International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 12, pp 158-163, December-2023

https://doi.org/10.47001/IRJIET/2023.712022

Using the Chaotic to Improve the RC4 Algorithm

[1]Noor M. Hussein, [2]Nadia M. Mohammed

[1,2]Software Engineering Department, College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq

***Abstract* - Image encryption plays a crucial role in ensuring the security and confidentiality of digital image data. This study explores an image encryption approach that employs the RC4 algorithm with the chaotic Logistic Map and with varying key lengths. The encryption quality assesses using metrics such as Peak Signal-to-Noise Ratio (PSNR), Mean Square Error (MSE), The Normalized Cross Correlation Coefficient (NCC), Structural Similarity Index Measure (SSIM), and Histogram. Results demonstrated that the Logistic Map-based encryption method successfully balanced image quality and security. The study's contributions lie in offering valuable insights into image encryption techniques, providing a practical method for secure image communication and storage.**

Keywords: Image Encryption, Logistic Map, Chaotic System.

I. INTRODUCTION

Advancements in communication and multimedia technologies in computer networks and the internet have raised concerns about unauthorized access to highly sensitive media data, necessitating a robust and efficient security system. Encryption is a vital method for safeguarding information. Image encryption, in essence, involves converting information into an unreadable format using an algorithm and a private key. This approach is consistent with the Kirchhoff principle, which entails having a known algorithm and a secret key [1] .Traditional encryption algorithms like AES, DES, and RSA were initially employed in image encryption. However, these methods proved to be less efficient for modern applications, primarily due to their extensive computational requirements, which resulted in substantial latency [2].

Chaos is a defining feature of systems where the current state is highly sensitive to prior states, initial conditions, or both. This sensitivity makes predicting the behavior of chaotic systems challenging. Chaos theory explains and formulates apparent disorder in chaotic systems [3]. Chaotic systems, maps, and sequences have been pivotal in various applications, including security in smart networks and communication systems. Chaotic encryption, in particular, has been used to secure different types of content, including images [4].

Encryption can be comprehensively arranged into two classifications: block encryption and stream encryption. In block encryption, a bunch of plaintext numbers is handled all the while, as exemplified by the High-level Encryption Standard (AES). While stream encryption scrambles each plaintext number separately, making it appropriate for ongoing information [5]. An example of stream encryption is RC4, known for its capacity to scramble pictures with lower asset prerequisites and clear intricacy quickly. Nonetheless, notwithstanding its benefits, RC4 has legitimate shortcomings [1]. The reason for this study is to distinguish and upgrade the security weaknesses present in the RC4 calculations.

In this work, we further enhance the RC4 encryption algorithm by incorporating it with the chaotic logistic map. The logistic map is a dynamic system that exhibits chaotic behavior and is known for its sensitivity to initial conditions and parameter values. By using the chaotic logistic map into RC4, we introduce an additional layer of complexity randomness and unpredictability to the encryption process. This enhanced RC4 encryption, which combines the strengths of the RC4 stream cipher with the chaotic dynamics of the logistic map, aims to provide improved security for image encryption and other applications.

II. RELATED WORKS

Several modern encryption techniques incorporate chaotic systems with the RC4 algorithm for image encryption. Here are some key studies in this field:

Kumari & Gupta (2018) introduced an image encryption scheme that combines chaotic maps (3D maps) with RC4 for confusion and diffusion [3]. While lani & Al Iesawi (2018) proposed a compression-based image encryption method that combines RC4 with the Henon map [4], also Sahib et al. (2018) enhanced RC4 key generation by using multiple chaotic maps, improving secrecy and randomness [5].

Jawad (2022) introduced a secure image encryption scheme based on RC4 and chaotic Henon and Sine maps for confusion and diffusion stages [6]. D. W. Ahmed et al. (2021) presented a dual-level multiple chaotic map-based algorithm that combines Linear Double Multiple Chaotic Maps with RC4, achieving strong resistance against various attacks [7]. Kumer et al. (2022) proposed an image encryption technique

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 12, pp 158-163, December-2023

https://doi.org/10.47001/IRJIET/2023.712022

that combines RC4 with the Arnold Chaotic Map, preserving pixel distribution and resisting differential attacks [8].

III. RC4 ENCRYPTION ALGORITHM

RC4, which stands for Rivest Cipher 4, is a widely used symmetric stream cipher encryption algorithm. It was developed by Ron Rivest in 1987 and has been employed in various security protocols and applications, including Secure Sockets Layer (SSL) and Wired Equivalent Privacy (WEP) for wireless networks. RC4 is known for its simplicity, efficiency, and speed, making it a popular choice for encrypting data. The algorithm operates by using a variable-length key (typically ranging from 40 to 256 bits) to generate a pseudorandom keystream. This keystream is then combined with the plaintext data using a simple bitwise XOR operation, resulting in ciphertext.

The strength of RC4 lies in the secrecy of the key, as the algorithm itself is relatively straightforward. However, its security has been a subject of debate and concern over the years, leading to the development of enhanced versions and recommendations for more robust encryption techniques in modern applications [9]. RC4 Methodology shown in Figure (1).

IV. LOGISTIC MAP CHAOS

The Chaotic Logistic Map, also known as the Logistic Map, is a mathematical function that exhibits chaotic and unpredictable behavior. It is used in various fields, including chaos theory, cryptography, and modeling complex systems. The logistic map is defined by the recurrence relation:

$$X_{n+1} = \lambda X_n (1 - X_n) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (1)$$

X_n represents the population at time and λ is a parameter that controls the rate of population growth. The logistic map is often used in the context of modeling population dynamics, where it illustrates how a population can exhibit chaotic and unpredictable behavior, even with simple rules [11].

In cryptography, chaotic logistic maps are utilized as a source of pseudorandom numbers, which are essential for encryption and data security. The chaotic behavior of the logistic map makes it difficult for an attacker to predict the sequence of numbers generated, adding an extra layer of security to encryption algorithms that rely on such maps.

Chaotic logistic maps are an example of how chaos theory can be harnessed for practical applications in various fields [12].

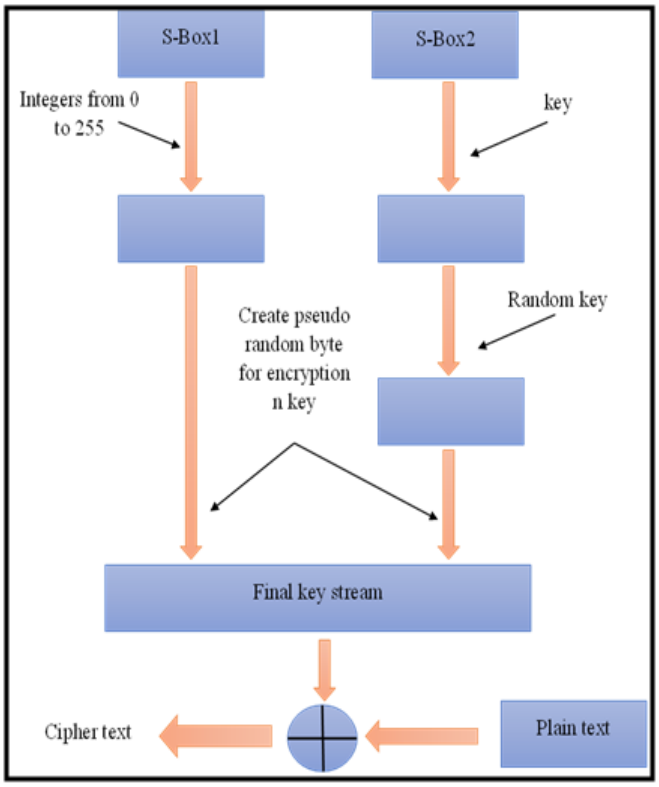

Figure 1: RC4 Encryption Methodology [10]

V. PROPOSED MODEL

5.1 Model Block Diagram

In proposed method, the using of chaotic logistic map in generating the key of RC4 encryption offers another level of complication in addition to increase security. The generated key by the chaotic is more resistant to cryptographic assaults by taking advantage of the chaotic map's unpredictability and sensitivity to initial conditions. This method improves the RC4 algorithm's overall security posture by adding a degree of unpredictability that guarantees a more resilient encryption scheme.

To encrypt an image using the proposed method, the user needs to perform the following steps:

1. Load the image.
2. Specify the Logistic Map parameters.
3. Select the length of key for RC4 Algorithm.
4. Generate the key based on the the Logistic Map.
5. Execute the RC4 algorithm (using the key in 4.) to encrypt the image.
6. Display the encrypted image.
7. Save the encrypted image. See the figure (2).

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 12, pp 158-163, December-2023

https://doi.org/10.47001/IRJIET/2023.712022

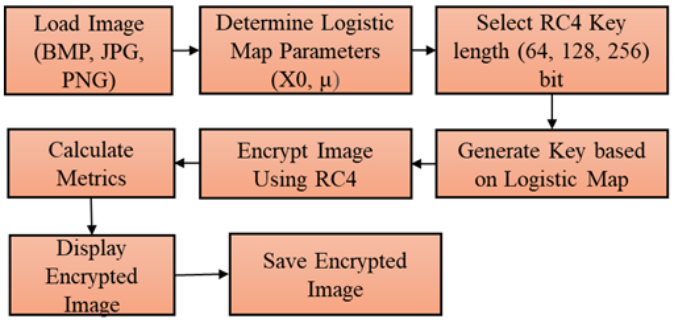

Figure 2: Encryption Process in a Proposed Method

Decryption is the reverse process of encryption and involves restoring the original information from its encrypted form. To decrypt an image using the proposed method, the user needs to perform the following steps:

1. Load the encrypted image.
2. Specify the same Logistic Map parameters that used in encryption process.
3. Select the same key length as used during encryption process.
4. Generate the decryption key using the same Logistic Map parameters as used during encryption process.
5. Execute the RC4 algorithm (using the key in 4.) to decrypt the encrypted image.
6. Display the decrypted image.
7. Save the decrypted image. See the figure (3).

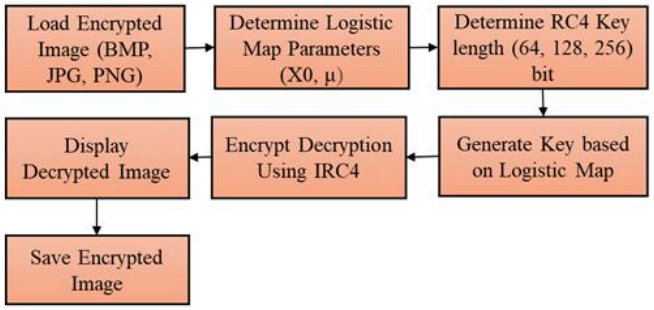

Figure 3: Decryption Process in a Proposed Method

5.2 Model Graphical User Interface

The GUI simplifies interaction with the software, enabling users to control and access different functions.

The proposed method's GUI workflow is illustrated in Figure (4) It begins by loading an image according to the chosen extension through the selection keys and then clicking the "Read Color Image" button to display the image on the axis. Logistic map parameters are then specified using text boxes for input. Key length selection is made through the selection keys. The "Generate Key" button generates the encryption key based on the logistic map parameters. The "Encryption" button encrypts the image using the generated key and displays it on the designated axis. The "Save Encrypted Image" button saves the encrypted image. After encryption, evaluation metrics are displayed. The decryption process is the reverse of encryption. In decryption process, we use the same key that used in encryption process and following a similar mechanism as explained in the decryption section.

Figure 4: Model GUI

VI. RESULTS

In order to test the designed system, a color images with dimensions of (512×512) pixels were used, as illustrated in Figure (5).

Figure 5: Sample images

The method's performance on a BMP image with different key lengths was assessed using several evaluation metrics. With a 64-bit key, the method achieved a low PSNR (Peak Signal-to-Noise Ratio) of 7.9707, a relatively high MSE (Mean Squared Error) of 10375.55, a negative NCC (Normalized Cross-Correlation) of -0.00127, and an SSIM (Structural Similarity Index) of 0.12077. Similarly, for a 128-bit key, the PSNR was 7.9710, the MSE was 10374.90, the NCC was -0.00220, and the SSIM was 0.12068. When using a

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 12, pp 158-163, December-2023

https://doi.org/10.47001/IRJIET/2023.712022

256-bit key, the value of PSNR increased slightly and become to 7.9799, the MSE equal to 10353.67, the NCC became positive at 0.001775, and the SSIM increased to 0.12142. These results indicate that the method's performance is consistent across different key lengths, with the PSNR and the MSE values reflecting the image quality. The NCC values suggest a weak correlation, while the SSIM values point to there is not moderate structural similarity. Table (1) & Figure (6) shows a comprehensive evaluation to the method's performance when applied to BMP images with varying key lengths.

Table 1: BMP Image Results

Key Length	PSNR	MSE	NCC	SSIM
64- Bit	7.9707	10375.55	-0.00127	0.12077
128- Bit	7.9710	10374.90	-0.00220	0.12068
256- Bit	7.9799	10353.67	0.00178	0.12142

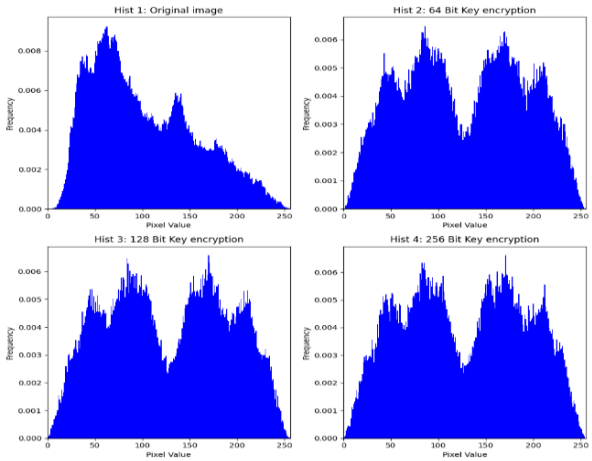

Figure 6: BMP Image Histogram Before and After Encryption using Different Key lengths

The method's performance was further assessed when applied to a PNG image with varying key lengths and using multiple evaluation metrics as shown in Table (2) and Figure (7). With a 64-bit key, the method achieved a PSNR (Peak Signal-to-Noise Ratio) of 7.9818, indicating relatively good image quality. The Mean Squared Error (MSE) was 10349.1148, which is relatively high and reflects distortion in the image. However, the Normalized Cross-Correlation (NCC) was -0.002005, suggesting a weak negative correlation, and the Structural Similarity Index (SSIM) was 0.084992, indicating structural differently.

When using a 128-bit key, the PSNR remained consistent at 7.9817, with an MSE of 10349.3454, reflecting the distortion. The NCC improved slightly to -0.000647, indicating a slightly weaker negative correlation, but the SSIM decreased to 0.008529, suggesting reduced structural similarity. With a 256-bit key, the PSNR increased to 7.9859, signifying improved image quality, while the MSE decreased to 10339.3955, indicating minimal distortion. The NCC became positive at 0.00036117, showing a weak positive correlation, and the SSIM increased to 0.085486.

Table 2: PNG Image Results

Key Length	PSNR	MSE	NCC	SSIM
64- Bit	7.9818	10349.11	-0.00201	0.08499
128- Bit	7.9817	10349.35	-0.00065	0.00853
256- Bit	7.9859	10339.40	0.00036	0.08549

The method's performance on a JPG image, assessed using various evaluation metrics, produced interesting results with different key lengths. When using a 64-bit key, the method achieved a PSNR (Peak Signal-to-Noise Ratio) of 8.0018, indicating relatively good image quality. The Mean Squared Error (MSE) was 10301.4693, signifying distortion in the image. However, the Normalized Cross-Correlation (NCC) was -0.0024724, indicating a weak negative correlation, and the Structural Similarity Index (SSIM) was 0.085124, indicating moderate structural differently.

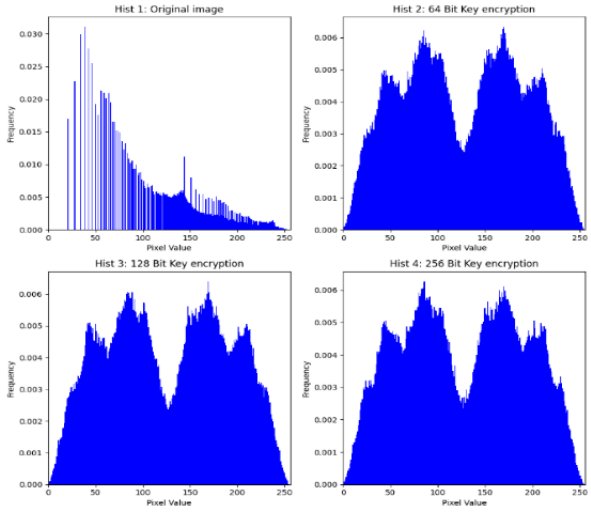

Figure 7: PNG Image Histogram Before and After Encryption using Different Jey lengths

With a 128-bit key, the PSNR remained quite high at 8.0005, with an MSE of 10304.6193, reflecting distortion. The NCC equal to -0.0011529, showing weak negative correlation, and the SSIM was 0.085385, indicating structural differently.

Using a 256-bit key resulted in a PSNR of 8.0065, which represents an improvement in image quality. The MSE was reduced to 10290.3304, indicating minimal distortion. The NCC became positive at 0.0002453, signifying a weak positive correlation, and the SSIM was 0.085473, indicating structural differently. Table (3) and Figure (8) shows results of evaluation metrics on JPG image.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 12, pp 158-163, December-2023

https://doi.org/10.47001/IRJIET/2023.712022

Table 3: JPG image Results

Key Length	PSNR	MSE	NCC	SSIM
64- Bit	8.0018	10301.4693	-0.0024724	0.085124
128- Bit	8.0005	10304.6193	-0.0011529	0.085385
256- Bit	8.0065	10290.3304	0.0002453	0.085473

The pronounced disparities in the histogram distribution of an image before and after encryption reveal a deliberate and effective disruption introduced by the encryption technique. This intentional alteration, attributed to chaotic systems, enhances the security of the image resist potential attacks.

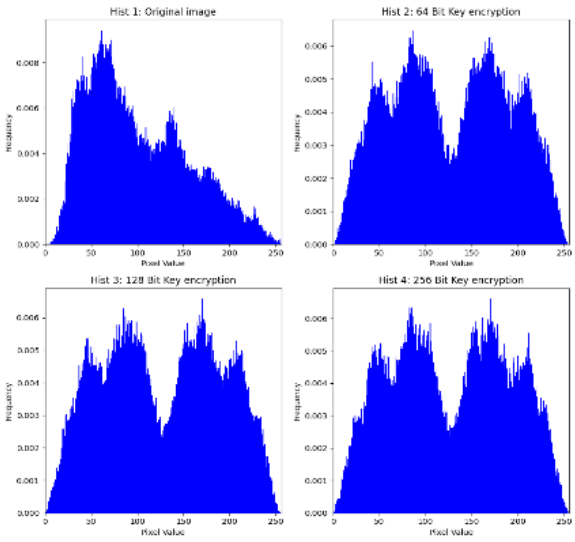

Figure 8: JPG Image Histogram Before and After Encryption using Different Jey lengths

In the realm of image encryption methodologies, several studies have explored diverse approaches, each implementing different chaotic systems and key lengths, with the primary evaluation metric being PSNR (Peak Signal-to-Noise Ratio) to gauge the quality of the encrypted images. To compare these approaches, we can observe the following results: Jawad (2022) employed a combination of the Henon and Sine maps with a 256-bit key, resulting in a PSNR of 8.5253 dB, which demonstrated superior image quality due to the innovative use of chaotic systems. D. W. Ahmed et al. (2021) used a Pseudo Random-Number-Generator (PRNG) with a 256-bit key, achieving a PSNR of 8.6675 dB, indicating strong image quality preservation.

In the present study, the logistic chaotic map was the focal point, and varying key lengths (64-bit, 128-bit, and 256-bit) were tested, yielding PSNR values of 8.002 dB, 8.001 dB, and 8.007 dB, respectively. This research underscored the importance of selecting an appropriate chaotic map and key length to balance image encryption security and quality, thereby ensuring the specific needs of the application are met.

Table (4) displays the comparison between the performances of different techniques with present study using key length is equal 256.

Table 4: Comparison between the Different Techniques

Researcher	Method	PSNR
[6]	Henon & Sine	dB8.5253
[7]	PRNG	8.6675 dB
Present Study	Logistic Map	8.007dB

VII. CONCLUSIONS

In conclusion, this study delved into the realm of image encryption, focusing on the utilization of a Logistic Map as the core chaotic system, exploring different key lengths (64-bit, 128-bit, and 256-bit) to evaluate its impact on the image encryption process. The results showed that the Logistic Map-based encryption approach achieved favorable PSNR values, indicative of maintaining image quality, while providing a robust security mechanism. These findings emphasize the importance of selecting an appropriate chaotic system and key length to strike a balance between encryption strength and image quality, aligning with the specific requirements of the application. Moreover, it was highlighted that the Logistic Map can be a viable choice for image encryption in scenarios where preserving image quality is of paramount importance alongside security. Overall, this research contributes valuable insights into image encryption methodologies and offers a practical method for secure image transmission and storage.

REFERENCES

[1] R. Mohammed, “Secure Image Encryption Scheme Using Chaotic Maps and RC4 Algorithm,” *Solid State Technol.*, 2020.

[2] B. Zolfaghari and T. Koshiba, “Chaotic Image Encryption: State-of-the-Art, Ecosystem, and Future Roadmap,” *Appl. Syst. Innov.*, vol. 5, no. 3, pp. 1–38, 2022, doi: 10.3390/asi5030057.

[3] M. Kumari and S. Gupta, “A Novel Image Encryption Scheme Based on Intertwining Chaotic Maps and RC4 Stream Cipher,” *3D Res.*, vol. 9, Mar. 2018, doi: 10.1007/s13319-018-0162-2.

[4] D. S. Alani and S. A. Al Iesawi, “Image encryption algorithm based on RC4 and Henon map,” *J. Theor. Appl. Inf. Technol.*, vol. 96, no. 21, pp. 7065–7076, 2018.

[5] N. M. Sahib, A. H. Fadel, and N. S. Ahmed, “Improved RC4 Algorithm Based on Multi-Chaotic Maps Improved RC4 Algorithm Based on Multi-Chaotic

ISSN (online): 2581-3048
Volume 7, Issue 12, pp 158-163, December-2023
https://doi.org/10.47001/IRJIET/2023.712022

Maps," *Res. J. Appl. Sci. Eng. Technol.*, vol. 15, no. 1, 2018, doi: 10.19026/rjaset.15.5285.

[6] L. M. Jawad, "Secure Image Encryption Scheme Using Chaotic Maps and RC4 Algorithm," *Solid State Technol.*, no. February, 2022.

[7] D. W. Ahmed, T. M. Jawad, and L. M. Jawad, "AN EFFECTIVE COLOR IMAGE ENCRYPTION SCHEME BASED ON DOUBLE PIECEWISE LINEAR CHAOTIC MAP METHOD AND RC4 ALGORITHM," *J. Eng. Sci. Technol.*, vol. 16, no. 2, pp. 1319–1341, 2021.

[8] N. Sharma, "Implementation of Stream Cipher RC4 in the Regime of Quantum Communication with QKD," pp. 0–7, 2023.

[9] M. Abd ulkareem and I. Q. Abduljaleel, "Analysis of a Modified on Rivets Cipher (RC4) Algorithm by Chaotic Algorithm," *J. Educ. Coll. Wasit Univ.*, vol. 1, no. 27, pp. 473–484, 2017, doi: 10.31185/eduj.vol1.iss27.72.

[10] H. Kholidy and K. Alghathbar, A New Accelerated RC4 Scheme using "Ultra Gridsec" and "HIMAN" and use this Scheme to secure "HIMAN" Data. 2009. doi: 10.1109/IAS.2009.140.

[11] S. Chen, S. Feng, W. Fu, and Y. Zhang, "Logistic map: Stability and entrance to chaos," *J. Phys. Conf. Ser.*, vol. 2014, no. 1, 2021, doi: 10.1088/1742-6596/2014/1/012009.

[12] Ł. Pawela and K. Życzkowski, "Matrix logistic map: fractal spectral distributions and transfer of chaos," arXiv:2303.06176v1, pp. 1–8, 2023, [Online]. Available: http://arxiv.org/abs/2303.06176.

Citation of this Article:

Noor M. Hussein, Nadia M. Mohammed, "Using the Chaotic to Improve the RC4 Algorithm" Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 7, Issue 12, pp 158-163, December 2023. Article DOI https://doi.org/10.47001/IRJIET/2023.712022

****

