BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **21**, No 3
Sofia • 2021 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2021-0032

# A Hybrid Method for Payload Enhancement in Image Steganography Based on Edge Area Detection

*Nadia A. Mohsin*[1], *Huda A. Alameen*[2]

[1]*Department of Computer Science, Faculty of Computer Science and Mathematics, University of Kufa, Najaf, Iraq*
[2]*Faculty of Computer Science and Mathematics, University of Kufa, Najaf, Iraq*
*E-mails: nadia.mohsin@uokufa.edu.iq hudaa.alameen@uokufa.edu.iq*

***Abstract*:** *In this research a new method for increasing the embedding capacity in images based on the edge area is proposed. The new approach combines Canny and Prewitt edge detection techniques using OR binary operation. The secret message is concealed using the Least Significant Bit* (*LSB*) *method. Embedding capacity, PSNR, SSIM, and MSE values are used as evaluation metrics. Based on the resulted values, the proposed method showed higher embedding capacity while keeping the PSNR, SSIM, MSE values without major changes of other methods which means keeping the imperceptibility quality of the stego image.*

***Keywords*:** *Data hiding, image steganography, edge detection, edge dilation, embedding capacity.*

## 1. Introduction

Information security and privacy have become a high priority due to urgent demand for data transmission over public channels, the use of cloud services, and multimedia transmission over social networks. Two main techniques to protect the confidentiality of the transmitted information, cryptography and steganography. Cryptography, on one hand, is the process of converting confidential information into a non-readable form however, the encrypted information raises the doubts of the intruders and reveals the importance of the data [1]. On the other hand, steganography is a technique for concealing secret information inside different types of media such as image, audio, video, etc., which is called the cover [2].

Images are considered as one of the most popular file formats in steganography, which is known as Image Steganography [3]. Three main factors to be considered in image steganography are imperceptibility, robustness, and capacity. Imperceptibility is used for image quality measurement by applying the Peak Signal-To-Noise (PSNR). The second factor Robustness refers to protecting the secret data against any manipulation or attacks by the eavesdropper. Capacity, also known as payload, refers to the secret information amount to be hidden in the cover image [2, 4]. Image Steganography is classified into spatial domain and frequency domain. In the spatial

domain, the secret information is embedded directly in the pixel intensity using Pixel Value Differencing (PVD) and Least Significant Bit (LSB). In the frequency domain, the transformation techniques are used like Fourier, cosine, and others.

As mentioned previously LSB is one of the techniques that is used in image steganography in a spatial domain. It is a very popular method that does not require high computational complexity and provides high embedding capacity [4]. The lower-order bits of the image pixels are substituted with the secret information bits [5]. Using classical LSB by itself in image steganography can be considered as a predictable and not a very strong method [5].

Some researchers work on combining LSB with other methods to ensure data secrecy, such as using encryption techniques to encrypt the secret information before embedding it in the cover image [6, 7]. Others work on hiding the information in specific areas of the cover image such as the image edges area [8]. Edge areas are extracted by applying one of the edge detection methods such as Canny, Sobel, Roberts, Prewitt, etc.

This paper introduces a data hiding method based on embedding the secret information in a binary form within the edge area of a cover image. The main goal is to hide confidential information in a way that any unauthorized receivers will not suspect the secret text's existence. The edges of the image are extracted using a technique that combines Canny and Prewitt methods together using OR operation. Edge dilation is applied on the acquired edges to increase the embedding capacity. LSB is used to embed the secret message.

## 2. Edge detection techniques and edge dilation

### 2.1. Edge detection

Edge detection techniques are used to identify and locate the sharp discontinuities in an image that occur due to changes in pixel intensity [9]. Edge detection process outlines and detects the image background, objects and object's boundaries. There are many methods for edge detection, but most of them can be grouped into two categories, search-based and zero-crossing-based algorithms [10]. In zero crossing the derivatives of second order is computed for detecting the edges. In search-based, the first-order derivatives are computed. The most popular methods are Sobel, Canny, Prewitt, Roberts, and Laplacian which belong to one of the above categories [11].

Each method has its pros and cons so the edges produced by these methods differ from one to another. Edge detection in image processing is used for many purposes such as image recognition, image segmentation, and even image steganography. Using edge detection in steganography means hiding the data within the edge area. Many studies have proven that image edge areas are a better option to embed secret information than any other part of the cover image as any small distortion can be noticeable [12]. One concern in using the edge area is the limited embedding capacity as not all the image pixels can be used for data hiding. The edge area size depends on the method used for detecting it, the larger the edge area size is the better.

Canny edge detector can be considered as the optimal edge detection method and it is better than many other edge detection methods [9]. Applying a canny edge

detector works on enhancing the signal to noise ratio but this is not why we choose canny to be one of the methods. Canny is known for detecting thick edges which means that the larger edge area size is better. Prewitt is another edge detection method that mainly depends on applying a pair of convolution masks of size 3×3. Prewitt is known for its sensitivity to noise. In image steganography based on edges, the edge detection method doesn't have to be accurate but it is important to detect more edges to increase the payload. So Prewitt sensitivity can work better in this case by detecting false edges and increase the edge area size. In our research, we combine Canny and Prewitt using OR binary operator which leds to increasing the embedding area as it will be shown in the experimental results section [13].

### 2.2. Edge dilation

In image steganography based on hiding the secret data in the edges usually one tries to expand the image edge area. This can be achieved by utilizing the process of dilation on the edge area to increase the number of pixels that can be used to hide the secret data. The edge dilation process is performed on binary images to enlarge the edge area [23].

The dilation process uses a dilation operator which takes two inputs, the image to be dilated and the structuring element. The structuring element is a matrix of coordinates points and has a default size of 3×3.

The dilation of the image $I$ by structuring element $S$ is denoted by $I \oplus S$ and calculated as

$$I \oplus S = \{z|(\hat{S})z \cap I \neq \emptyset\}, \quad (1)$$

where $I$ is any grayscale image and $S$ is a structuring element; $\hat{S}$ is a reflection of the structure element on pixel loop $z$.

## 3. Related work

De Rosal Ignatius Moses Setiadi, Jumanto [17], propose a method aimed to increase the payload of the secret message inside the image pixels. The reason behind using the edge area is to conceal extra secret bits without affecting the image quality because the image edge area can better tolerate the changes in the image pixels. A combination between Sobel and Canny edge detectors is used to get a thicker edge area. These two edge detection techniques methods provide a bigger edge area for more payload of messages while maintaining the stego-images imperceptibility.

Another study by S. Kumar, A. Singh and M. Kumar [4], also work on hiding the secret data in the edge area. In this article, an adaptive method based on a novel fuzzy edge identification is presented. The method works on inserting secret information in gray images. The insertion does not have a perceptible change in the cover images. The method locates the sharper edges of the cover images effectively and then hides the secret information. The image edges are kept after hiding the secret message to retrieve the data at the receiver side accurately. The experimental results have shown that the proposed method achieves a better quality of stego images than other methods if the same embedding capacity is used.

Arora and Anand [18], also propose a new technique for concealing the secret text data in the edges of a colored images edges. First, the edges are detected

by scanning the image using a 3×3 window. The text message is embedded in the edge area using the first component alteration technique. The results have shown a high quality of the encoded image and higher embedding capacity.

B a s s i l [19] proposes a technique based on the Canny edge detection method. The proposed technique work on embedding the secret information inside the pixels of the image object boundaries. More specifically, the bits of the three LSBs of each color channel of the colored image. The algorithm is parametrized using three parameters, first the Gaussian filter size, second a low threshold value, and a high threshold value. A different output has been yielded from the same input image and the same secret message due to these three parameters. As a result, finding the inner workings of the algorithm can be considered misleading and ambiguous from the exact location of the covert data.

H e m p s t a l k [20], propose two techniques for image steganography, BattleSteg, and FilterFirst. The two techniques work on improving the effectiveness of hiding the secret data edge detection filters. The techniques performance is evaluated against BlindHide and HideSeek which can be considered as more traditional techniques. A machine learner to predict steganography on images used and the results have shown that the proposed FilterFirst algorithm is more effective in hiding information than the BlindHide and HideSeek methods.

K a d h i m, P r e m a r a t n e and V i a l [21], propose an approach to enhance the image steganography that is based on the edge area by working on increasing the payload capacity and imperceptibility. Dual-Tree Complex Wavelet Transform (DT-CWT) is used with an adaptive embedding process over sub-band coefficients. Machine learning techniques are employed to conceal the secret information in blocks in the cover image with minimal retrieval error. A secret unique key is created and must be transmitted to the receiver side via a secret channel for data retrieval This improves the data security and prevents the intruders from hacking the data. The algorithm performance is evaluated with PSNR, SSIM, CF, Retrieval error, BPP, and Histogram.

## 4. The proposed approach

In this paper, we propose a new steganography method based on extracting the image edges then dilate the edge area. A hybrid edge detection method is proposed for detecting the image edges. The hybrid method is performed by combining Canny and Prewitt techniques then an edge dilation process is performed. The secret message is concealed in the least significant bits of the cover image pixels. It is embedded only in the pixels that are part of the edge area. The reason behind using this hybrid technique is to increase the number of pixels that can be used to hide the secret message.

The approach is divided into three main phases. In the first phase, the edges are detected and dilated to obtain the embedding area and saving it for later use as an extraction map. The second phase is about converting the secret text into binary form then hiding it in the edge area pixels of the cover image. The Least Significant Bit (LSB) method has been used to hide the text. The result of the previous phases is an image called the carrier image [22] which holds the secret message. In the final, phase

we extract the secret hidden text depending on the saved map. The three phases are explained briefly as it is follow.

4.1. Phase one-edge detection using a hybrid technique

The image edges in our method are detected by applying Canny and Prewitt edge detection methods then combining the resulted images. The combination is performed using OR operation. The following steps explain the edge detection phase:

**Step 1.** Reading the cover image.

**Step 2.** Extracting the edges using the Canny technique and saving the result in an image we call C. Fig. 1b shows the result of this step on the pepper image.

**Step 3.** Extracting the edges using the Prewitt technique and saving the result in an image we called *P* as in Fig. 1c.

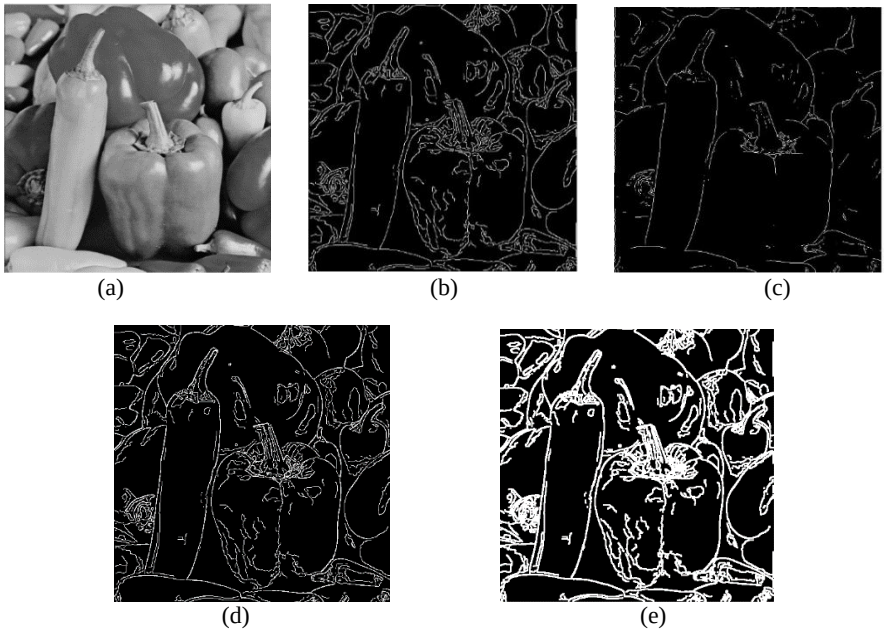
**Step 4.** Since both Canny and Prewitt are binary images, where the edges are represented by one and the non-edges are represented by zero, the OR operation will be applied to obtain a hybrid Canny-Prewitt denoted by HCP. The edge detection technique is

$$\mathrm{HCP} = C \ \mathrm{OR} \ P. \tag{2}$$

The resulted image is shown in Fig. 1d.

**Step 5.** Perform a dilation process to the edges of the HCP. The used element structure is of size 3×3. The dilation will increase the number of edge pixels that will be used in the next phase to hide the secret information, as shown in Fig. 1e.

**Step 6.** Save the resulted image as a map to be used later in the secret text extraction process.



(a) (b) (c) (d) (e)

Fig. 1. 512×512 pepper image as cover image (a); edge detection using Canny (b); edge detection using Prewitt (c); edge detection using Canny-Prewitt (d); edge detection using the proposed approach (e)

### 4.2. Phase two – secret message hiding

In this phase the following three steps are performed on the secret message.

**Step 1.** A specific symbol is added after the last letter of the secret message. This symbol is used to indicate the end of the message during the extraction process.

**Step 2.** Converted into a binary form based on the ASCII code.

**Step 3.** Hiding the secret message in the cover image using LSB method.

### 4.3. Phase three – secret message extracting

The last phase of the proposed approach is performed in the next five steps.

**Step 1.** Use the map saved from Step 6 (Phase one) to get the exact coordinates of the pixels that contain the secret message.

**Step 2.** Get the LSB of the stego image pixels based on the coordinates of the extraction map.

**Step 3.** Combine every 8 bits and convert to ASCII code.

**Step 4.** Convert the ASCII code to a character to get the secret text.

**Step 5.** Repeat Step 3 and Step 4 until finding the symbol that indicates the end of the secret message.

## 5. Experimental results

In this section, the experimental results are presented to evaluate the performance of the proposed approach. Our method has been tested and implemented using MATLAB R2014a. Standard 512×512-pixel grayscale images are used. We have evaluated the proposed method in comparison with other steganography approaches with different edge detection methods. All methods have been compared with and without the dilation process. The sizes of the embedded text messages are 512, 1024, 2048, and 4096 bytes.

Five evaluation metrics have been used, first the embedding capacity. The other factors are standard image quality assessment factors, the Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measurement (SSIM), and MSR for Mean Square Error. The last metric is the Character Error Rate (CER) which is the ratio of wrong extracted characters to the length of the secret message [23]. In this research, the embedding capacity is the number of edge pixels that can hold the secret information (one bit per pixel in our case).

PSNR is the most popular way for evaluating the quality of the reconstructed image. It is hard for the human visual system to see the difference between the original image and the stego image with PSNR values less than 35, which means a higher value of PSNR provides better image quality. It is a standard tool that is defined by a logarithmic scale [14, 23] as shown in the next equation:

$$\text{PSNR} = 10\log_{10}\left(\frac{255^2}{\sqrt{\text{MSE}}}\right). \tag{3}$$

MSE value is calculated by

$$\text{MSE} = \sum_{m=0}^{m}\sum_{n=0}^{n}\|c(m,n) - s(m.n)\|, \tag{4}$$

where $m$ and $n$ are the cover image width and height; $c$ and $s$ stand for the cover image and the Stego image [23]. As the stego image gets closer to the cover image, the MSE value gets lower and the PSNR value gets higher [14].

The SSIM calculates the similarity between the cover and the stego images. The similarity is calculated regarding contrasts, local luminance, and spatial structure. SSIM can be calculated [14, 23] using the next equation:

$$\text{SSIM}(C,S) = \frac{(2\mu_c\mu_s+y)(2\sigma_{cs}+y_2)}{(\mu_c^2+\mu_s^2+y_1)(\sigma_c^2+\sigma_s^2+y_2)}, \tag{5}$$

where each $C$ stands for the cover image and each $S$ stands for the stego image; $\mu_c$ and $\mu_s$ are the means of $C$ and $S$; $\sigma_{cs}$ is the covariance of the Cover and Stego images; $y_1$ and $y_2$ are the variables for stabilizing the division with a weak denominator; $\sigma_c^2$ and $\sigma_s^2$ are the variants of $C$ and $S$.

The last factor is the CER which can be calculated by

$$\text{CER} = \frac{\text{the Number of wrong character}}{\text{the Number of characters of the secret message}}. \tag{6}$$

As we can see, Fig. 2. and Table 1 illustrate the embedding capacity for different images using different techniques. Each technique has been evaluated with and without the dilation process. The proposed approach has a higher embedding capacity than the other approaches for all images as shown in Fig. 2. and Table 1. Table 1 shows the number of edge areas for each image using different edge detection techniques. Since our approach hides one bit in each edge area pixel that means the number of pixels in the edge area is the same as the number of allowed secret bits. For example, to hide a 512 Byte secret message, we need 4096 edge area pixels.

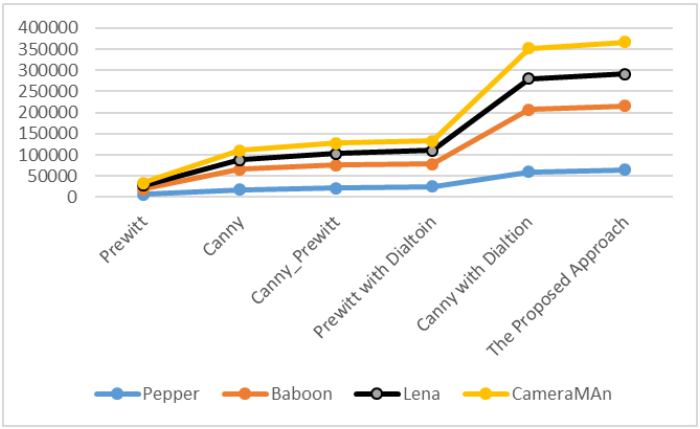


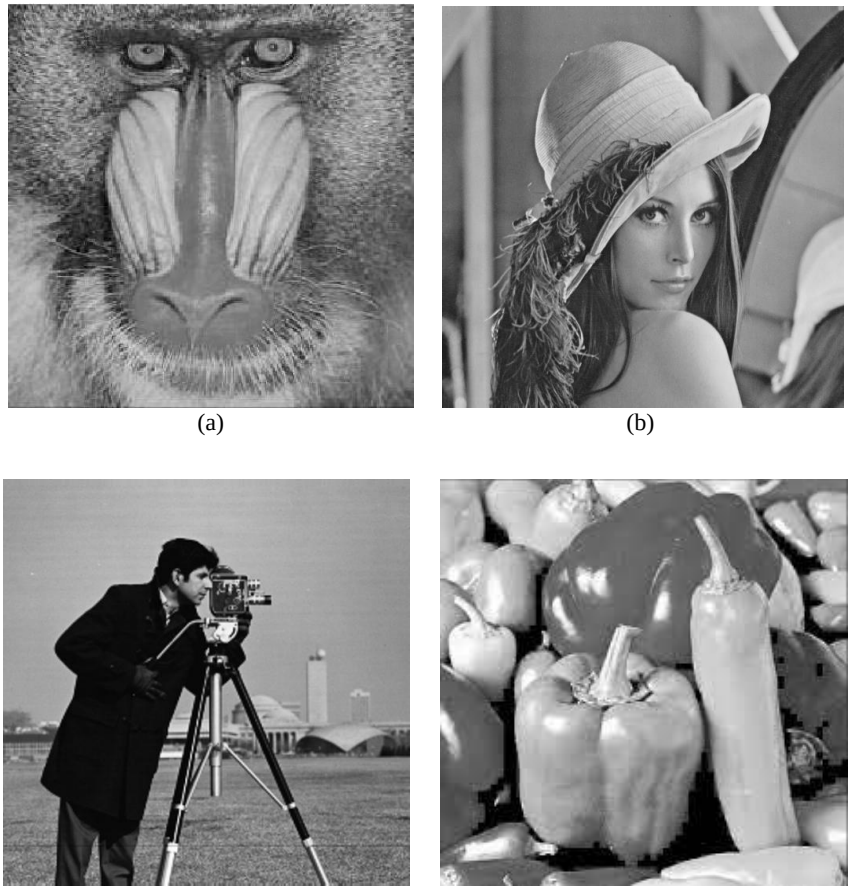

Fig. 2. Embedding capacity (number of edge area pixels)

Table 1. The number of the edge area pixels

| No | Edge detection method | Pepper | Baboon | Lena | Cameraman |
|---|---|---|---|---|---|
| 1 | Prewitt | 6715 | 12,438 | 8239 | 6397 |
| 2 | Canny | 18,254 | 47,464 | 22,549 | 22,650 |
| 3 | Canny-Prewitt | 21,868 | 53,786 | 26,961 | 25,874 |
| 4 | Prewitt with dilation | 25,082 | 53,594 | 31,607 | 22,964 |
| 5 | Canny with dilation | 60,371 | 146,495 | 72,811 | 72,285 |
| 6 | Proposed approach | 65,321 | 150,207 | 76,339 | 75,050 |

The PSNR, SSIM, and MSE values in Tables 2-5 represent the average values of the four used cover images and the stego images. The cover images used are Lena, Baboon, Pepper, and Cameraman. The used images are listed in Fig. 3.

As we can notice that Prewitt method appears only in Table 2. The reason is that Pepper and Cameraman images do not have enough edge pixels to hide the secret message. Since each byte consists of 8 bits and we are hiding only 1 bit in a pixel that means 1024 Byte secret message needs 1024×8 = 8192 edge area pixels. The same reason goes for the rest of the missing methods in Tables 3-5. While in our approach there is a huge number of edge area pixels that can conceal the secret text.



(a) (b)

(c) (d)

Fig. 3. The used cover images: baboon (a); Lena (b); cameraman (c); pepper (d)

The results of the PSNR values for the proposed approach in Tables 2, 4, and 5 are higher than the other approaches. While in Table 2 Canny has a slightly higher value than the proposed approach. MSE and SSIM values look to be the same if it is rounded three digits after the coma for all the methods and different secret message sizes, it means that embedding with the proposed approach is better due to the number of edge area pixels owned by our method is much higher than other approaches.

Table 2. PSNR, SSIM, and MSE results after embedding a 512-byte secret text message

| No | Edge detection Method | PSNR | SSIM | MSE |
|---|---|---|---|---|
| 1 | Prewitt | 69.7055308 | 0.999996 | 0.005652 |
| 2 | Canny | 69.789596 | 0.999986 | 0.005838 |
| 3 | Canny-Prewitt | 69.74364716 | 0.999989 | 0.005861 |
| 4 | Prewiit with dilation | 69.73372428 | 0.999995 | 0.00592 |
| 5 | Canny with dilation | 69.78235211 | 0.999988 | 0.005908 |
| 6 | Proposed Approach | 69.79523798 | 0.999989 | 0.005874 |

Table 3. PSNR, SSIM, and MSE results after embedding a 1024-byte secret text message

| No | Method | PSNR | SSIM | MSE |
|---|---|---|---|---|
| 1 | Canny | 66.77208064 | 0.999971 | 0.005838 |
| 2 | Canny_Prewitt | 66.76929813 | 0.999975 | 0.005861 |
| 3 | Prewiit with dilation | 66.73144196 | 0.99999 | 0.00592 |
| 4 | Canny with dilation | 66.74701707 | 0.999973 | 0.005908 |
| 5 | Proposed Approach | 66.7672965 | 0.999975 | 0.005874 |

Table 4. PSNR, SSIM, and MSE results after embedding 2048-byte secret text message

| No | Method | PSNR | SSIM | MSE |
|---|---|---|---|---|
| 1 | Canny | 63.75876159 | 0.999939 | 0.011683 |
| 2 | Canny_Prewitt | 63.75927277 | 0.999948 | 0.011686 |
| 3 | Prewiit with dilation | 63.75499056 | 0.99998 | 0.01173 |
| 4 | Canny with dilation | 63.75089407 | 0.999937 | 0.011814 |
| 5 | Proposed Approach | 63.77093533 | 0.999941 | 0.011726 |

Table 5. PSNR, SSIM, and MSE results after embedding 4096-byte secret text message

| No | Method | PSNR | SSIM | MSE |
|---|---|---|---|---|
| 1 | Canny with dilation | 60.75990313 | 0.999878 | 0.023484 |
| 2 | Proposed Approach | 60.76760591 | 0.999885 | 0.023426 |

Table 6. The CER values of the extracted messages using the proposed approach for different images and different message sizes

| No | Message size | Extracted message status | CER value |
|---|---|---|---|
| 1 | 512 Byte | Successfully extracted | 0 |
| 2 | 1024 Byte | Successfully extracted | 0 |
| 3 | 2048 Byte | Successfully extracted | 0 |
| 4 | 4096 Byte | Successfully extracted | 0 |

## 6. Conclusions

In this paper, a new approach for hiding a secret text in digital images is presented. The approach works by concealing the text in the edge area. A hybrid edge detection method consisting of combining Canny and Prewitt methods using OR binary operation followed by a dilation process to increase the edge area size is proposed.

According to the experimental results, it is proven that using the combination of the Canny-Prewitt method increases the embedding capacity more than using these methods separately. It is also proven by experiments that the dilation on the edge area increases the embedding capacity too. The edge area (embedding capacity) for Prewitt with dilation is 33,312 pixels and for Canny with dilation is 87,991 while in

the proposed approach it is 91,729 which means more pixels to hide secret data. The increment is done without affecting the imperceptibility quality of the image as we can see no major changes to the PSNR, SSIM, and MSE values.

## References


1. Ghosal, S. K., A. Chatterjee, R. Sarkar. Image Steganography Based on Kirsch Edge Detection. – Multimedia Systems, 2020, pp. 1-15.
2. Zhang, H., L. Hu. A Data Hiding Scheme Based on Multidirectional Line Encoding and Integer Wavelet Transform. – Signal Processing: Image Communication, Vol. **78**, 2019, pp. 331-344.
3. Hamid, N., et al. Image Steganography Techniques: An Overview. – International Journal of Computer Science and Security (IJCSS), Vol. **6**, 2012, No 3, pp. 168-187.
4. Kumar, S., A. Singh, M. Kumar. Information Hiding with Adaptive Steganography Based on Novel Fuzzy Edge Identification. – Defence Technology, Vol. **15**, 2019, No 2, pp. 162-169.
5. Alabaichi, Ashwak, M. A. Abid, A. K. Al-Dabbas, A. Salih. Image Steganography Using Least Significant Bit and Secret Map Techniques. – International Journal of Electrical & Computer Engineering, Vol. **10**, 2020, No 1 pp. 2088-8708.
6. Kordov, K., B. Stoyanov. Least Significant Bit Steganography Using Hitzl-Zele Chaotic Map. – International Journal of Electronics and Telecommunications, Vol. **63**, 2017.
7. Irawan, C., C. A. Sari, E. H. Rachmawanto. Hiding and Securing Message on Edge Areas of Image Using LSB Steganography and OTP Encryption. – In: Proc of 1st International Conference on Informatics and Computational Sciences (ICICoS'17), IEEE, 2017.
8. Tripathy, S. K., R. Srivastava. An Edge-Based Image Steganography Method Using Modulus-3 Strategy and Comparative Analysis. – In: Proc of International Conference on Computer Vision and Image Processing, Singapore, Springer, 2019.
9. Bhardwaj, S., A. Mittal. A Survey on Various Edge Detector Techniques. – Procedia Technology, Vol. **4**, 2012, pp. 220-226.
10. Zhang, H., L. Hu. A Data Hiding Scheme Based on Multidirectional Line Encoding and Integer Wavelet Transform. – Signal Processing: Image Communication, Vol. **78**, 2019, pp. 331-344.
11. Bassil, Y. Image Steganography Based on a Parameterized Canny Edge Detection Algorithm. – arXiv preprint arXiv:1212.6259, 2012.
12. Islam, S., M. R. Modi, P. Gupta. Edge-Based Image Steganography. – EURASIP Journal on Information Security, Vol. **2014**, 2014, No 1, pp. 1-14.
13. Maini, R., H. Aggarwal. Study and Comparison of Various Image Edge Detection Techniques. – International Journal of Image Processing (IJIP), Vol. **3**, 2009, No 1, pp. 1-11.
14. Gaurav, K., U. Ghanekar. Image Steganography Based on Canny Edge Detection, Dilation Operator and Hybrid Coding. – Journal of Information Security and Applications, Vol. **41**, 2018, pp. 41-51.
15. Kadhim, I. J., et al. Comprehensive Survey of Image Steganography: Techniques, Evaluations, and Trends in Future Research. – Neurocomputing, Vol. **335**, 2019, pp. 299-326.
16. Hameed, M. A., et al. An Adaptive Image Steganography Method Based on Histogram of Oriented Gradient and PVD-LSB Techniques. – IEEE Access, Vol. **7**, 2019, pp. 185189-185204.
17. Jumanto, J. An Enhanced LSB-Image Steganography Using the Hybrid Canny-Sobel Edge Detection. – Cybernetics and Information Technologies, Vol. **18**, 2018, No 2, pp. 74-88.
18. Arora, S., S. Anand. A Proposed Method for Image Steganography Using Edge Detection. – International Journal of Emerging Technology and Advanced Engineering, Vol. **3**, 2013, No 2, pp. 296-297.
19. Bassil, Y. Image Steganography Based on a Parameterized Canny Edge Detection Algorithm. – arXiv preprint arXiv:1212.6259, 2012.
20. Hempstalk, K. Hiding Behind Corners: Using Edges in Images for Better Steganography. – In: Proc of Computing Women's Congress, Hamilton, New Zealand. 2006.

21. K a d h i m, I. J., P. P r e m a r a t n e, P. J. V i a l. High Capacity Adaptive Image Steganography with Cover Region Selection Using Dual-Tree Complex Wavelet Transform. – Cognitive Systems Research, Vol. **60**, 2020, pp. 20-32.
22. S e t i a d i, D e R o s a l, I. M o s e s. PSNR vs SSIM: Imperceptibility Quality Assessment for Image Steganography. – Multimedia Tools and Applications, 2020.
23. S e t i a d i, D e R o s a l, I. M o s e s. Payload Enhancement on Least Significant Bit Image Steganography Using Edge Area Dilation. – International Journal of Electronics and Telecommunications, Vol. **65**, 2019.

*Received: 28.04.2021; Second Version: 05.07.2021; Accepted: 15.07.2021 (fast track)*