ISSN: 0067-2904

LUC Image Encryption Enhancement Using a Hyper-Chaotic System and DNA

Noor Muneam Abbas , Matheel E. Abdulmunim
Computer Science Department, University of Technology, Baghdad, Iraq

Received: 25/2/2023 Accepted: 29/10/2023 Published: 30/12/2024

Abstract

The digital image is considered one of the most heavily attacked and targeted data types because of its various applications and sensitive visual data. Many security techniques have been introduced to cover various image security needs, such as watermarking and steganography for intellectual property protection, authentication, and cryptography techniques for confidentiality and secrecy. In this paper, we present a technique that introduces enhancements to encryption quality and employs chaos theory to augment system robustness. The approach involves the generation of pseudo-random sequences through the utilization of the four-dimensional Lorenz system. These sequences serve a dual purpose: firstly, they facilitate image preprocessing by applying a scrambling and diffusion technique to eliminate intrinsic relationships between adjacent pixels. Secondly, a DNA-based codebook method is incorporated to further elevate encryption quality during the final step. This entails the utilization of the LUC public encryption algorithm, along with one of the generated pseudo-random sequences, as the encryption keys for safeguarding the image. The findings regarding encryption quality and the perceptibility of the resulting image yield promising outcomes. In terms of encryption quality, notably high values were attained. In regard to entropy, the achieved entropy for the encrypted images exceeded 7.9. Additionally, the range of PSNR between the plain and encrypted images remained below 8.7.

Keywords: Bioinformatics, Chaotic Map, DNA, Image Encryption, LUC.

تحسين طريقة LUC في تشفير الصور باستعمال نظام فوضوي هجين والحمض النووي

نور منعم عباس, مثيل عماد الدين عبدالمنعم
قسم علوم الحاسوب، الجامعة التكنولوجية، بغداد، العراق

الخلاصة

تعتبر الصور الرقمية أحد أكثر أنواع البيانات عرضة للهجمات السيبرانية لما تحمله هذه الصور من معلومات ولكثرة التطبيقات التي يمكن استعمالها فيها، لهذا السبب تم تطوير العديد من أنواع الحماية لهذا النوع من البيانات كالإخفاء والعلامة المائية لحماية حقوق الملكية او للتأكد من سلامة بيانات الصورة، وتقنيات التشفير لحماية محتويات الصور وللسرية. في هذا البحث نقدم تقنية جديدة تُقدم تحسينات على جودة التشفير وتستعمل نظرية الفوضى لزيادة صلابة النظام. تتضمن الطريقة إنتاج تسلسلات عشوائية زائفة باستعمال نظام لورينز رباعي الأبعاد. تخدم هذه التسلسلات غرضين: أولًا، تيسر تجهيز الصورة من خلال تطبيق تقنية الحركة

*Email: 110128@uotechnology.edu.iq

والانتشار للقضاء على العلاقات الجوهرية بين البكسلات المتجاورة. ثانيًا، يتم استعمال خصائص الحمض النووي لزيادة جودة التشفير أثناء الخطوة النهائية. ينطوي ذلك على استعمال خوارزمية التشفير العامة LUC، بالإضافة إلى أحد تسلسلات الأرقام العشوائية الزائفة المولَّدة كمفاتيح التشفير لحماية الصورة.
نتائج جودة التشفير واستدلال الصورة الناتجة تظهر نتائج واعدة. بالنسبة لجودة التشفير، تم تحقيق قيم مرتفعة بشكل ملحوظ. فيما يتعلق بالإنتروبي، تجاوزت القيمة المحققة للإنتروبي للصور المشفرة 7.9. بالإضافة إلى ذلك، تبقى نطاق نسبة الإشارة إلى الضوضاء (PSNR) بين الصورة العادية والصورة المشفرة دون الـ 8.7.

.

1. Introduction

The field of image security covers the security needs in storage, transmission, and distribution [1]. This leads to a large amount of development and research in this area. Digital images are ubiquitous, serving as communication tools, memory keepers, and marketing assets. Encryption is essential to protect privacy, intellectual property, and data integrity. They aid education, information sharing, and brand promotion. Moreover, images play a crucial role in legal cases and national security. As the world relies more on digital imagery, securing it becomes paramount. Balancing convenience with responsibility is key in a visually driven world. The most used security technique for securing images is image encryption, and it has been carried out in spatial and frequency domains. In color, gray, and binary form [2] [3], other techniques are used to protect the images, such as watermarking [4], secret sharing [5], and image hiding [6].

In general, encryption relies heavily on randomness, and the chaos system is considered one of the strongest random sequence generator systems [7] [8].
Recently, DNA's biological properties received high attention and was employed to enhance the security of cryptography and steganography systems [9] [10] [11]. Some others failed to combine DNA with a chaotic system [12] [13], as such a system needs to be built carefully to avoid worsening any encryption property (speed, randomness, correlation, statistical analysis, known attack method, etc.).
Many encryption techniques were used to encrypt an image; some proved efficient, and others failed to deal with images due to their speed, correlation, or perceptuality. One of these techniques is LUC, designed by P.J. Smith [14], a public key encryption technique based on Lucas sequences. Similar to RSA, it uses prime numbers and Lucas sequence properties for encryption and decryption. LUC offers a secure method for data protection, and its reliance on Lucas sequences adds diversity to cryptographic options, enhancing security.

The organization of the paper is as follows: the second section is about related work; the next sections are a brief description of DNA, the hyper-chaotic 4-D Lorenz system, and the LUC public encryption algorithm that were used in the proposed encryption technique, followed by a detailed description of the proposed encryption technique and the obtained results. The last section presents the conclusion and the suggested future works.

2. Related Work

In this section, a study of the related works in image encryption, bioinformatics, and image encryption using LUC is presented as follows:
Xingyaun et al. [15] proposed chaotic scrambling and RNA computing in image encryption to encrypt gray-level images. The analysis showed that this technique had a higher encryption quality compared to related works; the PSNR score was 8.8327 for the baboon image and 9.0422 for Lena.
Hua et al. [16] proposed another chaotic system and got a high encryption level. They presented a new chaotic map called 2D-LSCM derived from the existing Logistic and Sine

maps, then designed a 2D-LSCM-based image encryption algorithm (LSCM-IEA). The entropy in this work ranged around 7.901.
Pak et al. [17] employed bit-level encryption using a one-dimensional logistic map; they improved the 1D logistic map and sine map made by the output sequences of two of the same existing 1D chaotic maps. The entropy in this work was around 7.99.
Ratnadewi et al. [18] used LUC in their method but had an issue covering all the 256-pixel levels due to an inability to find sufficient prime numbers to grant the restoration of all pixel values. RSA, DES, AES [19], [20], and other techniques were used in image encryption, and good security levels were obtained.

3. DNA Concepts

Deoxyribonucleic Acid (DNA) is a genetic information container that holds and decides the genetic features in the field of molecular biology [21]. The properties of living organisms in terms of physical details and behavior status are determined by this genetic information [13] [22].

DNA is a long strand of double helix; each DNA helix is a compound called nucleotides. The triplet: deoxyribose sugar, four bases (A, G, C, and T), which stand for adenine, guanine, cytosine, and thymine, respectively, and a phosphate group are considered the components of the DNA, as illustrated in Figure 1 [23] [10].
Bioinformatics has been used in different computer science fields, including but not limited to sequence analysis, expression analysis, intellectual property rights (IPR) data, biological graphics, and security enhancement [13] [24].

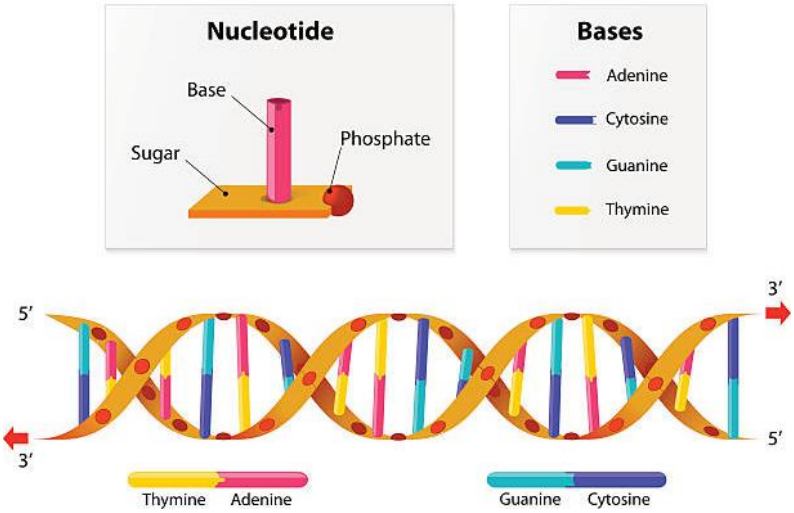


Figure 1: DNA Structure [15]

4. 4-D Lorenz System

Lorenz is one of the famous hyper-chaotic systems. It was found using ordinary differential equations studied by an American mathematician called Edward Lorenz. This system is hypersensitive to initial values and will generate totally different chaotic results by changing any initial parameter by a very small offset [25], [26], and [27].

Lorenz was used as a three-dimensional form by many subjects [28] [29], but then it was extended to cover many dimensions. A four-dimensional Lorenz system was used based on the following equations and initial parameters [25]:

$$\begin{cases} \frac{dx}{dt} = a(y-x) - ew & \dots\dots\dots(1) \\ \frac{dy}{dt} = xz - hy & \dots\dots\dots(2) \\ \frac{dz}{dt} = b - xy - cz & \dots\dots\dots(3) \\ \frac{dw}{dt} = ky - dw & \dots\dots\dots(4) \end{cases}$$

where x, y, z, and w are initial state variables and a, b, c, d, e, and h are parameters of the system [28].

5. LUC Algorithm

LUC is an RSA variant public key encryption algorithm built using Lucas function properties, and it's from this that it gets its name, "LUCas." LUC works like RSA in most details, with a difference in the way of calculating the cipher by replacing the exponential operation of RSA with the Lucas function. The following represent the general steps of the LUC encryption technique [14], [30], and [31]:

1- choose two large prime numbers p & q and $p \neq q$.
2- Compute $n = p \times q$.
3- Choose public key e where $e < n$ and e is relatively prime to s(n) where $s(n) = lcm\,(e, p-1, q-1, p+1, q+1)$
4- Calculate d where $ed \equiv 1\ mod\ s(n)$
5- Compute cipher c by repeating the following formula e times.

$$v_{i+1} = m \times v_i - Q \times v_{i-1}\ mod\ n$$

Where m= message, $v_1 = m$, $Q = 1$,
For the decryption process m is calculated using the following formula d times

$$v_{i+1} = c \times v_i - Q \times v_{i-1}\ mod\ n$$

6. Proposed Encryption Algorithm

The proposed algorithm consists of four phases: the first is to use a hyper-chaotic (4-D Lorenz system) to generate four chaotic sequences (X, Y, W, and Z); the second phase is to shuffle a pixel's positions using a shuffling technique based on X and Y chaotic sequences; the third phase is to use a DNA code book generated using W and Z chaotic sequences; and finally, the LUC encryption algorithm is used to encrypt the image. As shown in Figure 2 and stated in algorithm 1.

Algorithm 1: proposed work.
Input: Original image of size (N * M), encryption key.
Output: Encrypted image (N * M).
Begin Step 1: Generate 4-D Lorenz system hyper chaotic sequences (X_0, Y_0, Z_0, W_0). Step 2: Shuffle image pixels' rows and columns using algorithm 2, shuffle (Image, X, Y). Step 3: Code resultant image with algorithm 3, codebook (shuffled image, Z, W). Step 4: Encrypt coded image using LUC using algorithm 4, LUC (coded image, key). End.

After generating four hyper-chaotic sequences using four seeds (X, Y, Z, and W) supplied by the encryptor or derived from a key, X and Y sequences are used to generate rows and columns shuffling indices to shuffle image pixels, as stated in algorithm 2. This step can be carried out in two ways: either by splitting pixels into color bands (RGB) and swapping each color with its equivalent, or by swapping the pixel values (24 bits for color, 8 bits for grey) directly.

1- Create two indices, one for rows and one for columns (rows = {1, 2, 3, row number}).
2- Sort these indices according to the order of X and Y values incrementally.
3- Re-arrange the rows and columns of the image pixels based on these indices.

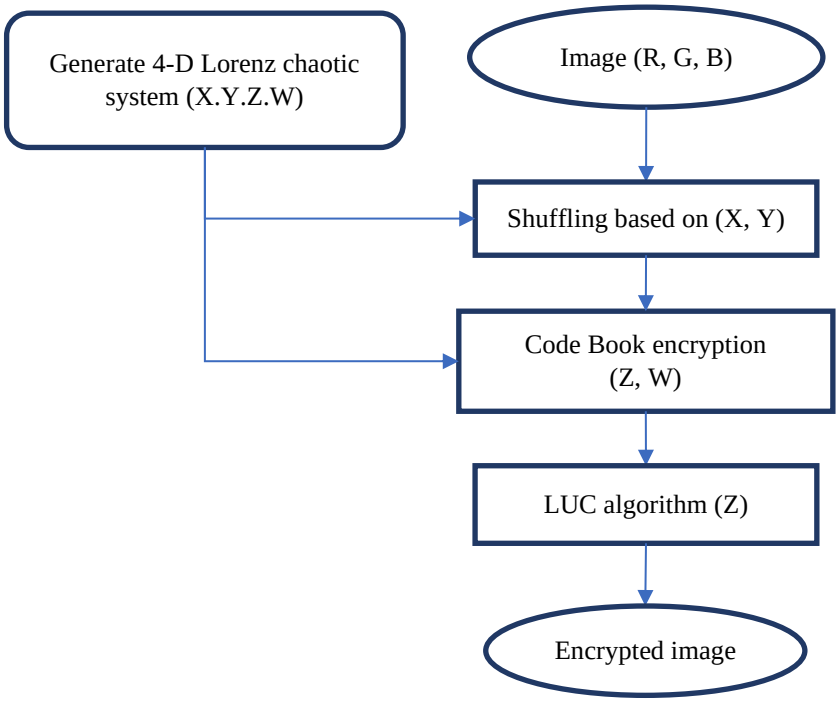


Figure 2: Proposed algorithm flowchart

Algorithms 2: Image shuffling.
Input: Original image, Hyper-chaotic sequences X and Y.
Output: Shuffled image.
Begin Step1: generate row shuffling index *row_index* table based on sequence X by sorting X incrementally and sort *row_index* accordingly. a- Create *row_index* table where table index represents image row and table value represent current row location. b- Sort X values and record each X sequence value new location in *row_index* table. c- Re-order image pixels rows based on *row_index* new values. Step2: perform same sub steps of step 1 on columns using *column_index* and Y sequence in same fashion to re-order image columns. End.

The third stage of this work starts by taking the output of the previous stage and splitting it into color bands (R, G, and B) and working on each band individually using the proposed DNA codebook as follows and as stated in algorithm 3:

1- Split the shuffled image into color bands R, G, and B.
2- Multiply sequence Z by 1012 mod 8 to generate a new random sequence of 0–7 values.
3- Encode each pixel color based on index and table 1 to convert the color value to four DNA nucleotides.
4- Multiply sequence W by 1012 mod 4 to generate a new random sequence of 0–4 values.
5- Use the DNA addition arithmetic stated in Table 2 to add pixel nucleotide values with sequence W nucleotide elements.
6- Decode nucleotides back to their color value using codebook table index 0.

Table 1: DNA codebook

	0	*1*	*2*	*3*	*4*	*5*	*6*	*7*
00	A	A	C	C	G	G	T	T
01	C	G	A	T	A	T	C	G
10	G	C	T	A	T	A	G	C
11	T	T	G	G	C	C	A	A

Table 2: DNA addition

+	*A*	*C*	*G*	*T*
A	A	C	G	T
C	C	G	T	A
G	G	T	A	C
T	T	A	C	G

Algorithm 3: Codebook coding.
Input: Shuffled image, Hyper-chaotic sequences Z and W.
Output: coded image.
Begin Step 1: generate codebook index table for every pixel color using Z sequence by multiplying Z by 10^{12} modulo 8 to get the codebook index from 0 to 7. Step 2: generate DNA addition table for every pixel color using W by multiplying W by 10^{12} modulo 4 to get the addition suffix index from 0 to 3. Step 3: divide each pixel color to 4 segments 2-bits each and refer to table 1 to code the bits using codebook index table and codebook table into nucleotides. Step 4: add each DNA nucleotide to its offset generated in step 2 using table 2.
Step 5: convert nucleotides back to pixel colors using codebook table index 0 to generate the output image. End.

Final stage is to encrypt the coded image using LUC encryption using algorithm 4.

Algorithm 4: LUC encryption.
Input: coded image, decryption key.
Output: encrypted image.
Begin Step 1: set P=23 and Q=11 and calculate n=P*Q = 253. Step 2: select encryption key e where e < n and relatively prime to S(n) where S(n) is the least common multiplier between P-1, P+1, Q-1, and Q+1. Step 3: Calculate decryption key d where $ed \equiv 1 \; mod \; S(n)$. Step 4: encrypt all pixel colors using the following formula: For i =1 to e $$v_{i+1} = m \times v_i - Q \times v_{i-1} \; mod \; n$$ Next $$c = v_i$$ End.

7. Decryption algorithm

The decryption process is straightforward for all stages except for the LUC, and it is carried out in reverse order; that is, for the first decryption stage, after generating Lorenz sequences using the same initial parameters, is LUC decryption using algorithm 5:

Algorithm 5: LUC decryption.
Input: encrypted image, decryption key.
Output: coded image.
Begin Step 1: set P=23 and Q=11 and calculate n=P*Q = 253. Step 2: select encryption key e where e < n and relatively prime to S(n) where S(n) is the least common multiplier between P-1, P+1, Q-1, and Q+1. Step 3: Calculate decryption key d where $ed \equiv 1 \; mod \; S(n)$. Step 4: decrypt all pixel colors using the following formula: For i =1 to d $$v_{i+1} = c \times v_i - Q \times v_{i-1} \; mod \; n$$ Next $$m = v_d$$ End.

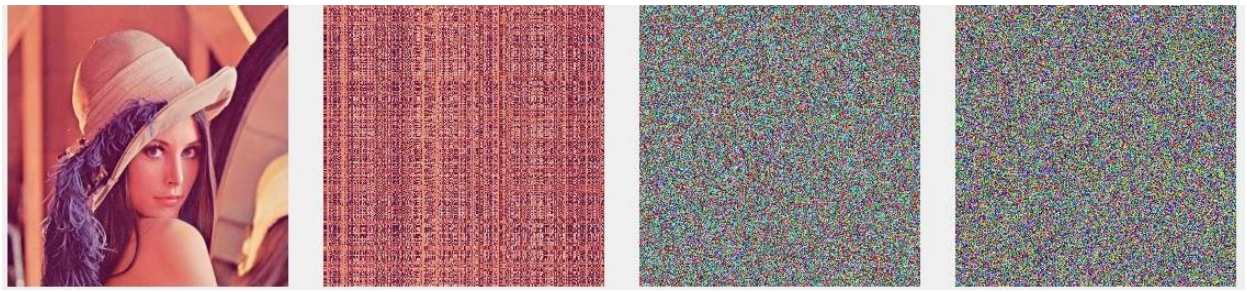
In the codebook decoding stage, all the steps are the same except for the addition, in which the complement of the addition offset is used. As stated in algorithm 6:

Algorithm 6: Codebook decoding.
Input: Coded image, Hyper-chaotic sequences Z and W.
Output: Suffled image.
Begin Step 1: generate codebook index table for every pixel color using Z sequence by multiplying Z by 10^{12} modulo 8 to get the codebook index from 0 to 7. Step 2: generate DNA addition table for every pixel color using W by multiplying W by 10^{12} modulo 4 to get the addition suffix index from 0 to 3. Step 3: divide each pixel color to 4 segments 2-bits each and refer to table 1 to code the bits using codebook index table and codebook table into nucleotides. Step 4: add each DNA nucleotide to its offset addition complement (3 - offset) generated in step 2 using table 2. Step 5: convert nucleotides back to pixel colors using codebook table index 0 to generate the output image. End.

In the de-shuffling stage, the steps are the same as the shuffling in the encryption; the result of this step is the original image.

8. Results

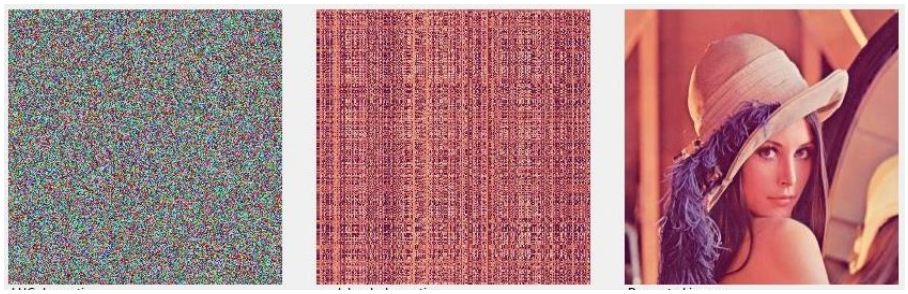
Following is the perceptual result of encryption and decryption for some images, along with the statistics of encryption and decryption.
The first image used was Lena 256*256, as shown with its encryption stages in figure 3.



(a) (b) (c) (d)

Figure 3: Lena image encryption (a- original image, b- shuffling process, c- code book DNA, and d- LUC algorithm)

The stages of the decryption process for this image are shown in Figure 4. The first image represents the inverse of LUC, while the second image represents the inverse of the codebook step, and finally, the third image represents the inverse of the shuffling step.



(a) (b) (c)

Figure 4: Lena image decryption, a-Luc decryption, b- decryption of code book, c- original image

Mean Square Error (MSE) [32], Peak Signal to Noise Ratio (PSNR) [32], Sum Absolute Difference (SAD), Encryption Quality (EQ) [32], Average Difference (AD) [32], Maximum Difference (MD), Normalized Cross-Correlation (NCC) [32], and Entropy [32] were used to assess the proposed encryption technique.

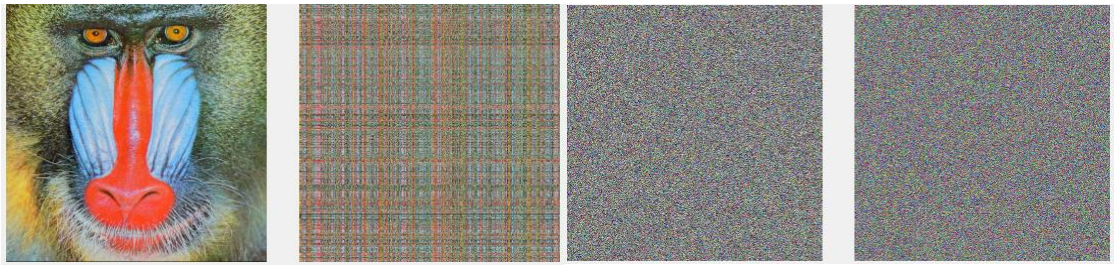
MSE for all images is 0 and PSNR is ∞, which means the decrypted image is an exact copy of the original. The same applies for SAD, MD, and AD, with a value of 1 for NCC, which is the optimal value. As seen in tables 3 and 4, the PSNR between the original and encrypted images was above 8, which means a huge difference between them.

The entropy of encrypted images is very close to the optimal distribution, which is 2^3=8.
The EQ of the image is 623.88 for a 256-pixel image (Table 3) and 1717.79 for a 512-pixel image (Table 4), which is considered a good value.

Table 3: Results of Lena 256*256

	PSNR (original & decrypted)	PSNR (original & encrypted)	MSE	*SAD*	EQ	*AD*	*MD*	*NCC* (original & decrypted)	Entropy (encrypted image)
Lena 256	∞	8.706	0	0	623.88	0	0	1	7.990

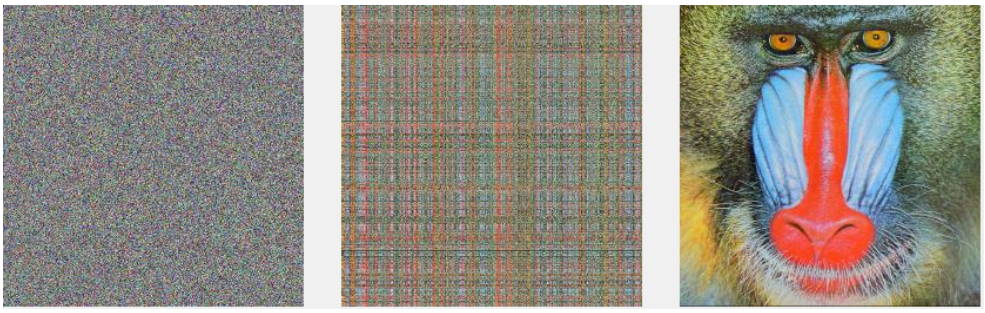
The second image used was of size 512 x 512, with its results in Figure 5 and its decryption process stages in Figure 6.



(a) (b) (c) (d)

Figure 5: 512 image encryption, a. original image, b. shuffling process, c. code book DNA, d. LUC algorithm

The decryption process is shown in Figure 6.



(a) (b) (c)

Figure 6: 512 image decryption, a-Luc decryption, b- decryption of code book, c- original image

Table 4: Results of Baboon 512 x 512

	PSNR (original & decrypted)	PSNR (original & encrypted)	MSE	*SAD*	EQ	*AD*	*MD*	*NCC* (original & decrypted)	Entropy (encrypted image)
Baboon 512	∞	8.816	0	0	1717.79	0	0	1	7.992

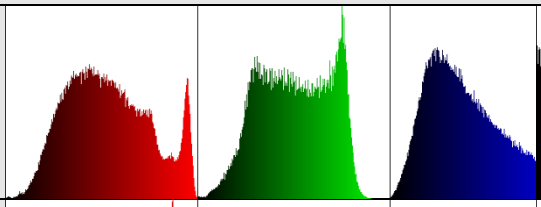
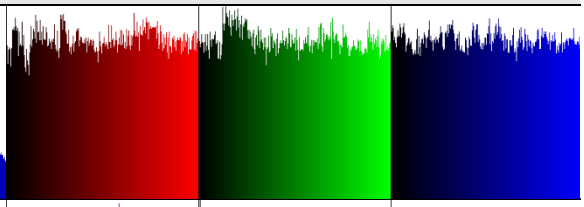
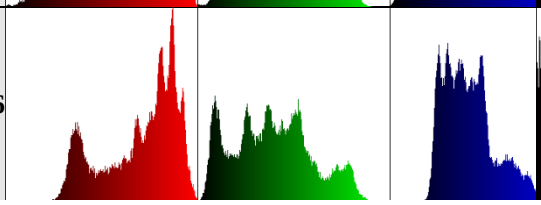
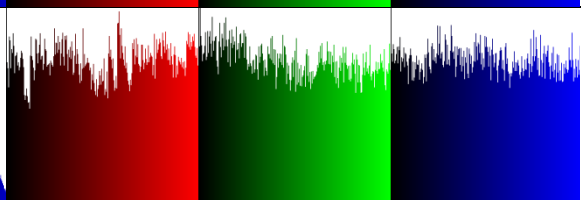
For the cross correlation measure [33], a very small value (close to zero or negative) indicates that there is no correlation within neighbor pixels. The same applies for the structural similarity index measure (SSIM) [33], which indicates that the encrypted image and the plain image are totally different if the value of SSIM is close to zero, as shown in table 5.

Table 5: Cross correlation and SSIM measures results

	VCC	HCC	DCC	SSIM
Baboon 512	-0.014	-0.0013	-0.102	0.0083
Lena 256	0.032	-0.029	0.0047	0.0088

The histogram of the plain/ciphered image shown in Table 6 shows a close-to-uniform figure, as indicated by the entropy measure, where an entropy of 8 would give a perfect uniform line in the histogram.

Table 6: Histogram of image before and after the encryption

	Plain image histogram	Encrypted image histogram
Baboon 512		
Lena 256		

According to the next table, the PSNR of the encrypted image is higher than the PSNR of a related work [34], and the entropy of the encrypted image is very close to or higher than that of other works and very close to the optimal value (8). For the encryption quality, the value is increased with the size of the image, and the achieved values are considered to be high.

Table 7: Comparison table

Encryption algorithm	*Test image*	*PSNR*	*entropy*	*EQ*
Proposed algorithm	Baboon	8.816	7.992	1717.79
REF [34]	Baboon	8.804	7.999	----------
Proposed algorithm	Lena	8.706	7.990	623.88
REF [13]	Lena	----------	7.989	----------

9. Conclusions and future work

The suggested method uses shuffling to get rid of the connection between pixels by spreading them out in the image; DNA coding makes things even more confusing by changing the values of pixels to completely different ones; and finally, the LUC algorithm makes the system more complicated and secure by adding the features of public key encryption. These techniques, combined, give the proposed algorithm more security and complexity.

For future work, a different method can be used to generate keys for each data value or combine LUC with elliptic curves to enhance security.

References

[1] G. Singh and S. Kinger, "A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security," *International Journal of Computer Applications,* vol. 67, no. 19, pp. 33-38, 2013.

[2] M. Subhedar and V. Mankar, "Current status and key issues in image steganography: A survey," *Computer Science Review,* vols. 13-14, pp. 95-113, 2014.

[3] G. Hamed, M. Marey, S. A. El-Sayed and M. F. Tolba, "Comparative study for various DNA based steganography techniques with the essential conclusions about the future research," in *11th International Conference on Computer Engineering & Systems (ICCES)*, Cairo, 2016.

[4] R. Abdulrida, M. E. Abdulmunem and A. M. Jaber, "Quantum image watermarking based on

wavelet and geometric transformation," *Iraqi Journal of Science,* vol. 61, no. 1, pp. 153-163, 2020.

[5] Z. Radeef and A. Hashim, "Multiple Image Secret Sharing based on Linear System," *Indian Journal of Science and Technology,* vol. 10, no. 33, pp. 1-17, 2017.

[6] M. Abdulmunim, N. F. Hassan and A. E. Ali, "Proposed advanced hiding method on color images based on DMWT," *European Journal of Scientific Research,* vol. 79, pp. 385-392, 2012.

[7] A. K. Jabbar, A. T. Hashim and Q. F. Al-Doori, "Secured Medical Image Hashing Based on Frequency Domain with Chaotic Map," *Engineering and Technology Journal,* vol. 39, no. 5A, pp. 711-722, 2021.

[8] A. Hamad and A. K. Farhan, "Image Encryption Algorithm Based on Substitution Principle and Shuffling Scheme," *Engineering and Technology Journal,* vol. 38, no. 3B, pp. 98-103, 2020.

[9] M. T. Sulaiman and N. F. Hassan, "Propose an Arabic CAPTCHA System," *Engineering and Technology Journal,* vol. 36B, no. 1, pp. 48-52, 2018.

[10] K. Sakib, B. Anam, M. A. Hossain and K. Dahal, "Review on the Advancements of DNA Cryptography," in *Proceedings of International Conference on Software, Knowledge, Information Management and Applications*, Paro, Bhutan, 2010.

[11] S. Hamad, A. Elhadad and A. Khalifa, "DNA Watermarking Using Codon Postfix Technique," *IEEE/ACM Transactions on Computational Biology and Bioinformatics,* vol. 15, no. 5, pp. 1605 - 1610, 2017.

[12] F. Elamrawy, M. Sharkas and . A. M. Nasser, "An Image Encryption Based on DNA Coding and 2DLogistic Chaotic Map," *International Journal of Signal Processing,* vol. 3, pp. 27-32, 2018.

[13] X. Zhang, Z. Zhou and Y. Niu, "An Image Encryption Method Based on the Feistel Network and Dynamic DNA Encoding," *IEEE Photonics Journal,* vol. 10, no. 4, pp. 1-14, 2018.

[14] P. J. Smith and M. J. J. Lennon, "LUC: A New Public Key System," *Tech. Reports,* vol. 1, no. 1, 1993.

[15] X. Wang and L. Liu, "Application of chaotic Josephus scrambling and RNA computing in image encryption," *Multimedia Tools and Applications,* vol. 80, no. 11, pp. 23337–23358, 2021.

[16] Z. Hua, F. Jin, B. Xu and H. Huang, "2D Logistic-Sine-Coupling Map for Image Encryption," *Signal Processing ,* vol. 149, pp. 148-161, 2018.

[17] C. Pak, K. An, P. Jang, J. Kim and S. Kim, "A novel bit-level color image encryption using improved 1D chaotic map," *Multimedia Tools and Applications,* vol. 78, pp. 12027–12042, 2019.

[18] R. Ratnadewi, C. D. Alpha G, D. Napitupulu and H. Nurdiyanto, "LUC Algorithm in Visual Cryptography," *Journal of Physics Conference Series,* vol. 1114, 2018.

[19] R. R. P. Adhie, Y. Hutama and A. S. Ahmar, "Implementation Cryptography Data Encryption Standard (DES) and Triple Data Encryption Standard (3DES) Method in Communication System Based Near Field Communication (NFC)," *Journal of Physics Conference Series,* vol. 954, 2018.

[20] R. Y. Hutama, R. Adhie, J. Christian and D. Wijaya, "Implementation and performance analysis of AES-128 cryptography method in an NFC-based communication system," *World Transactions on Engineering and Technology Education,* vol. 15, no. 2, pp. 178-183, 2017.

[21] K. Zhan, D. Wei, J. Shi and J. Yu, "Cross-utilizing hyperchaotic and DNA sequences for image encryption," *Journal of Electronic Imaging,* vol. 26, no. 1, pp. 1-11, 2017.

[22] H. B. Abdul Wahab and T. M. Abed, "Anti Phishing Based On Visual Cryptography And 4D Hyperchaotic System," *Iraqi Journal of Information Technology,* vol. 9, no. 1, pp. 1-27, 2018.

[23] R. A. Hussain, M. E. Abdulmunim and A. M. J. Abdul-Hossen, "Propose Image Encryption Watermarking Algorithm Based on Frequency and Geometric Transform," *2019 2nd Scientific Conference of Computer Sciences (SCCS),* pp. 143-147, 2019.

[24] A. T. Sadiq and H. S. Abdullah, "HDNA : Heuristic DNA Computing Algorithm," *Engineering and Technology Journal,* vol. 27, no. 6, pp. 1064-1073, 2009.

[25] R. Enayatifar, H. Abdullah and I. Isnin, "Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence," *Optics and Lasers in Engineering,* vol. 56, pp. 83-93, 2014.

[26] G. Alvarez and S. Li, "Some Basic Cryptographic Requirements for Chaos-Based Cryptosystems," *International Journal of Bifurcation and Chaos,* vol. 16, no. 8, pp. 2129-2151, 2006.

[27] E. N. Lorenz, "Deterministic Nonperiodic Flow," *Journal of the Atmospheric Sciences,* vol. 20, no. 2, p. 130–141, 1963.

[28] F. Zhang, X. Liao and G. Zhang, "Qualitative behaviors of the continuous-time chaotic dynamical systems describing the interaction of waves in plasma," *Nonlinear Dynamics,* vol. 88, no. 3, p. 1623–1629, 2017.

[29] O. E. Rossler, "An equation for hyperchaos," *Physics Letters A,* vol. 71, no. 2, pp. 155-157, 1979.

[30] P. Smith and C. Skinner, "A Public-Key Cryptosystem and a Digital Signature System Based on the Lucas Function Analogue to Discrete Logarithms," *ASIACRYPT,* vol. 917, pp. 355–364, 1995.

[31] Ø. Ihlen, B. v. Ruler and M. Fredriksson, Public Relations and Social Theory: Key Figures and Concepts, New York: Routledge, 2009.

[32] N. M. Abbas and M. E. Abdulmunim, "mRNA Approach Image Encryption Using LUC Algorithm," *Iraqi Journal of Science,* vol. 64, no. 5, pp. 2545-2560, 2023.

[33] G. A. Sathishkumar, K. Bhoopathy Bagan and N. Sriraam, "IMAGE ENCRYPTION BASED ON DIFFUSION AND MULTIPLE CHAOTIC MAPS," *International Journal of Network Security & Its Applications (IJNSA),* vol. 3, no. 2, 2011.

[34] X. Zhang and X. Yan, "Adaptive Chaotic Image Encryption Algorithm Based on RNA and Pixel Depth," *Electronics ,* vol. 10, no. 15, p. 1770, Jul. 2021.

[35] Z. Hua, B. Xu, F. Jin and H. Huang, "Image encryption using Josephus problem and filtering diffusion," *IEEE,* vol. 7, pp. 8660-8674, 2019.