



A combination of Hill CIPHER-LSB in RGB image encryption

Rama Aria Megantara^{*1}, Fauzi Adi Rafrastara², Syafrie Naufal Mahendra³

Universitas Dian Nuswantoro, Indonesia^{1,2,3}

Article Info

Keywords:

Cryptography, Steganography, Hill Cipher, LSB

Article history:

Received 12 March 2019

Revised 13 May 2019

Accepted 21 June 2019

Published 30 July 2019

Cite:

Megantara, R., Rafrastara, F., & Mahendra, S. (2019). A Combination of Hill CIPHER-LSB in RGB Image Encryption. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, 4(3).

doi:<http://dx.doi.org/10.22219/kinetik.v4i3.785>

* Corresponding author.

Rama Aria Megantara

E-mail address:

aria@dsn.dinus.ac.id

Abstract

The progress of the development of digital technology today, many people communicate by sending and receiving messages. However, along with extensive technological developments, many crimes were committed. In avoiding these crimes, data security needs to be done. Form of data security in the form of cryptography and steganography. One of the cryptographic techniques is the hill cipher algorithm. Hill ciphers include classic cryptographic algorithms that are very difficult to solve. While the most popular steganography technique is Least Significant Bit (LSB). Least Significant Bit (LSB) is a spatial domain steganography technique using substitution methods. This study discusses the merging of message security with hill cipher and LSB. The message used is 24-bit color image for steganography and text with 32, 64 and 128 characters for cryptography. The measuring instruments used in this study are MSE, PSNR, Entropy and travel time (CPU time). Test results prove an increase in security without too damaging the image. This is evidenced by the results of the MSE trial which has a value far below the value 1, the PSNR is > 64 db, the entropy value ranges from 5 to 7 and the results of travel time < 1 second.

1. Introduction

The progress of the development of digital technology today, many people communicate by sending and receiving messages. The message sent can be in the form of images, text, video or sound. However, along with extensive technological developments, many crimes are committed by people who are not responsible for carrying out digital crimes such as data theft (hacking). In avoiding these crimes, data security needs to be done [1]. One form of data security is by inserting a message against another message and making the message unrecognizable. This security technique is called cryptography and steganography [2]. Cryptography is a technique to change the original message by using a key into a message that cannot be recognized by someone who does not have the key [3]. Meanwhile, steganography is the technique of hiding messages in other messages so that the message cannot be recognized [4].

The steganography method in this study is the Least Significant Bit (LSB) method. LSB is a very simple method because it only changes the value of the last bit in a message with the message bit inserted [5]. This method produces a message that is similar to the original message so it is very difficult to know if there is a change in the message [6]. However, from the simplicity and ease of this LSB, this method is vulnerable to attacks [7].

In reducing the vulnerability of attacks on LSB, messages need to be secured before insertion using cryptography [8]. Many cryptographic methods are used to secure a message, namely by means of security with a symmetrical and asymmetrical key. One of the cryptographic methods with symmetric keys is hill cipher. Hill cipher has a key in the form of a matrix and is used to encrypt messages by multiplying the value of a character with that key [9]. Hill cipher is a very simple and fast processing method.

Previous research say that they have used a combination of LSB and Hill Cipher as Swain did by implementing a combination of LSB-Hill on the cover in the form of digital images but the message that is inserted in the form of text. The LSB used has been modified first by operating the last 3 bits of LSB [10]. Another study conducted by Wamiliana [11] used a cover in the form of an image by giving a difference in the size of the brightness change interval and where the message was inserted in the form of a txt file, resulting in a larger image file size with a significant KB difference. Whereas in the research conducted by Sari [12], proving that the use of short text characters in the message inserted into digital images does not change the file size at all.

In this study, proposed a combination of two methods, namely steganography using Least Significant Bit (LSB) and cryptography using hill cipher to provide multiple security of messages. The message used in this study is text and image. The images used are 256 × 256 pixels and 512x512 pixels, while the text used has a length of 32, 64, 128 characters.

2. Research Method

2.1 Hill Cipher

Hill Cipher was first introduced and developed by Lester Hill in 1929. Hill Cipher is a polyalphabetic cipher that is categorized as a block cipher, because the text to be processed will be divided into blocks of a certain size. Each character in one block will affect the other characters in the encryption and decryption process, so that the same character is not mapped to be the same character as well [13]. Hill Cipher includes classic cryptographic algorithms that are very difficult to solve [14]. The security process carried out by hill cipher uses multiplication and inverse of the matrix [15]. The key to hill cipher is a matrix measuring $m \times m$, it can be written in Equation 1.

$$Key = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{bmatrix} \tag{1}$$

Encryption technique in hill cipher is done per plaintext block, the size of the key matrix must be the same as the block size [16]. Before dividing text into rows of blocks, the plaintext is changed to a decimal number, such as A = 0, B = 1, C = 2 and show in Figure 1.

A	B	C	D	E	F	G	H	I	J	K	L	M	N
0	1	2	3	4	5	6	7	8	9	10	11	12	13
O	P	Q	R	S	T	U	V	W	X	Y	Z		
14	15	16	17	18	19	20	21	22	23	24	25		

Figure 1. Alphabeth Indexing in Hill Cipher

The encryption process on Hill Cipher can be done by Equation 2.

$$C = Key \times P \text{ mod } 26 \tag{2}$$

Where:

- C = The result of encryption with Hill Cipher
- Key = Key value in the form of matrix $m \times m$
- P = Plainteks is a decimal number

To return to the initial value, the decryption process is performed. This decryption process requires a key, the key used is the inverse of the key matrix. The key matrix must be obtained as in Equation 3.

$$Key \times Key^{-1} = 1 \tag{3}$$

The decryption process on Hill Cipher can be done by Equation 4.

$$D = Key^{-1} \times C \tag{4}$$

Where:

- D = Plaintext or initial character before being encrypted
- Key = Key value in the form of matrix $m \times m$
- C = Ciphertext is a decimal number

2.2 Least Significant Bit (LSB)

Message hiding is done by changing the bits in the image segment with secret message bits. Least Significant Bit (LSB) is a spatial domain steganography technique [17] with substitution method. LSB is one of the most popular techniques in steganography because it is easy to apply and relatively simple [18]. The one of media used in the LSB technique is image. Images that can be used are 1 bit (Binary), 3 bits (Grayscale) and 24 bits (colors). LSB must have an imperceptibility value [19], the image can reach the limits of the Human Visual System (HVS) which is above 40 dB.

In this study the image used for LSB technique is a color image (RGB) with a size of 24 bits. Color image (RGB) is an image consisting of 3 components. The component consists of Red (1 byte), Green (1 byte), and Blue (1 byte). To hide an image in an LSB on each byte of a 24-bit image, it can be stored 3 bytes in each pixel.

Changes from the inserted image cannot be seen significantly with human flushing. However, this change can be proven by testing using measuring instruments such as Peak Signal Noisy to Ratio (PSNR), Mean Square Error (MSE) and Entropy from the resulting image.

2.3 Proposed Scheme

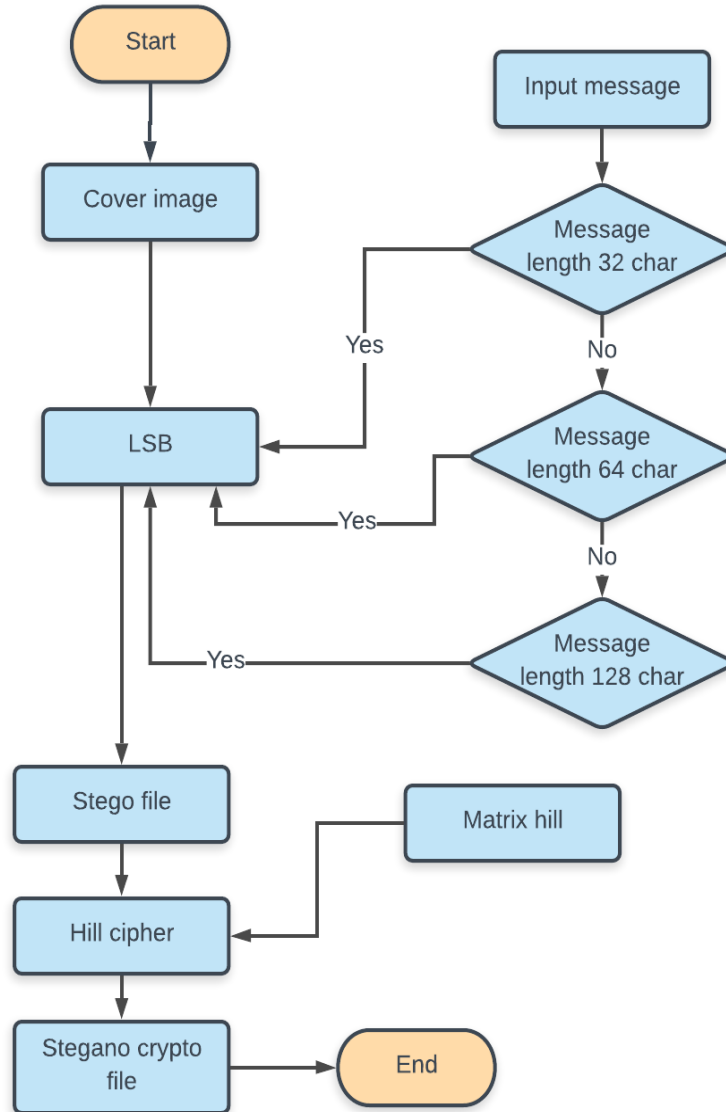


Figure 2. Proposed Scheme Using LSB-Hill Cipher









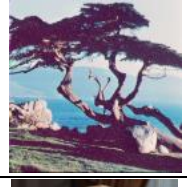
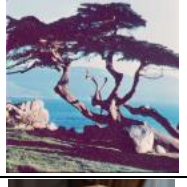
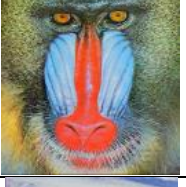

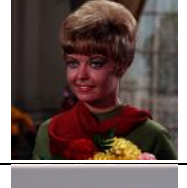



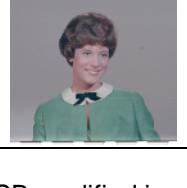
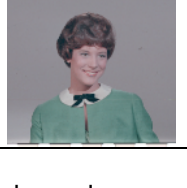


3. Results and Discussion

This study used digital images with image sizes of 256 × 256 pixels. The image used is a Red, Green, Blue (RGB) color image or a 24 bit image. This color image has tiff extension. The image is a standard color image that has been widely used in researches, this is to facilitate the process of comparing with the next study. The image in Table 1 is taken from the SIPI website.

The message encryption used is in the form of characters. This encryption aims to hide the message so that when the image is extracted to the original image form, the message cannot be recognized. So that increases the security of a message. The message used in this study amounted to 32, 64, and 128 characters which has been visualized in Figure 2.

This section will display the results of inserting text messages into the original image that will produce insertion images. The results of the insertion image will be tested for quality using PSNR, MSE, Entropy and insertion time (CPU time). The encryption results inserted into the original image do not significantly change the original image. This can be seen from the results of histogram in Table 2.

Table 1. Comparison Between Original and Result Image After Embedding Process

Name of Image	Original Image (512x512 pixels)	After Embedding	Name of Image	Original Image (256x256 pixels)	After Embedding
House.tiff			Peppers.tiff		
Candy.tiff			Droplets.tiff		
Tree.tiff			Babon.tiff		
Woman1.tiff			Jet.tiff		
Woman2.tiff			Lake.tiff		

Here, we proposed LSB modified in red channel:

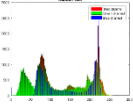
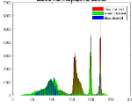
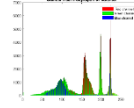
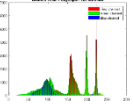
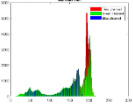
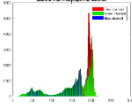
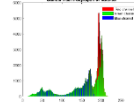
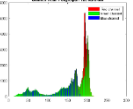
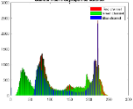
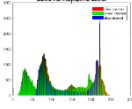
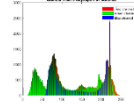
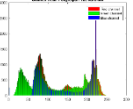
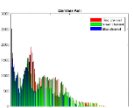
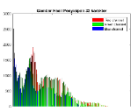
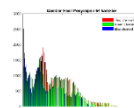
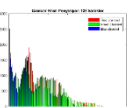
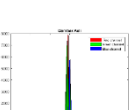
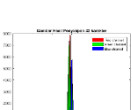
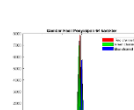
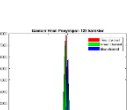
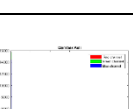
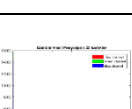

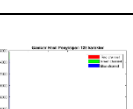
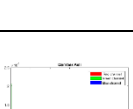
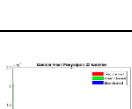
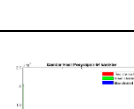
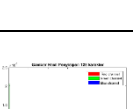
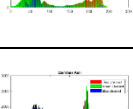
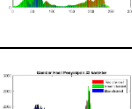
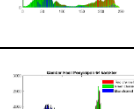
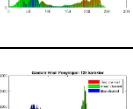
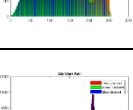
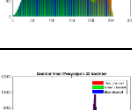
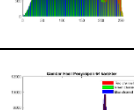
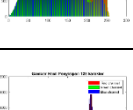
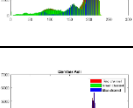
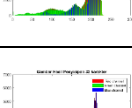
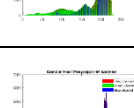
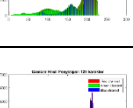
```

for i = 1 : height
    for j = 1 : width

        LSB = mod(double(rc(i,j)), 2);
        if (k>m || LSB == b(k))
            rs(i,j) = rc(i,j);
        else
            if(LSB == 1)
                rs(i,j) = rc(i,j) - 1;
            else
                rs(i,j) = rc(i,j) + 1;
            end
            k = k + 1;
        end
    end
end
end
end
    
```

From Table 2, it can be seen that the insertion results of 32, 64, and 128 images do not change significantly and cannot be compared visually to humans. Because the LSB algorithm functions only change the block value of an image with a difference of 1 or 0. So that the image changes do not change drastically and do not damage the quality of the original image. To see changes in the insertion image needs to be done by testing. This test uses MSE, PSNR, entropy, and travel time (CPU time).

Table 2. Histogram Comparison Between Original Image and After Embedding

Name of image	Original histogram	Histogram after embedding		
		32x32 pixels message	64x64 pixels message	128x128 pixels message
House.tif				
Candy.tif				
Tree.tif				
Woman1.tif				
Woman2.tif				
Peppers.tif				
Droplets.tif				
Babon.tif				
Jet.tif				
Lake.tif				

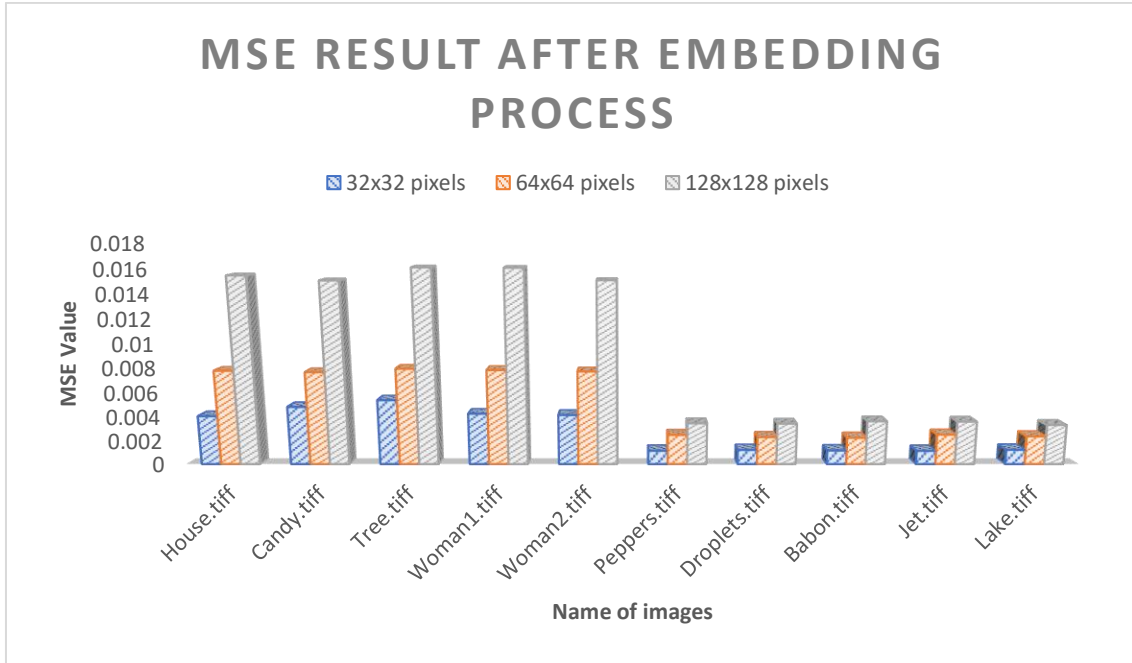


Figure 3. MSE Result After Embedding Process Using Several Size of Message

Based on Figure 3, it can be seen that the result of the insertion image with 32, 64 and 128 characters with a size of 256 × 256 and 512 × 512 pixels produces MSE that is far below 1. This shows that the result of the insertion image is not significantly damaged. MSE value is influenced by the pixel size of the image and many characters are inserted, the higher the image size and the less the insertion character, the smaller the MSE value. MSE with a low value is owned by an image measuring 512 × 512 pixels with a number of 32 characters. The MSE value in the color Red, Green, Blue (RGB) has the same value.

In another test, the calculation of the PSNR value has been done to determine the level of imperceptibility in the resulting image. This is indicated by the PSNR value in each color Red (R), Green (G) and Blue (B) which has a value > 64 db. The PSNR value is also influenced by the maximum pixel value of the image and MSE, the greater the maximum pixel value and the smaller the MSE value, the higher the PSNR result. The image that has the highest PSNR is the image of babon.tiff with a size of 512 × 512 with insertions of 32 characters that have a value of Red (R) of 78.233803, Green (G) of 77.561240, and Blue (B) of 75.223503. Whereas if viewed from each color Red (R), Green (G) and Blue (B), the image that has the highest color Red (R) PSNR with a value of 78.233803 is babon.tiff, while images that have a PSNR color the lowest red (R) with a value of 64.910253 is chocolate. tiff. For Green (G), the image that has the highest Green (G) PSNR with a value of 77.956939 is fruit. tiff. the image that has the lowest Green (G) PSNR with a value of 64.670075 is brown. tiff. Furthermore, in the color Blue (B), the babon.tiff image is an image that has the highest PSNR value that can be seen in graph 2 with a value of 78.233803 and a brown image. tiff is an image that has the lowest PSNR value with a value of 63.452059. due to achieve security, this study also tested using entropy as shown in Table 3.

According to Table 3, the results of entropy and travel time are obtained for each image. From the results of entropy testing it can be seen that the highest entropy value lies in the babon.tiff image with a size of 512 × 512 pixels which has an entropy value of each color > 7 indicated by entropy Red (R) of 7.706682, entropy Green (G) of 7.474479 and Blue entropy (B) amounting to 7.752192. This proves that the babon.tiff insert image has a high similarity to the original image. Whereas woman2.tiff image with a size of 256 × 256 pixels has a low entropy with the entropy value of each color > 5 is indicated by entropy Red (R) of 5.714986, entropy Green (G) is 5.373754 and entropy Blue (B) is 5.711664. If on the average of each color Red (R), Green (G) and Blue (B) the entropy value has a value of 6.763286 for Red (R), the value of 6.831159 for Green (G) and the value of 6.642764 for Blue (B). The number of insert characters does not significantly affect the image entropy value.

In Table 3, the travel time for inserting characters into the original image. The fastest travel time is in the brown image. Tiffs with a size of 256 × 256 pixels and insertions of 32 characters are executed for 0.065989 seconds, while the longest travel time is in the image .tiff with the size of 512 × 512 pixels and insertions of 32 characters that are executed for 0.221902 seconds. If the average travel time obtained when inserting text into an image can be reached in 0.141989 seconds. It can be obtained that the travel time is influenced by the pixel size of the image and the memory capacity on the computer.

Table 3. Entropy Value and CPU Time Results

Image	Character of message	Entropy			CPU Time (second)
		Red (R)	Green (G)	Blue (B)	
House.tiff	32	6.431178	6.539253	6.232546	0.067947
	64	6.431280	6.539520	6.232761	0.070593
	128	6.431413	6.539961	6.233543	0.073697
Candy.tiff	32	5.792081	6.219557	6.798634	0.065989
	64	5.792139	6.219618	6.798626	0.066984
	128	5.792168	6.219676	6.798522	0.067682
Tree.tiff	32	7.210443	7.413782	6.921459	0.074814
	64	7.210398	7.413890	6.922029	0.072338
	128	7.210382	7.414114	6.922788	0.071979
Woman1.tiff	32	6.441073	6.456180	6.388330	0.069047
	64	6.455471	6.466178	6.396295	0.069111
	128	6.483614	6.479776	6.407888	0.073810
Woman2.tiff	32	5.715411	5.374076	5.712213	0.068603
	64	5.715262	5.373923	5.711950	0.073329
	128	5.714986	5.373754	5.711664	0.067390
Peppers.tiff	32	7.338813	7.496247	7.058341	0.221902
	64	7.338811	7.496263	7.058351	0.212155
	128	7.338810	7.496245	7.058373	0.214257
Droplets.tiff	32	6.948074	6.884456	6.126479	0.205300
	64	6.948106	6.884599	6.126531	0.203571
	128	6.948129	6.884646	6.126577	0.202529
Babon.tiff	32	7.706672	7.474449	7.752217	0.217427
	64	7.706674	7.474465	7.752218	0.220340
	128	7.706682	7.474479	7.752192	0.219150
Jet.tiff	32	6.717771	6.798980	6.213793	0.208795
	64	6.717772	6.799018	6.213800	0.214365
	128	6.717772	6.799056	6.213840	0.216301
Lake.tiff	32	7.312406	7.642856	7.213642	0.217283
	64	7.312400	7.642858	7.213643	0.220395
	128	7.312402	7.642891	7.213660	0.212591
Average		6.763286	6.831159	6.642764	0.141989

4. Conclusion

Based on the results of the above tests it can be concluded that the hill cipher and LSB techniques provide more security for a message. The use of hill cipher for text encoding can increase security because messages are encrypted using a key. The hill cipher technique was also successfully combined with the LSB technique. In addition to encoding text messages, messages are also inserted using the LSB method. This merger was successfully applied to 24 bit Red, Green and Blue (RGB) color images. The result of combining the two techniques between hill cipher and LSB produces a very good image. The color image used has a size of 256 × 256 and 512 × 512 pixels. Meanwhile, the text messages used for insertion are 32, 64, and 128 characters.

The LSB method has advantages in the quality of insert images that cannot be proven by naked eye [20]. To prove the quality of the image, it is used a measuring instrument. Measuring instrument to measure good image quality is by using PSNR, MSE and entropy. This is evidenced by the value of MSE that is far below the value 1 that has been proven in graph 1. PSNR value > 64 db shown in graph 2, graph 3, and graph 4. The highest PSNR value is owned by the baboon image. Tiff with the color value Red (R) amounting to 78.233803, Green (G) of 77.561240, and Blue (B) of 75.223503. Entropy value in insert image > 5. The highest entropy value is obtained in babon.tiff image with the entropy value Red (R) of 7.706682, entropy Green (G) of 7.474479 and entropy of Blue (B) of 7.752192. The travel time obtained to run both techniques is <1 second. So that these two techniques can be used to protect from message theft. Both of these techniques have proven to have good image results, algorithms that are simple and fast in processing and have high similarities to the original image.

References

- [1] A. Cheddad, J. Condell, K. Curran dan P. Mc Kevitt, "Biometric Inspired Digital Image Steganography," in *International Conference and Workshop on the Engineering of Computer Based Systems*, 2008. <https://doi.org/10.1109/ECBS.2008.11>

- [2] C. Irawan, C. A. Sari, E. H. Rachmawanto dan S. D.R.I.M, "Hiding and Securing Message on Edge Areas of Image Using LSB Steganography and OTP Encryption," in *International Conference on Informatics and Computational Sciences (ICICoS)*, Semarang, 2017. <https://doi.org/10.1109/ICICoS.2017.8276328>
- [3] G. Ardiansyah, C. A. Sari, E. H. Rachmawanto dan S. D.R.I.M, "Hybrid Method Using 3-DES, DWT and LSB for Secure Image Steganography Algorithm," in *International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, Yogyakarta, 2017. <https://doi.org/10.1109/ICITISEE.2017.8285505>
- [4] A. E. Handoyo, D. Setiadi, E. H. Rachmawanto, C. A. Sari dan A. Susanto, "Teknik Penyembunyian dan Enkripsi Pesan pada Citra Digital dengan Kombinasi Metode LSB dan RSA," *Jurnal Teknologi dan Sistem Komputer*, Vol. 6, No. 1, Pp. 37-43, 2018. <https://doi.org/10.14710/jtsiskom.6.1.2018.37-43>
- [5] A. E. Handoyo, D. Setiadi, E. H. Rachmawanto, C. A. Sari dan A. Susanto, "Teknik Penyembunyian dan Enkripsi Pesan pada Citra Digital dengan Kombinasi Metode LSB dan RSA," *Jurnal Teknologi dan Sistem Komputer*, Vol. 6, No. 1, Pp. 37-43, 2018. <https://doi.org/10.14710/jtsiskom.6.1.2018.37-43>
- [6] D. Debnath, S. Deb dan N. Kar, "An Advanced Image Encryption Standard Providing Dual Security: Encryption Using Hill Cipher & RGB Image Steganography," in *International Conference on Computational Intelligence and Networks*, Bhubaneswar, 2015. <https://doi.org/10.1109/CINE.2015.41>
- [7] M. Jain dan S. K. Lenka, "Secret Data Transmission Using Vital Image Steganography Over Transposition Cipher," in *International Conference on Green Computing and Internet of Things (ICGCIoT)*, Noida, 2015. <https://doi.org/10.1109/ICGCIoT.2015.7380614>
- [8] G. Ardiansyah, C. A. Sari, E. H. Rachmawanto dan D. Setiadi, "Hybrid Method Using 3-DES, DWT and LSB for Secure Image Steganography Algorithm," in *2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, Yogyakarta, 2017. <https://doi.org/10.1109/ICITISEE.2017.8285505>
- [9] X. Zhang, Z. Zhou dan Y. Niu, "An Image Encryption Method Based on the Feistel Network and Dynamic DNA Encoding," *IEEE Photonics Journal*, Vol. 10, No. 4, 2018. <https://doi.org/10.1109/JPHOT.2018.2859257>
- [10] G. Swain dan S. K. Lenka, "A Dynamic Approach to Image Steganography Using the Three Least," *Advanced Materials Research*, Vol. 1 Dari 2403-408, Pp. 842-849, 2012. <https://doi.org/10.4028/www.scientific.net/AMR.403-408.842>
- [11] Wamiliana, R. Andrian dan E. F. Jayanti, "Implementasi Kriptografi Dan Steganografi Pada Media Gambar Menggunakan Hill Cipher Dan Least Significant Bit (LSB)," *Jurnal Komputasi*, Vol. 5, No. 1, Pp. 17-23, 2017. <http://dx.doi.org/10.23960%2Fkomputasi.v5i1.1411>
- [12] J. I. Sari, Sulindawaty dan H. T. Sihotang, "Implementasi Penyembunyian Pesan Pada Citra Digital Dengan Menggabungkan Algoritma Hill Cipher Dan Metode Least Significant BIT (LSB)," *Jurnal Mantik Penusa*, Vol. 1, No. 2, Pp. 1-8, 2017.
- [13] D. Nofriansyah, S. Defit, G. W. Nurcahyo, G. Ganefri, R. Ridwan, A. S. Ahmar dan R. Rahim, "A New Image Encryption Technique Combining Hill Cipher Method, Morse Code and Least Significant Bit Algorithm," *Journal of Physics: Conference Series*, Vol. 954, No. 1, 2018. <https://doi.org/10.1088/1742-6596/954/1/012003>
- [14] M. H. Qasem dan M. Qataweh, "Parallel Hill Cipher Encryption Algorithm," *International Journal of Computer Applications*, Vol. 179, No. 19, 2018.
- [15] W. K. Chen, "Image Sharing Method for Gray-Level Images," *Journal of Systems and Software*, Vol. 86, No. 2, Pp. 581-585, 2013. <https://doi.org/10.1016/j.jss.2012.09.040>
- [16] B. Karthikeyan, J. Chakravarthy dan V. Vaithyanathan, "An Enhanced Hill Cipher Approach for Image Encryption in Steganography," in *International Journal of Electronic Security and Digital Forensic*, Vol. 5, No. 3-4, 2014. <https://doi.org/10.1504/IJESDF.2013.058652>
- [17] K. R. Ilaga, C. A. Sari dan E. H. Rachmawanto, "A High Result for Image Security Using Crypto-Stegano Based on ECB Mode and LSB Encryption," *Journal of Applied Intelligent System*, Vol. 3, No. 1, Pp. 28-38, 2018. <https://doi.org/10.33633/jais.v3i1.1694>
- [18] E. J. Kusuma, C. A. Sari, E. H. Rachmawanto dan S. D.R.I.M., "A Combination of Inverted LSB, RSA, and Arnold Transformation to Get Secure and Imperceptible Image Steganography," *Journal of ICT Research and Applications*, Vol. 12, No. 2, Pp. 103-122, 2018. <http://dx.doi.org/10.5614%2Fitbj.ict.res.appl.2018.12.2.1>
- [19] S. M. M. Karim, M. S. Rahman dan M. I. Hossain, "A New Approach for LSB Based Image Steganography Using Secret Key," in *14th International Conference on Computer and Information Technology (ICCIIT 2011)*, Dhaka, Bangladesh, 2011. <https://doi.org/10.1109/ICCIITechn.2011.6164800>
- [20] E. Walia, P. Jain dan Navdeep, "An Analysis of LSB & DCT Based Steganography," *Global Journal of Computer Science and Technology*, 2010.