DOI: https://doi.org/10.52436/1.jutif.2024.5.2.1470
p-ISSN: 2723-3863
e-ISSN: 2723-3871

A ROBUST AND IMPERCEPTIBLE FOR DIGITAL IMAGE ENCRYPTION USING CHACHA20

Widhi Bagus Nugroho*[1], Ajib Susanto[2], Christy Atika Sari[3], Eko Hari Rachmawanto[4], Mohamed Doheir[5]

[1,2,3,4]Informatics Engineering, University Dian Nuswantoro, Indonesia
[5]Universiti Teknikal Malaysia Melaka, Malaysia
Email: [1*]widhi.bagoes.wb@gmail.com, [2]ajib.susanto@dsn.dinus.ac.id, [3]christy.atika.sari@dsn.dinus.ac.id, [4]eko.hari@dsn.dinus.ac.id, [5]doheir@utem.edu.my

(Article received: October 12, 2023; Revision: November 16, 2023; Published: April 04, 2024)

Abstract

*In the current era, data security is mandatory because it protects our personal data from being used by irresponsible people. The objective of this research is to show the robustness of the method we propose to encrypt images using the chacha20 algorithm which is included in the symmetric encryption cryptography technique and uses one key for both encryption and decryption processes. we use the encryption method by reading the bits from a digital image which is processed using the chacha20 algorithm to get the results of the digital image encryption***.** *The results of this study indicate that the Chacha20 algorithm is secure to use when encrypting and decrypting digital images. The average MSE value generated by the chacha20 algorithm is 0.1232. The average PSNR value is 57.4784. The average value of UACI is 49.99%. The average value of NPCR is 99.602%. The test values were acquired by executing encryption and decryption processes on 5 distinct colour digital images with different size. Additionally, this study displays histograms for the original digital image, as well as for the encrypted and decrypted digital images, illustrating the pixel distribution in each. The histogram also serves as material for analysis of the success of the encryption and decryption processes in digital images.*

Keywords: *Cryptography, Chacha20, MSE, NPCR, PSNR, UACI*

1. INTRODUCTION

Data has several types, for example text, images, audio and video. Data communication using digital media is now growing but occurs over an unsecured web network [1]. In communication, the data being transmitted is not always public; some of it is private due to confidentiality. There are several ways to hide secret data, namely cryptography and steganography [2], [3]. Cryptography is the mathematical art applied to the process of encryption and decryption. The goal of this mechanism is to encrypt secret data using different combinations of words, numbers, or phrases. The security of encrypted data depends entirely on two critical aspects: the strength of the cryptographic algorithm and the secrecy of the key [4].

Cryptography has two encryption techniques: stream cipher and block cipher. Based on the type of key, there is symmetric encryption and asymmetric encryption. Symmetric encryption uses one key to encrypt and decrypt data [5]. Asymmetric encryption uses a pair of private and public keys for encryption and decryption processes [6]. The difference between stream cipher and block cipher lies in the process of encrypting plaintext into ciphertext [7]. Stream ciphers encrypt and decrypt each bit of data, while block ciphers represent complete data blocks processed at once. The main focus in evaluating block ciphers is the block size, key size, number of rounds, and structure type [8]. ChaCha20 is a member of the stream cipher family [9]. This algorithm requires a key and a nonce as input to generate a keystream, which is then subjected to Exclusive OR (XOR) operations with the plaintext (original data) to produce ciphertext [10]. Encryption of digital images using the ChaCha20 algorithm involves reading the digital image as a bitstream, performing an Exclusive OR (XOR) operation between the bitstream and the already generated keystream, resulting in an encrypted digital image. The decryption process is carried out to restore the digital image to its original form using the same encryption key.

Research with a similar theme has been carried out previously, such as research conducted by Mustafa Hussein Taha, and Jamal Mustafa Al-Tuwaijari in 2021, in this research proposing improvements to the stream cypher chaha20 algorithm based on the tent and Chebyshev functions with the aim of increasing the security layer of the ChaCha20 algorithm [11]. The results of this research's histogram analysis of images of baboon and pappers show that the histogram is distributed randomly, the MSE value is between 7690-7740, and the PSNR value is between 9.24-9.26. Another research conducted by Mohammed Salih Mahdi, Raghad Abdulaali Azeez, and Nidaa Falih Hassan in

2020, they used the ChaCha20 algorithm to encrypt images combined with hyperchaotic maps [7]. The results of this research's histogram analysis of baboon images show that the histograms are distributed randomly.

This study perform encryption and decryption of digital images using ChaCha20 stream cipher algorithm. The way a stream cipher works is bit by bit in encrypting and decrypting data, is similar to a flowing data stream. Their main advantage lies in processing speed [12]. The study aims to determine the performance of the ChaCha20 algorithm in encrypting and decrypting digital images. There are three security analyses of encryption and decryption in digital images. The security analyses encompass differential attack analysis using parameters such as the Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI). Additionally, an analysis of encryption quality is conducted using parameters like Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). Statistical analysis employing image histogram parameters is also included in this study. [13].

2. RESEARCH METHOD

This research utilized the ChaCha20 algorithm to encrypt images. ChaCha20 encrypts images using a combination of a key and a nonce, performing a series of operations like XOR, rotation, and addition to generate a keystream. The generated keystream is used for the encryption process by XOR-ing it with the plain image, resulting in the encrypted image.

The process of decryption closely resembles that of encryption. The keystream generated during the encryption stage is repurposed for decryption. The encrypted image undergoes XOR operation with the keystream to obtain the plain image again, also known as the decrypted image. ChaCha20 is a stream cipher, enabling encryption and decryption through the same XOR operations.

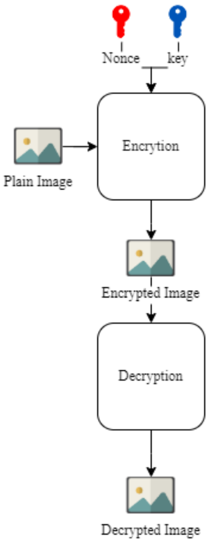

Figure 1. General Process

The research method can be seen in Figure 1. First, the image will be read as a bitstream, then the image bits will be encrypted using the ChaCha20 algorithm by entering the key and nonce to produce an encrypted image. with the same key, the decryption process for the encrypted image is carried out to return the image to its original form.

2.1. Chacha20

Chacha20 is one of the stream cipher members known for its excellent performance, fast, secure, and simple encryption [9]. The digital image will be converted into a bit form and then processed using the Chacha20 algorithm. For Chacha20, a 32-byte key, a 4-byte counter, and a 12-byte nonce are necessary inputs for the input matrix. These inputs collectively transform into a keystream, as illustrated in Table 1. [9].

Table 1. Input Matrix

Const	Const	Const	Const
Key	Key	Key	Key
Key	Key	Key	Key
Count	Nonce	Nonce	Nonce

Chacha20 has three main processes that involve addition operations, XOR processes, and rotations. The total number of rotations in Chacha20 is 20 rotations, which is the maximum of security level of the Chacha algorithm. The amount of diffusion in each rotation in Chacha increases. Chacha20 uses identical computational processes in column and diagonal forms as shown in Figure 2 [7], [14].

Column Form	Diagonal Form
QR(x 0 , x 4 , x 8 , x 12)	QR(x 0 , x 5 , x 10 , x 15)
QR(x 1 , x 5 , x 9 , x 13)	QR(x 1 , x 6 , x 11 , x 12)
QR(x 2 , x 6 , x 10 , x 14)	QR(x 2 , x 7 , x 8 , x 13)
QR(x 3 , x 7 , x 11 , x 15)	QR(x 3 , x 4 , x 9 , x 14)

Figure 2. Quarter Function

During the encryption process, the variables 'a, b, c, and d' represent intermediate values that have been calculated. Using XOR operations and left rotation on the ChaCha20 matrix elements these intermediate values are derived. The equation calculates the value of a new variable based on the value of the previous variable as shown in Figure 3.

$$\begin{array}{ll} a = a + b, & d = (d \oplus a) \ll 16 \\ c = c + d, & b = (b \oplus c) \ll 12 \\ a = a + b, & d = (d \oplus a) \ll 8 \\ c = c + d, & b = (b \oplus c) \ll 7 \end{array}$$

Figure 3. Process Function

2.2. Number of Pixels Change Rate (NPCR)

The Number of Pixels Change Rate (NPCR) is a criterion used to evaluate the encryption of digital images against differential attacks [15]–[17]. This testing is conducted to examine the effect of pixel changes on the entire digital image during the encryption process [18] NPCR demonstrates the percentage of differing pixels between E1 and E2 and can be calculated using the formula below [19]:

$$D(i,j) = \begin{cases} 0 & E1(i,j) = E2(i,j) \\ 1 & E1(i,j) \neq E2(i,j) \end{cases} \quad (1)$$

$$NPCR = \frac{1}{W \times H} \sum_{i=1}^{H} \sum_{j=1}^{W} D(i,j) \times 100\% \quad (2)$$

The height and width of the digital image are denoted by the symbols H and W, respectively. The original digital image is represented by the symbol E1, while the encrypted digital image is represented by the symbol E2. The letter D represents the total pixel discrepancy between the original digital image E1 and E2. The pixel values of original and encrypted digital images are denoted by 'i' and 'j', respectively, in row i and column j. The ideal value for NPCR testing should be greater than 99.6% or close to 1[20].

2.3. Unified Avarage Changing Intensity (UACI)

Unified Average Changing Intensity (UACI) is a criterion utilized to evaluate the encryption of digital images against differential attacks [15]–[17]. UACI measures the intensity of the difference between E1 and E2 and can be calculated using the formula below [19]:

$$UACI = \frac{1}{W \times H} \sum_{i=1}^{H} \sum_{j=1}^{W} \frac{|E1(i,j) - E2(i,j)|}{255} \times 100\% \quad (3)$$

Symbol H represent the height of the digital image and symbol W represent the width of the digital image. The original digital image is represented by the symbol E1, while the encrypted digital image is represented by the symbol E2. The pixel values in the original and encrypted digital images are denoted by 'i' and 'j', respectively.

2.4. Mean Squared Error (MSE)

Mean Squared Error (MSE) is method used for testing digital images that employs specific parameters to calculate the approximate difference in pixel values between the original digital image and the encrypted digital image at corresponding pixel locations [4]. The accuracy is considered good when the MSE value is low and poor when the MSE value is high [21], [22]. The calculation of MSE for a digital image can be expressed as follows [4]:

$$MSE = \sum_{m=1}^{M} \sum_{n=1}^{N} [P(m,n) - P'(m,n)]^2 \quad (3)$$

Symbols M represent length of digital image and symbol N represent width of the digital image. P (m, n) represents the original image with M as length and N as width. P' (m, n) represents the encrypted image with M as length and N as width.

2.5. Peak Signal-to-Noise Ratio

Peak Signal-to-Noise Ratio (PSNR) is a method used for testing digital images obtained from the MSE values of the digital image [4]. The relationship between PSNR and MSE is such that a higher PSNR value corresponds to a lower MSE value, indicating a higher accuracy level [13]. In summary, the PSNR value directly correlates with the accuracy of the digital image, whereas the MSE value is inversely proportional to the accuracy of the digital image. The accuracy is considered good when the PSNR parameter is high and poor when the PSNR parameter is low [21], [22]. The calculation of the PSNR value for a digital image can be expressed as follows [4]

$$PSNR = 20 \times \log_{10}\left(\frac{255}{\sqrt{MSE}}\right) \quad (3)$$

2.6. Histogram Analysis

Histogram is used to visually analyze digital images [23]. Moreover, the histogram displays the pixel distribution in a digital image. The histogram of an encrypted digital image should exhibit an even distribution, making it difficult to discern any clear data pattern. This prevents third parties from making informed guesses about the information contained in the digital image. A good encryption process will produce a histogram that is different from the original image’s histogram. Conversely, the decrypted digital image’s histogram should resemble the original digital image’s histogram [24], [25].

3. RESULT

In this study, 5 digital image datasets with varying pixel sizes were utilized. All of these digital images are colored. Each of these digital images will undergo a series of tests to determine the performance of the ChaCha20 algorithm in encrypting and decrypting digital images.

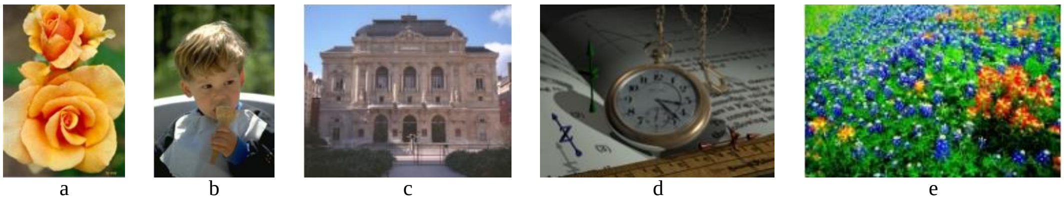

Figure 4. Dataset Images (a) brandyrose, (b) kid, (c) opera, (d) watch, (e) wildflowers

Before testing, the five digital image datasets will undergo an encryption process using the ChaCha20 algorithm. The results of digital image encryption from this dataset can be seen in Figure 5. Visually, the five digital image datasets are well encrypted. Every pixel of each digital image is properly scrambled. the image will be difficult to guess visually.

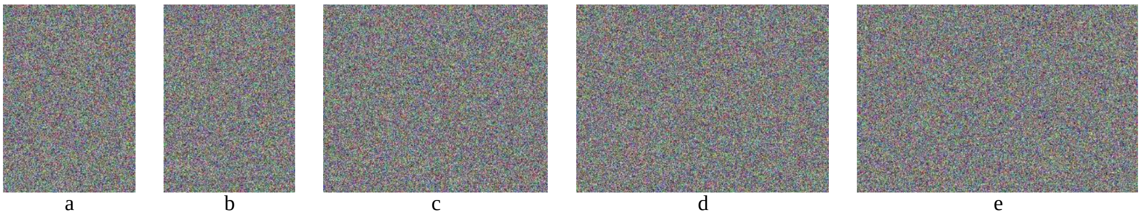

Figure 5. Encrypted Images (a) brandyrose, (b) kid, (c) opera, (d) watch, (e) wildflowers

After carrying out the encryption process, the five digital image datasets will undergo a decryption process. The results of digital image decryption can be seen in Figure 6. Visually, the decryption process went well. At first glance, the decryption results of digital images are no different from the original digital images of the dataset. After carrying out both the encryption process and the decryption process with the ChaCha20 algorithm, the digital image will be a test object to determine the performance of the ChaCha20 algorithm in encrypting and decrypting digital images.

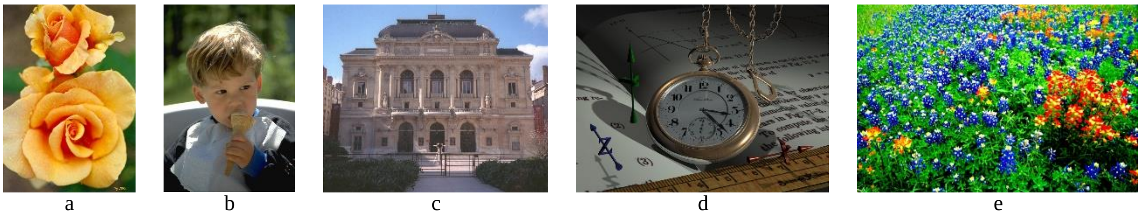

Figure 6. Decrypted Images (a) brandyrose, (b) kid, (c) opera, (d) watch, (e) wildflowers

3.1. Measurement of Encryption Quality Testing

In this study, various techniques were employed to evaluate digital images concerning the measurement of encryption quality, namely Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR).

Table 2. MSE and PSNR Test Results

Digital Image	Size (pixel)	MSE	PSNR
brandyrose	518 x 744	0.0907	58.5528
kid	487 x 703	0.0928	58.4542
opera	695 x 586	0.1054	57.8994
watch	1024 x 768	0.1096	57.7318
wildflowers	594 x 400	0.2176	54.7539

Based on the results presented in Table 2 from the encryption process and decryption process experiments conducted on 5 dataset images, it is evident that the decrypted digital images exhibit good quality. This good quality is indicated by the low MSE values approaching zero across all digital images. These MSE values are further supported by relatively high PSNR values for all the tested digital images. This signifies that the quality of the digital images that have undergone encryption and decryption using the ChaCha20 algorithm is indeed good.

3.2. Differential Attack Testing

In this study, various techniques were employed to evaluate digital images against differential attacks, namely Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI).

Table 3. UACI and NPCR Test Results

Digital Image	Size (pixel)	UACI	NPCR
brandyrose	518 x 744	49.96%	99.60%
kid	487 x 703	50.01%	99.60%
opera	695 x 586	50.03%	99.61%
watch	1024 x 768	50.02%	99.60%
wildflowers	594 x 400	49.94%	99.61%

The results of the encryption and decryption processes on the five database images shown in table 3 show that the quality of the encrypted digital images is good. This good quality is demonstrated by the NPCR values approaching approximately 100% across all experiments. As for the UACI values, they varied in the range of 49.94% to 50.03% across all experiments.

3.3. Histogram Testing

This test will also compare the histogram of the original digital image, The encrypted digital image histogram, and the decrypted digital image histogram. Figure 4 displays the original digital image histogram, encrypted digital image histogram, and decrypted digital image histogram for the brandyrose image.

DOI: https://doi.org/10.52436/1.jutif.2024.5.2.1470
p-ISSN: 2723-3863
e-ISSN: 2723-3871

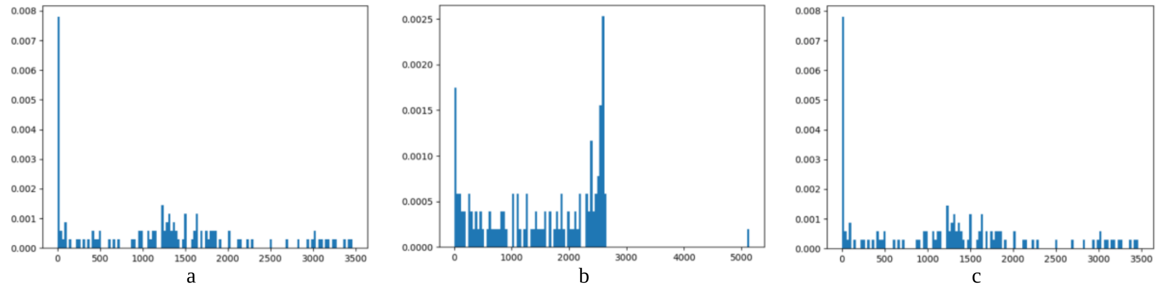

Figure 7. Brandyrose Histogram (a) original, (b) encrypted, (c) decrypted

Based on Figure 7, the encrypted digital image histogram shows a different pixels distribution compared to the original digital image's histogram. In the original digital image's histogram, there are few evenly distributed peaks between 0(x) and 3500(x). In the encrypted digital image's histogram, the small peaks in the original digital image have disappeared, so that the pixel distribution is evenly distributed. This indicates that the original digital image has been encrypted effectively. In the decrypted digital image's histogram, there is minimal disparity when compared to the histogram of the original digital image. This indicates that the digital image has been decrypted effectively.

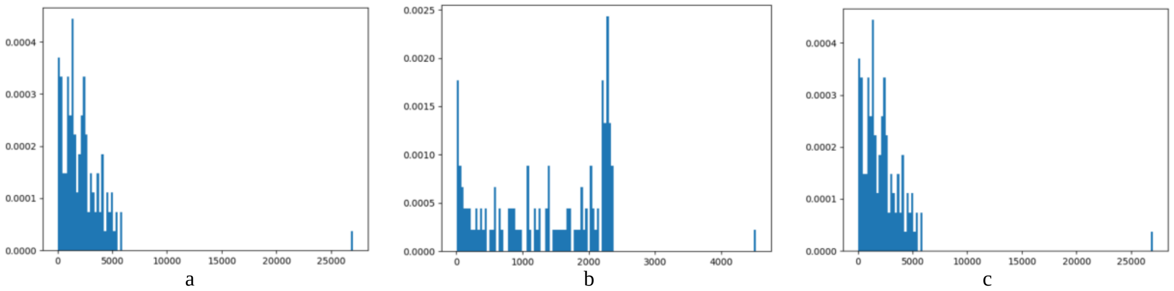

Figure 8. Kid Histogram (a) original, (b) encrypted, (c) decrypted

Based on Figure 8, the encrypted digital image histogram displays a different pixel distribution compared to the original digital image's histogram. The even distribution of pixels in the encrypted digital image suggests that the original digital image has been successfully encrypted. Similarly, the decrypted digital image's histogram also confirms that digital image has been decrypted effectively. This can be seen from the almost imperceptible difference between decrypted digital image histogram and original digital image histogram.

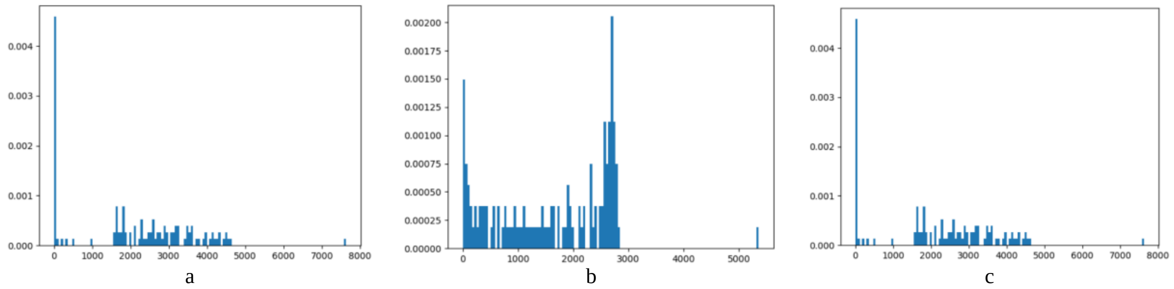

Figure 9. Opera Histogram (a) original, (b) encrypted, (c) decrypted

Based on Figure 9, the original digital image's histogram shows small spikes between 1500(x) and 4500(x). In contrast, the histogram of the encrypted digital image shows a consistent pixel distribution, visually distinct from the histogram of original digital image. This suggests that the original digital image has been effectively encrypted. In the decrypted digital image's histogram, there is minimal difference compared to the histogram of the original digital image, affirming the effective decryption of the digital image.

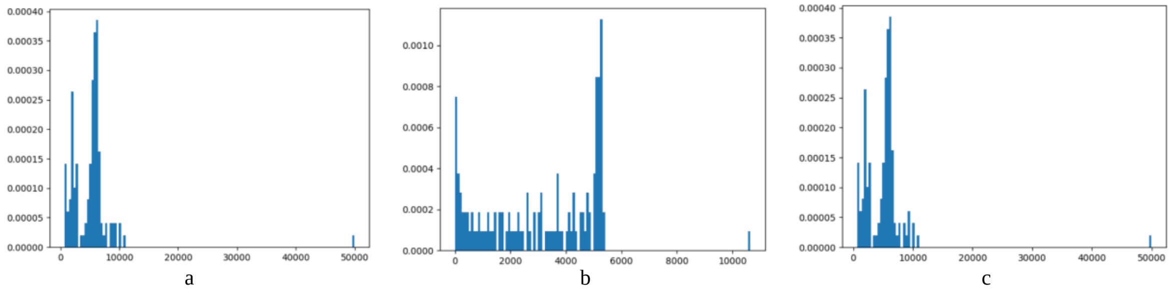

Figure 10. Watch Histogram (a) original, (b) encrypted, (c) decrypted

Based on Figure 10, the encrypted digital image’s histogram displays a different pixel distribution compared to the original digital image’s histogram, indicating effective encryption of the original digital image. Similarly, the decrypted digital image’s histogram shows minimal difference in pixel distribution compared to the original digital image, affirming the effective decryption of the digital image.

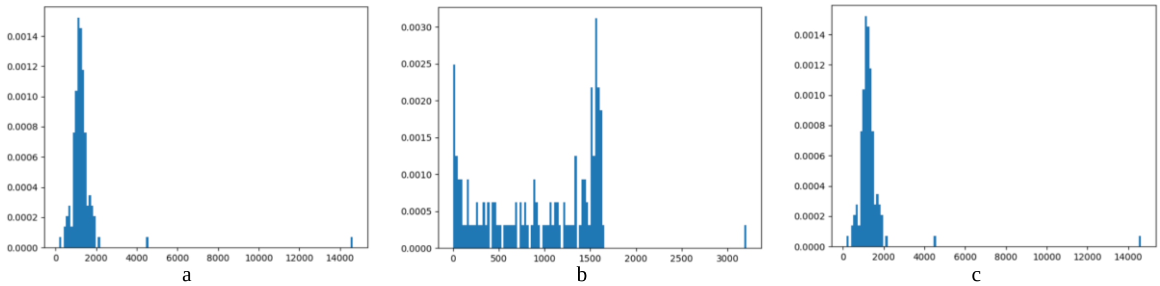

Figure 11. Wildflowers Histogram (a) original, (b) encrypted, (c) decrypted

Based on Figure 11, The histogram of the original digital image exhibits spikes in the range of 0(x) to 2000(x). Conversely, the histogram of the encrypted digital image displays an even distribution of pixels, visually signifying the successful encryption of the original digital image. In the histogram of the decrypted digital image, there is barely any noticeable difference compared to the histogram of the original digital image, providing confirmation of the effective decryption of the digital image.

The testing on the 5 digital images yielded good quality results. This is evidenced by the testing result using four techniques: MSE, PSNR, UACI, and NPCR, all of which demonstrated good results. The MSE testing showed promising results, with the lowest value being 0.0907 for the brandyrose digital image. PSNR testing yielded the highest value of 58.5528 for the brandyrose digital image. NPCR testing also displayed good results, with the highest value being 99.61% for the opera and wildflowers digital images. UACI testing revealed values in the range of 49.94% to 50.03%. These findings were further supported by the histogram results. Therefore, Encrypting and decrypting digital images using the ChaCha20 algorithm are secure.

4. DISCUSSION

Based on the test data obtained, the evaluation value of image encryption using the ChaCha20 algorithm is very satisfactory. The evaluation value is also better than previous research which used the ChaCha20 algorithm for encryption of baboon and pappers images. The average MSE value of previous research was 7715 and the average PSNR value was 9.25 [11]. This research obtained an MSE evaluation value of 0.1232 and PSNR 57.4784, which is better than previous research. This is also supported by a UACI value of 49.99% and NCPCR of 99.602% to ensure good research results.

5. CONCLUSIONS

In this research, a series of tests were carried out to assess the performance of the ChaCha20 algorithm in encrypting and decrypting digital images. Based on testing of 5 color digital images of different sizes in .tif format, the average MSE value obtained was 0.1232. MSE value that is close to zero means that very few errors occur. The average result of PSNR value is 57.4784, this value is included in the ideal category. The average UACI value is 49.99%. This UACI percentage value is relatively high, which means that the change in pixel intensity in the encryption image is also relatively high. The average NPCR percentage is 99.602%. This high NPCR percentage value indicates that the algorithm is very effective in randomizing image pixels. The success of digital image encryption and decryption can also be observed through the histogram obtained which shows a good pixel distribution in the digital image. All these test results confirm that digital image encryption using the ChaCha20 algorithm is safe and of high quality.

REFERENCES

[1] A. Tiwari, G. Shankar, B. Bhusan Jain, and Mt. Scholar, “Comparative Analysis of Different Steganography Technique for Image Security,” *International Journal of Engineering Trends and Applications (IJETA)*, vol. 8, no. 2, pp. 6–9, 2021, doi: 10.33144/23939516/IJETA-V8I2P2.

[2] D. Sinaga, E. H. Rachmawanto, C. A. Sari, D. R. I. M. Setiadi, and N. A. Setiyanto, “An Enhancement of Data Hiding Imperceptibility using Slantlet Transform (SLT),” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 87–98, Nov. 2018, doi: 10.22219/kinetik.v4i1.702.

[3] C. A. Sari, M. H. Dzaki, E. H. Rachmawanto, R. R. Ali, and M. Doheir, “High PSNR Using Fibonacci Sequences in Classical Cryptography and Steganography Using LSB,” *International Journal of Intelligent Engineering and Systems*, vol. 16, no. 4, pp. 568–580, 2023, doi: 10.22266/ijies2023.0831.46.

[4] A. A. Rashid and K. A. Hussein, “Image encryption algorithm based on the density and 6D logistic map,” *International Journal of Electrical and Computer Engineering*, vol. 13, no. 2, pp. 1903–1913, Apr. 2023, doi: 10.11591/ijece.v13i2.pp1903-1913.

[5] C. AtikaSari, D. W. Utomo, and M. A. S. Doheir, “Visual Analysis Based on CMY and RGB Image Cryptography Using Vigenere and Beaufort Cipher,” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, May 2023, doi: 10.22219/kinetik.v8i2.1664.

[6] R. Imam, Q. M. Areeb, A. Alturki, and F. Anwer, “Systematic and Critical Review of RSA Based Public Key Cryptographic Schemes: Past and Present Status,” *IEEE Access*, vol. 9. Institute of Electrical and Electronics Engineers Inc., pp. 155949–155976, 2021. doi: 10.1109/ACCESS.2021.3129224.

[7] M. Salih Mahdi, R. Abdulaali Azeez, and N. Falih Hassan, “A proposed lightweight image encryption using ChaCha with hyperchaotic maps,” *Periodicals of Engineering and Natural Sciences*, vol. 8, no. 4, pp. 2138–2145, 2020, doi: 10.21533/pen.v8i4.1708.

[8] J. Waleed, A. Noori Mazher, and A. Tariq MaoLood, “Developed Lightweight Cryptographic Algorithms for The Application of Image Encryption: A Review,” *Journal of Al-Qadisiyah for Computer Science and Mathematics*, vol. 13, no. 2, p. 11, 2021, doi: 10.29304/jqcm.2021.13.2.788.

[9] A. T. Maolood, E. K. Gbashi, and E. S. Mahmood, “Novel lightweight video encryption method based on ChaCha20 stream cipher and hybrid chaotic map,” *International Journal of Electrical and Computer Engineering*, vol. 12, no. 5, pp. 4988–5000, Oct. 2022, doi: 10.11591/ijece.v12i5.pp4988-5000.

[10] A. R. Alawi and N. F. Hassan, “A Proposal Video Encryption Using Light Stream Algorithm,” *Engineering and Technology Journal*, vol. 39, no. 1B, pp. 184–196, Mar. 2021, doi: 10.30684/etj.v39i1b.1689.

[11] M. H. Taha and J. M. Al-Tuwaijari, “Improvement of Chacha20 algorithm based on tent and Chebyshev chaotic maps,” *Iraqi Journal of Science*, vol. 62, no. 6, pp. 2029–2039, Jul. 2021, doi: 10.24996/ijs.2021.62.6.29.

[12] S. M. S. Reza *et al.*, “Salsa20 based lightweight security scheme for smart meter communication in smart grid,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 1, pp. 228–233, Feb. 2020, doi: 10.12928/TELKOMNIKA.V18I1.14798.

[13] Y. Q. Zhang, J. L. Hao, and X. Y. Wang, “An Efficient Image Encryption Scheme Based on S-Boxes and Fractional-Order Differential Logistic Map,” *IEEE Access*, vol. 8, pp. 54175–54188, 2020, doi: 10.1109/ACCESS.2020.2979827.

[14] L. E. Kane, J. J. Chen, R. Thomas, V. Liu, and M. McKague, “Security and Performance in IoT: A Balancing Act,” *IEEE Access*, vol. 8, pp. 121969–121986, 2020, doi: 10.1109/ACCESS.2020.3007536.

[15] O. Dişkaya, E. Avaroğlu, H. Menken, and A. Emsal, “A New Encryption Algorithm Based on Fibonacci Polynomials and Matrices,” *Traitement du Signal*, vol. 39, no. 5, pp. 1453–1462, Nov. 2022, doi: 10.18280/ts.390501.

[16] G. Shengtao, W. Tao, W. Shida, Z. Xuncai, and N. Ying, “A Novel Image Encryption Algorithm Based on Chaotic Sequences and Cross-Diffusion of Bits,” *IEEE Photonics J*, vol. 13, no. 1, Feb. 2021, doi: 10.1109/JPHOT.2020.3044222.

[17] P. Parida, C. Pradhan, X. Z. Gao, D. S. Roy, and R. K. Barik, “Image Encryption and Authentication with Elliptic Curve Cryptography and Multidimensional Chaotic Maps,” *IEEE Access*, vol. 9, pp. 76191–76204, 2021, doi: 10.1109/ACCESS.2021.3072075.

[18] M. K. Hussein, K. R. Hassan, and H. M. Al-Mashhadi, “The quality of image encryption techniques by reasoned logic,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 6, pp. 2992–2998,

Dec. 2020, doi: 10.12928/TELKOMNIKA.v18i6.14340.

[19] W. Jang and S.-Y. Lee, “Partial image encryption using format-preserving encryption in image processing systems for Internet of things environment,” *Int J Distrib Sens Netw*, vol. 16, no. 3, p. 155014772091477, Mar. 2020, doi: 10.1177/1550147720914779.

[20] S. A. Mehdi and Z. latif Ali, “Image Encryption Algorithm Based on a Novel Six-Dimensional Hyper- Chaotic System,” *Al-Mustansiriyah Journal of Science*, vol. 31, no. 1, pp. 54–63, Mar. 2020, doi: 10.23851/mjs.v31i1.739.

[21] S. A. Shawkat and I. Al-Barazanchi, “A proposed model for text and image encryption using different techniques,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 20, no. 4, pp. 858–866, Aug. 2022, doi: 10.12928/TELKOMNIKA.v20i4.23367.

[22] A. A. Alhijaj and M. Kamil Hussein, “Stereo Images Encryption by OSA &amp; RSA Algorithms,” *J Phys Conf Ser*, vol. 1279, no. 1, p. 012045, Jul. 2019, doi: 10.1088/1742-6596/1279/1/012045.

[23] X. Xue, D. Zhou, and C. Zhou, “New insights into the existing image encryption algorithms based on DNA coding,” *PLoS One*, vol. 15, no. 10, Oct. 2020, doi: 10.1371/journal.pone.0241184.

[24] S. T. Kamal, K. M. Hosny, T. M. Elgindy, M. M. Darwish, and M. M. Fouda, “A new image encryption algorithm for grey and color medical images,” *IEEE Access*, vol. 9, pp. 37855–37865, 2021, doi: 10.1109/ACCESS.2021.3063237.

[25] X. Huang, Y. Dong, G. Ye, W. S. Yap, and B. M. Goi, “Visually meaningful image encryption algorithm based on digital signature,” *Digital Communications and Networks*, vol. 9, no. 1. KeAi Communications Co., pp. 159–165, Feb. 01, 2023. doi: 10.1016/j.dcan.2022.04.028.