Received: June 10, 2020. Revised: July 14, 2020.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Combination of Block-Based Chaos with Dynamic Iteration Pattern and Stream Cipher for Color Image Encryption

Fikri Budiman[1]* **De Rosal Ignatius Moses Setiadi[2]**

[1]*Department of Computer Science, Dian Nuswantoro University, Semarang, Indonesia*
[2]*Department of Informatics Engineering, Dian Nuswantoro University, Semarang, Indonesia*
* Corresponding author's Email: fikri.budiman@dsn.dinus.ac.id

Abstract: This research proposes an encryption method on images using a combination of chaotic methods, streams, and hash functions. SHA-1 is used as a hash function to encrypt key inputs to be more secure and can produce more dynamic keys at chaotic and stream encryption stages. Chaos encryption is done by dividing the image into small blocks where each encrypted block differs based on a dynamic key pattern based on chaotic keys. At the last stage, all blocks are made as whole images again to be encrypted by the stream method. Tests carried out on standard RGB images and Indonesian batik images. Encryption quality measurements using entropy, histogram analysis, UACI, NPCR, SSIM, PSNR, and the avalanche effect. Based on the results of trials the proposed method is proven to be resistant to various attacks such as statistics as evidenced by the average entropy value of 7.9996, avalanche effect value of 50.0366 and a relatively uniform histogram, while differential attack as evidenced by the value of UACI 33.5716 and NPCR 99.6082 where this value is very close to ideal. Also visually the results of the encryption look very chaotic and very different from the original image, which is evidenced by the value of PSNR 8.0191 and SSIM 0.0081. The decryption process can also be done perfectly wherein the resulting infinity value on PSNR and value 1 on SSIM.

Keywords: Dynamic encryption, Chaos method, Dynamic pattern, Image cryptography, Batik.

1. Introduction

Data security is very much needed in long-distance communication especially through internet media, especially in the transmission and storage of personal and confidential telemedicine data, military communications, trade data, consumer information, data that has legal force, etc.[1-4]. Various methods can be applied in security such as steganography, watermarking, and cryptography. Steganography and Watermarking are hiding messages in certain media, the difference is that steganography is used to protect messages that are hidden while watermarking is used to protect media covers that have been embedded with watermarks[2].

Cryptography is the science used to encode to protect it from being easily discovered by unauthorized people[5]. Several methods of steganography have been used on digital media, both in the form of text, images, audio, video, and other shapes. Cryptography in digital images requires a special method because the image has two dimensions. Chaos-based cryptography is an algorithm that is considered suitable for encoding digital images and has been widely applied in various previous studies as in research [3, 5-11]. The chaos method has many advantages: it has excellent intrinsic characteristics, including ergodicity, aperiodicity, high sensitivity to initial conditions and control parameters, and pseudorandom behavior[11]. Chaos-based cryptography has many methods such as the Arnold chaotic map, logistic map, henon map, hyperchaotic map, and so on. Chaos methods generally combine permutation and diffusion operations, where traditional permutation operations only change the pixel position, whereas in the chaos method the pixel position is randomized based on

diffusion operations. Diffusion operations can increase resistance to differential attacks[12].

The quality of image encryption can be measured by 1) Peak Signal to Noise Ratio (PNSR) used to measure the quality of visual encryption on researches [3-5, 13], 2) Entropy used to measure the level of randomness and represent the intensity of pixel distribution, this measurement tool is used on researches [3, 5, 7, 10-11, 14-16] 3) Histogram analysis used to determine the encryption resistance against statistical attacks, where a good histogram must show uniformity, this measurement tool is used in researches [3-5, 7, 10-11, 14-15], 4) Avalanche effect used to meausure the level of sensitivity of the keys and plaintext to slight changes, which are generally 1-bit, this measurement tool is used in researches [5, 12, 17-18], 5) the number of pixels changes rate (NPCR) and unified average changing intensity (UACI) used to determine the level of key sensitivity and image encryption resistance from differential attacks, this measurement tool is used in researches [5, 7, 10-11, 14-15], where the ideal value is close to 99.6093% for NPCR and close to 33.4635% for UACI[19]. To improve the quality of good encryption results, we need an algorithm to combine several existing methods.

In previous research contained in [20] has made comparisons of some chaotic cryptographic methods, streams, and classic substitution, where the chaotic algorithm used is the Arnold chaotic map (ACM). ACM is popular chaos cryptography[17, 21-22] and is suitable for image encryption, but has drawbacks on the results of entropy and histogram calculations, where both are the same value as the original image even though it is encrypted and visually very different. So in various technical papers, many proposed a combination of chaotic methods and other methods to improve image encryption security.

In some researches like [5, 8, 10-11, 23] chaotic methods are combined with stream methods to increase security. The stream cipher is a method for generating keystream, where then the key is used for substitution operations, which generally use XOR operations. Stream ciphers have the advantage of resisting statistical attacks where this method can produce high entropy values and make significant histogram changes. To strengthen the results of encryption, some research like [15, 24-25] uses a hash function to encrypt key input. The hash function is a one-way encryption method that is widely used to encrypt passwords. This method cannot be used to encrypt images directly, usually, this function is used to encrypt the keys used for the image encryption method. In general, the key entered by the user is generally a key that can be read either in the form of the name of the item, the name of a person, the name of something or in the form of a number such as a date of birth that is easy to remember and possibly has a small number of characters, then the hash function of the key length will be extracted at least 40 encrypted characters. If several of these methods are combined, it will make multiple layers of security from key input to encryption of the image itself. Image encryption can also be done on a block basis, which is an image divided into smaller blocks, and then each block is encrypted using a key that has a specific pattern. Encryption per block with a logical sequential pattern can also increase security, some encryption research using encryption per block has been carried out by [6, 14] and proven to improve encryption quality.

From some of the literature, this research proposes an image encryption method based on chaos encryption combined with stream ciphers, where chaotic encryption is performed on each image block with dynamic encryption patterns generated from key input encrypted with hash functions. This method will produce a strong image encryption method and is resistant to various attacks such as statistics and differentials. Furthermore, the theories of chaos, streams, and hashes methods will be explained in part two, the methods proposed in section 3, implementation, and discussion in section 4, wherein this section there is also a comparison with the previous and final methods in section 5.

2. Preliminaries

2.1 Chaos method

Chaos method is one of the developments of permutation techniques in image cryptography. Permutation of the image is done by randomizing the pixel position without changing the pixel value. Before the chaotic method, there were permutations based on columns, rows, or both, where this method was the simplest permutation method[12]. In its development chaos methods have been widely used. This method is a randomization technique with formulas that make certain maps and with the number of iterations that can be determined. Unlike encryption methods such as AES, DES, and Blowfish, chaotic methods are suitable for digital image encryption, because they do not require large data capacities, high redundancies and strong correlations between adjacent pixels [11]. The Chaos method also has many intrinsic properties that are very good for image encryption, such as high sensitivity to initial conditions, ergodicity, aperiodicity, unpredictability, control parameters, and random behavior based on

parameters[5, 7, 11, 26]. One method of chaos that is widely used is Arnold's chaotic map (ACM). This method has been widely applied in various studies such as image encryption [3, 5-6, 17, 22]. This method is quite popular because it can increase security, especially from differential attacks. Even this method is also applied to other topics such as image watermarking [2, 27], and steganography [28] to increase the security of messages embedded in the host image. ACM has a fairly simple algorithm, so it's easy to develop and combine with other algorithms. In some researches like [13, 17, 21-22], by combining the ACM method and other methods resulted in a significant increase in encryption performance, and strong against various attacks. Chaotic encryption methods used in this research can be calculated by Eq. (1), while the decryption process can be calculated with Eq. (2).

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

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix}^{-1} \begin{bmatrix} x' \\ y' \end{bmatrix} mod\ M \quad (2)$$

Where the input image has dimensions $M \times M$; x and y are the pixel coordinates of the image; $x'$and y' are the coordinates of the encrypted image; p and q are positive integers. Both encryption and decryption processes are carried out in iterations that can be determined. In this research, the number of iterations for each image block will be dynamic based on the key entered, which will be discussed in detail in the next chapter.

2.2 Stream cipher

A stream cipher is one of the symmetric cryptographic methods performed by permutation and substitution operations to bind the diffusion characters. The characteristic of this method is to generate a stream key based on the input key, which is then performed substitution encryption using XOR operation based on the stream key. The stream method has been combined with chaotic methods as in research [5, 8, 11, 23, 29, 30]. Combining these two methods has proven effective in increasing the encryption security of statistical attacks, producing good process diffusion, increasing entropy, and increasing histogram uniformity. In general, the keystream generation process is carried out through three stages, namely the permutation process of the input key to produce a stream key (S-Box), then a pseudo-random process is carried out to produce a pseudo key, and the last process is a substitution process using XOR operations. In detail, the encryption stage using the stream cipher method consists of three stages, viz.

1. Generate S-Box based on a key with the permutation of 256 iterations, see the pseudo-code below.

```
i=1;j=1; Sbox = 0:255;
while (i<=256){
  j=1 + mod(j + Sbox (i) + key(1 +
  mod(i, keylength)), 256);
  swap(Sbox(i), Sbox(j));
  i++;
}
```

   Where Sbox was originally an array with an index number of 256, which initially contained 0 to 255; mod is a function to get the remainder of the quotient; and swap is a function to swap variables
2. Generate pseudo-random key (P_k) based on S-Box

```
i=1;j=1; k=1; n = totaPixelImage;
while (k<=n){
  i = 1 + mod(i + 1, 256);
  j = 1 + mod(j + Sbox (i), 256);
  swap(Sbox(i), Sbox(j))
  pKey = 1 + mod(Sbox(i) + Sbox(j),
  256);
}
```

3. Perform the XOR operation on the input images and P_k to produce encryption

2.3 Hash function

The hash function is one method of unidirectional encryption that is widely used to encrypt keywords or passwords. This method makes the results of encryption cannot be decrypted or returned to its original form. Some of the hash methods that are widely used are MD5 and SHA-1. This method is also widely used in some research such as image encryption [15, 24, 31-32]. By encrypting the key using the hash function, the input key will be safe. In some of these studies, the hash function was used to increase the security of the input key. In this research, the hash function encryption results will be utilized to form chaos lock patterns and more dynamic key streams, to increase image encryption security. SHA-1 is a hash function that is used to encrypt the input key, resulting in a 40 char key. Later this 40 char will be extended according to the number of image blocks and used as a different chaotic encryption pattern for each image block.

3. Proposed method

This section explains the encryption and decryption paths of the proposed method. In the case of encryption, the proposed method requires input in the form of a text key and color image and produces an encrypted image as its output, as an illustration, can be seen in Fig. 1, wherein more detail is described as follows:

1. The text key is read and then the SHA-1 hash function is generated to produce the hash key, then convert the hash key to an ASCII number. The purpose of using the hash function is here so that chaotic encryption keys that are later created have a more dynamic pattern based on hash keys.
2. Color images are read, then separated each color channel, where later each color channel will be processed with the same steps.
3. Each color channel is divided into small blocks of 64 × 64 pixels. If the image size is 512 × 512, there will be 64 blocks on each channel.
4. Create a key for chaotic encryption by performing modulus operations and key extensions according to the number of blocks available. This key is used to determine the number of chaotic encryption iterations per block, see pseudocode below.

$$
\begin{aligned}
&for(i = 1{:}n)\\
&\quad j = mod(i, lk) + 1;\\
&\quad ck(i) = mod(hk(j), m) + 1;\\
&end
\end{aligned}
$$

Where i = variable for iteration/index; n = number of blocks; lk= length of the key; mod = modulus operation; ck = chaos key; m =max chaos iteration; hk=ASCII number of the hash key; j= index of the hash key

5. Encrypt the chaotic of each block according to the chaos key and Eq. (1) to compute, where the encryption results will again produce a random image.
6. Create a stream key based on the hash key according to the theory presented in section 2.2
7. Use the stream key to re-encrypt the whole image to produce the final encrypted image.

A description of the decryption stage has been presented in Fig. 2, where at this stage it takes input in the form of the same key as the encryption and encrypted image process. In detail the stages of decryption are explained as follows:

1. The key input is read then the hash function is applied to generate the hash key. Change the hash key to an ASCII number.

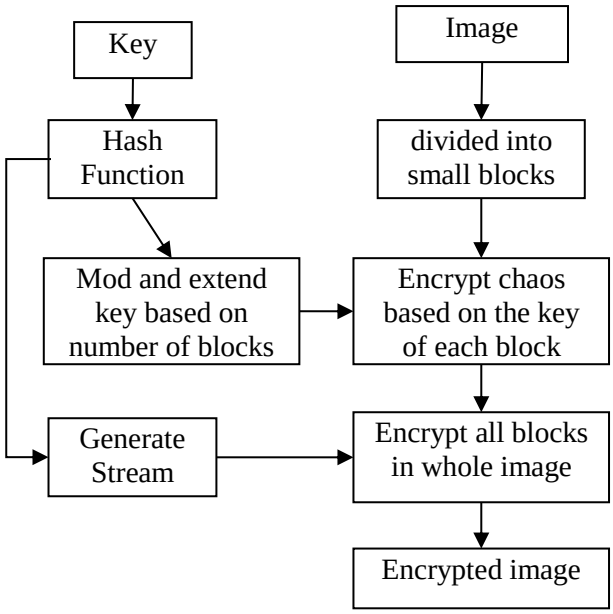

Figure. 1 Proposed encryption method

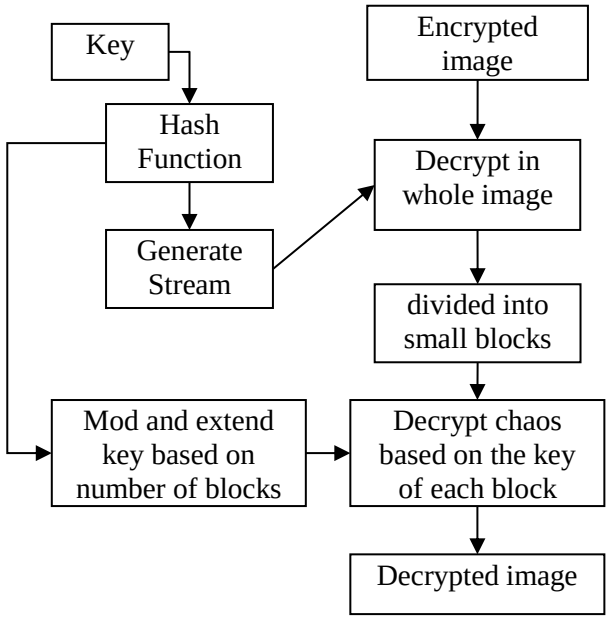

Figure. 2 Proposed decryption method

2. Create a stream key based on the hash key, according to the theory in section 2.2.
3. Read the encrypted image, then decrypt it based on the keystream that was created.
4. The decryption results are divided into 64 × 64 small blocks.
5. Create chaotic keys based on hash keys, in the same way as the encryption process in step four.
6. Decrypt chaos on each block based on chaotic keys and Eq. (2) to compute.
7. Decryption results for each block are then rearranged according to their location to produce decrypted images.

4. Implementation and discussions

At this stage, the proposed method will be tested on several standard images so that it is easy to make comparisons with previous research and easier to measure its contribution. In addition to being tested on standard images, this method has also been applied as security for image data in applications that are the result of research from research [33, 34]. Fig. 3 shows the image data set that was tested in this study.

Based on Fig. 3 there are three standard image samples and three batik images with 512 × 512 dimensions. The standard image used can be downloaded from [35], where all original images are in RGB format, then to convert to grayscale images Eq is used. (3). As for the batik image taken from the same dataset as the study [33, 34]. Especially in the batik image, a preprocessing process with imcrop and imresize functions in Matlab is used to change the image dimensions to 512 × 512.

$$GS = 0.2989 \times R + 0.5870 \times G + 0.1140 \times B \quad (3)$$

Where GS = grayscale image, R =red channel, G=green channel, B=blue channel

As has been explained in section 3, the proposed method uses three encryptions, namely hash for key

Figure. 3 Dataset image used: (a) Lena_gray, (b) Baboon_grey, (c) Peppers_gray, (d) Lena_color, (e) Baboon_ color, (f) Peppers_ color, (g) Batik Nitik, (h) Batik Kawung, and (i) Batik Parang

Table 1. The key text used to encrypt the image

Text	Hash results
password	5baa61e4c9b93f3f0682250b6cf8331b7ee68fd8
passwor**c**	6ef65c000bbf482f374f90406f203e4724066132
pas**t**word	59a6ce614876a6dec4472913e1f83489325863ea
password	fb75d844432e2448bf5a604e47dbdc06a91be4d0

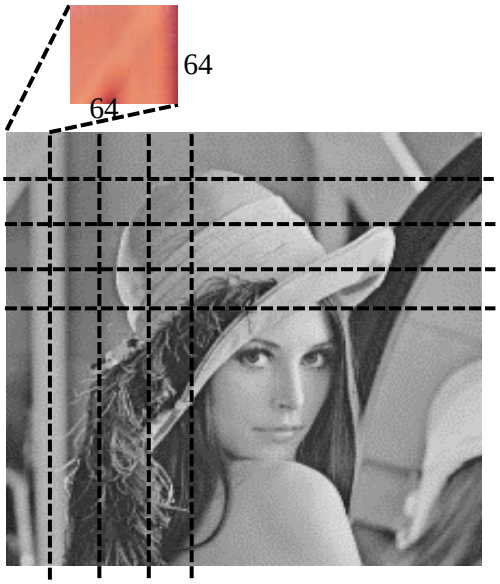

Figure. 4 Image zoning with dimensions of each zone

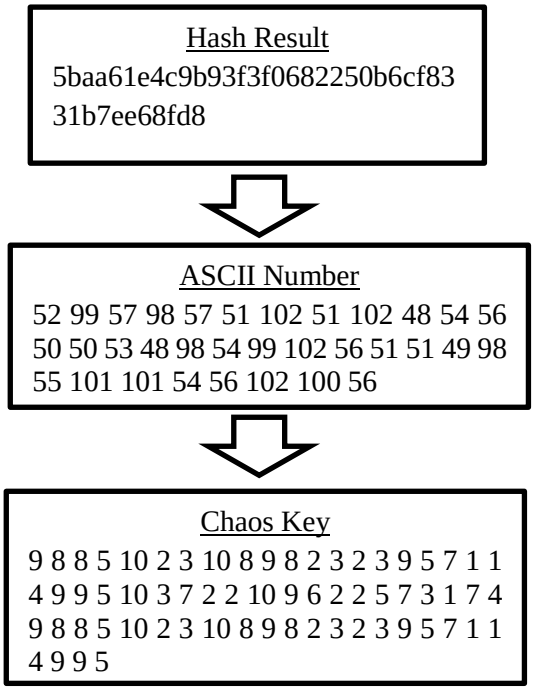

Figure. 5 Step by step to generate chaos key

encryption and a combination of chaotic encryption and streams for images based on encrypted keys. In this research, four key types and hash encryption

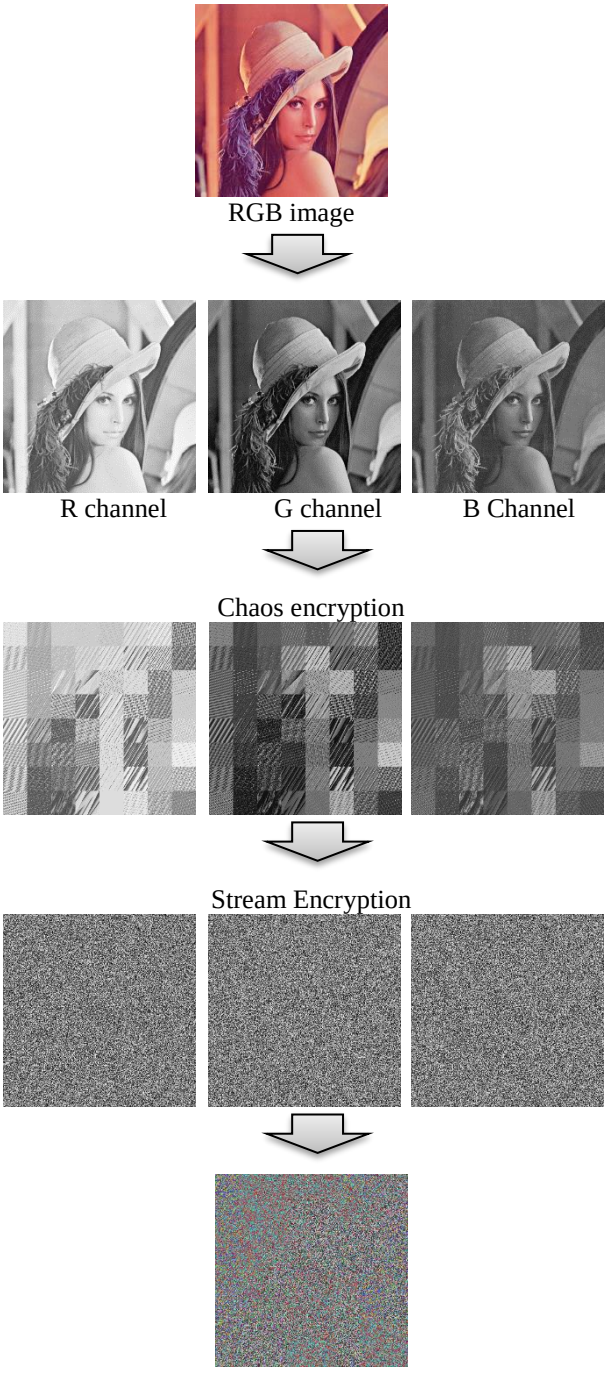

Figure. 5 Step by step to the encryption process

results are presented in Table 1. The standard text key used is "password", while "passworc", "pastword", "qassword" will be used to measure the avalanche effect.

The result of hash encryption on the password will change the password character length to 40. The forty characters are then converted to ASCII numbers. To generate chaotic keys, a mod operation and extend key are used using the pseudocode in step 4 of the encryption process.

The chaos key length will be adjusted according to the number of blocks divided by the image. In images with dimensions of 512 × 512, and blocks with dimensions of 64 × 64, a total of 64 is generated, see Fig. 4. Note that Fig. 4 only describes the process of dividing blocks or zoning, on an RGB image before the zoning process is carried out, the image is broken up into three channels, respectively red, green, and blue. Then the key hash results are extended from 40 to 64, the results of the extension process are presented in Fig. 5, where the maximum chaos iteration value specified in the example above is 10. For RGB images, it means that one chaos key generated is used for all channels. Furthermore, after chaotic encryption is complete, a stream key is produced based on a hash key to encrypt the image on the second layer. Step by step the color image encryption process is presented in Fig. 6.

After the encryption results are obtained, the image encryption quality is tested by several measuring devices, the first is entropy information. Entropy is the encryption standard measurement tool used to determine the quality of the randomness of image encryption statistically. Image encryption will certainly have a random character with a certain texture, entropy measurement is used to know this. With Entropy value getting closer to 8, statistically, it will make encrypted images more difficult to be decoded by irresponsible parties[11-12, 22]. Entropy (ε_m) can be calculated with Eq. (4).

$$\varepsilon_m = -\sum_{i=0}^{N-1} \rho(m_i) log_2 \rho\,(m_i) \qquad (4)$$

Where $\rho\,(m_i)$ and $log_2 \rho\,(m_i)$ refer to the probability of the occurrence of the symbol m_i and base 2 logarithms, respectively.

The second measuring instrument is SSIM and PSNR which function is to measure the quality of encryption visually, the more random the encryption visually it certainly will not show the smaller the value of PSNR and SSIM.[5, 20]. SSIM can be calculated by Eq. (5), whereas PSNR can be calculated by Eq. (6). The results of the measurement of image quality based on entropy, SSIM, and PSNR are presented in Table 1. Whereas in Tables 2 and 3 present a comparison of entropy values with some previous research.

$$SSIM(O,E) = \frac{(2\mu_O\mu_E+v_1)(2\sigma_{OC}+v_2)}{(\mu_O^2+\mu_E^2+v_1)(\sigma_O^2+\sigma_E^2+v_2)} \qquad (5)$$

$$PSNR\,(O,E) = 10\,log_{10}\frac{255}{MSE} \qquad (6)$$

Table 1. Entropy, SSIM, and PSNR measurement of encryption results

Image	Entropy	SSIM	PSNR
Baboon gray	7.9993	0.0091	9.1351
Lena gray	7.9993	0.0094	9.1910
Peppers gray	7.9993	0.0103	8.8759
Baboon color	7.9998	0.0091	8.7918
Lena color	7.9997	0.0093	8.6258
Peppers color	7.9998	0.0085	8.0772
Batik Kawung	7.9998	0.0067	6.7336
Batik Nitik	7.9998	0.0055	6.6552
Batik Parang	7.9998	0.0047	6.0861

Table 2. Entropy comparison in Lena color image

Method in	Entropy
Chai et al [36]	7.9993
Halagowda and Lakshminarayana [3]	7.5937
Sravanthi et al [37]	7.9993
Rehman et al [38]	7.6635
This Method	7.9997

Table 3. Entropy comparison in Lena gray image

Method in	Entropy
Bakhshandeh and Eslami [16]	7.9696
Li et al [11]	7.9972
Susanto et al [5]	7.9976
Babaei et al [39]	7.9993
This Method	7.9993

Where O is an original image; E is an encrypted image; w and h are the width and height; x, y are pixel locations; μ_O is mean of the O; μ_E is mean of the E; σ_{OC} is the covariance O against E; σ_O^2 is a variant of O; σ_E^2 is a variant of E; $v_1 = (l_1 D)^2$ and $v_2 = (l_2 D)^2$are dynamic range $(2^{bits} - 1)$ with the default value $l_1 = 0.01$ and $l_2 = 0.03$, and

$$MSE = \frac{1}{wh}\sum_{w=0}^{W-1}\sum_{h=0}^{H-1}[O(x,y) - E(x,y)]^2 \quad (5)$$

Based on the results presented in Table 1, it appears that the SSIM and PSNR values are very minimal, this shows the encryption quality is visually very good. While the entropy value also indicates an excellent encryption result, because the entropy value is close to 8. Comparison of the entropy value carried out in several related studies before, where Lena Grayscale's image was used in the test. These four studies also use chaotic methods combined with several other methods that have characteristics close to the proposed method, especially in research conducted by Rehman et al [38], that uses hash functions, so that comparison can be done fairly. The entropy value produced by this method looks superior. Similarly, the comparison of entropy values presented in Table 3, where the Lena color image is used as a comparison. It appears that the proposed method has the same entropy value as the research of Babaei et al [39], in his research also proposed a method with a composition similar to the proposed method, where there are methods of chaos, hashes, diffusion, and substitution.

Table 4. NPCR and UACI measurement of encryption results

Image	NPCR	UACI
Baboon gray	99.6057	33.8666
Lena gray	99.6212	33.4192
Peppers gray	99.6066	33.2432
Baboon color	99.6019	33.8556
Lena color	99.6009	33.2349
Peppers color	99.6104	33.2508
Batik Kawung	99.6120	33.8906
Batik Nitik	99.6033	33.7398
Batik Parang	99.6117	33.6435

Table 5. UACI and NPCR comparison in Lena color image

Method in	UACI	NPCR
Chai et al [36]	33.2800	99.6000
Sravanthi et al [37]	33.4707	99.6098
Rehman et al [38]	33.4254	99.6082
This Method	33.2349	99.6009

Table 6. UACI and NPCR comparison in Lena gray image

Method in	UACI	NPCR
Bakhshandeh and Eslami [16]	33.2161	99.4602
Susanto et al [5]	28.6600	99.6400
Babaei et al [39]	33.4197	99.6213
This Method	33.4192	99.6212

The next measuring instrument is UACI and NPCR, where both of these gauges are used to determine the quality of encryption against differential attacks. NPCR can be calculated with Eq. (7) and UACI on Eq. (8). The measurement results of the NPCR and UACI values are presented in Table 4.

$$NPCR = \left(\frac{1}{W \times H}\sum_{x=0}^{W}\sum_{y=0}^{H} D(x,y)\right) \times 100\%,$$
$$D(i,j) = \begin{cases} 0, O(x,y) = E(x,y) \\ 1, O(x,y) \neq E(x,y) \end{cases} \quad (7)$$

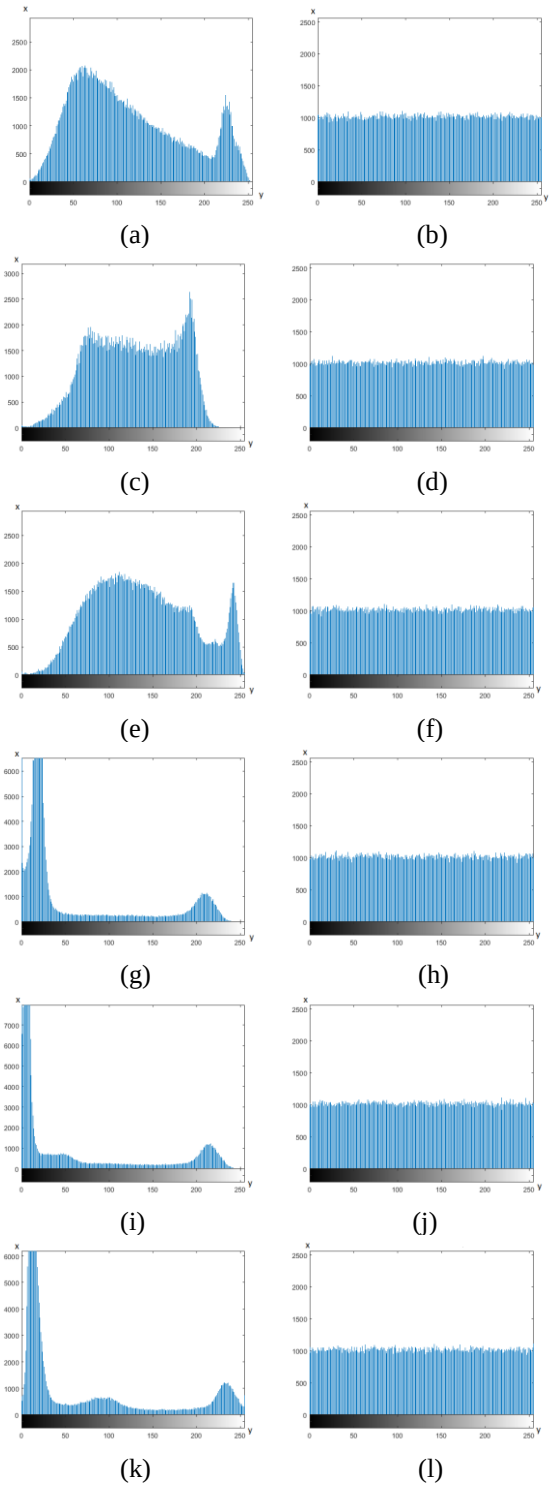

Figure. 6 Sample of histogram results: (a) baboon blue original, (b) baboon blue encrypted, (c) baboon green original, (d) baboon green encrypted, (e) baboon red original, (f) baboon red encrypted (g) batik parang blue original, (h) batik parang blue encrypted, (i) batik parang green original, (j) batik parang green encrypted, (k) batik parang red original, and (l) batik parang red encrypted

$$UACI = \left(\frac{1}{W \times H}\sum_{x=0}^{W}\sum_{y=0}^{H}\frac{|O(x,y)-E(x,y|}{(2^8)-1}\right) \times 100\% \quad (8)$$

Table 7. Avalanche effect measurement of encryption results

Image	passworc	pastword	qassword
Baboon gray	50.0109%	49.9842%	49.9460%
Lena gray	50.0202%	49.9826%	49.9815%
Peppers gray	50.0068%	50.0105%	49.9838%
Baboon color	49.9607%	50.0043%	49.9911%
Lena color	49.9900%	50.0038%	49.9529%
Peppers color	49.9850%	50.0565%	49.9879%
Batik Kawung	49.9753%	49.9801%	49.9956%
Batik Nitik	50.0259%	50.0538%	50.0300%
Batik Parang	50.0675%	49.9857%	49.9531%

Table 8. Avalanche effect comparison in Lena image

Method in	Avarege Avalanche Effect (%)
Seyedzadeh and Mirzakuchaki [18] for Lena Color	50.0123
This Method for Lena Color	49.9822
Susanto et al [5] for Lena Gray	49.9767
This Method for Lena Gray	50.0366

Based on the results presented in Table 4, it appears that the UACI and NPCR values are of excellent quality because the values are very close to the ideal values. Besides, Table 5 and Table 6 also performed a comparison of UACI and NPCR values with fairly similar previous studies. The results of UACI and NPCR scores indicate that the proposed method has very good performance because the value is close to ideal, but it does not show the best results, although it also does not become the worst when compared with comparative research.

The next measurement tool is the histogram analysis. Ideally, the encrypted image produces a histogram that is different from the original image, besides that the encrypted image histogram should ideally be uniform. A sample histogram of the resulting image is presented in Fig. 6.

Based on Fig. 6, it appears that the resulting histogram looks uniform and ideal. The last test of encrypted images in this research was the avalanche effect. This is a measurement tool to find out the change in the value of the pixel bits of the encrypted image with other encrypted images when there is a small change in the key. Key changes are generally 1 bit, where the ideal value of the avalanche effect is 50% or close to that value[12]. It has been explained in Table 1, that four types of keys are used, namely "password" as the initial key, while "#passworc", "pastword", "qassword" are used as comparison keys

Table 9. SSIM and PSNR measurement of decryption results

Image	SSIM	PSNR
Baboon gray	1	inf
Lena gray	1	inf
Peppers gray	1	inf
Baboon color	1	inf
Lena color	1	inf
Peppers color	1	inf
Batik Kawung	1	inf
Batik Nitik	1	inf
Batik Parang	1	inf

for measuring the avalanche effect. The avalanche effect measurement results are presented in Table 7.

Based on Table 7, it appears that the value of the avalanche effect is ideal, this is indicated by all AE values approaching 50%. Table 8 also presents a comparison of the avalanche effect with the previous method. Comparisons are made on the Lena image in grayscale type, where it appears that the proposed method also has very good performance and is not worse when compared to the previous method.

Research conducted by Seyedzadeh and Mirzakuchaki [17] and Susanto et al [5] also carried out an avalanche effect measurement that was equally doing a 1-bit modification on the key, both of these studies also used chaos methods combined with other methods to improve performance. The dataset used is the same, namely Lena's image, except that Susanto et al's research use grayscale image and other research uses color imagery. Based on the results presented in Table 8, it appears that the resulting method has fairly similar and has excellent performance because the values are both close to 50%.

The last step taken for measurement is the image decryption process, to ensure the decryption process runs perfectly. The encryption process will be in vain if the decryption process does not work perfectly. Measuring instruments used to determine the perfection of the decryption results are SSIM and PSNR, where infinity values must be obtained at PSNR and 1 at SSIM. The results of the measurement of these values in image decryption are presented in Table 9.

5. Conclusions

This study proposes a combination of chaotic and stream methods based on hash functions to form dynamic, strong, and layered encryption patterns. The hash function is unidirectional encryption commonly used to encrypt keys or passwords. In this research, the hash function is not only used to encrypt keys but with a few modifications, the hash function also has a role in shaping chaotic and stream encryption patterns, so that chaos and stream encryption patterns become more dynamic. Another innovation proposed in this research is encryption technique based on blocks, so it does not encrypt the whole image, with this model each image block has a different encryption key than the other blocks, so logically it will produce stronger and more robust encryption against attack. This method is implemented in digital images and has been tested on grayscale and color images. Based on the test results, the proposed method is proven to have excellent performance which has been proven by various measurement instruments such as PSNR, SSIM, entropy, histogram analysis, UACI, NPCR, and avalanche effects. All measurement instruments show that the proposed method has excellent performance as evidenced by the ideal values obtained. The results of comparison with previous methods also show that the proposed method has a relatively better performance than previous studies, especially in increasing the strength of statistical attacks as evidenced by the entropy value.

Conflicts of Interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome. We confirm also that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed.

References

[1] C. Edward Jaya Singh and E. Baburaj, “XOR Reformed Paillier Encryption Method with Secure De-duplication for Image Scaling and Cropping in Reduced Cloud Storage”, *International Journal of Intelligent Engineering and Systems*, Vol. 12, No. 4, pp. 328-337, 2019.

[2] A. Setyono and D. R. I. M. Setiadi, “An Image Watermarking Method Using Discrete Tchebichef Transform and Singular Value Decomposition Based on Chaos Embedding”, *International Journal of Intelligent Engineering and Systems*, Vol. 13, No. 2, pp. 140–150, 2020.

[3] S. R. M. Halagowda and S. K. Lakshminarayana, “Image Encryption Method based on Hybrid Fractal-Chaos Algorithm”, *International Journal of Intelligent Engineering and Systems*, Vol. 10, No. 6, pp. 221-229, 2017.

[4] S. Koppu and M. Viswanatham, “2D Chaotic

Map Based on 2D Adaptive Grey Wolf Algorithm for Ultra Sound Medical Image Security", *Image Encryption Method based on Hybrid Fractal-Chaos Algorithm*, Vol. 10, No. 1, pp. 104-113, 2017.

[5] A. Susanto, D. R. I. M. Setiadi, E. H. Rachmawanto, I. U. W. Mulyono, C. A. Sari, M. K. Sarker, and M. R. Sazal "Triple layer image security using bit-shift, chaos, and stream encryption", *Bull. Electr. Eng. Informatics*, Vol. 9, No. 3, pp. 980–987, 2020.

[6] E. H. Rachmawanto, D. R. I. M. Setiadi, C. A. Sari, H. A. Santoso, F. A. Rafrastara, and E. Sugiarto, "Block-based Arnold chaotic map for image encryption", In: *Proc. of International Conf. on Information and Communications Technology*, pp. 174–178, 2019.

[7] J. A. P. Artiles, D. P. B. Chaves, and C. Pimentel, "Image encryption using block cipher and chaotic sequences", *Signal Process. Image Commun.*, Vol. 79, pp. 24–31, 2019.

[8] K. U. Shahna and A. Mohamed, "A novel image encryption scheme using both pixel level and bit level permutation with chaotic map", *Appl. Soft Comput. J.*, Vol. 90, p. 106162, 2020.

[9] X. Zhang, L. Wang, G. Cui, and Y. Niu, "Entropy-Based Block Scrambling Image Encryption Using DES Structure and Chaotic Systems", *Int. J. Opt.*, 2019.

[10] H. Liu, A. Kadir, and J. Liu, "Color pathological image encryption algorithm using arithmetic over Galois field and coupled hyper chaotic system", *Opt. Lasers Eng.*, Vol. 122, pp. 123–133, 2019.

[11] Y. Li, C. Wang, and H. Chen, "A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation", *Opt. Lasers Eng.*, Vol. 90, pp. 238–246, 2017.

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

[13] N. A. Abbas, "Image encryption based on Independent Component Analysis and Arnold's Cat Map", *Egypt. Informatics J.*, Vol. 17, No. 1, pp. 139–146, 2016.

[14] R. M. Rad, A. Attar, and R. E. Atani, "A New Fast and Simple Image Encryption Algorithm Using Scan Patterns and XOR", *Int. J. Signal Process.*, Vol. 6, No. 5, pp. 275–290, 2013.

[15] 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.

[16] A. Bakhshandeh and Z. Eslami, "An authenticated image encryption scheme based on chaotic maps and memory cellular automata", *Opt. Lasers Eng.*, Vol. 51, No. 6, pp. 665–673, 2013.

[17] S. Farwa, N. Muhammad, T. Shah, and S. Ahmad, "A Novel Image Encryption Based on Algebraic S-box and Arnold Transform", *3D Res.*, Vol. 8, No. 3, pp. 1–14, 2017.

[18] S. Mohammad Seyedzadeh and S. Mirzakuchaki, "A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map", *Signal Processing*, Vol. 92, No. 5, pp. 1202–1215, 2012.

[19] Y. Jain, R. Bansal, G. Sharma, B. Kumar, and S. Gupta, "Image Encryption Schemes: A Complete Survey", *Int. J. Signal Process.*, Vol. 9, No. 7, pp. 157–192, 2016.

[20] D. R. I. M. Setiadi, E. H. Rachmawanto, C. A. Sari, A. Susanto, and M. Doheir, "A Comparative Study of Image Cryptographic Method", In: *Proc. of 2018 5th International Conf. on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, pp. 336–341, 2018.

[21] M. Bi, X. Zhou, G. Yang, M. Hu, B. Fan, X. Yang, and W. Hu, "Chaotic Arnold transform and chirp matrix encryption scheme for enhancing the performance and security of OFDM-PON", *Opt. Fiber Technol.*, Vol. 51, pp. 64–70, 2019.

[22] N. R. Zhou, T. X. Hua, L. H. Gong, D. J. Pei, and Q. H. Liao, "Quantum image encryption based on generalized Arnold transform and double random-phase encoding", *Quantum Inf. Process.*, Vol. 14, No. 4, pp. 1193–1213, 2015.

[23] A. Jolfaei and A. Mirghadri, "An Image Encryption Approach using Chaos and Stream Cipher", *J. Theor. Appl. Inf. Technol.* , Vol. 19, no. 2, pp. 117–128, 2010.

[24] C. Chen, K. Sun, and S. He, "An improved image encryption algorithm with finite computing precision", *Signal Processing*, Vol. 168, p. 107340, 2020.

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

[26] 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.

[27] T. Venugopal, V. Siva, and K. Reddy, "Image Watermarking Using Two Level Encryption Method Based on Chaotic Logistic Mapping and Rivest Shamir Adleman Algorithm", *Int. J.*

Intell. Eng. Syst., Vol. 11, No. 6, 2018.

[28] 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, p. 103, 2018.

[29] X. Wang, X. Zhu, and Y. Zhang, "An Image Encryption Algorithm Based on Josephus Traversing and Mixed Chaotic Map", *IEEE Access*, Vol. 6, pp. 23733–23746, 2018.

[30] 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", In: *Proc. of J. Phys. Conf. Ser*, Vol. 1201, No. 1, p. 012022, 2019.

[31] J. C. Dagadu, J. Li, and F. Shah, "An efficient di-chaotic diffusion based medical image cryptosystem", In: *Proc. of 2016 13th International Computer Conf. on Wavelet Active Media Technology and Information Processing, ICCWAMTIP 2017*, Vol. , pp. 206–210, 2018.

[32] N. Wang, G. Di, X. Lv, M. Hou, D. Liu, J. Zhang, and X. Duan "Galois Field-Based Image Encryption for Remote Transmission of Tumor Ultrasound Images", *IEEE Access*, Vol. 7, pp. 49945–49950, 2019.

[33] F. Budiman, A. Suhendra, D. Agushinta, and A. Tarigan, "Determination of SVM-RBF kernel space parameter to optimize accuracy value of Indonesian Batik images classification", *J. Comput. Sci.*, Vol. 13, No. 11, pp. 590–599, 2017.

[34] F. Budiman, "SVM-RBF parameters testing optimization using cross validation and grid search to improve multiclass classification", *Sci. Vis.*, Vol. 11, No. 1, pp. 80–90, 2019.

[35] Ming Hsieh Department of Electrical Engineering USC Viterbi School of Engineering, "SIPI Image Database", [Online]. Available: http://sipi.usc.edu/database/. [Accessed: 2019].

[36] X. L. Chai, Z. H. Gan, Y. Lu, M. H. Zhang, and Y. R. Chen, "A novel color image encryption algorithm based on genetic recombination and the four-dimensional memristive hyperchaotic system - IOPscience", *Chinese Phys. B*, Vol. 25, 2016.

[37] D. Sravanthi, K. Abhimanyu Kumar Patro, B. Acharya, and S. Majumder, "A secure chaotic image encryption based on bit-plane operation", in *Advances in Intelligent Systems and Computing*, Vol. 758, Springer Verlag, pp. 717–726, 2018.

[38] A. ur Rehman, X. Liao, R. Ashraf, S. Ullah, and H. Wang, "A color image encryption technique using exclusive-OR with DNA complementary rules based on chaos theory and SHA-2", *Optik (Stuttg).*, Vol. 159, pp. 348–367, 2018.

[39] A. Babaei, H. Motameni, and R. Enayatifar, "A new permutation-diffusion-based image encryption technique using cellular automata and DNA sequence", *Optik (Stuttg).*, Vol. 203, p. 164000, 2020.