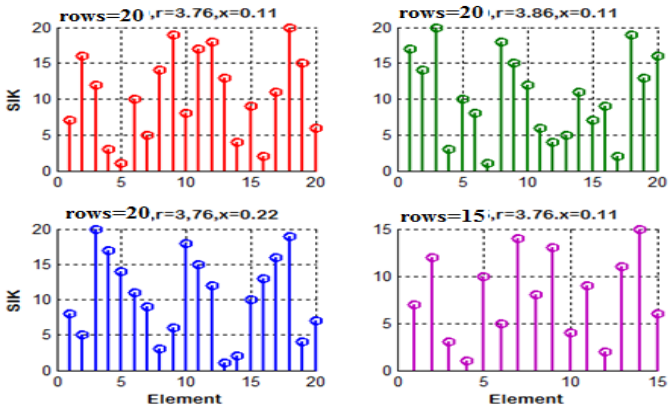


Figure 4. CLMM generated IK sensitivity

2.2. The proposed method

We introduce comprehensive encryption method which can treat all forms of media including text, images, and speech. In this paper research we will classify the digital data to the following categories:

- Text messages, text message is one-dimension array of characters. Messages with length less than 1 K bytes (characters) will be considered as short messages (SMs) and messages with length greater or equal 1 K bytes will be considered as long messages (LMs).
- Digital images, gray and color images are most popular and used images. Gray image is a collection of pixel unsigned integer values arranged in 2D matrix which may be converted to vector and vice versa. True color image is also a set of colors unsigned integer values arranged three 2D matrices, the first 2D matrix will represent the red colors; the second 2D matrix will represent the green colors, while the third 2D matrix will represent the blue colors. Color images usually have a big size and it can be treated as long messages.
- Digital speech signal (DSS) is a collection of samples (representing the speech amplitude) arranged in one or two column matrix (mono or stereo DSS), and each sample value is a decimal fraction within the range -1 to +1 with a double data type 8 bytes are used to represent each sample value. DSS matrix can be easily reshaped to one row matrix and vice versa, the size of DSS is usually big and it can be treated as long message.

The proposed method intends to improve the performance of the traditional chaos-based crypto methods by providing the following enhancements:

- Applying cryptography with and without data blocking and variable block size.
- One or more rounds can be used; each round will be treated as an independent task.
- Using variable length of PK which is based on number of rounds.
- Applying optimized CLMM to produce secret keys with minimum time.
- Crypto process will be applied by rearranging the data items based on the contents of generated PK. Unlike traditional methods that utilize logical or mathematical operations to change data item values, it focuses on changing the order of the data items without changing the actual values of the data items. Thus, the processes will be speeded up.

Data rearrangement can be implemented at the data item (byte or sample) level, and at the block level, here the block will have a fixed number of bytes or sample (block size), the two levels of data rearrangement will be executed using the same sequence of operations, in the byte level the number of blocks must equal the data length (NB=L, and the block BS size must=1) this will be considered as no blocking. Figure 5 shows the data rearrangement process sequences.

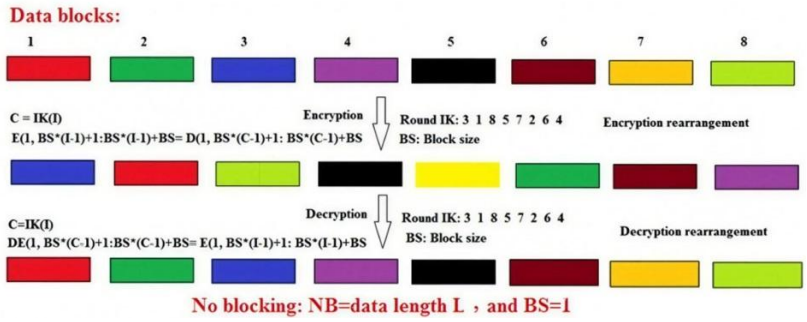


Figure 5. Data rearrangement process

The total of encryption-decryption time consists of the following components:

- Key generation time (KGT), which is the time required to generate the secret indices keys.
- Rearrangement time (RT).

Decreasing KGT and/or decreasing the RT will decrease the total encryption time (TET), thus the speed of data cryptography will be increased. The secret image key (SIKs) generation process uses a CLMM, which uses the values of the chaotic parameters x and r and the value of L, which defines the length of the generated SIK. For CLMM increasing the key length will rapidly increase the KGT, when L is large a KGT will be big, thus the method will be inefficient. To show how the KGT will increase when increasing L different keys were generated and the KGT for each key was measured, and Table 1 shows the required KGT for each key.

Table 1. KGT when varying the key length (L) (r=3.61; x=0.18)

L	KGT (second)	L	KGT (second)
100	0.001000	10000	0.105000
250	0.002000	15000	0.173000
500	0.005000	25000	0.383000
750	0.008000	40000	0.925000
1000	0.014000	50000	1.888000
2000	0.052000	75000	4.754000
5000	0.070000	100000	11.680000

As it shown from Table 1 KGT for short SIKs requires a few milliseconds, when key length increased the KGT rapidly increased (see Figure 6), and from these results we can use a key length up to 2000 to minimize the KGT and to optimize the crypto method.

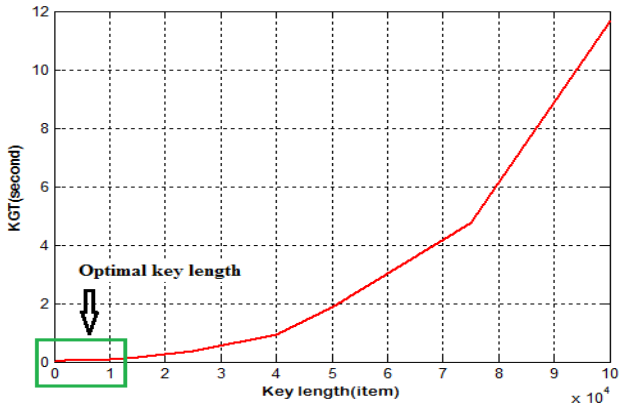


Figure 6. KGT vs L

From the obtained KGT results and to optimize the method of cryptography, the optimal recommended length of the generated SIK must be less than 2000, so if the digital data has a bigger size it must be divided into blocks, with NB not greater than 2000.

Digital data blocking is required for data with big size in order to use the optimal SIK length, this will decrease the KGT and speed up the crypto method. To show the effects of data blocking a message of 10000 characters was selected and encrypted–decrypted varying the NB value (BS value), the KGT, RT, and the TET were measured and Table 2 shows the encryption speed results depending on NB.

Table 2. Encryption speed values depending on NB value ((r=3.61; x=0.18)

NB (key length)	BS	KGT (second)	RT (second)	TET (second)	Encryption speed (K bytes per second)
10000	1 (no blocking)	0.1090	0.0130	0.1220	80.0461
5000	2	0.0670	0.0060	0.0730	133.7757
2500	4	0.0580	0.0030	0.0610	160.0922
2000	5	0.0560	0.0040	0.0600	162.7604
1000	10	0.0180	0.0020	0.0200	488.2813
500	20	0.0060	0.001	0.00700	1395.1
250	40	0.0030	0.001	0.0040	2441.4
100	100	0.0020	0.000001	0.002001	4882.8
50	200	0.0020	0.000001	0.002001	4882.8
5	2000	0.0020	0.000001	0.002001	4882.8

From Table 2 the following can be seen:

- Decreasing the number of blocks (key length) will decrease the KGT time, and the RT time.
- KGT has a bigger effect on the speed; it required more time than RT, so optimizing the KGT will be an important issue.
- The selected values of r and x did not affect the speed; they affect only the contents of the generated SIK.
- Data blocking must be used to optimize the speed of data cryptography.

The PK needed to apply data rearrangement must contain the values of NB, r and x, thus the length of this key will equal 192 bits (3×64), this key will be good enough to provide a key space capable to resist hacking attacks, the key space will be calculated using in (2):

$$Key\ space\ = 2^{192} = 6.277101735386680763835789423207 7 \times 10^{57} Combinations \quad (2)$$

The PK can be expanded to be used to apply one or more rounds of data rearrangements, each round will be used as an independent task, the obtained outputs of each round can be considered as final results. Expanding the PK will not affect the algorithms of the method; the sequence of used operations will be repeated for each round using various different input data. For each round the NB, r, and x values will be used thus the PK structure will be as shown in Figure 7.

```
%Expanded PK example
Last = 1;
First = 1;
NB = [L 4 6 5 100 103 1066 104 2000 1980 2015 2200];
r = [3.77 3.52 3.88 3.71 3.65 3.67 3.84 3.89 3.75 3.68 3.66 3.82];
x=[0.11 0.2 0.23 0.15 0.17 0.25 0.19 0.13 0.12 0.164 0.175 0.201];
```

Figure 7. Expanded PK structure

The user must select the maximum number of rounds in in this example we select 12, and the PK will consist of the following components:

- First: first round in the selected sequence of rounds to be executed, it must be less or equal the last.
- Last: the last round in the selected sequence of rounds to be executed, it must be less or equal maximum selected number (in our case 12).
- NB: array of 12 integer values, each of them points to the number of blocks to be used in each selected round, NB value must be less or equal the data size (L).

  Chaotic array of parameters r: an array of 12 elements.

  Chaotic array of parameters x: an array of 12 elements.

Using the expanded PK will give the method the following improvements:

- The PK length will equal expanded and it will equal 2304 bits (192×12), thus the key space will be increased and it will be very big and capable to resist any hacking attacks, the key space will be calculated using in (3):

$$Key\ space\ = 2^{2304} = 3.74205365089213342367999773538 3 \times 10^{693} Combinations \quad (3)$$

- The PK will allow using no-data-blocking and data-blocking, if the NB equal L (data size), then no blocking for this round.
- NB values can be arranged as in Figure 7 by dividing the values into groups: rounds 1 to 4 can be used to treat short messages, rounds 5 to 8 can be used to treat data with medium size, while rounds 9 to 12 can be used to treat images and DSSs, this NB grouping will give us the beast speed, and this grouping is not a must, the user will have the freedom to select any NBs values.
- Using multiple rounds will not much affect the quality of the encrypted data, it will affect the speed. Comparing the speed result with other existing methods speeds it will be acceptable and higher than the existing methods speed.
- The method will be very flexible, and it can be executed using one or more rounds, and the selected sequence of rounds will be defined by setting the values of last and first in the x=expanded PK.

Using expanded PK will not change the encryption and decryption process, each of them will use a simplified sequence of operations implemented by MATLAB codes shown in Figure 8.

```
for i = First : Last
        BS(i) = fix(L/NB(i));
end
for i = First : Last
        mm1 = mm;
        LL = NB(i);      xx=x(i);          rr = r(i);
        for j = 1 : LL
                xx = xx × rr × (1-xx);     ck(j) = xx;
        end
        [rr1 IK] = sort(ck);
        clear xx;          clear rr;
        dd = BS(i);
        for ii = 1 : LL
        c = IK(ii);
%for encryption:
mm1(1, (i - 1) × dd + 1 : (i - 1) × dd + dd) = mm(1, (c - 1) × dd + 1 : (c - 1) × dd + dd);
% for decryption
mm1(1, (c - 1) × dd + 1 : (c - 1) × dd + dd) = mm(1, (ii - 1) × dd+1 : (i - 1) × dd + dd);
end
mm = mm1;
clear IK
clear ck;
end
```

Figure 8. Encryption-decryption process

3. RESULTS AND DISCUSSION

The presented method is capable to handle multimedia data, and the input data (if it is image or DSS) must be reshaped to one row matrix before processing and reshaped back to original size after processing.

3.1. Quality requirements

The presented method satisfies the quality requirements when using any sequence of rounds, the encrypted data will be always have a low quality and it will be damaged, while the decrypted data will always have an excellent quality and it will be the same as the source data. To show this the following message was encrypted using the PK shown in Figure 7 by varying some values and Table 3 shows the obtained encrypted messages with the values of MSE and PSNR calculated between the source and the encrypted messages: Message: 'Optimized method of data cryptography'.

Table 3. Obtained encrypted messages

First	Last	Sequence of rounds	Encrypted message	MSE	PSNR
1	1	1	rdrdmOtf ozh tpaotm ehyadpitaopi cyeg	1055.4	26.4636
1	2	1-2	m ehyadpirdrdmOtf taopi cyeozh tpaotg	1338.4	24.0879
1	3	1-3	cyeozh tpaotdmOtf m ehyadpirdrtaopi g	1015.8	26.8463
1	4	1-4	Otf m ecyeozh hyadpirdrtaopitpaotdm g	1177.7	25.3670
2	3	2-3	aph method oimizedf data crOptyptogry	662.3243	31.1229
2	4	2-4	izedf daph metata crOptyptoghod oimry	1349.8	24.0034
3	4	3-4	od oOpt cryptoimized mef datgraphthay	1013.7	26.8670
4	4	4-4	od of dOptimizata cryptograped methhy	1452.1	23.2727
Remarks			Damaged	High	Low

The presented optimized method (POM) is a multipurpose method, it was used to treat color images, several images were selected and they were encrypted-decrypted using POM and using the PK shown in Figure 7 (by selecting rounds 9 to 12, Last=12, and First=9), the obtained encrypted images were damaged, while the obtained decrypted images were the same as the source images, decrypted image always had a low quality, while encrypted image always had excellent quality, Figure 9 shows sample outputs, and these outputs proved that POM satisfied the quality requirements of good crypto method.

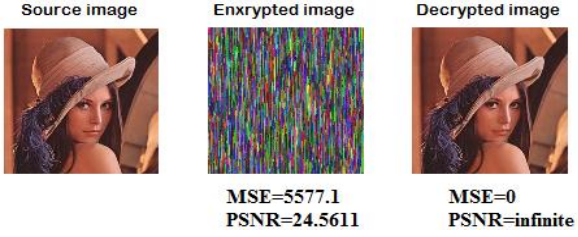


Figure 9. Images sample outputs

POM was implemented using various DSS using the PK shown in Figure 7 (by selecting rounds 9 to 12, last=8, and first=5), the ciphered DSSs were damaged, while the deciphered DSSs were the same as the source DSSs, encrypted DSS always had a low quality, while decrypted DSS always had excellent quality, Figure 10 shows sample outputs, and these outputs proved that the quality of encrypted/decrypted digital data remains acceptable.

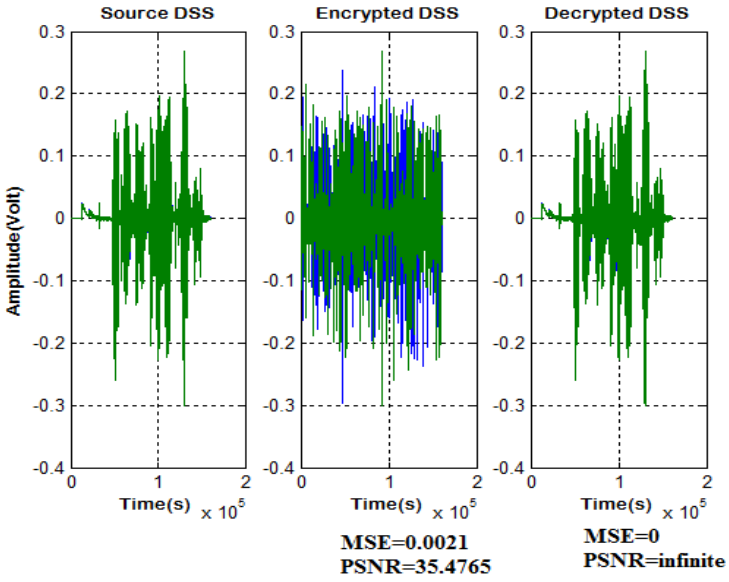


Figure 10. DSS sample outputs

3.2. Key sensitivity

The POM is very sensitive initial values of the PK, and slight changes in the PK in the decryption function will lead to damaged decrypted data. As shown in Figure 11, the encrypted image was decrypted using PK2 shown in Figure 12 (r9 was changed from 3.75 to 3.65), the obtained outputs shown in Figure 13 proved that POM is sensitive, here the decrypted image is a damaged one:

```
%PK1:
Last = 12;
First = 9;
NB = [L 4 6 5 100 103 1066 104 2000 1980 2015 2200];
r = [3.77 3.52 3.88 3.71 3.65 3.67 3.84 3.89 3.75 3.68 3.66 3.82];
x = [0.11 0.2 0.23 0.15 0.17 0.25 0.19 0.13 0.12 0.164 0.175 0.201];
```

Figure 11. Used PK1

```
%PK2:
Last = 12;
First = 9;
NB = [L 4 6 5 100 103 1066 104 2000 1980 2015 2200];
r = [3.77 3.52 3.88 3.71 3.65 3.67 3.84 3.89 3.65 3.68 3.66 3.82];
x = [0.11 0.2 0.23 0.15 0.17 0.25 0.19 0.13 0.12 0.164 0.175 0.201];
```

Figure 12. Used PK2

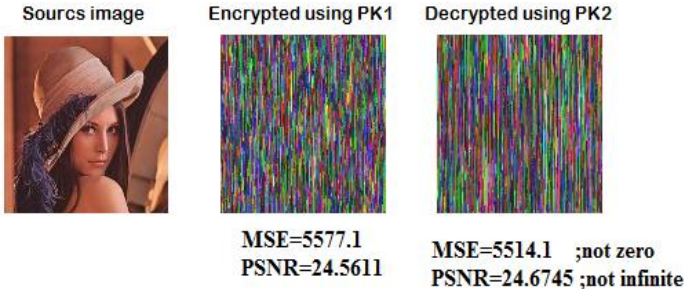


Figure 13. POM sensitivity

3.3. The effect of blocking on encryption speed

The speed of POM was tested, a message with 500 characters, color image with size equal 6119256 bytes and a DSS with size equal 321536 samples were processed using the PK shown in Figure 14 by varying the values of first and last, the cipher time was measured and the speed was evaluated, and Tables 4 to 6 show the obtained speed results:

```
%Used PK in speed tersting:
Last = ?
First = ?
NB = [L 4 6 5 100 103 1066 104 2000 1980 2015 2200];
r = [3.77 3.52 3.88 3.71 3.65 3.67 3.84 3.89 3.75 3.68 3.66 3.82];
x = [0.11 0.2 0.23 0.15 0.17 0.25 0.19 0.13 0.12 0.164 0.175 0.201];
```

Figure 14. Used PK in speed testing

Table 4. Speed results for short message

First value	Last value	Number of used rounds	Encryption time (second)	Encryption speed (K bytes per second)
1(no blocking)	1	1	0.0060	81.3802
1	2	2	0.0090	54.2535
1	3	3	0.0100	48.8281
1	4	4	0.0110	44.3892
2(with blocking)	2	1	0.0030	162.7604
2	3	2	0.0050	97.6563
2	4	3	0.0050	97.6563
3(with blocking)	3	1	0.0030	162.7604
3	4	2	0.0050	97.6562
	Average		0.0063	94.1490

Table 5. Speed results for image

First value	Last value	Number of used rounds	Encryption time (second)	Encryption speed (K bytes per second)
9	9	1	0.2620	22809
9	10	2	0.5150	11604
9	11	3	0.7790	7671.2
9	12	4	1.0290	5807.4
10	10	1	0.2640	22636
10	11	2	0.5140	11626
10	12	3	0.7810	7651.5
11	11	1	0.2660	2246.6
11	12	2	0.5720	10447
	Average		0.5536	11389

Table 6. Speed results for DSS

First value	Last value	Number of used rounds	Encryption time (second)	Encryption speed (K samples per second)
1	1	1	180.0590	1.7439
No blocking, data with big size (not optimized)				(Excluded)
5	5	1	0.0220	14273
5	6	2	0.0380	8263.2
5	7	3	0.0820	3829.3
5	8	4	0.0930	3376.3
6	6	1	0.0190	16526
6	7	2	0.0660	4757.6
6	8	3	0.0790	3974.7
7	7	1	0.0500	6280.0
	Average		0.0561	7660.0

From Tables 4 to 6 we can see the following:

- Short messages can be encrypted-decrypted with blocking and without blocking, they can be treated using one or more rounds, increasing the number of rounds will not much affect the speed.
- Data with big size must be blocked to optimize the speed, cryptography without blocking will negatively affect the speed, see Table 6 row 1.

– In general POM gave a good speed up while comparing with other existing methods as shown in Tables 7 and 8.

Table 7. POM average speed (in K bytes/samples per second)

Data type	Average speed
Short message	94.1490
Color image	11389
DSS	7660.0
Average	6381.0

Table 8. POM speed up

Method	Speed (K samples per second)	Speed up
Proposed method	6381.0	1.0000
DES	10.6276	600.4178
3DES	9.1430	697.9110
AES	11.0623	576.8240
RC2	7.5821	841.5874
RC6	19.0539	334.8921
BF	68.6234	92.9858

4. CONCLUSION

In this work, multimedia data crypto method was proposed which is based on optimized CLMM and simple rearrangement operations with data blocking. Our proposed method provided high time efficiency, high-security features and low correlation between the original and encrypted media data. The proposed method enhanced the traditional crypto methods by using a variable number of PK length, block size and number of rounds. Multi rounds were necessary to increase the security level, and to optimize the speed when using big data such as images and DSS. The simulation results showed that proposed method provided a good speed up comparing with conventional method, flexible in key handling and used a long PK with enough key space that made the produced decrypted data is very sensitive to slight changes in PK key parameters. The future work of this method is to extend the proposed solution to include various types of multimedia data in fog computing application.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Rashad J. Rasras	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Mutaz Rasmi Abu Sara		✓				✓		✓	✓	✓	✓	✓		
Jihad Nader	✓		✓	✓			✓			✓	✓		✓	✓
Ziad Alqadi	✓		✓	✓			✓			✓	✓		✓	✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

DATA AVAILABILITY

The authors confirm that the data supporting the findings of this study are available within the article.

REFERENCES

[1] O. M. Al-Hazaimeh, A. A. Abu-Ein, M. M. Al-Nawashi, and N. Y. Gharaibeh, "Chaotic based multimedia encryption: a survey for network and internet security," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 4, pp. 2151–2159, Aug. 2022, doi: 10.11591/eei.v11i4.3520.

[2] Y. Li, H. Yu, B. Song, and J. Chen, "Image encryption based on a single-round dictionary and chaotic sequences in cloud computing," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 7, p. 1, Apr. 2021, doi: 10.1002/cpe.5182.

[3] M. S. Azzaz, C. Tanougast, S. Sadoudi, and A. Bouridane, "Synchronized hybrid chaotic generators: Application to real-time wireless speech encryption," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 8, pp. 2035–2047, Aug. 2013, doi: 10.1016/j.cnsns.2012.12.018.

[4] I. Yasser, F. Khalifa, M. A. Mohamed, and A. S. Samrah, "A New Image Encryption Scheme Based on Hybrid Chaotic Maps," *Complexity*, vol. 2020, pp. 1–23, Jul. 2020, doi: 10.1155/2020/9597619.

[5] Y. Zheng, "Chaotic Phenomenal and One-dimensional Logistic Map," *Highlights in Science, Engineering and Technology*, vol. 72, pp. 638–644, Dec. 2023, doi: 10.54097/wd5v9r36.

[6] Y. Su, X. Wang, S. Unar, X. Zhao, and P. Liu, "Secure image storage system based on compressed sensing and 2D-SLLIM in cloud environment," *Nonlinear Dynamics*, vol. 111, no. 3, pp. 2779–2814, Feb. 2023, doi: 10.1007/s11071-022-07930-5.

[7] S. F. Yousif, "Speech Encryption Based on Zaslavsky Map," *Journal of Engineering and Applied Sciences*, vol. 14, no. 17, pp. 6392–6399, 2019, doi: 10.36478/jeasci.2019.6392.6399.

[8] S. Amina and F. K. Mohamed, "An efficient and secure chaotic cipher algorithm for image content preservation," *Communications in Nonlinear Science and Numerical Simulation*, vol. 60, pp. 12–32, Jul. 2018, doi: 10.1016/j.cnsns.2017.12.017.

[9] D. Valli and K. Ganesan, "Chaos based video encryption using maps and Ikeda time delay system," *European Physical Journal Plus*, vol. 132, no. 12, p. 542, Dec. 2017, doi: 10.1140/epjp/i2017-11819-7.

[10] X. Hu, L. Wei, W. Chen, Q. Chen, and Y. Guo, "Color Image Encryption Algorithm Based on Dynamic Chaos and Matrix Convolution," *IEEE Access*, vol. 8, pp. 12452–12466, 2020, doi: 10.1109/ACCESS.2020.2965740.

[11] Z. Hua, Y. Zhou, and H. Huang, "Cosine-transform-based chaotic system for image encryption," *Information Sciences*, vol. 480, pp. 403–419, Apr. 2019, doi: 10.1016/j.ins.2018.12.048.

[12] M. Asgari-Chenaghlu, M. A. Balafar, and M. R. Feizi-Derakhshi, "A novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation," *Signal Processing*, vol. 157, pp. 1–13, Apr. 2019, doi: 10.1016/j.sigpro.2018.11.010.

[13] S. Zhou, Y. Qiu, X. Wang, and Y. Zhang, "Novel image cryptosystem based on new 2D hyperchaotic map and dynamical chaotic S-box," *Nonlinear Dynamics*, vol. 111, no. 10, pp. 9571–9589, May 2023, doi: 10.1007/s11071-023-08312-1.

[14] N. Rani, V. Mishra, and S. R. Sharma, "Image encryption model based on novel magic square with differential encoding and chaotic map," *Nonlinear Dynamics*, vol. 111, no. 3, pp. 2869–2893, Feb. 2023, doi: 10.1007/s11071-022-07958-7.

[15] X. Zhang and X. Wang, "Multiple-image encryption algorithm based on DNA encoding and chaotic system," *Multimedia Tools and Applications*, vol. 78, no. 6, pp. 7841–7869, 2019, doi: 10.1007/s11042-018-6496-1.

[16] M. Brindha, "Multiple stage image encryption using chaotic logistic map," in *2017 International Conference on Intelligent Sustainable Systems (ICISS)*, Dec. 2017, pp. 1239–1243, doi: 10.1109/ISS1.2017.8389384.

[17] S. Mazloom and A. M. Eftekhari-Moghadam, "Color image cryptosystem using chaotic maps," in *IEEE SSCI 2011 - Symposium Series on Computational Intelligence - CIMSIVP 2011: 2011 IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing*, Apr. 2011, pp. 142–147, doi: 10.1109/CIMSIVP.2011.5949254.

[18] K. S. Sankaran and B. V. S. Krishna, "A New Chaotic Algorithm for Image Encryption and Decryption of Digital Color Images," *International Journal of Information and Education Technology*, pp. 137–141, 2011, doi: 10.7763/ijiet.2011.v1.23.

[19] Z. Yan-Bin and D. Qun, "A new digital chaotic sequence generator based on logistic map," in *Proceedings - 2011 2nd International Conference on Innovations in Bio-Inspired Computing and Applications, IBICA 2011*, 2011, pp. 175–178, doi: 10.1109/IBICA.2011.48.

[20] E. Agrawal and P. R. Pal, "A Secure and Fast Approach for Encryption and Decryption of Message Communication," *International Journal of Engineering Science and Computing*, vol. 7, no. 5, pp. 11481–11485, 2017.

[21] A. M. Qadir and N. Varol, "A Review PapGao Hui, Research on computer network information security and protection strategies in the era of big data," *Science and Technology Innovation*, pp. 76–77, 2018.

[22] A. Majeed, M. L. M. Kiah, H. T. Madhloom, B. B. Zaidan, and A. A. Zaidan, "Novel approach for high secure and high rate data hidden in the image using image texture analysis," *International Journal of Engineering and Technology*, vol. 1, no. 2, pp. 70–76, 2009.

[23] M. M. Abu-Faraj, K. Aldebei, and Z. A. Alqadi, "Simple, Efficient, Highly Secure, and Multiple Purposed Method on Data Cryptography," *Traitement du Signal*, vol. 39, no. 1, pp. 173–178, Feb. 2022, doi: 10.18280/ts.390117.

[24] K. Sharma, "Taxonomy of Cryptography Techniques for Network Security," *International Journal of Engineering and Computer Science*, vol. 5, no. 8, pp. 17787–17793, 2016, doi: 10.18535/ijecs/v5i8.60.

[25] E. Ashraf, N. F. A. Takieldeen, and M. Abd-elazeem, "Novel Cryptographic Algorithm for 4G / LTE-A," *International Journal of Computer Applications*, vol. 163, no. 1, pp. 5–9, Apr. 2017, doi: 10.5120/ijca2017912921.

[26] S. Manku and K. Vasanth, "Blowfish encryption algorithm for information security," *ARPN Journal of Engineering and Applied Sciences*, vol. 10, no. 10, pp. 4717–4719, 2015.

[27] I. Yasser, M. A. Mohamed, A. S. Samra, and F. Khalifa, "A chaotic-based encryption/decryption framework for secure multimedia communications," *Entropy*, vol. 22, no. 11, pp. 1–23, 2020, doi: 10.3390/e22111253.

[28] M. Abomhara, O. Zakaria, O. O. Khalifa, A. A. Zaidan, and B. B. Zaidan, "Enhancing Selective Encryption for H.264/AVC Using Advanced Encryption Standard," *arXiv preprint*, Jan. 2022, doi: 10.48550/arXiv.2201.03391.

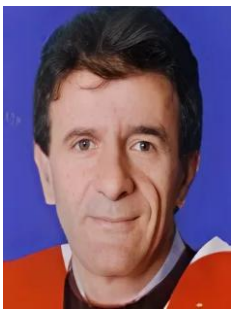
[29] T. Umar, M. Nadeem, and F. Anwer, "Chaos based image encryption scheme to secure sensitive multimedia content in cloud storage," *Expert Systems with Applications*, vol. 257, 2024, doi: 10.1016/j.eswa.2024.125050.

BIOGRAPHIES OF AUTHORS

Rashad J. Rasras received the Ph.D. degree from National Technical University (Kharkov Polytechnic Institute) 2001, with research in automated intelligent control systems. Currently. He is an associate professor at Department of Electrical Engineering, Al-Balqa Applied University. His research interests include image processing, machine learning, signal processing, and advanced computer architecture. He can be contacted at email: rashad.rasras@bau.edu.jo.

Mutaz Rasmi Abu Sara received the Master's degree in computer science (database systems) in 2006 and Ph.D. from Saint Petersburg Electro technical University in 2010 with research and development of integrated database circuit components for CAD schematic, his research interest includes database systems, algorithms and machine learning. 2011-2020 he worked as assistant professor at Taibah University in K.S.A, currently he works as assistant professor at Palestine Ahliya University. He can be contacted at email: moutaz.a@paluniv.edu.ps.

Jihad Nader received the Ph.D. degree from Southwestern State University (Kursk State Technical University) 2006 with research in Components and devices for computing and control systems. Currently, he is an associate professor at Department of Electrical Engineering, Al-Balqa' Applied University. His research interests include signal processing, parallel processing, and image processing. He can be contacted at email: jihadnader@bau.edu.jo.



Ziad Alqadi received the Ph.D. degree from National Technical University (Kiev Polytechnic Institute) 1986 with research in parallel computer architecture. Currently, he is a professor at Department of Electrical Engineering, Al-Balqa' Applied University. His research interests include signal processing, parallel processing, and image processing. He can be contacted at email: dr.ziad.alqadi@bau.edu.jo.