Bulletin of Electrical Engineering and Informatics
Vol. 9, No. 3, June 2020, pp. 980~987
ISSN: 2302-9285, DOI: 10.11591/eei.v9i3.2001

Triple layer image security using bit-shift, chaos, and stream encryption

Ajib Susanto[1], De Rosal Ignatius Moses Setiadi[2], Eko Hari Rachmawanto[3], Ibnu Utomo Wahyu Mulyono[4], Christy Atika Sari[5], Md Kamruzzaman Sarker[6], Musfiqur Rahman Sazal[7]
[1,2,3,4,5]Informatics Engineering Department, Dian Nuswantoro University, Indonesia
[6]Department of Computer Science, Kansas State University, USA
[7]School of Computing and Information Sciences, Florida International University, USA

Article Info

Article history:

Received Aug 15, 2019
Revised Nov 2, 2019
Accepted Dec 10, 2019

Keywords:

Arnold chaotic maps
Cyclic bit-shift
Hybrid encryption
Rivest cipher
Super-encryption

ABSTRACT

One popular image security technique is image encryption. This research proposes an image encryption technique that consists of three encryption layers, i.e. bit-shift encryption, chaos-based encryption, and stream encryption. The chaos algorithm used is Arnold's chaotic map, while the stream cipher algorithm used is RC4. Each layer has different cryptology characteristics in order to obtain safer image encryption. The characteristics of cryptology are permutation, confusion, diffusion, and substitution. The combination of the proposed encryption method aims to secure images against various attacks, especially attacks on statistics and differentials. The encryption method testing is done by various measuring instruments such as statistical analysis, i.e. entropy information, avalanche effect, and histogram, differential analysis, i.e. UACI and NPCR, visual analysis using PSNR and SSIM, and bit error ratio. Based on the results of experiments that the encryption method that we propose can work excellently based on various measurement instruments. The decryption process can also work perfectly this is evidenced by the ∞ value based on PSNR, and zero value based on SSIM and BER.

Corresponding Author:

De Rosal Ignatius Moses Setiadi,
Informatics Engineering Department, Dian Nuswantoro University,
Imam Bonjol Street, no. 207, Semarang 50131, Indonesia.
Email: moses@dsn.dinus.ac.id

1. INTRODUCTION

The development of digital technology has changed the lifestyle of humans. Storing data manually has been replaced by digital storage, even a few old data also been replaced in the form of hard scanned forms of files or digital images. Digital data is used more widely because it can be easily archived, rediscovered, and transacted through the internet network. The internet is a global network that is accessed by audiences throughout the world [1-3]. Digital data transmission that has privacy and confidentiality needs special handling, namely security.

Cryptography is a powerful way to provide security to digital data [4]. At present, many cryptographic algorithms can be used to secure digital files. However, along with the times, cryptographic algorithms must also be developed and maintained to provide better security [5-9]. Cryptographic algorithms that are widely implemented in digital files are AES, DES, RC4, OTP, RSA, and Chaotic Map [4, 5, 8, 10-13]. Cryptographic research on images has been widely developed as in research carried out in [14-23]. Cryptography in images uses more permutation and diffusion techniques such as chaotic or shifting algorithms [14, 24, 25], this is because methods such as AES, DES and RSA

are considered inefficient because they have complex computing and enable over-computing of extensive multimedia data [26]. Chaotic algorithms have deterministic values such as sensitivity to control and initial values, ergodicity and unpredictability [9, 23]. However, in the research carried out on [4, 25, 26] it is said that diffusion and permutation techniques such as chaotic maps do not change the encrypted image histogram, so it needs to be combined with other encryption methods to increase its security.

Some research on encryption in images has combined permutation methods, diffusion, and stream ciphers to increase security. As in the research conducted by Irawan et al. [27], his research combined Arnold's chaotic map and RC4 methods to improve security in medical images. Arnold chaotic maps are used, so that image encryption is resistant to attacks by differential attacks and brute force. While the RC4 method is used to increase resistance to statistical attacks, RC4 belongs to the stream cipher, which has the advantage of fast computing [28]. Shifting techniques are also proposed in research [15, 26, 29]. This technique is usually used with the cyclic method or shifting bits. This technique can work faster than chaos techniques that may have many iterations. Iterations in chaotic algorithms determine the randomness of pixels, but it affects the speed of computing. So in this research, we propose a combination of three bit-shifting encryption algorithms, chaos and stream ciphers using the RC4 algorithm to produce strong image encryption and resistance to various attacks.

2. RESEARCH METHOD

The method proposed in this study is to combine three encryption techniques, namely bit-shifting, chaotic system, and stream cipher. Our proposed method is illustrated in Figure 1.

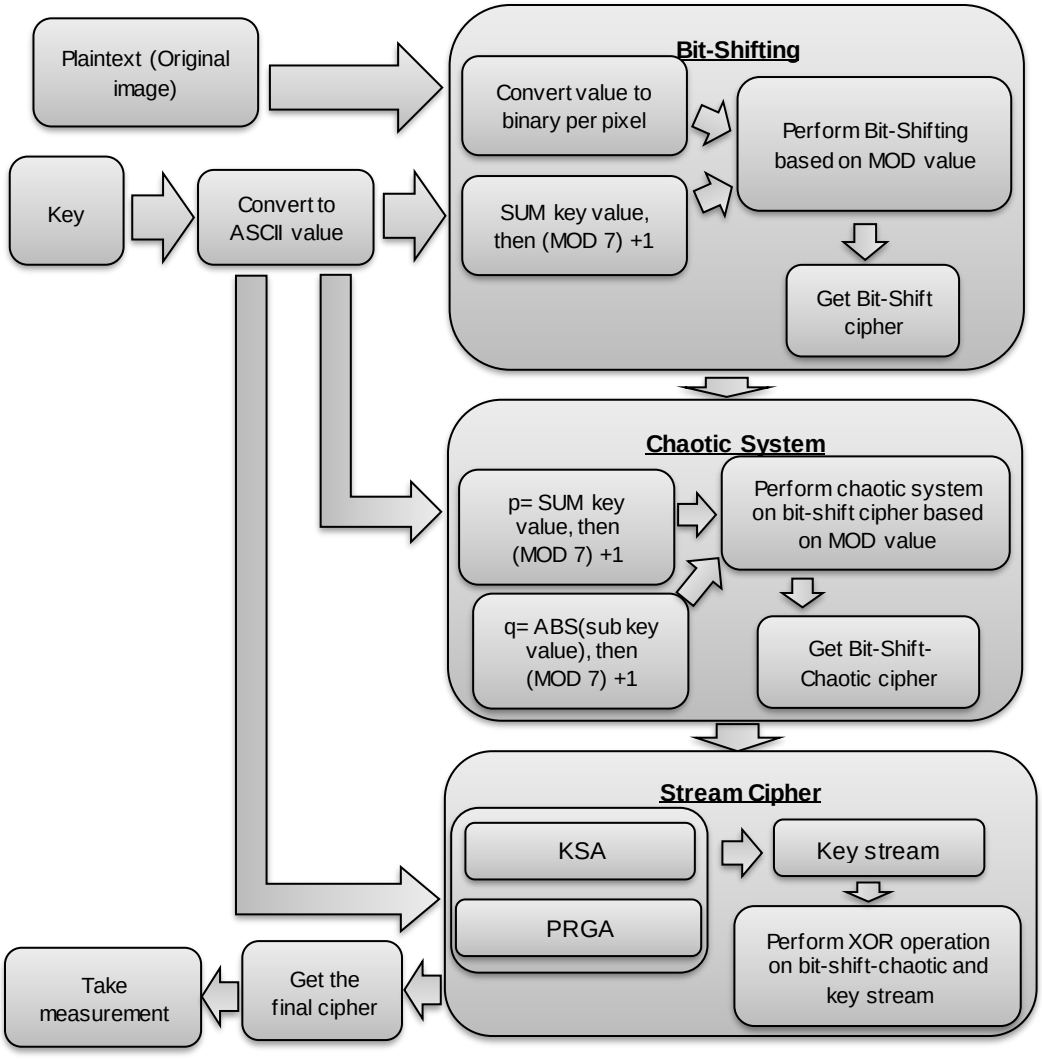


Figure 1. The proposed hybrid encryption algorithm

2.1. Bit-shifting

Bit-shifting is the first layer that is applied to the proposed hybrid encryption algorithm. Bit-shifting is done on the original image based on the key, where the shifting process is circular. For example, there is an image with a pixel value as follows [123 231; 51 222], and the key is 'abc', the bit-shifting process is described as follows:

1. Change the pixel value: $\begin{bmatrix}123 & 231\\ 51 & 222\end{bmatrix}$ into a binary form: $\begin{bmatrix}01111011 & 11100111\\ 00110011 & 11011110\end{bmatrix}$
2. Change the key 'abc' to the ASCII value :$\begin{bmatrix}97 & 98 & 99\end{bmatrix}$
3. Sum the key ASCII value, then the result is modulated 7 plus 1, (294 mod 7) +1 = 1. Note that modulus 7 is added to 1 so that the shift is only 1 to 7 because the pixel value of the image consists of only 8-bits.
4. Perform bit-shifting on image pixels that are already binary form, like below.
   $\begin{bmatrix}01111011 & 11100111\\ 00110011 & 11011110\end{bmatrix} \rightarrow \begin{bmatrix}10111101 & 11110011\\ 10011001 & 01101111\end{bmatrix}$
5. Transform the pixel value that has been shifted into an integer value $\begin{bmatrix}189 & 243\\ 153 & 111\end{bmatrix}$

2.2. Chaotic system

In the second layer, the results of bit-shifting encryption are chaotic with the Arnold chaotic map (ACM) algorithm. ACM is an encryption technique by scrambling 2-D on image matrix pixels. Formula (1) is used to encrypt ACM [12].

$$\begin{bmatrix}x'\\ y'\end{bmatrix} = \begin{bmatrix}1 & p\\ q & pq+1\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix} \ mod\ N \quad (1)$$

Where x and y are pixel coordinates, N is the image dimension, p and q are chaotic keys, where p is generated from the sum of the key ASCII values then the results are modulated 7 plus 1, while q is the absolute value of the key-value reduction then the result is modulated 7 plus 1. For example, if the key is 'abc' then the value of p is 1 (equal to the bit-shift key), while q is (abs (-100) mod 7) + 1 = 3.

2.3. Stream cipher

At the last layer, the results of bit-shift and ACM encryption are encrypted again with the RC4 algorithm, which is a stream cipher. The RC4 algorithm consists of three stages, namely the key scheduling algorithm (KSA), pseudo-random generator algorithm (PRGA), and XOR operation [30]. The key used in this layer is the same as the key used in the previous two layers, but the key used does not go through the preprocessing process such as SUM, SUB, ABS, and MOD operations. At the KSA permutation stage is created using two substitution boxes (S-Box), where the first S-Box (S) has 256 lengths and is given an initial value of 0 to 255, and the second S-Box (K) is given an extended ASCII key-value up to 256. Then do an S-Box randomization with the algorithm below.

```
For I = 0 to 255 {
          j = (j + S[i] + K[i]) mod
255
         swap S[i] dan S[j]
}
```

Furthermore, in the PRGA permutation stage, a keystream (KS) array will be obtained with a length equal to the dimensions of the image. Where each KS element is obtained by the algorithm below.

```
I = (I + 1) mod 255
j = (j + S[i]) mod 255
swap S[i] dan S[j]
t = (S[i] + S[j] mod 255
KS = S[t]
```

The last stage of this process is the XOR operation to get the final cipher and XOR operations on the bit-shift-ACM cipher and KS.

2.4. Measurements

Several methods were applied, such as entropy, avalanche effect, NPCR, UACI, histogram analysis, BER, PSNR, and SSIM, to compare and evaluate the quality of encryption.

2.4.1. Entropy analysis

Entropy analysis is useful for measuring the level of ciphertext randomness in each image will be calculated the spread of pixels for each color channel. In this case, it serves to analyze the resistance level of ciphertext against statistical attacks. In RGB images, it would be better if a uniform distribution was produced [21]. The ideal ciphertext entropy score is close to 8 [4, 27]. The entropy formula can be calculated by (2).

$$E = -\sum_{r=0}^{R=255} c(r)\ log_2(c(r)) \quad (2)$$

Where E is Entropy; R is a range of the pixel value; $c(r)$ is the chance of probability.

2.4.2. Avalanche effect (AE) analysis

Avalanche effect analysis is to find out the change in the proportion value of the cipher bits compared to other ciphers with little modification to the key or plaintext. The ideal AE value is around 50% or higher [16, 21]. The higher the AE value, the better the ciphertext quality. Generally, the key or plaintext modification for AE calculations is done at 1-bit at the beginning, middle, or end. The value of AE can be calculated by (3).

$$AE = \frac{diffbits(cipher1, cipher2)}{totalbits} \times 100\% \quad (3)$$

2.4.3. Number of pixels change rate (NPCR) and unified average changing intensity (UACI)

NPCR and UACI are standard measurement tools that are widely used to determine the level of resistance of a cipher against differential attacks [19, 20]. NPCR and UACI are also used to analyze the sensitivity of ciphertext [15, 21]. A good NPCR value should approach 100%. While the UACI value is the opposite, smaller values represent better performance.

$$NPCR = \left(\frac{1}{M \times N}\sum_{i=0}^{M}\sum_{j=0}^{N} D(i,j)\right) \times 100\%, D(i,j) = \begin{cases} 0, P(i,j) = C(i,j) \\ 1, P(i,j) \neq C(i,j) \end{cases} \quad (4)$$

$$ACI = \left(\frac{1}{M \times N}\sum_{i=0}^{M}\sum_{j=0}^{N} \frac{|P(i,j) - C(i,j)|}{255}\right) \times 100\% \quad (5)$$

Where P is the plain image; C is the cipher image; both images have the same dimension which is $M \times N$; i, j are pixel locations.

2.4.4. Histogram analysis

The histogram is the distribution of distribution and the statistical intensity of the pixels of an image. The histogram analysis also functions to determine the statistical performance of the image encryption algorithm [24]. The cipher image histogram must be very different from the plain image and have a uniform distribution of each pixel intensity. The uniformity of pixel intensity displayed on histogram images and histogram differences in image ciphers that are very different from plain images shows that the quality of image encryption is categorized as good [19].

2.4.5. PSNR and SSIM

Measurement of image encryption using PSNR and SSIM is actually not widely used in image encryption research. This measurement is done with the aim of knowing and analyzing the visual display of the encrypted image [31]. Visually the results of image encryption must be very different from the original image so that the image of the cipher is not easily guessed [27]. Where PSNR and SSIM are two of the most popular measuring instruments for measuring visual image quality. Where, in this case, proper image encryption should produce the lowest PSNR and SSIM values possible. On the other hand, the PSNR and SSIM decryption process can be used to measure the perfection of the image decryption process, where the PSNR and SSIM values are infinity and 1 respectively, to get the perfect image decryption. PSNR can be calculated by (6), whereas SSIM can be calculated by (7).

$$PSNR = 10\ log_{10}\left(\frac{255}{\frac{1}{M\ x\ N}\sum_{i}^{M}\sum_{j}^{N}[P(i,j) - C(i,j)]^2}\right) \quad (6)$$

$$SIM\ (P,C) = \frac{(2\mu_P\mu_C + v_1)(2\sigma_{PC} + v_2)}{(\mu_P^2 + \mu_C^2 + v_1)(\sigma_P^2 + \sigma_C^2 + v_2)} \quad (7)$$

P is a plain image; C is a cipher image; M and N are the image dimension; i, j are pixel locations; μ_P is mean of the P; μ_C is mean of the C; σ_{PC} is the covariance P against C; σ_P^2 is a variant of P; σ_C^2 is a variant of C; $v_1 = (l_1 D)^2$ and $v_2 = (l_2 D)^2$; D is a dynamic range $(2^{bits} - 1)$ with the default value $l_1 = 0.01$ and $l_2 = 0.03$.

2.4.6. Bit error ratio (BER)

Similar to PSNR and SSIM, the measurement of BER in image encryption is also rarely used. However, this also needs to know the error bit level after it is encrypted. The calculation method is almost the same as the avalanche effect, but the tang is compared to the plain image bits and the cipher image. The more different bits in the two images, the more uncorrelated images, meaning that encryption is getting better. BER can also be used as a measurement tool in the image decryption process where the BER value must be zero, meaning that there is no difference at all between the plain image and the decryption image. The formula for calculating BER can be seen in (8).

$$BER = \frac{diffBits\,(oriBits, encBits)}{totalBits} \times 100\% \quad (8)$$

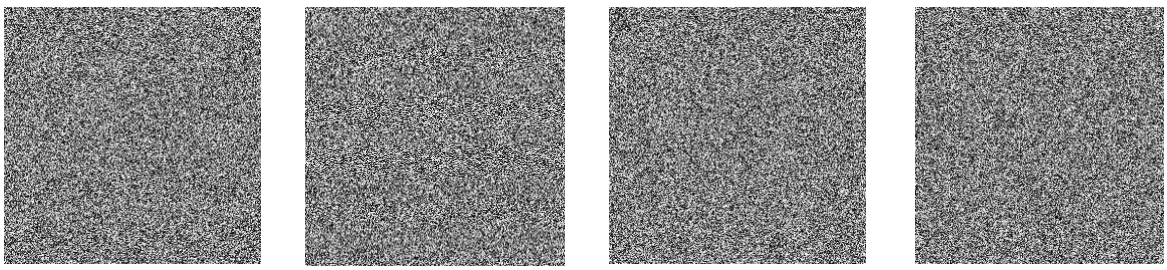
3. RESULTS AND DISCUSSION

At this stage, the test is carried out on several standard images that are widely used in various image processing research. Three images, namely Peppers, Lena, and Cameraman, can be downloaded from the website in reference [32], whereas for Lichtenstein imagery can be downloaded through the website in reference [33]. These images are then preprocessed, which consists of the process of resizing to dimensions 256×256 and the process of converting colors from RGB (for color images) to grayscale. The preprocessed image used in this study is presented in Figure 2.

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

Figure 2. Original image used (after preprocessing), (a) Cameraman, (b) Lena, (c) Lichtenstein, (d) Peppers

Then the encryption process is carried out using the proposed method with key 'password' in the four images so that the image is generated as shown in Figure 3. As discussed earlier, this key is used in all encryption layers, however, the key in each encryption layer will be different due to processing different key keys in each layer of encryption.



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

Figure 3. Encrypted image used, (a) Encrypted-Cameraman, (b) Encrypted-Lena, (c) Encrypted-Lichtenstein, (d) Encrypted-Peppers

Based on the image shown in Figure 3, it appears that the results of image encryption visually appear different. However, from these results, it is necessary to process the measurement of the encryption using the measurement tools discussed earlier; Table 1 presents the results of measurement of Entropy, SSIM, PSNR, UACI, NPCR, and BER values. Table 2 presents the results of the avalanche effect, and Table 3 presents the histogram analysis.

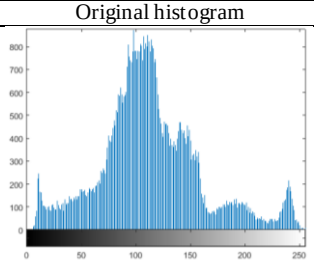
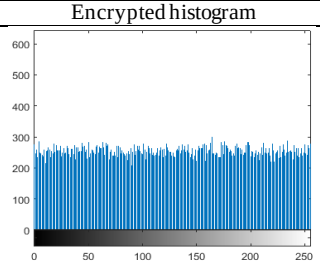
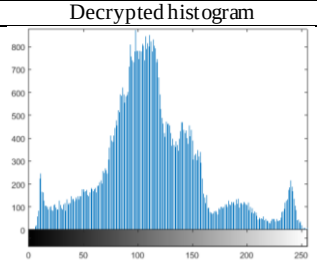
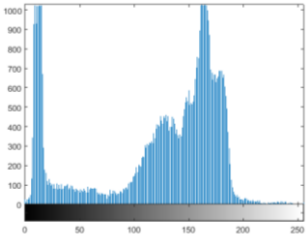
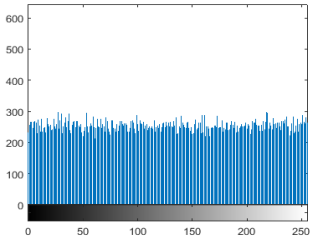
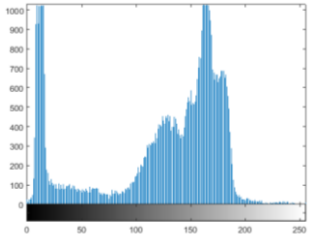
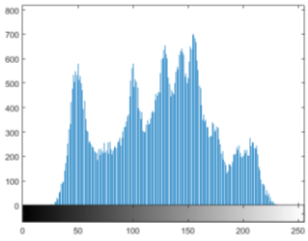
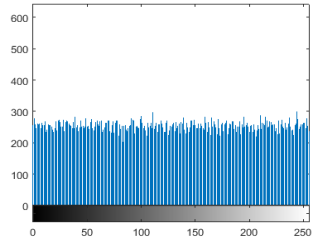
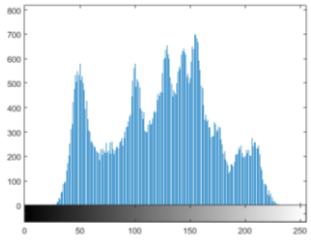
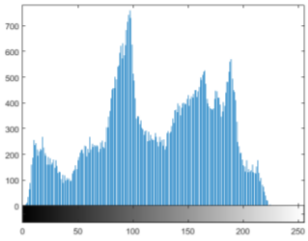
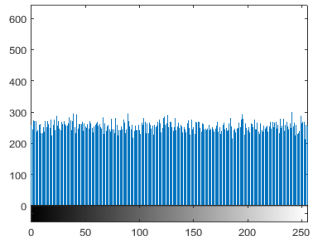
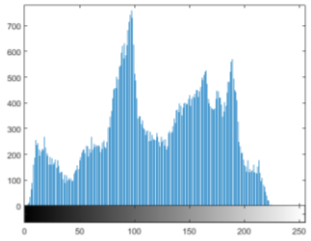
Table 1. Entropy, SSIM, PSNR, UACI, NPCR, and BER measurement results from encrypted images

Image	Entropy	NPCR	UACI	PSNR	SSIM	BER
Lichtenstein	7.9974	99.63%	28.72%	9.1870	0.0107	50.04%
Cameraman	7.9973	99.58%	31.01%	8.3925	0.0069	50.09%
Lena	7.9976	99.64%	28.66%	9.2292	0.0084	50.02%
Peppers	7.9973	99.63%	29.48%	8.9254	0.0097	50.00%
Average	7.9974	99.62%	29.47%	8.9335	0.0089	50.04%

Table 2. Avalanche effect

Image	Modified key: passwor**c**		Modified key: pas**t**word		Modified key: **q**assword	
	Different bit	Percentage	Different bit	Percentage	Different bit	Percentage
Lichtenstein	262581	50.08%	261309	49.84%	262534	50.07%
Cameraman	262361	50.04%	261823	49.94%	262194	50.01%
Lena	262311	50.03%	261865	49.95%	261882	49.95%
Peppers	262435	50.06%	262733	50.11%	261740	49.92%
Average	262422	50.05%	261932.5	49.96%	262087.5	49.99%

Table 3. Histogram analysis

Image	Original histogram	Encrypted histogram	Decrypted histogram
Lichtenstein			
Cameraman			
Lena			
Peppers			

Based on the values presented in Table 1, it appears that the resulting entropy value is near perfect, which is 7.99 [34], the NPCR and UACI values are also in the excellent category. Likewise, with the visual measurement of PSNR and SSIM, the value is very poor; this means that the results of the visual encryption are very different from the original image. In Table 2, the value of the avalanche effects every key modification ranges from 50%, meaning that the encryption algorithm produced is very good. The cipher image histogram presented in Table 3 also changes significantly by distributing relatively uniform pixel intensity values. Figure 4 presents the results of image decryption, in Figure 4 visually looks the same as the original image. Based on the measurement results of BER, PSNR, and SSIM between the original image and the decrypted image, it is proven true that there is indeed no difference in the results of decryption, this is evidenced by all the PSNR values generated are infinity, while the SSIM and BER values are 0.

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

Figure 4. Decrypted images, (a) Cameraman, (b) Lena, (c) Lichtenstein, (d) Peppers

Based on the results of the comparison with the method proposed by Irawan et al. [27] presented in Table 4 and Table 5, it appears that the proposed method is in terms of Entropy, NPCR, UACI, and SSIM values. This proves that the proposed method has advantages over the previous method.

Table 4. Comparison Entropy, NPCR, and UACI results with method [27]

Image	Entropy		NPCR		UACI	
	Method [27]	Proposed	Method [27]	Proposed	Method [27]	Proposed
Lichtenstein	7.9970	7.9974	99.60%	99.63%	28.86%	28.72%
Cameraman	7.9978	7.9973	99.59%	99.58%	31.00%	31.01%
Lena	7.9971	7.9976	99.60%	99.64%	28.69%	28.66%
Peppers	7.9971	7.9973	99.65%	99.63%	29.45%	29.48%
Average	7.9973	**7.9974**	99.61%	**99.62%**	29.50%	**29.47%**

Table 5. Comparison PSNR, SSIM, and BER results with method [27]

Image	PSNR		SSIM		BER	
	Method [27]	Proposed	Method [27]	Proposed	Method [27]	Proposed
Lichtenstein	9.1542	9.1870	0.0092	0.0107	50.05%	50.04%
Cameraman	8.4321	8.3925	0.0097	0.0069	49.93%	50.09%
Lena	9.2166	9.2292	0.0114	0.0084	50.00%	50.02%
Peppers	8.9266	8.9254	0.0088	0.0097	50.05%	50.00%
Average	**8.9324**	8.9335	0.0098	**0.0089**	50.01%	50.04%

4. CONCLUSION

There are various image encryption algorithms, where each algorithm has its own advantages. Image encryption must stage various attacks such as statistical attacks, differential, and brute force attacks. This research proposes three layers of encryption algorithms to increase image security to be more resistant to various attacks to decrypt cipher images. Based on testing with various measuring instruments such as Entropy, NCPR, UACI, histograms, SSIM, PSNR, and BER, it appears that the proposed method works well and can improve security. The decryption process also works perfectly, where this becomes a condition of the encryption process working correctly.

REFERENCES

[1] C. A. Sari, G. Ardiansyah, D. R. I. M. Setiadi, and E. H. Rachmawanto, "An improved security and message capacity using AES and Huffman coding on image steganography," *TELKOMNIKA*, vol. 17, no. 5, pp. 2400–2409, 2018.

[2] A. Setyono and D. R. I. M. Setiadi, "Image watermarking using discrete wavelet-tchebichef transform," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 16, no. 3, pp. 1416–1423, 2019.

[3] D. R. I. M. Setiadi, "Improved payload capacity in LSB image steganography uses dilated hybrid edge detection," *J. King Saud Univ. -Comput. Inf. Sci.*, 2019.

[4] D. R. I. M. Setiadi, E. H. Rachmawanto, C. A. Sari, A. Susanto, and M. Doheir, "A comparative study of image cryptographic method," *5th Int. Conf. on Inf. Tech., Comp., and Elec. Eng. (ICITACEE)*, pp. 336–341, 2018.

[5] A. E. Mezher, "Enhanced RSA cryptosystem based on multiplicityof public and private keys," *Int. J. Electr. Comput. Eng.*, vol. 8, no. 5, pp. 3949–3953, 2018.

[6] Z. Kasiran, H. F. Ali, and N. M. Noor, "Time performance analysis of advanced encryption standard and data encryption standard in data security transaction," *Ind. J. Electr. Eng. Comp. Sci.*, vol. 16, no. 2, pp. 988-994, 2019.

[7] S. D. Putra, M. Yudhiprawira, S. Sutikno, and Y. Kurniawan, "Power analysis attack against encryption devices: A comprehensive analysis of AES, DES, and BC3," *TELKOMNIKA*, vol. 17, no. 3, pp. 1282–1289, 2018.

[8] A. Setyono, D. R. I. M. Setiadi, and Muljono, "Dual encryption techniques for secure image transmission," *J. Telecommun. Electron. Comp. Eng.*, vol. 10, no. 3-2, pp. 41-46, 2018.

[9] R. Lan, J. He, S. Wang, T. Gu, and X. Luo, "Integrated chaotic systems for image encryption," *Signal Processing*, vol. 147, pp. 133–145, 2018.

[10] W. S. Sari, E. H. Rachmawanto, D. R. I. M. Setiadi, and C. A. Sari, "A good performance OTP encryption image based on DCT-DWT steganography," *TELKOMNIKA*, vol. 15, no. 4, pp. 1982–1989, 2017.

[11] E. J. Kusuma, O. R. Indriani, C. A. Sari, E. H. Rachmawanto, and D. R. I. M. Setiadi, "An imperceptible LSB image hiding on edge region using DES encryption," *Int. Conf. on Innov. and Creative Inf. Tech.*, pp. 1-6, 2017.

[12] E. J. Kusuma, C. A. Sari, E. H. Rachmawanto, and D. R. I. M. Setiadi, "A combination of inverted LSB, RSA, and Arnold Transformation to get secure and imperceptible image steganography," *J. ICT Res. Appl.*, vol. 12, no. 2, pp. 103-122, 2018.

[13] H. V. Gamido, M. V. Gamido, and A. M. Sison, "Developing a secured image file management system using modified AES," *Bull. Electr. Eng. Informatics*, vol. 8, no. 4, pp. 1461–1467, 2019.

[14] X. Wang and H. Zhang, "A color image encryption with heterogeneous bit-permutation and correlated chaos," *Opt. Commun.*, vol. 342, pp. 51–60, 2015.

[15] X.-Y. Wang, S.-X. Gu, and Y.-Q. Zhang, "Novel image encryption algorithm based on cycle shift and chaotic system," *Opt. Lasers Eng.*, vol. 68, pp. 126–134, 2015.

[16] M. Alawida, A. Samsudin, J. Sen Teh, and R. S. Alkhawaldeh, "A new hybrid digital chaotic system with applications in image encryption," *Signal Processing*, vol. 160, pp. 45–58, 2019.

[17] G. Ke, H. Wang, S. Zhou, and H. Zhang, "Encryption of medical image with most significant bit and high capacity in piecewise linear chaos graphics," *Measurement*, vol. 135, pp. 385–391, 2019.

[18] C. Cao, K. Sun, and W. Liu, "A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map," *Signal Processing*, vol. 143, pp. 122–133, 2018.

[19] L. Xu, Z. Li, J. Li, and W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Opt. Lasers Eng.*, vol. 78, pp. 17–25, 2016.

[20] K. A. Kumar Patro and B. Acharya, "An efficient colour image encryption scheme based on 1-D chaotic maps," *J. Inf. Secur. Appl.*, vol. 46, pp. 23–41, 2019.

[21] H. M. Ghadirli, A. Nodehi, and R. Enayatifar, "An overview of encryption algorithms in color images," *Signal Processing*, vol. 164, pp. 163–185, 2019.

[22] A. Hasheminejad and M. J. Rostami, "A novel bit level multiphase algorithm for image encryption based on PWLCM chaotic map," *Optik (Stuttg).*, vol. 184, pp. 205–213, 2019.

[23] Ü. Çavuşoğlu, S. Kaçar, I. Pehlivan, and A. Zengin, "Secure image encryption algorithm design using a novel chaos based S-Box," *Chaos, Solitons & Fractals*, vol. 95, pp. 92–101, 2017.

[24] M. Ge and R. Ye, "A novel image encryption scheme based on 3D bit matrix and chaotic map with Markov properties," *Egypt. Informatics J.*, vol. 20, no. 1, pp. 45–54, 2019.

[25] A. Broumandnia, "Designing digital image encryption using 2D and 3D reversible modular chaotic maps," *J. Inf. Secur. Appl.*, vol. 47, pp. 188–198, 2019.

[26] S. Kandar, D. Chaudhuri, A. Bhattacharjee, and B. C. Dhara, "Image encryption using sequence generated by cyclic group," *J. Inf. Secur. Appl.*, vol. 44, pp. 117–129, 2019.

[27] C. Irawan, D. R. I. M. Setiadi, E. H. Rachmawanto, C. A. Sari, and M. Doheir, "Hybrid encryption using confused and stream cipher to improved medical images security," *J. Phys. Conf. Ser.*, vol. 1201, no. 1, pp. 1-9, 2019.

[28] A. J. Abboud, A. N. Albu-Rghaif, and A. K. Jassim, "Balancing compression and encryption of satellite imagery," *Int. J. Electr. Comput. Eng.*, vol. 8, no. 5, pp. 3568–3586, 2018.

[29] X. Wang, X. Zhu, X. Wu, and Y. Zhang, "Image encryption algorithm based on multiple mixed hash functions and cyclic shift," *Opt. Lasers Eng.*, vol. 107, pp. 370–379, 2018.

[30] A. Bhowmick, N. Sinha, R. V. Arjunan, and B. Kishore, "Permutation-Substitution architecture based image encryption algorithm using middle square and RC4 PRNG," *Int. Conf. on Inv. Syst.s and Cont.*, pp. 1–6, 2017.

[31] A. Houas, Z. Mokhtari, K. E. Melkemi, and A. Boussaad, "A novel binary image encryption algorithm based on diffuse representation," *Eng. Sci. Technol. an Int. J.*, vol. 19, no. 4, pp. 1887–1894, 2016.

[32] Ming Hsieh Department of Electrical Engineering, "The USC-SIPI image database." USC Viterbi School of Engineering, [Online]. Available at: http://sipi.usc.edu/database/. [Accessed: 27-Mar-2019].

[33] A. Tille and A. Damato, "File:Lichtenstein img processing test.png - wikimedia commons," [Online]. Available at: https://commons.wikimedia.org/wiki/File:Lichtenstein_img_processing_test.png. [Accessed: 09-Jul-2019], 2001.

[34] S. Fadhel Hamood, M. S. Mohd Rahim, and O. Farook Mohammado, "Chaos image encryption methods: A survey study," *Bull. Electr. Eng. Informatics*, vol. 6, no. 1, pp. 99–104, 2017.