

Implementation of a Logistic Map to Calculate the Bits Required for Digital Image Steganography Using the Least Significant Bit (LSB) Method

¹Wikky Fawwaz Al Maki, ²Indra Bayu Muktyas, ³Samsul Arifin, ⁴Suwarno and ⁵Mohd Khairul Bazli Mohd Aziz

¹Department of Informatics, Telkom University, Bandung, 40257, Indonesia

²Mathematics Education Department, STKIP Surya, Tangerang, 15115, Indonesia

³Department of Statistics, School of Computer Science, Bina Nusantara University, Jakarta, 11480, Indonesia

⁴Primary Teacher Education Department, Faculty of Humanities, Bina Nusantara University, Jakarta, Indonesia

⁵Faculty of Industrial Sciences and Technology, University Malaysia Pahang, 26300 Gambang, Pahang, Malaysia

Article history

Received: 10-03-2023

Revised: 16-04-2023

Accepted: 28-04-2023

Corresponding Author:

Samsul Arifin

Department of Statistics,
School of Computer Science,
Bina Nusantara University,
Jakarta, 11480, Indonesia

Email: samsul.arifin@binus.edu

Abstract: The LSB method in steganography usually only uses the last bit or the last few bits that are the same for all pixels. This is very easy to solve by using a bitwise shift left operation so that the last bit becomes the leading bit (MSB). Some techniques combine steganography and cryptography through two different processes. In this study, a new technique is proposed to perform steganography and cryptography together. The random sequence obtained from the logistic map is used to determine the number of bits in the LSB method. Furthermore, testing was carried out on several grayscale images. The result obtained is that the hidden images cannot be opened easily. The level of sensitivity is very small, reaching 10^{-15} .

Keywords: Steganography, Digital Images, Logistic Map, LSB

Introduction

In today's digital era, data security is becoming increasingly important. This is especially related to digital communication which often occurs in the exchange of information and data. One technique to maintain the confidentiality and security of data is steganography. Steganography is a technique of hiding secret messages on other media, such as digital images. Data security has become more crucial in recent years due to technology's quick development, particularly in the internet, communication, and data processing (Arifin and Muktyas, 2021). Two ways can be done to secure data, namely by cryptography and steganography. Cryptography changes the shape of the data, while steganography hides the data invisible so that it doesn't look different from the original data. The data referred to in this study is digital image data (Sari *et al.*, 2020; Agustini and Kurniawan, 2019).

The algorithm employed in digital picture steganography, particularly in the spatial realm, modifies the cover image's pixel intensity value. One of these is the Least Significant Bit (LSB), which is well-known. The secret data is concealed in this manner using the last bit value. In this study, there are investigations using the LSB approach (Delmi *et al.*, 2020). But this method is not difficult to solve. One of the ways to open the secret data is by using the left operation. There are several solutions offered by several previous researchers, such

as (Agustini and Kurniawan, 2019; Delmi *et al.*, 2020) namely by randomizing the pixels of the cover image and placing the secret data. This is done using the chaos function. Chaos theory has been studied for a long time, namely in the 1970s. Chaos is deterministic, dynamic, nonlinear, and sensitive to initial values. This means that even if there is a small change in the initial value, it will have a big effect on the output. Chaos also has a random nature, so it is used to generate random numbers. From the random behavior of this chaotic sequence, it is then used to secure steganography (Schuster and Just, 2006; Kocarev and Lian, 2011).

We will first examine the chaos and LSB functions in this study. The chaos function is used to generate chaotic sequences and is used to hide data. The nature of chaos is random and unpredictable behavior because it is sensitive to changes in initial values. This means that even a slight difference from the initial value will result in a very large change in the output (Kocarev and Lian, 2011; Alligood *et al.*, 1998). Several chaotic functions can produce chaotic sequences, including logistic maps, Bernoulli maps, tent maps, and circle maps. recursively, the logistic map is written in Eq. (1):

$$x_{n+1} = \lambda x_n (1 - x_n) \quad (1)$$

where, x_n is the value of the variable at time n , x_{n+1} is the value of the variable at time $n + 1$ and λ is the parameter that influences the behavior of the system. The logistic

map will map from (0,1) to (0,1). The sequence x_i will be random at time $\lambda \in (3,6,4)$. The resulting random sequence is the real numbers in the interval (0,1). In use in cryptography or steganography, then this random sequence is changed with a real to integer function to match what is desired (Arifin *et al.*, 2022).

Usually, in steganography, the chaos function is combined with the LSB method. The random sequence obtained from the chaos function is used to randomize the secret data. Then the random data is inserted using the LSB method, which is placing the last bit of each pixel in the cover image. Sari has conducted research using insertion in the last 1 bit and 2 bits of the cover image (Sari and Siahaan, 2018). The result for the last 1 bit is good for small information, while the last 2 bits are good for larger information. Furthermore, based on (Thakur and Saravanan, 2016), the LSB method is still optimal for the last 4 bits of digital color images in terms of the comparison between the Root Mean Squared Error (RMSE) and Peak Signal Noise Ratio (PSNR) values. It will also still be good on grayscale digital images. From this came the idea to use the random sequence generated by the chaos function to determine the last 1-4 bits of insertion in the cover image (Ariyus, 2019).

Furthermore, will be discussed regarding the concept of Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). To test how similar two digital images are, *MSE* and *PSNR* are used (Kocarev and Lian, 2011). This is to find out how big the difference is between the secret image and the open image, as well as the cover image and stego image (inserted image). *MSE* and *PSNR* are metrics used to measure the quality of images or videos that have been compressed or given noise. *MSE* is the average of the squared difference between the original pixel values and the decompressed or noise pixel values. The smaller the *MSE* value, the better the quality of the compressed or noisy image or video. The formal definition of *MSE* is as follows:

$$MSE = \frac{1}{mn} \sum_{j=1}^n \sum_{i=1}^m [I(i, j) - I'(i, j)]^2 \quad (2)$$

where, $I(i, j)$ is the original pixel value, $I'(i, j)$ is the decompressed or noise pixel value, and m, n is the number of pixels in the image or video. *PSNR* is the ratio between the maximum value of a pixel and the *MSE*, measured in decibels (dB). The higher the *PSNR* value, the better the quality of the compressed or noisy image or video. The formal definition of *PSNR* is as follows:

$$PSNR = 10 \log \left(\frac{255^2}{MSE} \right) \quad (3)$$

where, *MSE* is the pre-computed *MSE* value.

Steganography is a technique of hiding secret messages or information in a cover image, video, or audio file. Many researchers have proposed various methods for steganography and some of them are reviewed below

(Shankar and Azhakath, 2023; Yang and Liao, 2023). In a study, steganography is a technique to hide data and information on other media, such as digital images, and has become an important method for secure communication. This study used the Discrete Wavelet Transform (DWT) and Singular Value Decomposition (SVD) algorithms, both of which use domain transforms to process digital images and have high speeds in embedding secret messages into an image. However, one of the problems in steganography is that the quality of the stego image produced may degrade. The DWT method was implemented in Python 2 and showed better image quality than the SVD method, with an average *MSE* value of 0.0046 dB and an average *PSNR* value of 63.47 dB (Darwis and Pasaribu, 2020; Arifin and Garminia, 2019). On the other hand, Hafiz (2019) said that data confidentiality is a necessity principal in computer systems, networks computers, and the internet because of the threat of piracy or data theft. Steganography can be a solution to maintain confidentiality and data security by hiding data in digital images. One method that can be used is the least significant bit, where the data bits will be embedded in the digital image bits. Steganography and least significant bit methods allow data to be hidden and then retrieved to be read by the data owner (Yang and Liao, 2023).

In steganography, the most used method is the Least Significant Bit (LSB) because it is easy to implement, and the results are quite good. LSB works by inserting message bits into the digital image bits that will be hidden. However, one problem that occurs is the number of bits that can be inserted into a digital image without destroying the image quality (Novianto and Setiawan, 2018). To overcome this problem, this research tries implementing a logistic map in determining the number of bits that can be inserted in the LSB. A logistic map is a mathematical model that can produce marks random. In this study, the value of the random number is used to determine how many bits can be inserted into the digital image without destroying its quality image. This research uses language Python programming to implement a logistic map on LSB (Muktyas *et al.*, 2021).

This article introduces a novel approach by incorporating the logistic map in determining the number of bits to be used for steganography (Kanwal *et al.*, 2021). This approach improves the security of data by enhancing the quality of the stego image and minimizing the chances of detection by unauthorized parties. This article has the potential to impact the field of steganography by introducing a new and effective technique for securing digital information. This research aims to improve the quality of steganography images and enlarge the capacity storage of secret messages on digital images (Hafiz, 2019). Thus, it is hoped that this research can contribute to the development technique of steganography to keep confidentiality and data security on digital images. This study is structured with the following framework. Moreover, some introductions and previous research related to this study are presented. Second, information about the chaos and LSB functions used in steganography. Second, explain

the proposed method. Third, discussion and experimentation on several grayscale digital images. Further, we talk about the analysis of results in this research and close with conclusions and suggestions for the future.

Materials and Methods

As indicated by a survey of the literature, the research style used here is one of exploration and application of previously obtained results. This section looks at the theories that were used in this inquiry. The methodology used in this study includes the following stages. (a) Data collection: The data used in this study are digital images that have been prepared beforehand. (b) Steganography process: The steganography process is carried out using the Least Significant Bit (LSB) method which will insert secret message data into a digital image. This process is done by changing the LSB bit (the least significant bit) in each image pixel so that the difference is not visible. (c) Logistic map implementation: Logistic maps are used to generate a series of random numbers which will determine the number of message bits to be inserted in each digital image pixel. This implementation is done by calculating the iteration value of the logistic map equation at each iteration. (d) Evaluation of results: The results of the steganography process will be evaluated using the Mean Square Error (MSE) and Peak Signal Noise Ratio (PSNR) methods to measure the quality of images that have been modified with the LSB method using a logistic map. (f) Analysis of results: Analysis of results will be carried out by comparing the results of experiments using the LSB and logistic map methods with experiments using the LSB method without using the logistic map. The results obtained will be compared and analyzed to determine differences in image quality and the number of message bits embedded in digital images (Rustad *et al.*, 2022; Hussein and Amintoosi, 2023).

First, the secret image is converted into a gray degree value from 0-255. These numbers are then converted into binary and combined into one long binary series (Kanwal *et al.*, 2021). On the other hand, generate an integer sequence with a length 8 times the number of pixels in the secret image. The chaos function will generate a sequence of random numbers from 0-1. The sequence of real numbers is then converted into an integer sequence with the function real to an integer. The real-to-integer function is a mathematical function that takes real numbers as input and returns integers as output. In simpler language, this function maps real numbers to integers. In mathematical notation, the real to integer function will be defined as follows as $f(x)$ in Eq. (4):

$$f(x) = 1 + 10^3 x \text{ mod } 3 \quad (4)$$

where, x is the input real number and $f(x)$ is the output integer. An example of a real-to-integer function is the rounding function, in which real numbers are converted to the nearest integer. For example, if $f(3.7) = 4$, then the number 3.7 is rounded to 4. Other functions that fall into the category of real to integer functions are the floor function, which returns the largest integer that is smaller than the input real number, and the function ceiling (ceiling function), which produces the smallest integer that is greater than the input real number.

Next, cut the long binary series based on the integer sequences that have been obtained. Then paste it on the cover image. The process of inserting a cover image is by using the right shift (\gg), left shift (\ll), and or (\mid) operations. To zero out the last n bits of pixel x , we perform the operation $x \gg n \ll n$. Meanwhile, to write a new bit (y) to x , the operation is performed $x \mid y$. The insertion process scheme Fig. 1 (Handayani *et al.*, 2019).

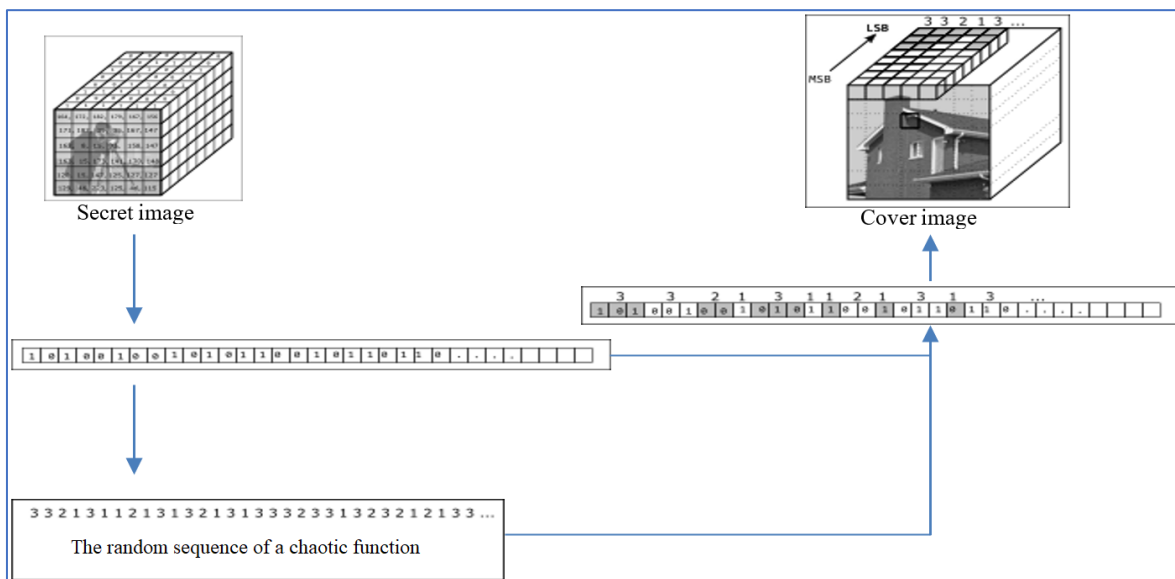


Fig. 1: The proposed image insertion scheme

Pseudocode algorithms for generating random sequences, insertion algorithms, and opening algorithms can be seen in Algorithms 1-3 respectively. Algorithm 1 contains techniques for generation of random integer sequences (Essaid *et al.*, 2019).

Algorithm 1: Random integer sequence generation.

1. **Inputs:** x_0, n, lambda s
2. $\text{bars} = [0]*n$
3. **for** i in $(1..100)$ **do**
4. $x_0 = \text{lambda} * x_0 * (1 - x_0)$
5. **done**
6. **for** i in $(1..n)$ **do**
7. $x_0 = \text{lambda} * x_0 * (1 - x_0)$
8. $\text{bar}[i] = \text{int}(1 + 1000 * x_0 \% 3)$
9. **done**
10. **Outputs:** Bars.

Next is Algorithm 2, which is about methods of the insertion filed as follows (Sari and Siahaan, 2018).

Algorithm 2: The proposed insertion algorithm is as follows.

1. **Input:** Secret Image, Cover Image, x_0
2. Convert cover and secret images to 1-dimensional matrices (mc and mr)
3. Save the size original secret image into row_rhs and column_rhs
4. Change every number decimal of each entry in the secret matrix into 1-byte binary form (mr_bin)
5. each byte together into a shape long binary string (mr_string)
6. Awaken A line random with a chaos function (input: x_0, n) which contains the numbers 1 through 3 that are equal to 8 times as many in-length entries in the secret matrix (take) with algorithm 1.
7. $\text{nbit_lsb} =$ amount of the required pixels of mc to be able to insert the whole mr_string,

8. $\text{nbit_lsb} = i$ if cumulative sum from take $[0:i]$ over long mr_string,
9. otherwise $\text{nbit_lsb} = \text{long_mr_string}$
10. **Do** this until all the entries in the long binary string (mr_string) are inserted:
11. $\text{nbit} = \text{take}[i]$
12. cut mr_string throughout nbit (bit_rhs)
13. Insert bit_rhs to each entry on mc with the method replace the last nbit of each number in mc with bit_rhs
14. **Output:** $x_0, \text{row_rhs}$ and $\text{column_rhs}, \text{nbit_lsb}$

Finally, Algorithm 3 is about the opening method proposed in this article as follows (Lone *et al.*, 2022).

Algorithm 2: The proposed opening algorithm is as follows.

1. **Input:** Stego Image, x_0, b, k, nbit
2. Generate a random sequence with algorithm 1.
3. $\text{bar}(x_0, \text{nbits} + 1)$
4. Take the last n bits of each pixel in the Stego Image according to a random sequence
5. Concatenate to one long bitstring
6. Cut it to 8-8 bits of the long string
7. Convert to decimal
8. **Output:** Open image.

Furthermore, the following Figs. 2-3 respectively are about the scheme opening of the submitted image and an example of the insertion process (Jatmoko *et al.*, 2018).

One of the properties of the chaotic function is ergodicity, meaning that the resulting points will spread evenly throughout all spaces (Kocarev and Lian, 2011). From here, if the scatter of the chaotic sequences is from 1-3, then the average of the scatter is 2. So that out of the 8 bits in the cover image, only 2 bits are occupied by the bits of the secret image on average. As a result, the size of the cover image must be 4 times the size of the secret image.

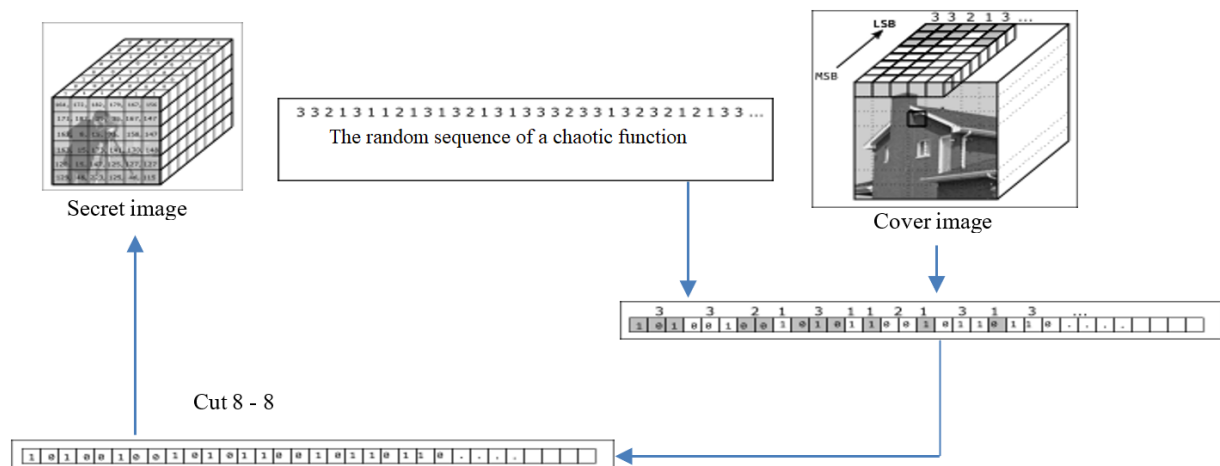


Fig. 2: Schematic opening of the proposed drawing

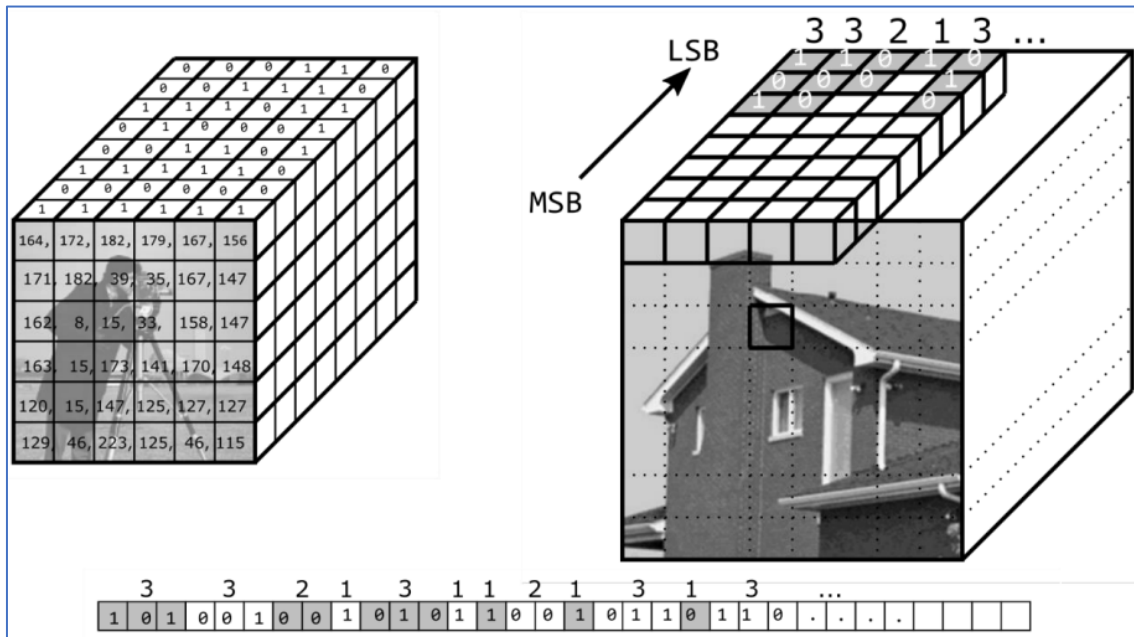


Fig. 3: Example of the insertion process

Results and Analysis

The proposed method is then applied to several standard grayscale images in png format (Gonzalez *et al.*, 2020). Details can be seen in Table 1. The insertion process is performed with the keys: $\lambda = 3.9$ and $x_0 = 0.7189$. At the end of the secret image insert process, important information is available to open it, namely value x_0 , row, column, and $nbit_lsb$:

$$x_0: 0.7189; row, column : 100, 100; nbit_lsb : 39277$$

In the process of opening the image, enter the x_0 corresponding, row, column, and $nbit_lsb$ values. The *MSE* value obtained from all data is zero. This means that the image that opens with the original image is the same. So, the authenticity of the secret image is completely preserved (Yahia and Abushaala, 2022). This can be seen in more detail in Table 2. When tested using the following key:

$$x_0: 0.71891; row, column : 100, 100; nbit_lsb : 39277$$

which only adds 1 to its initial value (x_0), apparently it can't open the secret image. This is due to the sensitivity to the initial value of the chaotic function. So, it is difficult to solve by brute force method. We'll talk about the PSNR value next. After insertion, the cover picture and stego image have a PSNR value of about 50. It can be seen from the naked eye that there is no difference between the cover image and the stego image (Delmi *et al.*, 2020; Suryadi *et al.*, 2021). Details can be seen in Table 3.

Table 1: Standard image as the dataset used




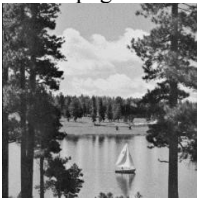



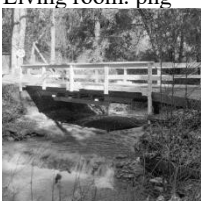
No.	Secret image (100 × 100 px)	Cover image (512 × 512 px)
1	 Cameraman. png	 House. png
2	 Jet plane. Png	 Lake. png
3	 Peppers. png	 Living room. png
	 Pirate. Png	 Walk bridge. png

Table 2: MSE value of secret image and open image











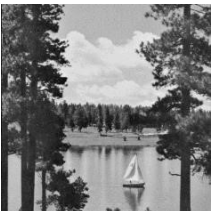
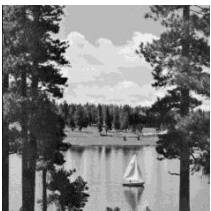


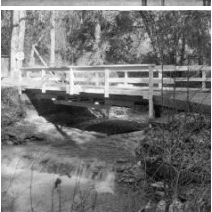
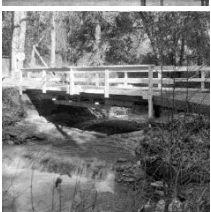
No.	Secret image (100 × 100 px)	Open image (100 × 100 px)	MSE
1			0
2			0
3			0
4			0

Table 3: PSNR value between the cover image and stego image

No.	Cover image (512 × 512 px)	Stego image (512 × 512 px)	PSNR
1			50.82
2			49.78
3			49.80
4			49.87

Discussion

In this research, we have successfully implemented a logistic map to calculate the number of bits needed to hide secret messages using the LSB method in digital image steganography. The test results show that the higher the key ratio value, the fewer bits needed to hide the secret message in the image. In addition, we also found that the higher the key ratio value, the lower the image quality of the steganography results. Based on the test results, we can conclude that the implementation of a logistic map on digital image steganography using the LSB method can provide quite good results in terms of efficient use of bits to hide secret messages.

However, this must be balanced with the quality of the resulting image so that the secret message can be hidden well enough without significantly reducing image quality. In addition, we also compared the results between the LSB method without using the logistic map and the LSB method with the logistic map on the same test. The results show that the use of logistic maps can reduce the number of bits needed to hide secret messages in images, so that the efficiency of using bits can increase. Even so, there are several factors that influence the success of embedding a secret message in an image, such as the size of the image, the type of image file, and the type of secret message you want to insert. Therefore, it is necessary to carry out further research to optimize the use of logistic maps in digital image steganography using the LSB method, as well as evaluate their effectiveness on various types of images and types of secret messages.

Conclusion

In this study, a method has been proposed to carry out the steganography process by utilizing the chaos function and the LSB method together. The chaos function determines how many bits are inserted into the cover image. The authenticity of the secret image with the proposed method is highly maintained because there is no difference between the secret image and the opened image. The cover image and the stego image are also invisible to the naked eye so that suspicion from other parties will be reduced. The security of this method is also good, judging from the initial values that differ only slightly, you cannot open secret images properly.

In this research, implementation of the logistic map has been carried out in determining the number of bits to be inserted in the Least Significant Bit (LSB) method for digital image steganography. Based on the test results, it is known that the method used can hide secret messages well in digital images. Tests also show that the use of the logistic map algorithm in determining the number of bits to be inserted in the LSB method can improve image

quality in stego images. The test results show that the MSE value of the resulting stego image is lower and the PSNR value is higher than the stego image produced without using the logistic map algorithm. In addition, the implementation of the logistic map algorithm in the LSB method can increase the level of data security hidden in digital images. This is because the use of a logistic map in determining the number of bits to be inserted arranges the hidden data bits more randomly, making it difficult for unauthorized persons to guess.

In future research, it is possible to develop this method using a larger digital image and carry out further analysis of the proposed method. In addition, this research can also be carried out by combining several other steganographic methods to strengthen the level of hidden data security. In conclusion, the implementation of the logistic map in determining the number of bits in the LSB method for digital image steganography has shown positive results in increasing the level of security and image quality in the resulting stego image. This method is expected to make a significant contribution to the development of data security and privacy technologies in the current digital era.

Acknowledgment

The authors would like to thank the reviewers for their informative comments, suggestions, and ideas, which have helped mold this manuscript into something that is worthy of publication.

Funding Information

This study is supported and funded by Telkom University campus and the Bina Nusantara University research and technology transfer office under the terms of the university's international research grant (PIB 2023) under contract number 029/VRRTT/III/2023.

Author's Contributions

Wikky Fawwaz Al Maki and Mohd Khairul Bazli Mohd Aziz: Simulated the data, tidying up the theoretical basis and the methods we use.

Indra Bayu Muktyas: Coding the Python program, written and finalized the manuscript.

Samsul Arifin: Coding the program, written, and finalized the manuscript.

Suwarno: Written and finalized the manuscript.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that there is no conflict of interest in this study and no ethical issues involved.

References

- Agustini, S., & Kurniawan, M. (2019). Peningkatan Keamanan Teks Menggunakan Kriptografi Dan Steganografi. *Scan: Jurnal Teknologi Informasi Dan Komunikasi*, 14(3), 33-38.
<http://www.ejournal.upnjatim.ac.id/index.php/scan/article/view/1685>
- Alligood, K. T., Sauer, T. D., Yorke, J. A., & Chillingworth, D. (1998). Chaos: An introduction to dynamical systems. *SIAM Review*, 40(3), 732-732.
- Arifin, S., & Garminia, H. (2019). Uniserial dimension of module $zm \times zn$ over Z using python. *International Journal of Scientific & Technology Research* 8(7), 194-199.
https://www.researchgate.net/publication/334769797_Uniserial_Dimension_Of_Module_ZmxZn_Over_Z_Using_Python
- Arifin, S., & Muktyas, I. B. (2021, April). Generate a system of linear equation through unimodular matrix using Python and Latex. In *AIP Conference Proceedings* (Vol. 2331, No. 1, p. 020005). AIP Publishing LLC. <https://doi.org/10.1063/5.0041651>
- Arifin, S., Muktyas, I. B., & Mandei, J. M. (2022, May). Graph coloring program for variation of exam scheduling modeling at Binus University based on Welsh and Powell algorithm. In *Journal of Physics: Conference Series* (Vol. 2279, No. 1, p. 012005). IOP Publishing.
<https://doi.org/10.1088/1742-6596/2279/1/012005>
- Ariyus, D. (2019, May). Optimization substitution cipher and hidden plaintext in image data using LSB method. In *Journal of Physics: Conference Series* (Vol. 1201, No. 1, p. 012033). IOP Publishing.
<https://iopscience.iop.org/article/10.1088/1742-6596/1201/1/012033/meta>
- Darwis, D., & Pasaribu, A. F. O. (2020). Komparasi Metode Dwt Dan Svd Untuk Mengukur Kualitas Citra Steganografi. *Network Engineering Research Operation*, 5(2), 100-108.
<http://dx.doi.org/10.21107/nero.v5i2.175>
- Delmi, A., Suryadi, S., & Satria, Y. (2020). Digital image steganography by using edge adaptive based chaos cryptography. In *Journal of Physics: Conference Series* (Vol. 1442, No. 1, p. 012041). IOP Publishing.
<https://doi.org/10.1088/1742-6596/1442/1/012041>
- Essaid, M., Akharraz, I., & Saaidi, A. (2019). Image encryption scheme based on a new secure variant of Hill cipher and 1D chaotic maps. *Journal of Information Security and Applications*, 47, 173-187.
<https://doi.org/10.1016/j.jisa.2019.05.006>
- Gonzalez, R. C., Woods, R. E., & Eddins, S. (2020). Image databases. *Digital Image Processing Using MATLAB, 3rd Ed.* ISBN-10: 9780982085417.
<http://www.imageprocessingplace.com>

- Hafiz, A. (2019). Steganografi Berbasis Citra Digital Untuk Menyembunyikan Data Menggunakan Metode Least Significant Bit (LSB). *Jurnal Cendikia*, 17(1 April), 194-198.
<https://core.ac.uk/download/pdf/267922086.pdf>
- Handayani, T., Arifin, S., & Surgandini, A. (2019). Penerapan Model Pembelajaran Penemuan Terbimbing Untuk Meningkatkan Kemampuan Pemahaman Konsep Matematis Siswa SMA. *Wacana Akademika: Majalah Ilmiah Kependidikan*, 3(2), 151-164. <https://doi.org/10.30738/wa.v3i2.4407>
- Hussein, M. K., & Amintoosi, H. (2023). Protection of images by combination of vernam stream cipher, AES and LSB steganography in a video clip. *Bulletin of Electrical Engineering and Informatics*, 12(3), 1578-1585. <https://doi.org/10.11591/eei.v12i3.4039>
- Jatmoko, C., Handoko, L. B., & Sari, C. A. (2018). Uji Performa Penyisipan Pesan Dengan Metode LSB dan MSB Pada Citra Digital Untuk Keamanan Komunikasi. *Dinamika Rekayasa*, 14(1), 47-56.
- Kanwal, S., Inam, S., Cheikhrouhou, O., Mahnoor, K., Zaguia, A., & Hamam, H. (2021). Analytic study of a novel color image encryption method based on the chaos system and color codes. *Complexity*, 2021, 1-19. <https://doi.org/10.1155/2021/5499538>
- Kocarev, L., & Lian, S. (Eds.). (2011). *Chaos-based cryptography: Theory, algorithms and applications* (Vol. 354). Springer Science & Business Media. ISBN-10: 9783642205415.
- Lone, P. N., Singh, D., Stoffová, V., Mishra, D. C., Mir, U. H., & Kumar, N. (2022). Cryptanalysis and Improved Image Encryption Scheme Using Elliptic Curve and Affine Hill Cipher. *Mathematics*, 10(20), 3878. <https://doi.org/10.3390/math10203878>
- Muktyas, I. B., Sulistiawati, & Arifin, S. (2021, April). Digital image encryption algorithm through unimodular matrix and logistic map using Python. In *AIP Conference Proceedings* (Vol. 2331, No. 1, p. 020006). AIP Publishing LLC.
<https://doi.org/10.1063/5.0041653>
- Novianto, D., & Setiawan, Y. (2018). Aplikasi Pengamanan Informasi Menggunakan Metode Least Significant Bit (LSb) dan Algoritma Kriptografi Advanced Encryption Standard (AES). *Jurnal Informatika Global*, 9(2).
<https://doi.org/10.36982/jiig.v9i2.561>
- Rustad, S., Andono, P. N., & Shidik, G. F. (2022). Digital Image Steganography Survey and Investigation (Goal, Assessment, Method, Development, and Dataset). *Signal Processing*, 108908.
<https://doi.org/10.1016/j.sigpro.2022.108908>
- Sari, I. Y., Jamaluddin, J., Simarmata, J., Rahman, M. A., Iskandar, A., Pakpahan, A. F., ... & Manullang, S. O. (2020). Keamanan data dan informasi. Yayasan Kita Menulis. ISBN-10: 978-623-6761-80-9.
<http://repo.handayani.ac.id/147/>
- Sari, R. D., & Siahaan, A. P. U. (2018). Least Significant Bit Comparison between 1-bit and 2-bit Insertion. *Int. J. Innov. Res. Multidiscip. F*, 4(10), 110-113.
- Schuster, H. G., & Just, W. (2006). *Deterministic chaos: An introduction*. John Wiley & Sons. ISBN-10: 9783527606412.
- Shankar, D. D., & Azhakath, A. S. (2023). Random embedded calibrated statistical blind steganalysis using cross validated support vector machine and support vector machine with particle swarm optimization. *Scientific Reports*, 13(1), 2359.
<https://doi.org/10.1038/s41598-023-29453-8>
- Suryadi, M. T., Satria, Y., & Hadidulqawi, A. (2021, March). Implementation of the Gauss-Circle Map for encrypting and embedding simultaneously on digital image and digital text. In *Journal of Physics: Conference Series* (Vol. 1821, No. 1, p. 012037). IOP Publishing.
<https://doi.org/10.1088/1742-6596/1821/1/012037>
- Thakur, R. K., & Saravanan, C. (2016, March). Analysis of steganography with various bits of LSB for color images. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* (pp. 2154-2158). IEEE.
<https://ieeexplore.ieee.org/abstract/document/7755073>
- Yahia, F. F., & Abushaala, A. M. (2022, May). Cryptography using Affine Hill Cipher Combining with Hybrid Edge Detection (Canny-LoG) and LSB for Data Hiding. In *2022 IEEE 2nd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA)* (pp. 379-384). IEEE.
<https://ieeexplore.ieee.org/abstract/document/9837714>
- Yang, J., & Liao, X. (2023). ACGIS: Adversarial Cover Generator for Image Steganography with Noise Residuals Features-Preserving. *Signal Processing: Image Communication*, 113, 116927.
<https://doi.org/10.1016/j.image.2023.116927>