Advanced Science Letters
Vol. 23, 3, 2017

Mobile Base Least Significant Bit Method For Steganography

Fransiskus Xaverius Kurniawan Malo[1], Albertus Joko Santoso[2], Pranowo[3]

[1] *Student at Magister Informatics Engineering, Universitas Atma Jaya Yogyakarta, 55281, Indonesia*

[2] *Lecture at Magister Informatics Engineering, Universitas Atma Jaya Yogyakarta, 55281, Indonesia*

[3] *Lecture at Magister Informatics Engineering, Universitas Atma Jaya Yogyakarta, 55281, Indonesia*

Corresponding author Email: ***kurnia.joseph@gmail.com***

Security and confidentiality are important aspects needed to exchange messages or information through the Internet. Development of technology for the Internet network allows everyone to exchange data and information without limitation of time and distance. Without the guarantee of security, other parties can easily access the information transmitted over a network or Internet. The issue raised in this research is on how to secure messages or information so that safety can be assured. The purpose of this research is to apply the right and quality steganography technique. This study uses Least Significant Bit (LSB) algorithm applied to mobile applications. The indicators used in the algorithm are color, processing time and size of the image carrier. The results of this study indicate that the Least Significant Bit algorithm can run on mobile applications with good quality. This algorithm is able to process color and grayscale images quickly and able to increase slightly the size of the original image. These algorithms can contribute to the message or information security techniques, but it can be used also by the secret services, companies and government agencies.

Keywords: Message, Image, Security, Steganography, Least significant bit

1. Introduction

Security and confidentiality are very important aspects in message or information exchange process through network or internet. It is because there is development of cybercrime with various interruption techniques, modification tappings, or fabrication[,1]. Without security guarantee, other people can easily get message or information sent through internet network. Various security techniques have been developed to protect and keep message confidentiality from irrelevant people, one of the techniques is steganography technique[,2].

Steganography is science in which it can hide text on a certain existing media by uniting the text and the media[,3]. Steganography has two processes; encoding and decoding. Encoding is a process of inserting messages into media container in this case it is image/digital image, while decoding is a process of displaying message hidden in a picture[,4,11].

Problem discussed in this research is how to secure messages or information so that the security is ensured This research uses *least significant bit algorithm* (LSB) applied on mobile application. Indicators used in this algorithm are processing time, carrier image size and PSNR value.

In this research, text messages will be inserted in images/digital images. After that the image will have information or confidential message as the result of encoding process. Information that

 1936-6612/2012/4/3398/005 doi: 10.1166/asl.2017.8773

has been inserted can be read and processed with steganography application made using *least significant bit* (LSB) algorithm through decoding process.

The purpose of this research is to implement the right steganography which has good quality. Besides, this application is expected to increase confidential message security so that the information given can only accessed by the limited intended receiver.

2. Literature Review

Lenti (2000) analyzed and tested some steganography on digital images. One of techniques in her research was using Least Significant Bit to insert a message into a digital image that does not change the performance of digital image significantly when the image had been inserted a message[5].

Amin (2014) hid message in form of confidential text in true colour 24 bit digital image in RGB format. Algorithm utilized to insert confidential message was Least Significant Bit algorithm (LSB) by changing the last *bit* or the 8th *bit* in each RGB color component[6].

Dewi (2007) developed a software of steganography on AVI file named AVISteg. The method developed in this research was *Least Significant Bit* (LSB) Modification. AVISteg was implemented in post programming language with Borland Delphi 7 compiler and operated in windows operation system. AVISteg was successful to insert data into BMP file group, however it could not be transferred into AVI file[7].

According to Suko (2011), embedding process of confidential message in steganography system was done by identifying media cover, which was redundant bit in which it could be modified without destroying integrity of the medium. Embedding process would create a stego medium with bit-bit redundant replacement with data from the confidential message. Steganography technique can be used to hide data from digital image with less or without change appeared on the image and can be exploited to export confidential message[8].

In Sari's (2012) research, she built an application using Least Significant Bit algorithm (LSB) in which the steganography had two processes. The first was hiding a message in a message storing media (*encoding*). In doing this, the message is hidden in a media that had been encrypted before. The second process was confidential message detection from message storing media *(*decode*)*[9].

3. Research Method

Least Significant Bit (LSB) algorithm is message hiding technique done by inserting messages on low bit or high bit in *byte* in the media to hide the message[5]. This trial uses image files with extension .PNG. in 1 pixel, the color is arranged from three color components, which are Red, Green and Blue. Each of them has decimal value from 0 to 255 decimal or with binary format that has 8 bit length, which is 00000000 to 11111111 binary. Therefore, in each pixel we can insert 3 data bites[8].

4. Analysis

In this research, the author designs a solution for steganography case. Steganography process with Least Significant Bit can be done at any pixels in *(*R) or (G) or (B) are.

Solution 1 example :

In this solution, the author selects position to put the data started from the 11th *pixel* in (R) element (Figure 1).

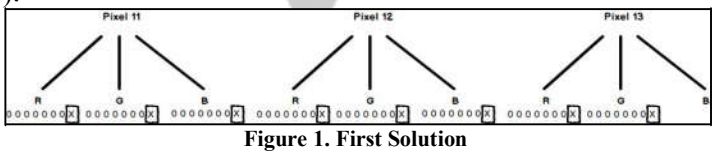

Figure 1. First Solution

Solution 2 example :

In this solution, the author selects this position to put the data started from the 17th pixel on (G) element (Figure 2).

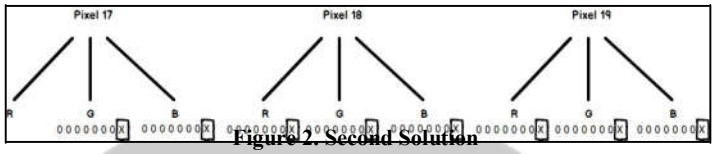

Figure 2. Second Solution

Solution 3 example :

In this solution, the author selects position to put data started from the 77th pixel on (B) element (Figure 3).

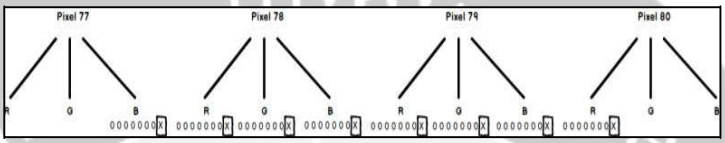

Figure 3. Third Solution

For the line, the flow chart can be showed as follow:

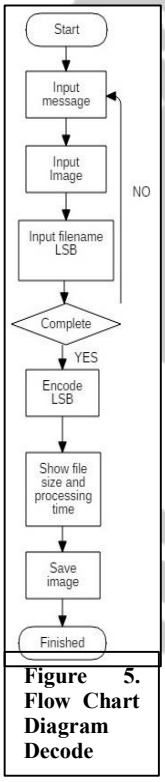

Figure 5. Flow Chart Diagram Decode

Encode process:

Started when the user running the program (select the process encode). Then the user inserts a message that we want to delivered to the recipient. After that the user choose the photo to insert the messages and user name images resulted from the developed process. Then encoding process starts when the user pressing the process button. After that, the system checks the results of the input that has been done, if all the process has met and developed by the process or not, if all the process has not been met and developed the process could not run and the system will display a warning message that there is a lack of input. The system will display the image size and the results of the process developed by the process and then save the image automatically in the internal storage. This process is completed (Figure 4).

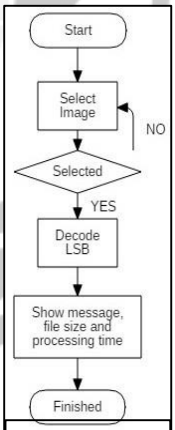

Figure 4. Flow Chart Diagram

Decode Process:

Started when the user running the program (select the process decoded). Then the user selects the image resulted from the developed process. The user pressing the button process decoded, then the system will check if all the input has been met. If it has not met the criteria, then the system will display a warning message that there is a lack of input and will return to the process of "select image". Then the system will display the message, image size and time process results of the decoded. This process is completed (Figure 5).

5. Results and Discussion

The results of research are:

1) Steganography is a method of inserting a message into an image by using Least Significant Bit (LSB) algorithm through encoding process, so that the message is unknown by other people except the limited intended receiver.
2) Steganography displays hidden message in the image through decoding process, so that the message content is known.

3) An image that is inserted by message is not broken or changes the quality. It has still similarity level.

In the discussion, the author takes an example of first solution in (Figure 1).

For encode algorithm process:

1) Insert image
2) Choose *carrier* image
3) Set file name
4) Encode process:
   a. Make container image which is similar to *carrier* image.
   b. Take text length, then change the decimal value of text length to be 8 bit binary, or that is called bit hider.
   c. Each character will be inserted on the last bit of R, G and B in each pixel.
   d. Each character in the message is changed into 8 bit binary (input prefix O if the number is less than 8 bits, so it becomes 8 bites). After that, combine all characters bit in container variable.
   e. Start from the 11st pixel, the color element (R) to (11 + (character length *8)/ 3), do LSB process with the existing bits in container variable.
   f. This process is finished.

For decode algorithm process:

1) Choose image containing hidden message.
2) Decode process: Take LSB from the 11st pixel, element (R) to (11 + (character length *8) /3) into container variable.
3) Cut or separate container variable 8 bit.
4) The process is finished.

After encoding and decoding process, the system will check the image size. It is intended to save the time of message insertion and insertion result display process because the image size is too big. Furthermore, if the image has been inserted by a message, the system will save the file as the name inputted by the user.

No.	Image Name	Original Image Size	Carrier Image Size	Encode Process Time	Decode Process Time
1.	Putri	26.974 bytes	23.521 bytes	0.025 second	0.003 second
2.	Lena	133.426 bytes	134.320 bytes	2.077 second	0.004 second
3.	Fran	172.034 bytes	149.766 bytes	2.692 second	0.003 second

Table 1. The result of encode and decode image processing

Original image before encoding process:

Figure 6. Putri *Encode*

Figure 7. Lena *Encode*

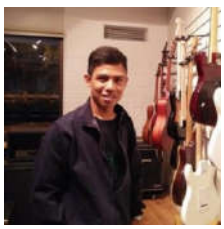

Figure 8. Fran *Encode*

Carrier image after decoding process:

Figure 9. Putri *Decode*

Figure 10. Lena *Decode*

Figure 11. Fran *Decode*

In order to determine the image quality, it uses *peak signal-to-noise ratio* (PSNR) calculation as image quality comparison of reconstruction result (Stego image) with original image (cover image). The term of *peak signal-to-noise ratio* (PSNR) is a term used in technique field which states comparison between maximum signal powers which may come from a digital signal with *derau* which influences the reliability of the signal.

Therefore, many signals have large dynamic range, so PSNR is usually expressed in logarithmic decibel scale. The formula to calculate PSNR is as follow[6] :

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \quad \text{..... (1)}$$

Image Name	Original Image Size	Carrier Image Size	Stegano Text Message	PSNR
Putri	26.974 bytes	23.521 bytes	Hallo Putri	77.850516
Lena	133.426 bytes	134.320 bytes	Hallo Lena	89.605535
Fran	172.034 bytes	149.766 bytes	Hallo Fran	89.222248

Table 2. PSNR value result with message and output of different image

As the calculation result on (Table 2), it shows that insertion of text message with different size will result different PSNR value. The bigger message file is, the more effects PSNR value change. If PSNR value is <50, it can be said that physically the image quality is bad. If the PSNR ≥ 50, the image quality is good, which means that there is only a small image damage[10].

6. Conclusion

Of the research that has been done, it can be concluded that:

1) Steganography is a very efficient and strong technique that enables people to send message in secure and hidden circumstances.
2) Least Significant Bit algorithm which is implemented on the hidden process does not influence the quality of cover image significantly.
3) This application is implemented on platform android mobile because tools for the development have supported in form of object and built-in function which is ready to use.

7. References

1. B. Rakhmat and M. Fairuzabadi, "STEGANOGRAPHY USING LEAST SIGNIFICANT BIT METHOD WITH VIGENERE AND RCA CRYPTOGRAPHY ALGORITHM COMBINATION," Informatic Dynamism Journal, (2010).

2. J. LIU and G. TANG, "Stego Key Estimation in LSB Steganography," JOURNAL OF MULTIMEDIA, vol. 7, (2012).
3. T. B. Harjo, M. Kapriati and D. A. Susanto, "Steganografi Application Using LSB (Least Significant Bit) and Tripple Des Encryption Using Programming Language C#," GLOBAL SISFOTEC JOURNAL, vol. 6, (2016).
4. R.F. Sannawira and A.S. Purnomo, "Inserting Message Image into Colorful Image Using Least Significant Bit and Redundant Pattern Encoding Method," Informatic Journal, vol 1, (2016).
5. J. Lenti, "Steganographic Methods," PERIODICA POLYTECHNICA SER. EL. ENG., vol. 44, (2000).
6. M. M. Amin, "IMAGE STEGANOGRAPHY DENGAN METODE LEAST SIGNIFICANT BIT (LSB)," Computer Science research and Its Development Journal, vol. 6, (2014).
7. K. Dewi, "STUDY AND IMPLEMENTATION OF DATA HIDING IN VIDEO DIGITAL FILE USING LEAST SIGNIFICANT BIT MODIFICATION METHOD," (2007).
8. R. S. Basuki and E. N. Maranggani, "EMBEDDING CONFIDENTAL MESSAGE IN AN IMAGE USING LEAST SIGNIFICANT BIT INSERTION METHOD (LSB)," SEMANTIC, (2011).
9. S. P. Sari, W. and D. Z. Sudirman, "Steganografi Implementation Using Least Significant Method and Advanced Encryption Standard Cryptography," in Universitas Multimedia Nusantara, Tangerang, Indonesia, (2012).
10. L. O. Sari, "The Implementation of CIELab and Chaos as Cipher on Digital Image Cryptography Application," Electric Engineering Journal, vol. 10. Page 115-159, (2013).
11. Pratiarso, M. Yuliana, M. Z. S. Hadi, F. B. H. and B. W., "PNSR Analysis on Steganography Technique Using Spread Spectrum," The 14th Industrial Electronics Seminar 2012 (IES 2012) , (2012).