8th IEEE International Conference and Workshops on Recent Advances and Innovations in Engineering- ICRAIE 2023
(IEEE Record #59459)

An Improved RC4 Algorithm Based on Multi Chaotic Map for Image Encryption

Rinaldi Munir
Informatics Research Group, School of Electrical Engineering and Informatics
Institute Teknologi Bandung, Jalan Ganesha 10 Bandung, Indonesia
rinaldi@staff.stei.itb.ac.id

***Abstract*— RC4 algorithm is a famous stream cipher. It could be used to encrypt images quickly and efficient. However, the RC4 has been proved has some drawbacks, therefore it is considered not secure anymore. Therefore, RC4 algorithm needs to be improved the security. This paper describes an improved RC4 algorithm based on multi chaotic map (Logistic map, Henon map, and Sine map) to and use it to encrypt the images. RC4 algorithm consists of KSA stage and PRGA stage. Modification of KSA stage has been done using the chaotic maps. Performances of the improved RC4 for image encryption were measured by some metrics (histogram, correlation, and entropy). Experiment results show that the improved RC4 has good performance for image encryption.**

Keywords—RC4 algorithm, image encryption, KSA, multi chaotic maps

I. INTRODUCTION

Image play important role in the digital information era, because it shows information visually. Digital images can be stored in the storage media or transmitted via public public channels such as internet. Some images are secret or confidential, only the owner or authorized party could access and view them. To protect the secret images, encryption algorithms could be used to encode secret images into a form that is hard to interpret.

RC4 is one of symmetric-key encryption algorithm. It could be used to encrypt any kind of information, including digital images. RC4 was a famous algorithm because the algorithm is simple and the encryption/decryption is fast (only XOR operation). It is used to secure data communication in network. RC4 was used in SSL (Secure Socket Layer), TLS (Transport Layer Security), WEP (Wired Equivalent Privacy) and WPA(Wi-Fi Protected Access). RC4 belongs to the stream cipher, it means RC4 encrypt one byte of plaintext at a time using XOR operation between plaintext and pseudorandom byte. RC4 algorithm has two stages: Key Scheduling Algorithm (KSA) and Pseudo Random Generator Algorithm (PRGA). In KSA stage, a state table is initialized and then scrambled based on an external key. Then, the scrambled state table is used by PRGA to generate a random byte through a simple computation. Finally, the random byte is used to mask the plaintext by XOR operation.

RC4 has attracted many cryptanalysts to analyze its security. They identified some drawbacks in RC4 algorithm. The weaknesses have been found both in KSA and PRGA. The weakness of KSA is the first few bytes of output of the generator related to several bytes of secret key, therefore, by analyzing these bytes makes it possible to attack RC4 [1]. Therefore, it was proposed to discard the first bytes of the output of PRGA. The weakness of PRGA is there are relations between the state table in different time [2].

After found some drawbacks of RC4, many researchers proposed variants of RC4 to strengthen RC4. Some variants of RC4 are Improved RC4 [2], RC4A [3], RC4+ [4], VMPC [5], FJ-RC4 [6], Block-RC4 [7] etc. Generally these variants of RC4 modified KSA or PRGA or both. However, some weakness also have been found in the algorithms.

Meanwhile, the usage of chaos theory for encryption has been widely studied. Chaos is an attractive topic in cryptography for two reasons: (1) sensitivity to the initial conditions, (2) random behavior. The usage of chaos in encryption can produce diffusion effect as stated by Shanon. Chaotic maps such as Logistic map, Henon map, Arnold Cat's map, Chebyschev map, Baker map, etc, have been used to generate pseudorandom numbers.

Related to the RC4 algorithm and image encryption, some researchers have been proposed modified RC4 algorithms based on chaos. Some of them are as follows. Gaata and Hantoosh [8] proposed an image encryption based on the RC4 algorithm and logistic map. The logistic map produced a chaotic sequence 256-bytes-length as initial permutation stage of the RC4 algorithm. Alani et.al [9] proposed an image encryption which is a combination between the RC4 algorithm and Henon_map. The Henon_map generated a chaotic 256-bytes-length for using in PRGA stage. Kumari and Gupta [10] proposed an image encryption which is a combination 3D_Logistic maps and the RC4 algorithm. The chaotic map is used to generate the key for RC4 cipher. Sahib et.al [11] proposed an improved RC4 based on multi-chaotic maps (Logistic map, Tent map, and Chebyschev map). They modified KSA into improved KSA (IKSA), which is the permutation of the state table depend on the random numbers generator based on three chaotic maps. Mohammed and Jawad [12] proposed a secure image encryption scheme using two chaotic maps (Henon map and Sine map) and RC4 algorithm. The chaotic maps are used for confusion and diffusion stage.

In this paper we proposed an improved RC4 algorithm based on three chaotic maps (Logistic Map, Henon map, and Sine map) for image encryption. The chaotic maps are used to improve the weakness of KSA stage. This paper has five sections. First section describes this introduction. The second section is preliminary study, it reviews RC4 algorithm and some related theories. The third section proposes an improved RC4 algorithm based on multi chaotic map for image encryption. The fourth section explains the experiments results of image encryption and discussion. The last section, i.e the fifth section, concludes the gained results and suggestion for future research.

2023 IEEE 8th International Conference on Recent Advances and Innovations in Engineering (ICRAIE) | 979-8-3503-1551-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICRAIE59459.2023.10468437

II. PRELIMINARIES

This section describes some material related to the proposed method, namely the RC4 algorithm, Logistic map, Henon map, and sine map.

A. RC4 Algorithm

Actually RC4 is not an encryption algorithm, but rather a keystream (pseudorandom) generator. The keystream is used to mask plaintext or ciphertext by XOR operation. As mentioned before, RC4 algorithm has two stages: KSA and PRGA. RC4 has a state table *S* and two indices *i* and *j*. In KSA stage, first the state table *S* is initialized and then it is scrambled based on an external key. Then, the scrambled state table is used by PRGA to generate a random through a simple computation .Fig. 1 shows pseudo-code of RC4 algorithm.

```
//KSA:
for i ← 0 to 255       //initialization step
   S[i] ← i
   T[i] ← K[i mod length(K)]
end for

j ← 0
for i ← 0 to 255      //Scramble the state table S
   j ← (j + S[i] + T[i]) mod 256
   swap(S[i], S[j]);
end for

//PRGA:
i ←0; j ← 0
while (true)    //generate keystreams. Repeat as needed
    i ← (i + 1) mod 256
    j ← (j + S[i]) mod 256
    swap(S[i], S[j]);
    t ← S[i] + S[j] mod 256
    keystream ← S[t]
   // encryption: C ← P ⊕ keystream
   // decryption: P ← C ⊕ keystream
end while
```

Fig. 1. Pseodo-code of *RC4* algorithm

B. Logistic Map

Logistic map is a simple nonlinear dynamical system that exhibit chaotic behavior. Mathematically, the equation of logistic map is

$$z_{n+1} = r z_n (1 - z_n) \quad (1)$$

where z_n is a real number $\in$ [0, 1], r is a parameter in the interval [0, 4]. The map is in chaotic state when $r > 3.56995$.

C. Henon Map

Henon map is a 2D chaotic map. It maps a point (x_n, y_n) to a new point (x_{n+1}, y_{n+1}) in the plane by equation:

$$\begin{cases} x_{n+1} = 1 - a x_n^2 + y_n \\ y_{n+1} = b x_n \end{cases} \quad (2)$$

In equation 2 above a and b are two parameters of Henon map. Henon map is in chaotic state when $a = 1.4$ and $b = 0.3$. The Henon map can also be written into a one-dimensional map,

$$x_{n+1} = 1 - a x_n^2 + b x_{n-1} \quad (3)$$

D. Sine Map

The sine map is similar to the logistic map, only different in its mathematical equation.. Equation of sine map is

$$x_{n+1} = \mu \sin(\pi x_n) \quad (4)$$

where x_n is a real number $\in$ [0, 1], μ is a parameter in the interval [0, 1]. The map reaches a chaotic state when $\mu > 0.87$.

E. Henon-Sine Map

Although the Henon map and sine map are simple, they have weaknesses, because of the trajectories of the maps can be estimated easily. Therefore, Wu et.al [13] proposed a new hyperchaotic map that named 2D Henon-Sine map as follows:

$$\begin{cases} x_{n+1} = 1 - a (\sin x_n)^2 + y_n \bmod 1 \\ y_{n+1} = b x_n \bmod 1 \end{cases} \quad (5)$$

The trajectory of 2D Henon-Sine map shown in Fig. 2 shows that the map distributes in the entire range of plane.

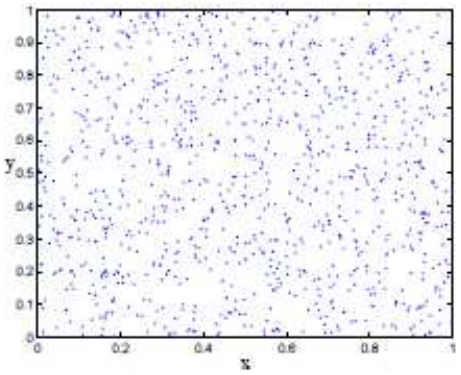


Fig. 2. Phase diagram of 2D Henon-Sine map [13]

III. PROPOSED SCHEME OF IMPROVED RC4

We proposed a modified RC4 algorithm based on multi chaotic maps. We used logistic map and Henon-Sine map to generate pseudorandom numbers. We choosed the chaotic maps because they generate random numbers in floating point number between 0 and 1. The Henon-Sine map in a one-dimensional map is

$$x_{n+1} = 1 - a (\sin x_n)^2 + b x_{n-1} \bmod 1 \quad (6)$$

As mentioned in Section 1, one of the weaknesses of algorithm A is in the KSA stage, namely the first few bytes of output of the generator related to several bytes of secret key, therefore, by analyzing these bytes makes it possible to attack RC4. To solve this problem, Therefore, after initialization step, we first encrypt the secret key K with random number sequence that generated from Logistic map. Length of K after padding is 256 characters, so with length of random number sequence.

```
//Generate 256 random number using Logistic map
R[0..255] ← Encode(LogisticMap(r, x0))

//Encrypt secret key K with R
T ← T ⊕ R
```

Next, in scrambling the state table *S*, we repleace indice *i* and *j* in statement

```
j ← (j + S[i] + T[i]) mod 256
```

with two random numbers that generated from combination Henon-Sine map as follows:

```
n1 ← Encode(HenonSineMap(a, b, μ, x0, x–1 ))
n2 ← Encode(HenonSineMap(a, b, μ, x0, x–1 ))
j ← (n1 + S[n2] + T[n2]) mod 256
```

For increasing security, in PRGA we generate an additional random number to be masked with plaintext and keystream:

$n3 \leftarrow Encode(HenonSineMap(a, b, \mu, x_0, x_{-1}))$
// *encryption:* $C \leftarrow P \oplus keystream \oplus n3$
// *decryption:* $P \leftarrow C \oplus keystream \oplus n3$

Fig. 3 shows pseudo-code of the improved RC4 algorithm. The pseudo-code in red represents additional computation or modification for the improved RC4 algorithm.

```
//KSA:
for i ← 0 to 255        //initialization
   S[i] ← i
   T[i] ← K[i mod length(K)]
end for

// Generate 256 random number using Logistic map
R[0..255] ← Encode(LogisticMap(r, x0))

// Encrypt secret key K with R
T ← T ⊕ R

j ← 0
for i ← 0 to 255     //Scramble the state table S
   n1 ← Encode(HenonSineMap(a, b, μ, x0, x–1 ))
   n2 ← Encode(HenonSineMap(a, b, μ, x0, x–1 ))
   j ← (n1 + S[n2] + T[n2]) mod 256
   swap(S[i], S[j]);
end for

//PRGA:
i ←0; j ← 0
while (true)    //generate keystreams. loop as needed
   i ← (i + 1) mod 256
   j ← (j + S[i]) mod 256
   swap(S[i], S[j]);
   t ← S[i] + S[j] mod 256
   keystream ← S[t]
   n3 ← Encode(HenonSineMap(a, b, μ, x0, x–1 ))
   // encryption: C ← P ⊕ keystream ⊕n3
   // decryption: P ← C ⊕ keystream⊕n3
end while
```

Fig. 3. Pseudo-code of the improved *RC4* algorithm

Logistic map and Henon-Sine map always generate random numbers in floating point number between 0 and 1. Because of cryptography operates in integer, first we have to encode the random numbers into integer. The encoding method is as follows: take n significant digits from digits of floating number (digits after decimal point). For examples, 0.23056291 and n = 4, we take 2305, 0.00156891 and n =5, we take 1568. All integers must be represented in modulus 256. This method is described in function *Encode*.

IV. Experiment Results and Discussion

We have programmed the improved RC4 algorithm using Matlab. Next we tested the program to encrypt 10 standard images (actually there are so many standard images, in this experiment we choose ten images as the test images). All of images are grayscale and they have size 512 x 512 pixels. For logistic map we used $r = 3.999$, $z_0 = 0.14444449$, while for Henon-Sine map we used $a = 1.4$, $b = 0.3$, $x_0 = 0.2999999$, and $x_1 = 0.244449$. All parameters can be considered as secret keys other than an external key of RC4. We used K = '12345678' as the external key. This algorithm successfully encrypts and decrypts images well. Due to space limitations, only examples of encrypting two test images ('house' and 'bird') is shown here, as shown in Fig. 4. The encrypted images looks unrecognizable because visually it looks as random images.

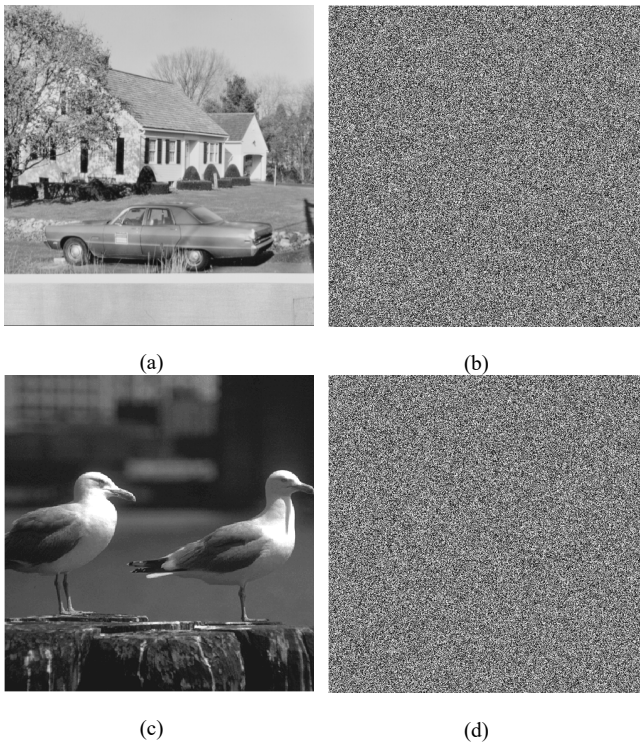
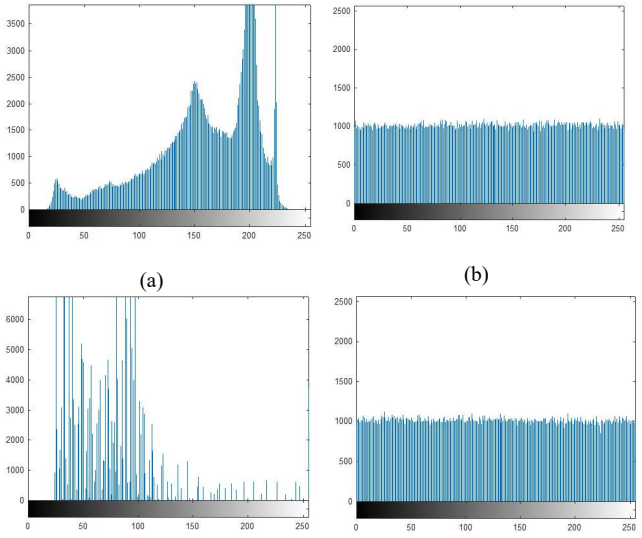


Fig. 4. Plain-images (a and c) and encrypted images (b and d) of 'house' and 'bird' image

We have performed some experiments and the results are discussed below.

A. Histogram Analysis

The histogram described distribution of the pixels in a image. Usually an attacker uses histogram to analyze the frequency of occurrence pixel values in the encrypted image to estimate the encryption key or pixels in the plain-image. To prevent the attack, histogram of the cipher image have no similarity with histogram of the plain-image. Therefore, the pixels in the encrypted image should have a (relatively) uniform distribution or shown with a histogram that looks flat.



(c) (d)

Fig. 5. Histogram of plain-images (a and b) and histogram of encrypted images (b and d) of 'house' and 'bird' image

Fig. 5 shows the histogram of plain-images and the encrypted images of 'house' and 'bird' image, respectively. The histogram of encrypted images looks flat, making it more difficult to attacker to deduce pixel values or encryption key(s).

B. Correlation Analysis

In statistics, correlation is a statistical relationship that refers to the degree to which a pair of variables are linearly related. Given two random variables X and Y, each variable has n elements, we calculate correlation coefficient X and Y by formula:

$$r_{XY} = \frac{\text{cov}(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}} \quad (7)$$

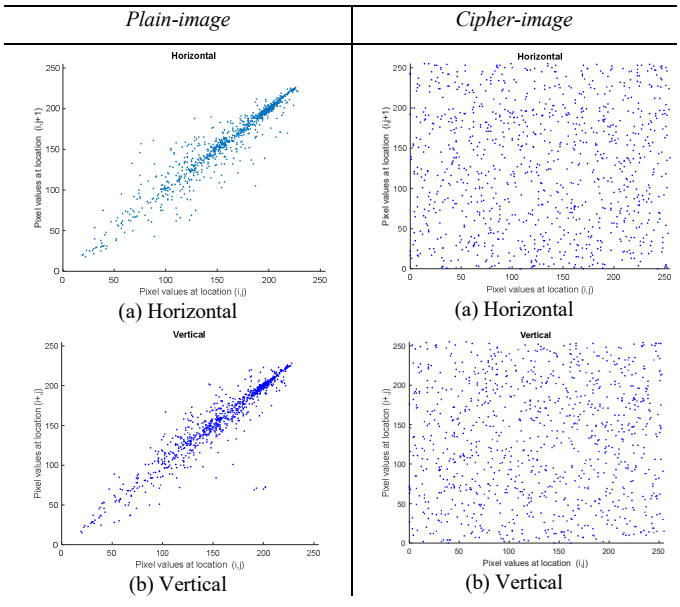
where

$$\text{cov}(X,Y) = \frac{1}{n}\sum_{i=1}^{n}[x_i - E(X)][y_i - E(Y)] \quad (8)$$

$$D(X) = \frac{1}{n}\sum_{i=1}^{n}[x_i - E(X)]^2 \quad (9)$$

$$E(X) = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (10)$$

Neighboring pixels in most plain-image images have a high correlation coefficient (around +1 or -1). Image encryption makes the correlation between pixels disappear or reduces the correlation coefficient to zero. We calculated the correlation coefficient between horizontally neighboring pixels, between vertically neighboring pixels, and between diagonally neighboring pixels. We do these for the plain-images and the encrypted images.



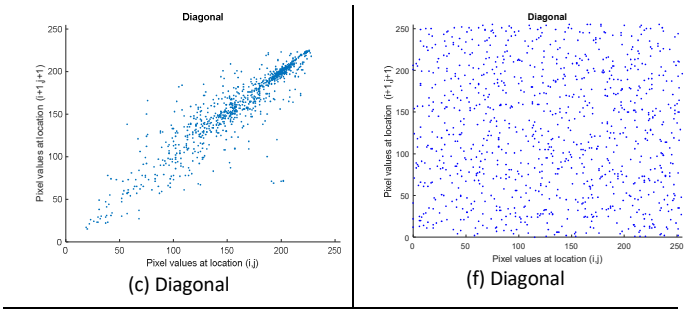


Fig. 6 Correlation distribution of neighboring pixels on plain-image and encrypted image of 'house' image

In this experiment, we randomly select 1000 pairs of neighboring pixels, one in each vertical, horizontal and diagonal direction. The correlation coefficients were calculated by equation (7). Fig. 6 and 7 display distribution of the correlation coefficient in plain-images (left column) and encrypted images (right column) of 'house' and 'bird' image respectively. Observe that in the plain-images, neighboring pixels have strong correlation between the pixels, the correlation values are around the diagonal line. However, in the encrypted images the correlation values are spread in the entire range of plane, indicating correlation of the pixels are disappear or the correlation is reduced.

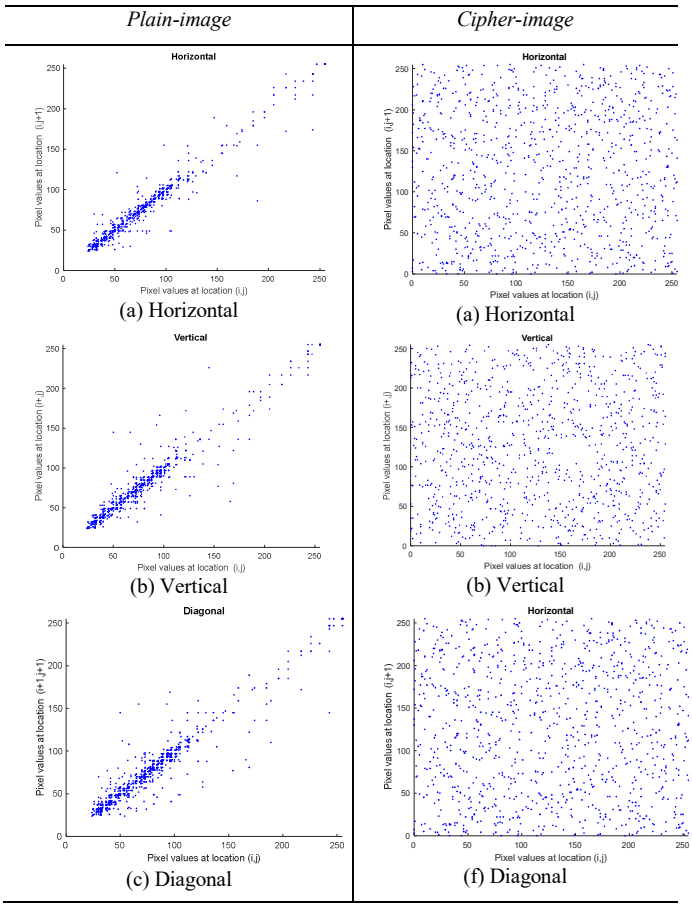


Fig. 7 Correlation distribution of neighboring pixels on plain-image and encrypted image of 'bird' image

Table 1 resumes correlation coefficient values of the plain-images and the encrypted images for each direction. In all plain images, the correlation coefficient are around 1. On the contrary, in all encrypted images, the correlation coefficient

are around 0. Therefore, the improved RC4 algorithm successfully makes the correlation pixels in the encrypted image decrease or close to zero.

TABLE 1. COMPARISON OF CORRELATION COEFFICIENT

Image		Correlation coefficient		
		Horizontal	***Vertikal***	***Diagonal***
Village	Plain image	0.9703	0.9599	0.9435
	Cipher image	*0.0095*	*0.0385*	*0.0185*
Roman	Plain image	0.9702	0.9649	0.9410
	Cipher image	*0.0226*	*0.0141*	*0.0119*
Mandrill	Plain image	0.8802	0.7836	0.7388
	Cipher image	*-0.0330*	*0.0647*	*0.0086*
House	Plain image	0.9536	0.9622	0.9169
	Cipher image	*0.0136*	*0.0530*	*-0.0328*
Bird	Plain image	0.9704	0.9598	0.9412
	Cipher image	*0.0182*	*0.0170*	*-0.0271*
Boat	Plain image	0.9545	0.9773	0.9465
	Cipher image	*-0.0127*	*0.0151*	*0.0039*
Lenna	Plain image	0.9647	0.9806	0.9414
	Cipher image	*0.0039*	*0.0498*	*0.0246*
Couple	Plain image	0.9324	0.8625	0.8149
	Cipher image	*-0.0102*	*0.0812*	*0.0040*
Barbara	Plain image	0.8940	0.9565	0.8845
	Cipher image	*0.0242*	*0.0286*	*-0.0113*
Peppers	Plain image	0.9764	0.9699	0.9480
	Cipher image	*0.0241*	*0.0216*	*-0.0054*

C. *Entropy Analysis*

Entropy is the measure of the disorder of a system.. Given a message M and the probability of symbol m_i in the message is $p(m_i)$, then the entropy of M is

$$H(M) = -\sum_{i=0}^{2M-1} p(m_i) \log_2 p(m_i) \qquad (11)$$

The entropy is measured in unit of bit(s). The higher the entropy, the bigger the uncertainty in the message. The encrypted image can be viewed as a random message. A grayscale image has 256 graylevels, therefore $m_0 = 0$, $m_1 = 1$, …, $m_{255} = 255$, and $p(m_i)$ can be calculated from its histogram, then entropy of the encrypted image is

$$H(M) = \sum_{i=0}^{255} p(m_i) \log_2 p(m_i) \qquad (12)$$

The encrypted image should have the ideal entropy is equal to eight, while the plain image have an entropy less than eight. If the entropy is less than eight, then there is a threat to security because a degree of predictability of the message increases.

In our experiment we calculated entropy of the plain-images and the encrypted images. Table 2 summarized the entropies and compared thems when calculated with the RC4 algorithm. From the table we can observe that the entropy values of encrypted images are very close to eight, both for RC4 and improved RC4, which means both RC4 algorithm and improved RC4 algorithm are secure from entropy attacks to predict the information in the image.

TABLE 2. ENTROPY OF THE PLAIN-IMAGES AND THE ENCRYPTED IMAGES

Image	Entropy		
	Plain-image	***Encrypted image (RC4)***	***Encrypted image (Improved RC4)***
Village	7.4778	7.9993	7.9993
Roman	6.1808	7.9991	7.9992
Mandrill	7.3579	7.9994	7.9992
House	7.2416	7.9993	7.9991
Bird	5.8484	7.9993	7.9991
Boat	7.1238	7.9992	7.9991
Lenna	7.4451	7.9993	7.9991
Couple	7.2010	7.9994	7.9994
Barbara	7.6321	7.9993	7.9992
Peppers	7.5712	7.9993	7.9993

D. *Sensitivity Analysis*

As mentioned in Introduction, a chaos system is sensitive to small changes in the initial conditions. To find out the sensitivity of chaos to small changes in initial conditions, an experiment was carried out by changing an initial value of Henon-Sine map, $x_0 = 0.2999999$, to $x_0 = 0.2999998$ (the difference is 10^{-7}). Decryption result of the cipher-image with the new value is shown in Fig. 7(d), which remains like a random image (does not return to the original image). These experiments show that chaos satisfies the diffusion principle from Shanon, so that brute force attacks will fail because changing just one bit in the key causes the decryption result to still be wrong.

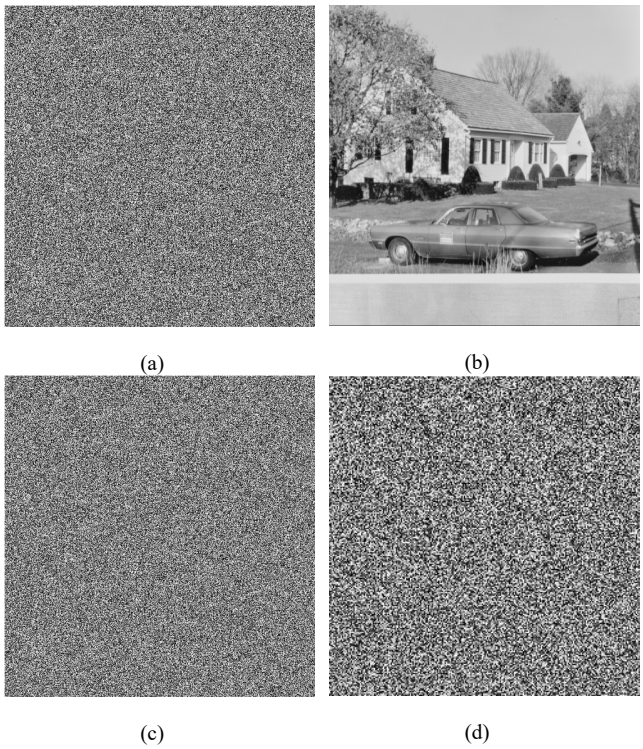


Fig. 6. Decryption with small changes of initial condition. (a) encrypted image of 'house' image, (b) decrypted image with original $x_0 = 0.2999999$, (c) cipher- image of 'house' image, (d) decrypted image with new value $x_0 = 0.299999$.

E. *Key Space Analysis*

The attacker tries to find all possible keys to decrypt the encrypted image by a brute force method. To overcome the brute force attack, we have to make the key space verry large. The key space means the total number of different keys for encryption/decryption.

In the improved RC4 algorithm, there are several multi chaotic map parameters that can be considered as secret keys. The parameters are from logistic map: r, z_0, and from Henon-Sine map: a, b, x_0, and x_1, and an external key K. Except K, all of these parameter values are real numbers. Refer to IEEE standard for 64-bit floating point, the precision of floating point computation is 10^{-15} [14], therefore number of possible values of each r, z_0, a, b, x_0, and x_1 is 10^{15}. While for external key K, length of K is maximal 256 characters, so number of possible values of K is 256^{256}. Therefore, the entire key space is

$$H(r, z_0, a, b, x_0, x_1, K) = 10^{15}.\times 10^{15}.\times 10^{15}.\times 10^{15} \times 10^{15}.\times 256^{256}$$
$$= 10^{75}.\times 256^{256}$$

This key space, $10^{75}.\times 256^{256}$, is very large, therefore the brute force attack against the RC4 algorithm is inefficient.

F. *Correlation Test between Key and Output of RC4*

As mentioned before, the drawback of RC4 is first few output bytes of RC4 (keystream) are highly correlated to several bytes of secret key. Therefore, we want to show that in the modified RC4 algorithm the the correlation is reduced. To do that, we run a experiment as follows:

(i) Original RC4
Key: 01234567
First 4 bytes of key: 0, 1, 2, 3
First 4 bytes of output of RC4: 254, 78, 64, 118
Correlation coefficient: abs(-0.3715) = 0.3715

(ii) Modified RC4
Key: 01234567
First 4 bytes of key: 0, 1, 2, 3
First 4 bytes of output of M-RC4: 88, 60, 194, 88
Correlation coefficient: 0.1413

We can see that in this experiment the correlation coefficient of modified RC4 less than original RC4, therefore, in this case, modified RC4 algorithm is more secure than original RC4. However, it needs to be proven mathematically whether the results like this apply in general.

V. CONCLUSION

An improved RC4 algorithm based on multi chaotic map has been generated and has been used for image encryption. Algorithm performance for image encryption can be measured from several aspects of analysis (histogram, correlation, entropy, and key space analysis). First, histogram of all encrypted images are flat, therefore it is difficult to attack with frequency analysis. Second, pixels in the encrypted images are no longer related to each other because the correlation coefficient of neighboring pixels close to zero. Third, entropy values in all encrypted images are very close to 8, which means it is secure from entropy attacks to predict the information in the image. Fourth, key space of the improved RC4 algorithm is relatively large, making this algorithm resistant to brute force attack. The whole analysis show that the improved RC4 algorithm is very secure to encrypt the images. For future works, we need to develop an algorithm to ensure that there is no correlation between output of the RC4 algorithm and bytes of the key, or the correlation between them has decreased.

ACKNOWLEDGMENT

Thank to Institut Teknologi Bandung (ITB), Indonesia. This research is funded by Penelitian, Pengabdian Masyarakat dan Inovasi (P2MI) ITB 2023.

REFERENCES

[1] L. Stosic, and M. Bogdanovic, “RC4 stream cipher and possible attacks on WEP”. Editorial Preface, 2012.

[2] J. Xie, X. Pan, “An Improved RC4 Stream Cipher”, 2010 International Conference on Computer Application and System Modeling”, ICCASM 2010

[3] S. Paul, and B. Preneel, "A New Weakness in the RC4 Keystream Generator and an Approach to Improve the Security of the Cipher", Fast Software Encryption, FSE 2004, Lecture Notes in Computer Science, vol. 3017, Springer-Verlag, pp. 245–259.

[4] S. Maitra, and G. Paul, "Analysis of RC4 and Proposal of Additional Layers for Better Security Margin", Progress in Cryptology – INDOCRYPT 2008 (PDF), Lecture Notes in Computer Science, vol. 5365, Springer-Verlag, pp. 27–39.

[5] Bartosz Zoltak, "VMPC One-Way Function and Stream Cipher", Fast Software Encryption, FSE 2004, Lecture Notes in Computer Science, vol. 3017, Springer-Verlag, pp. 210–225.

[6] F. Javdan Kherad, “A New Symmetric Cryptography Algorithm to Secure E-Commerce Transactions”, 2010 International Conference on Financial Theory and Engineering.

[7] L. Lae Khine, “A New Variant of RC4 Stream Cipher”, International Journal of Physical and Mathematical Sciences Vol:3, No:2, 2009

[8] M. Talib Gaata, and F. Fouad Hantoosh, "An Efficient Image Encryption Technique using Chaotic Logistic Map and RC4 Stream Cipher", Vol.3, pp.213–218, 2016.

[9] 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.

[10] M. Kumari, and S. Gupta, “A Novel Image Encryption Scheme Based on Intertwining Chaotic Maps and RC4 Stream Cipher,” 3D Res., vol. 9, no. 1, 2018.

[11] N. Mutar Sahib, A. Hussein Fadel, N. Shihab Ahmed, “Improved RC4 Algorithm Based on Multi-Chaotic Maps”, Research Journal of Applied Sciences, Engineering and Technology 15(1): 1-6, 2018

[12] R. Mohammed, L. Mohammed Jawad, “Secure Image Encryption Scheme Using Chaotic Maps and RC4 Algorithm”, Solid State Technology, Vol. 63 Issue: 3, 2020

[13] J. Wu, X. Liao, B. Yang, “Image Encryption Using 2D Henon-Sine Map and DNA Approach”, Signal Processing, Vol. 153, December 2018, Pages 11-23.

[14] C. Fu, J. Chen, H. Zou, W. Meng, Y. Zhan, Y. Yu. 2012. “A Chaos-based Digital Image Encryption Scheme with an improved Diffusion Strategy”. *Journal Optic Express 2363*, *Vol. 20. No. 3*.