RESEARCH ARTICLE OPEN ACCESS

Securing Message Using Steganography with Pixel Value Differencing Method

Indah Puspasari Handayani[1], Pinan[2], Carudin[3], Maisaroh[4]
1(Master of Computer Science, Budi Luhur University, Indonesia
Email: puspasarindahandayani@gmail.com)
2 (Master of Computer Science, Budi Luhur University, Indonesia
Email: p1mpinan789@gmail.com)
3(Master of Computer Science, Budi Luhur University, Indonesia
Email: carudin2905@gmail.com)
4 (Master of Computer Science, Budi Luhur University, Indonesia
Email: maisaroh1205@gmail.com)

Abstract:

Steganography is a technique used in the process of sending secret messages. A cover file is used as a medium to hide the secret message. On the use of the image medium as a cover the message hiding process is done by replacing each pixel bit of the cover image with the secret message bits. LSB (Least Significant Bit) is a method that is often used by replacing the bits in the LSB cover position with secret message bits. Increasing message capacity and reducing the level of distortion in steganography have been carried out by developing PVD (Pixel Value Differencing) techniques. In the technique before the message bits are inserted into the LSB cover, first the cover image is divided into non-overlapping pixel blocks. Each block contains two pixels that are close together. Then, the difference in the values of the two pixels is used to determine the size of the message bits that can be inserted. A large pixel difference can accommodate many such messages otherwise a small difference can only be accommodated by a small message.

Keywords — **Steganography, Pixel Value Differencing**

I. INTRODUCTION

Security techniques cannot only rely on physical security, but also need to be supported by information security techniques that are non-physical [1]. Therefore, in the process of sending messages, it requires a good safety technique.

The way steganography works is the confidentiality of messages to parties that are not entitled to do by hiding on a media called "cover" into pixel blocks that are not non-overlapping blocks [2].

II. LITERATURE REVIEW

Digital images can be displayed as two-dimensional matrices with rows and columns containing the position (x, y) of the image, while the element states the color value in that position. Elements in digital images are called pixels. Each pixel consists of 3 color components, namely R (Red), G (Green) and B (Blue) [3].

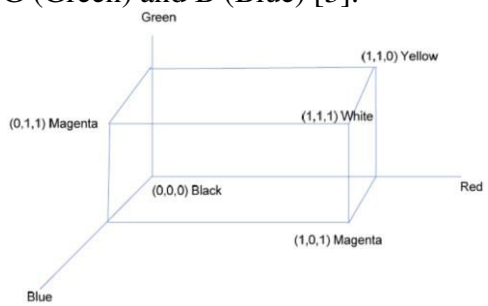

Fig. 1Cartesian coordinates of RGB colour

Pixel Value Differencing scheme uses a difference value between two successive pixels blocked to determine how many secret bits must be embedded.The process of message insertion is done in zigzag, starting from left to right then going down then to the right and going down again from left to right and so on [2]. Then in the message extraction process, to retrieve messages in the stego

image file, the message extraction process is carried out. The extraction process is done by reading (stego object) by using the message extraction algorithm implementation so that the message contained in the stego object will be known.

III. RESEARCH METHODS

The following is a method that the author applies in research papers about securing messages using steganography using the Pixel Value Differencing method [3]:

A. Material

The material in this study is the bmp 4 bit format image file and text file. BMP images are used as message media, while text files are messages to be inserted. The number of BMP images is eight, image selection is based on the diversity of images, especially seen from the spread of colour. While the text file is adjusted to the capacity of each of the BMP images.

B. Method

The research begins with several preliminary experiments to study the characteristics of pixel spread in an image.The next step is to learn how the pixel value differencing algorithm works.

C. Trial

The algorithm is made by the program with software and hardware specifications, namely as follows. Software: (1) Microsoft Windows 10, (2) Microsoft Excel 2010, and (3) Matlab R2017a. Second, hardware: (1) Intel Core i7 Processor, (2) 16 GB, (3) 8350 GB HDD Hard Drive, and (4) Keyboard and Monitor.

IV. RESULTS

This method works on a pair of adjacent pixels. The process of information insertion is done by modifying the difference in pixel value (pixel value difference) according to the message bit value and the quantization table of the grey value difference. After inserting the message, an extraction process is needed to be able to read secret messages by the recipient of the message. Quantization of grey difference is used to determine the number of bits that will be inserted at a certain difference in value. One of the ranges of grey values proposed by Wu and Tsai is [8 8 16 32 64 128]:

TABLE I
QUANTIZATION

Number of quantization (k)	Lower limit - upper limit (l_k- U_k)	Range of values	Number of bits (n)
1	0-7	8	3
2	8-15	8	3
3	16-31	16	4
4	32-63	32	5
5	64-127	64	6
6	128-255	128	7

Step of message insertion:

1) ***Change the message to an 8 bit binary number***

2) ***Calculate the difference between 2 neighboring pixels***

$$g_i, g_{i+1} \rightarrow d_1 = g_{i+1} - g_i$$

3) ***Specify top down (I_k) and number of bits n***

$$I_k \leq d_i < I_{k+1}$$

4) ***Take the message as many as n bits, then change it to decimal (b)***

5) ***Calculate the new value difference:***

$$d^1 = \begin{cases} I_k + b, d \geq 0 \\ (I_k + b), d < 0 \end{cases}$$

6) ***Count*** $\mathbf{m = d^1 - d}$

7) ***Calculate the new pixel value***

$$f(g_i^1, g_{i+1}^1) = \begin{cases} \left(g_i - \left\lceil \frac{m}{2} \right\rceil, g_{i+1} + \left\lfloor \frac{m}{2} \right\rfloor, m = odd \right) \\ \left(g_i - \left\lfloor \frac{m}{2} \right\rfloor, g_{i+1} + \left\lceil \frac{m}{2} \right\rceil, m = even \right) \end{cases}$$

Note:

- The sign ⌈ ⌉ is rounding up, for example 5.3 = 6 and -5.8 = -5
- The sign⌊ ⌋is rounding down, for example 5.3 = 5and -5.8 = -6

Example of message insertion:

A message "154" will be inserted into an 8-bit 4x4 pixel grayscale image using the PVD method.

1) ***First Message Insertion Iteration***

- Message 154 = 1 0 0 1 1 0 1 0
- $d = 200 - 105 = 95$
- $64 \le d \le 127, I_k = 64, n = 6$, (see quantization of grey value difference)
- Message = 1 0 0 1 1 0 1 0, take 6 bit, b = 1 0 0 1 1 0 = 38
- $\boldsymbol{d \ge 0}$,then the new difference is:
  $d^1 = I_k + b = 64 + 38 = 102$
- $m = d^1 - d$
  $m = 102 - 95 = 7$
- Because m is odd, then:

$$f\left(g_i^1, g_{i+1}^1\right) = \left(105 - \left\lceil \frac{7}{2} \right\rceil, 200 + \left\lfloor \frac{7}{2} \right\rfloor\right)$$

$$f\left(g_i^1, g_{i+1}^1\right) = (105 - 4, 200 + 3)$$

$$f\left(g_i^1, g_{i+1}^1\right) = (101, 203)$$

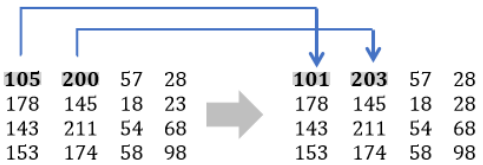

Fig. 2 First Iteration

2) ***Second Message Insertion Iteration (for the remaining bits)***

- Message 1 0
- d = 28 – 57 = -29
- $16 \le d \le 31, I_k = 16, n = 4$, (see quantization of grey value difference)
- Message = 1 0, take 4 bit,because it is not enough, 2 bits are added b = 1 0 0 0 = 8
- $\boldsymbol{d \ge 0}$,then the new difference is:
  $d^1 = I_k + b = -(16 + 8) = -24$
- $m = d^1 - d$
  $m = -24 - (-29) = 5$
- Because m is odd, then:

$$f\left(g_i^1, g_{i+1}^1\right) = \left(57 - \left\lceil \frac{5}{2} \right\rceil, 28 + \left\lfloor \frac{5}{2} \right\rfloor\right)$$

$$f\left(g_i^1, g_{i+1}^1\right) = (57 - 3, 28 + 2)$$

$$f\left(g_i^1, g_{i+1}^1\right) = (54, 30)$$

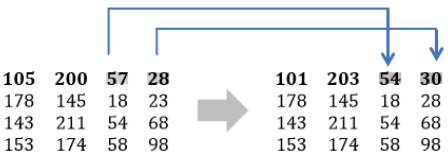

Fig. 3 Second Iteration

The message is up, the iteration is stopped and the stego image results:

101	**203**	**54**	**30**
178	145	18	28
143	211	54	68
153	174	58	98

Message extraction formula:

1) ***Calculate the difference between 2 neighboring pixels***

$$g_i, g_{i+1} \rightarrow d_1 = g_{i+1} - g_i$$

2) ***Specify top down (I_k) and number of bits n***

$$I_k \le d_i < I_{k+1}$$

3) ***Countb*** $= |d| - I_k$

4) ***Change decimal b to binary n bits***

5) ***Take message = n bits***

Example of message extraction:

The following stego image is known. Specify the 8 bits of messages contained in it.

101	**203**	**54**	**30**
178	145	18	28
143	211	54	68
153	174	58	98

1) ***Illustration of First Message Extraction***

- $d = 203 - 101 = 102$
- $64 \le d \le 127, I_k = 64, n = 6$(see quantization of grey value difference)
- b = |102|-64 = 38
- b = 38 = 1 0 0 1 1 0
- Message 6 *bit* = 1 0 0 1 1 0

2) ***Illustration of Second Message Extraction (Message's still 2 bits)***

- $d = 30 - 54 = -24$
- $16 \le |d| \le 31, I_k = 16, n = 4$ (see quantization of grey value difference)
- b = |-24|-16 = 8
- b = 8 = 1 0 0 0
- Message 2 bit = 1 0
  Message = message 6 bit ∪ message 2 bit
  = 1 0 0 1 1 0 1 0
  = 154

V. CONCLUSIONS

BMP format image file can be used as a medium for sending secret messages using the Pixel Value Differencing (PVD) method steganography. Pixel Value Differencing (PVD) method is better than the Least Significant Bit (LSB) method if the number the message to be hidden is large, the Pixel Value Differencing (PVD) method can work well for sending large amounts of messages, the workings of the Pixel Value Differencing (PVD) method is to divide the cover image into pixel blocks that are non-overlapping MathworksMatlab 2017 block and programming can be used to create a message delivery application using the Pixel Value Differencing (PVD) Steganography technique.

Notes:

- A large number of messages, merged message bit values in 1 bit stream.
- If there are pairs of pixels that experience a fall of boundary (out of the range of values) [0 255] after the embedding process, the pixel pairs will be ignored and the embedding process will be done on the next pixel block.
- In PVD, the quantization value determines the ratio between capacity and imperceptibility. Capacity is the maximum message capacity that can be accommodated and imperceptibility is image quality (difference in value before and after the embedding process).
- Quantization [8 8 16 32 64 128] produces more capacity.
- Quantization [2 2 4 4 4 8 8 16 16 32 32 64 64] produces better imperceptibility.

REFERENCES

[1] D.-C. Wu and W.-H. Tsai, "A Steganographic Method For Images By Pixel-Value Differencing," *Pattern Recognit. Lett.*, vol. 24, no. 9–10, pp. 1613–1626, 2003.

[2] Rojali, S. Guritman, and H. T. Natalisa, "PerbaikandanEvaliasiKinerjaAlgoritma Pixel Value Differencing (PVD)," *J. Mat Stat*, vol. 9 No. 2, pp. 143–156, 2009.

[3] R. Rahim, "PenyisipanPesanDenganAlgoritma Pixel Value Differencing DenganAlgoritma Caesar ChiperPada Proses Steganografi," *TIMES J.*, vol. 5, no. 2 ISSN: 2337-3601, p. No. 6-11, 2016.

[4] M. Azhari, "PengembanganAplikasiSteganografi Pixel Value Differences (PVD)," *ISSN 2407-1846*, 2014.

[5] S. Guritman, *PengantarKriptografi*. Bogor: FakultasMatematikadanIlmuPengetahuanAlam IPB, 2003.

[6] R. Amirtharajan, R. Akila, and P. Deepikachowdavarapu, "A Comparative Analysis of Image Steganography," *Int. J. Comput. Appl.*, vol. 2, pp. 41–47, 2010.