Implementation of Steganography Modified Least Significant Bit using the Columnar Transposition Cipher and Caesar Cipher Algorithm in Image Insertion

Handrizal1*, Jos Timanta Tarigan1, Doni Irwansyah Putra1

1 Department of Computer Science, Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Jl. University No. 9-A, Medan 20155, Indonesia

*handrizal@usu.ac.id

Abstract. Steganography technique is a technique to hide data or information into other media such as digital images, text, sound, or video. One of the simplest methods of Steganography in the concept of Steganography is the Least Significant Bit. The Least Significant Bit (LSB) method hides the message by inserting the message at the lower or rightmost bits in the cover work file as a medium to hide the message. Caesar Cipher is one method that replaces each letter in the plaintext with letters that are at odds with certain numbers in the alphabet. Columnar transposition Cipher is one method in which messages are written in rows of a specified length, then are read column by column in the order of reading based on a keyword. The message is inserted into the cover image with the target pixel determination to be inserted based on the results of the Caesar Cipher process of the Columnar Transposition Cipher index to the random value of the Multiply With Carry Generator algorithm. The system implementation uses the C# programming language. The results of this study indicate that the implementation of a combination of Caesar Cipher, Columnar Transposition Cipher, and Multiply With Carry Generator can maintain the confidentiality, integrity, and security of data.

1. Introduction

Computer technology that is developing rapidly in the digital era now makes it easy for us to communicate and exchange data or information. The impact of these developments makes the dissemination of data or information very prone to abuse and manipulation by unauthorized parties. One of the data or information is carried out in the form of image files. Image (image) can be represented in a flat area that has two sizes (width and height). In the world of image, computation consists of pixels where the pixel value shows the color of the image (image). One technique to protect it with the Steganography Technique [1].

Steganography technique is a technique of hiding data or information into other media such as digital images, text, sound, or video. One of the simplest Steganography methods in the concept of Steganography is the Least Significant Bit [2].
The Least Significant Bit (LSB) method is a message hiding technique by inserting messages in the low or rightmost bit of the cover work file as a medium for hiding messages [3].

Caesar Cipher is a cryptographic method that replaces each letter in plain text (plaintext) with a letter that is different from a certain number in the alphabet [4]. The advantage of the Caesar Cipher is one of the most well-known encryption techniques and easy to learn.

Columnar transposition Cipher is a cryptographic method in which messages are written in a series of a specified length, then read back column by column in the order of reading based on a keyword. The length of the string is determined by the length of the keyword [4]. The advantage of the Columnar transposition Cipher is has a better mixing of letters than some ciphers and the main benefit that the transposition cipher has over substitution cipher is that transposition cipher can be applied more than once.

The Steganography Modified Least Significant Bit using the Columnar Transposition Cipher and Caesar Cipher Algorithm has been widely used to solve various problems such as, insert text to an image [5], png image security [6], secure the image file [7], image hiding [8], hiding and data safety [9]. There are also study in using another steganography method, such as Max-Plus Algebra, to implement an image based steganography as proposed in [10].

Therefore, with the above background, the writer wants to research with the title "Implementation of Steganography Modified Least Significant Bit using Columnar Transposition Cipher Cryptography Algorithm and Caesar Cipher in Image insertion". These algorithms need to be combined to cover the disadvantages of each algorithm, so it is hoped that file security can be improved.

2. Method
In this paper, we perform several stage as follow:

2.1. Least Significant Bit (LSB)
The least significant bit is often used for inserting data into digital media, where the message data bits that are inserted will be replaced with the lowest bit [3]. The lowest bit position is in the last bit, can be seen in figure 1:

2.2. Modified Least Significant Bit (MLSB)
The Modified Least Significant Bit method is a development of the Least Significant Bit method. This method is used to produce a method that is better than the existing LSB method. In this study, the author will use the MLSB Pseudo Random Number Generator with the Multiply with Carry Algorithm, because Multiply with Carry Algorithm has better performance and, random numbers are also produced at a higher speed. [3].

2.3. Caesar Cipher Algorithm
This algorithm uses a substitution technique where each letter on the plaintext is replaced by another letter with a certain position in the alphabet. This algorithm was invented and used by Julio Caesar and his soldiers, to encode messages to his governors [3].

2.4. Columnar Transposition Cipher Algorithm
The Columnar Transposition Cipher is one of the classic algorithms using the transposition technique. The way the Columnar Transposition Cipher works is to write plaintext characters with line orientation with the same character length and then the ciphertext is obtained by rewriting with column orientation [4].

2.5. Mean Squared Error (MSE)
MSE is the average error squared value to compare and measure the accuracy between the image before insertion and the image after insertion. The lower the MSE value the better, the MSE value can be calculated by the following equation:

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

2.6. Peak Signal to Noise Ratio (PSNR)
PSNR is a parameter used for a comparison between the maximum value of the measured signal and the amount of noise that affects the signal. The greater the PSNR value, the better the compression quality. The PSNR value can be calculated by the following equation:

\[
PSNR = 20 \log_{10} \frac{b}{\sqrt{MSE}}
\]

3. Results and Discussion
At this stage, the system will test an image that has a length and width of 20x20 pixels and a cover image measuring 1000x1000 pixels. The display message image and cover image to be tested can be seen in figure 2 and figure 3.

![Figure 2. Image message of 20x20 pixels](image_url)
3.1. Embedding Process Testing
In the testing phase of the embedding process, the user first enters the message image file and the cover image where the message image size must be smaller than the cover image. Then press the insert button to carry out the process of inserting the message image into the cover image where before inserting the Multiply with Carry Generators and Columnar Transposition Cipher keys will be randomized first, which is then used to process the message image pixel randomization. After getting the stego image results, press the save button to save the stego image results and the keys obtained. The display of the embedding process results can be seen in Figure 4.

3.2. Extracting Process Testing
At the extracting stage, the first thing to do is to input the stored stego image file and input the previously stored key. Then press the extract button, and the system will perform the process of extracting the stego image to retrieve the message image that was inserted in the previous embedding process. After the message image is obtained, press the save button to save the message image. The extracting process display on the stego image can be seen in Figure 5.
3.3. **MSE-PSNR Process Testing**

In the MSE-PSNR process, a comparison of the similarity of the original message image to the message image that has been processed and the cover image and the stego image will be carried out. The display of MSE-PSNR results on the message image with the original message image and the cover image with the stego image can be seen in Figure 6 and Figure 7.
3.4. Testing Running Time System and MSE-PSNR
In this system, a processing time testing process will be carried out with a pixel size comparison. This
test will be carried out in 3 parts, namely the time of the embed, extract, and MSE-PSNR process. This
test uses 4 sample image messages with a size of 10x10,20x20,30x30,40x40 and 3 sample cover images
with a size of 1000x1000,2000x2000. The results of the system running time test and the MSE-PSNR
value can be seen in Table 1.

Table 1. Results of Running Time and MSE-PSNR Value

| Message | Cover | Insert Running Time (ms) | Extract Running Time (ms) | MSE | PSNR  |
|---------|-------|--------------------------|---------------------------|-----|-------|
| 10 x 10 | 1000  | 30.9182 33.9084          | 0.00764933333            | 69.29456774377 |
| 20 x 20 | 1000  | 323.1693 301.2319        | 0.00909866667            | 68.54102606100 |
| 40 x 40 | 1000  | 4794.1905 323.1642      | 0.01482000000            | 66.4223157225  |
| 80 x 80 | 1000  | 81956.3130 5118.3233    | 0.03822433333            | 62.30740441203 |
| 10 x 10 | 2000  | 39.9317 56.8417         | 0.00049225000            | 81.20894636058 |
| 20 x 20 | 2000  | 313.1642 317.6084       | 0.00085966667            | 78.78750273450 |
| 40 x 40 | 2000  | 5118.3233 487.7131      | 0.00232766667            | 74.46159573547 |
| 80 x 80 | 2000  | 87168.4805 87497.4085   | 0.00827200000            | 68.95469835118 |
| 10 x 10 | 3000  | 31.9547 92.7521         | 0.00715674074            | 69.58365076070 |
| 20 x 20 | 3000  | 315.0979 359.0742       | 0.00731696296            | 69.48749504143 |
| 40 x 40 | 3000  | 5486.3359 5466.4115     | 0.00796729630            | 69.11769392396 |
| 80 x 80 | 3000  | 85555.1237 87321.2564   | 0.01055685185            | 67.89545933883 |

4. Conclusion
After conducting research through literature studies, designing, implementing, analyzing, and testing
applications on the implementation of Steganography Modified Least Significant Bit using the
Columnar Transposition Cipher, Caesar Cipher, and Multiply With Carry Generator Algorithm on
image insertion, it can be concluded:
1. Based on the tests that have been done. Testing the Implementation of Steganography Modified Least Significant Bit using the Columnar Transposition Cipher Algorithm, Caesar Cipher, and Multiply with Carry Generator on image insertion works well.

2. The factor that affects the time of the embedding process is the size of the message, the larger the message size, the more time it takes to carry out the embedding process. The embedding time for message size 10x10 cover 1000 is 30.9182 ms, while the embedding time for message size 80x80 cover 3000 is 85555.1237 ms.

3. The factor that affects the extracting process time is the size of the message, the bigger the message size, the more time it takes to do the extracting process. The extracting time for message size 10x10 cover 1000 is 33.9084 ms, while the extracting time for message size 80x80 cover 3000 is 87321.2564 ms.

4. Factors that affect the MSE and PSNR values are the size of the message inserted and the size of the cover object. The smaller the message that is inserted and the bigger the cover object, the better the MSE and PSNR values.

5. Acknowledgments
The authors gratefully thank all who have supported the completion of this research and Universitas Sumatera Utara that has allowed me to publish this paper.

References
[1] Champakamala B.S, Padmini. K. Radhika D.K 2014 Least Significant Bit Algorithm for Steganography Int. J. of Advance Computer Technology 3 4
[2] Emam, Marwa M, Abdelmgeid A Aly, and Fatma A Omara. 2016 An Improved Image Steganography Method Based on LSB Technique with Random Pixel Selection.” Int. J. of Advanced Computer Science and Applications 7 3 17-22.
[3] Naveen K Nishchal 2019, Digital techniques of data and image encryption. IOP Publishing Ltd.
[4] D Rachmawati, S M Hardi and R P Pasaribu 2019 Combination of columnar transposition cipher caesar cipher and lempel Ziv welch algorithm in image security and compression. Journal of Physics: Conference Series 1339 012007
[5] R R A Lubis, S M Hardi, M Zarlis, I Jaya and J T Tarigan, 2019, Analysis on Combination of Watermarking Algorithm: Modified Least Significant Bit Algorithm with Least Significant Bit+1, Journal of Physics: Conference Series 1235 012081
[6] S M Hardi, D Rachmawati, F Chairinnisa, I Jaya and J T Tarigan, 2019 Combination of myszkowski transposition algorithm and modified least significant bit (MLSB) green channel on png image security, Journal of Physics: Conference Series 1235 012080.
[7] D Rachmawati, M A Budiman and A Yusuf 2020 Combination of Rail Fence Cipher Algorithm and Least Significant Bit Technique to Secure the Image File J. of Physics: Conference Series 851 012069
[8] D Rachmawati, A Amalia and J Surya, 2017, Combination of Huffman Coding Compression Algorithm and Least Significant Bit Method for Image Hiding. J. of Physics: Conference Series 801 012059
[9] S M Hardi, M Masitha, M A Budiman and I Jaya, 2020, Hiding and Data Safety Techniques in Bmp Image with LSB and RPrime RSA Algorithm J.of Physics: Conference Series 1566 012084
[10] K.A. Santoso, Fatmawati, H. Suprajitno, 2018, On Max-Plus Algebra and its Application on Image Steganography Scientific World Journal 6718653