LSB-based Bit Flipping Methods for Color Image Steganography
LSB-based Bit Flipping Methods for Color Image Steganography

E Z Astuti\(^1\), D R I M Setiadi\(^1\), E H Rachmawanto\(^1\), C A Sari\(^1\) and Md K Sarker\(^2\)

\(^1\) Department of Informatics Engineering, Dian Nuswantoro University, 207 Imam Bonjol Street, Semarang 50131, Indonesia.
\(^2\) Department of Computer Science, Kansas State University, Manhattan, Kansas 66502, USA.

Abstract. Imperceptibility is one of the most important aspects of the steganography method. Specifically, for steganography in images, embedded messages must not be felt by the human visual system. Humans have a higher sensitivity in color images compared to grayscale images, therefore the method that is still being tested on grayscale images needs to be retested on color images. Previous research has shown that the bit flipping method can increase imperceptibility to around 9dB in grayscale images. In this research the bit flipping method is tested on color images in the RGB format, the message capacity embedded in the image is 1 bit per pixel. The results of the imperceptibility test turned out to result in a more varied increase due to the greater number of layers in the color image. However, the bit flipping method is also proven to work well on color images, and even a maximum PSNR increase of more than 13dB. At the extraction stage, the message image can also be extracted perfectly, namely with the value NC = 1.

1. Introduction

Internet technology and the digital world continue to develop until now, where data security has a very important role. Several digital data security methods have been implemented such as cryptography, steganography, watermarking, digital signatures, etc. for data protection. These methods are implemented depending on the needs and objectives, where protection can be done for a variety of digital data\(^[1–5]\). Specifically, we will discuss steganography in this research. Digital images are digital files that are formed from the color spectrum taken from a color sensor\(^[6]\). Digital images are stored in binary form with a variety of bit depths. Digital images are generally stored in 1-bit, 8-bits, 24-bits, and 32-bits depth formats. 1-bit depth images are binary images, 8-bit images are often called grayscale images, 24-bit images are color images that have three color layers of 8 bits each, while 32-bit images are color images with additional opacity layers. 8-bits and 24-bits images are objects that are often used in steganography research in digital images.

Image steganography is a technique for embedding secret messages on an image as a storage medium. In the science of steganography, there is one important aspect, namely imperceptibility, this aspect means that the secret message must not be felt by the human sensory system\(^[7–9]\). Because the media in the form of images that appear visually, the human sense of sight is used as a benchmark. Please note that the human visual system is more sensitive to color images than grayscale images\(^[6]\), this means the methods proposed in grayscale images and have been proven good, still need to be tested again on color images to ensure the results.
One method of steganography in digital images that have long been used is the least significant bit (LSB). Although this method is still very popular and continues to be used today. This method works in the spatial domain, which is the process of embedding messages directly by manipulating the smallest bit value of pixels [10, 11]. LSB also has advantages in the aspect of imperceptibility and has a relatively large message storage capacity. In previous research conducted by Astuti et al. [12] stated that the LSB-based bit flipping method can increase the imperceptibility of a stego image of around 9dB if measured by the PSNR measurement tool. The 9dB difference is a great value and contributes well to the imperceptibility aspect, but in testing the method still uses grayscale imagery. The bit flipping method is a technique used to change the message bit value with the negation value of the bit. Bit flipping is done when the cover image bit changes by more than 50%. In digital image storage systems on computers generally use red, green, and blue (RGB) color formats. The RGB color format has a depth of 24 bits, where each color layer has 8 bits [13]. In the RGB color format, the spread of the intensity is relatively equal on each color layer, which intensity is the energy carried on a digital image [6]. This study discusses the analysis of the effect of the method of flipping bits on color images with the LSB-based RGB format.

2. Research Method

This research discusses steganography in images using the bit flipping method which will be applied in the LSB embed method. The method of bit flipping on message bits serves to reduce changes in the value of bits in the cover image so that the aspect of imperceptibility can be increased. There are several stages carried out in this research, namely data gathering, method analysis, embedding, extraction, and measurement.

2.1. Data Gathering

![Image Cover Dataset]

Figure 1. Image cover dataset.
Research on image processing especially image steganography continues to grow until now. But the research developed by different authors, sometimes becomes hard to replicate or compare, due to the lack of detailed explanation of the method, or because the use of nonstandard data. The selection and collection of data are very important and becomes one of the determinants of the quality of research, therefore in this research, we selected standard images as a storage medium. These images can be downloaded at reference no [14], [15] and [16]. The images used are presented in Fig. 1. These images are color images in the RGB model.

Furthermore, the entire cover image is preprocessed using the Matlab R2015a application. There are two things, we did in the preprocessing stage, namely, resize and crop, so the image has dimensions of $512 \times 512$. After the pre-processing is complete, the image is saved in the .bmp extension format. This format was chosen so that no compression and no change in the pixel value occurs. As for the secret message that is used is in the form of images, this message is the same image as the research [12].

The message image presented in Fig. 2 is an image with a bitmap format with the .bmp extension. This image has a dimension of $256 \times 128$ and has a depth of 8 bits or grayscale. The size of $256 \times 128$ was chosen because it can be fully embedded in a cover image with a size of $512 \times 512$ with the same bit depth.

2.2. Method Analysis

The LSB method is one of the steganographic methods that have an embedding technique by directly changing the smallest bit of image pixel. Steganography image research that uses grayscale images as collector images does not have image layer selection because grayscale images only have one image layer. Unlike the RGB color image which has three color layers R, G, and B.
Some previous research embedded messages on one particular layer based on color theory supports aspects of imperceptibility, some are embedded by dividing the message into three parts or embed randomly[17–19]. This research used an RGB cover image with a total depth of 24 bits and each layer of depth of 8 bits, the message image is shown in Fig. 2 can be completely embedded in just one cover image layer. But if it embeds only one layer, if only one-third of the container capacity is filled, then the message image in Fig. 2 pinned three times, each for each layer R, G, and B, more details are shown in Fig. 3. Embedding with maximum capacity is done with the aim to find out the quality of imperceptibility. Please note that the embedded message can also be replaced with a different image at each layer, or it can be replaced with text or other multimedia files.

2.3. Embedding Steps
Embedding is the first major step carried out in this steganography method. At this stage, the message image is converted into binary form to be embedded in the cover image. The binary value is checked with the smallest bit of the cover image, if the change is more than 50%, then the bit flipping is performed before it is pinned[12]. In detail, the following are the steps proposed in this research:

1. Read the pixel value of the message image then save it in an integer type matrix.
2. The pixel value of a message image that has base 10 is converted to base 2 or binary.
3. Change the binary matrix value of the message image into an array.
4. Read one of the preprocessing cover images, then split the image into three matrices R, G, and B.
5. Perform the following steps in three iterations to enter the message pixel value in each matrix R, G, and B.
   a. Change the pixel value of the designated cover image layer to binary
   b. Create a counter variable and give the initial value 0 to count the number of bit changes.
   c. Perform a loop that reads the pixel values on the designated layer from the initial pixel to the final pixel. Then compare the smallest bit value of each pixel with the corresponding message image bits. If the bit values are different then increment the counter variable.
   d. After the counter value is obtained, compare the counter value with the number of pixels of the cover image. If the counter value is more than 50%, do a bit flipping. Save the flip value for the extraction process.
   e. Embed the message image bits on all pixel layer image cover with the LSB substitution method.
   f. Get the image layer that has the message image embedded.
6. Combine all layers into an RGB stego image.
7. Measure the quality of the stego image imperceptibility with the specified measurement tools.

2.4. Extraction Steps
Extraction is the stage to get the message image that has been embedded in the cover image. The extraction process is lossy on the cover image, meaning that the pixel value of the cover image cannot return to the original. But for the image of the message should be extracted perfectly, so there is no change in meaning in the message image. In this process, the input of the stego image and the flip key is needed. Here are the detailed steps:

1. Read one of the stego images, then split the image into three matrices R, G, and B.
2. Perform the following steps in three repetitions to extract the message pixel bit values for each R, G, and B matrix.
   a. Change the pixel value of the designated stego image layer to binary
   b. Extract the message image bits from the smallest pixel layer of the cover image, then store it in an array.
   c. Do the bit flipping if necessary (according to the key).
   d. Convert each 8-bit array to an integer base value of 10. So that you get an array of messages with an integer type.
   e. Reshape the message array into a matrix according to the dimensions of the message image.
3. In the extraction process, three message images will be obtained.

4. Measure and compare the image of the extracted message with the specified measuring instrument.

2.5. Measurement Tools

Stego image is a cover image that has been embedded with the message image. A good stego image must have an imperceptibility aspect. Imperceptibility means that embedded messages should not be detected by the human senses. Measuring instruments that can be used to determine the quality of imperceptibility are the peak signal to noise ratio (PSNR) and structural similarity index (SSIM). This measuring instrument is used as a standard in various digital image processing. PSNR and SSIM values are generated by comparing the original image and the reconstructed image or in this case the stego image [2,20]. SSIM can be calculated with the formula (1), while PSNR can be calculated with the formula (2).

\[
SSIM(c, s) = \frac{(2\mu_c\mu_s + C_1)(2\sigma_{cs} + C_2)}{\mu_c^2 + \mu_s^2 + C_1(\sigma_c^2 + \sigma_s^2 + C_2)} \tag{1}
\]

\[
PSNR = 20 \log_{10} \left( \frac{255}{\sqrt{MSE}} \right) \tag{2}
\]

Where, \( c \) the cover image, \( s \) the stego image, \( \mu_c \) is the average light intensity of the cover image, \( \mu_s \) is the average light intensity of the stego image, \( \sigma_c \) is the standard deviation of the cover image, \( \sigma_s \) is the standard deviation of the stego image, \( C_1 = C_2 = 0 \) and the MSE can calculate by formula (3).

\[
MSE = \frac{1}{O \times P} \sum_{k=1}^{O} \sum_{l=1}^{P} [c(k, l) - s(k, l)]^2 \tag{3}
\]

Where \( O \) and \( P \) are the dimensions of the image, \( k \) and \( l \) are the pixel coordinates of the image.

3. Results and Analysis

This section presents results on the embedding and extraction stages of the proposed steganography method. As explained earlier that in the test in this research the message image is embedded with a maximum capacity where each pixel cover image is embedded with a 1-bit pixel message image. The resulting stego image quality is measured by three measuring instruments namely MSE, PSNR, and SSIM which are presented in Table 1. Whereas the stego image is visually presented in Fig. 4. Based on the numbers of the results presented in Table 1 and the visual appearance of the results presented in Fig 4 it appears that the results of the proposed method are very good. The minimum PSNR value generated is more than 55dB even on Monarch images showing the results of PSNR inf. PSNR value = inf means that no noise occurs in the stego image, the SSIM value = 1 also shows that the similarity of structure between the original image and the stego image is the same. This should have a very small possibility because the message bit value should change the image pixel value, but inadvertently with the proposed method and the message bit value used, incidentally did not change the original image pixel value. This is actually very, very rare, but the facts speak thus and there is no engineering of the results presented.

| Image     | MSE   | PSNR       | SSIM   |
|-----------|-------|------------|--------|
| Baboon    | 0.1692| 55.8461    | 0.9998 |
| Cablecar  | 0.1668| 55.9082    | 0.9986 |
| Lena      | 0.0181| 65.5438    | 0.9999 |
| Lichtenstein | 0.0185| 65.4543    | 0.9998 |
| Monarch   | 0.0000| Inf        | 1.0000 |
| Yacht     | 0.0183| 65.4948    | 0.9996 |
Figure 4. Stego image results

Table 2. MSE, PSNR and SSIM results of stego image

| Image     | Standard Method | Proposed Method |
|-----------|-----------------|-----------------|
|           | MSE  PSNR  SSIM | MSE  PSNR  SSIM |
| Baboon    | 0.1845  55.4718  0.9997 | 0.1692  55.8461  0.9998 |
| Cablecar  | 0.1848  55.4633  0.9982 | 0.1668  55.9082  0.9986 |
| Lena      | 0.1817  55.5362  0.9998 | 0.0181  65.5438  0.9999 |
| Lichtenstein | 0.4451  51.6458  0.9995 | 0.0185  65.4543  0.9998 |
| Monarch   | 0.4458  51.6394  0.9997 | 0.0000  Inf      1.0000 |
| Yacht     | 0.4448  51.6487  0.9991 | 0.0183  65.4948  0.9996 |

Another thing that needs to be discussed is the significant difference in the value of PSNR, which is almost 9dB. This is because there are three layers in a colored image, where there are layers that do bit-flip and there are also layers that don't do bit flip. For more details, it can be seen the results of comparing MSE, PSNR and SSIM values of stego images with the proposed method and with the traditional LSB method in Table 2.

Based on the results presented in Table 2, it appears that two images have increased PSNR values around 0.5dB, one image has increased more than 9dB, two other images have more than 13dB, and one image has a PSNR value inf. This shows that this method works very well, although not on all images.

Meanwhile, to answer the statements and theories contained in the introduction that is about colored images more sensitive to the human visual system if there is a change. It is true that the imagery of grayscale and color images has a different character, with the same method and payload resulting in an increase in different imperceptibility. In the study conducted by Astuti et al. [12] the increase in PSNR can reach around 9dB during the flipping process, otherwise, the PSNR value is the same. The difference in value was generated with a grayscale type image dataset, while in this research using a
color image dataset the increase in PSNR values was relatively varied, ranging from 0.5dB, 9dB and 13dB. This variation in value is caused by differences in the message bits that the flip does. These results indicate that the bit flipping method can work better in color images because all values and indicators of the human visual system measure a better trend.

At the extraction stage, the flip key is used as a tool for the extraction process that has been previously saved at the embedding stage. The flip key by default has a size of three bits, where each bit for each layer R, G and B. Extraction results are the secret message image that has been previously pinned. At each layer, the extraction of message images is generated, so that there are three secret message images from stego images. The message image must be extracted perfectly from all three layers so that there is no change in meaning. To measure the extraction of the message image NC measurement tools are used, and based on the NC measurement results, a value of 1 is obtained, which means that the extraction results are exactly the same as the original message image. The extraction results in the form of images and NC values are presented in Table 3.

Table 3. MSE, PSNR and SSIM results of stego image

| Image   | R         | G         | B         |
|---------|-----------|-----------|-----------|
| Baboon  | 1.0000    | 1.0000    | 1.0000    |
| Cablecar| 1.0000    | 1.0000    | 1.0000    |
| Lena    | 1.0000    | 1.0000    | 1.0000    |
| Lichtenstein | 1.0000 | 1.0000    | 1.0000    |
| Monarch | 1.0000    | 1.0000    | 1.0000    |
| Yacht   | 1.0000    | 1.0000    | 1.0000    |

4. Conclusion
This research carries out further analysis of the implementation of LSB-based bit flipping methods. Theories regarding digital images say that changes in the pixel value of color images are more
sensitive to human vision sensors when compared to grayscale images. In previous research, the bit flipping method has been proven to increase imperceptibility by around 9dB in grayscale images but has not been tested in color images. In this research, the bit flipping method is applied to color images by embedding messages with maximum capacity and it turns out that in color images with the RGB model the bit flipping method works well too, although the increase in imperceptibility is quite varied a maximum increase of about 13dB is obtained. This shows that the proposed bit flipping method works well on both grayscale and color images.

5. References

[1] Jiao S, Zhou C, Shi Y, Zou W and Li X 2019 Review on optical image hiding and watermarking techniques Opt. Laser Technol. 109 370–80
[2] Setiadi D R I M 2019 Payload Enhancement on Least Significant Bit Image Steganography Using Edge Area Dilation Int. J. Electron. Telecommun. 65 295–300
[3] Setyono A and Setiadi D R I M 2019 Image Watermarking using Discrete Wavelet-Tchebichef Transform Indones. J. Electr. Eng. Comput. Sci. 16 1416–23
[4] Ardy R D, Indriani O R, Sari C A, Setiadi D R I M and Rachmawanto E H 2017 Digital image signature using triple protection cryptosystem (RSA, Vigenere, and MD5) Proceeding of 2017 International Conference on Smart Cities, Automation and Intelligent Computing Systems, ICON-SONICS 2017 ed IEEE (Yogyakarta)
[5] Kusuma E J, Indriani O R, Sari C A, Rachmawanto E H and Setiadi D R I M 207AD An imperceptible LSB image hiding on edge region using des encryption Proceedings - 2017 International Conference on Innovative and Creative Information Technology: Computational Intelligence and IoT, ICITech 2017 (Salatiga: IEEE)
[6] Sundararajan D 2017 Color Image Processing Digital Image Processing (Singapore: Springer Singapore) pp 407–38
[7] Tushara M and Navas K A 2016 Image Steganography Using Discrete Wavelet Transform-A Review Int. J. Innov. Res. Electr. Electron. Instrum. Control Eng. nCORETech LBS Coll. Eng. Kasaragod 3 2321–5526
[8] Kadhim I J, Premaratne P, Vial P J and Halloran B 2019 Comprehensive survey of image steganography: Techniques, Evaluations, and trends in future research Neurocomputing 335 299–326
[9] Hussain M, Wahab A W A, Idris Y I Bin, Ho A T S and Jung K-H 2018 Image steganography in spatial domain: A survey Signal Process. Image Commun. 65 46–66
[10] Nain S and Kumar S 2014 Steganography and Its Various Techniques Int. J. Enhanc. Res. Sci. Technol. Eng. 3 241–5
[11] Setyono A and Setiadi D R I M 2019 Securing and hiding secret message in image using xor transposition encryption and LSB method J. Phys. Conf. Ser. 1196
[12] Astuti E Z, Setiadi D R I M, Rachmawanto E H, Sari C A and Sarker M K 2019 Flipping the Message Bits to Increase Imperceptibility in the Least Significant Bit Image Steganography J. Phys. Conf. Ser. 1201 012030
[13] Azad M M, Hasan M M and K M N 2017 Color Image Processing in Digital Image Int. J. New Technol. Res. 3 56–62
[14] Ming Hsieh Department of Electrical Engineering USC Viterbi School of Engineering SIPI Image Database
[15] Levkine G http://hlevkin.com/06testimages.htm
[16] Tille A and Damato A 2001 File: Lichtenstein img processing test.png - Wikimedia Commons
[17] Viji G and Balamurugan J 2011 LSB Steganography in Color and Grayscale Images without using the Transformation Bonfring Int. J. Adv. Image Process. 1 11–5
[18] Ali U A M E, Sohrawardi M and Palash Uddin M 2019 A Robust and Secure Image Steganography using LSB and Random Bit Substitution Am. J. Eng. Res. (AJER) 39–44
[19] Muhammad K, Ahmad J, Rehman N U, Jan Z, and Sajjad M 2017 CISSKA-LSB: color image steganography using stego key-directed adaptive LSB substitution method Multimed. Tools Appl. 76 8597–626
[20] Akbar J M and Setiadi D R I M 2019 Joint Method using Akamatsu and Discrete Wavelet Transform for Image Restoration Appl. Comput. Informatics