International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V72I10P107

Volume 72 Issue 10, 64-72, October 2024

Original Article

Comparison of Edge Detection Methods Using Road Images

Naufal Rizqullah Pratama Hidayat[1], Iman Herwidiana Kartowisastro[2,3]

[1]*Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia.*
[2]*Computer Science Department, BINUS Graduate Program-Doctor of Computer Science, Bina Nusantara University, Jakarta, Indonesia.*
[3] *Computer Engineering Department, Faculty of Engineering, Bina Nusantara University, Jakarta, Indonesia*.

[1]*Corresponding Author : Naufal.hidayat@binus.ac.id*

Received: 01 April 2024 Revised: 09 August 2024 Accepted: 29 September 2024 Published: 25 October 2024

Abstract *- Edge detection is a method in image processing that provides valuable information about images. This paper focuses on edge detection in the context of infrastructure, specifically on asphalt roads. It compares the Canny, Prewitt, Sobel, Roberts, and Laplacian of Gaussian algorithms as image processing techniques. Each algorithm yields distinct results, which are evaluated by comparing the processed images to the original images. The assessment utilizes Mean Squared Error (MSE), Peak Signal to Noise Ratio (PSNR), and Structural Similarity Index Measure (SSIM) to measure the algorithms' performance. By employing road images as data for processing, this study aims to identify the algorithm that produces the clearest edges in road images. The experimental results indicate that the Roberts algorithm demonstrates superior accuracy, achieving MSE values of 0.176, PSNR of 7.5, and SSIM of 0.001.*

Keywords *- Edge detection, Image processing, PSNR, MSE, SSIM.*

1. Introduction

Image processing has become integral across various fields. Images convey information that is essential for practical applications. However, to extract the necessary information, images must undergo processing to reveal details that can be informative. The points in images are identified by the indices of the rows and columns, which are treated as matrices. Image processing employs computer algorithms to process digital images using various techniques [1]. With advancements in the internet and imaging technology, digital images are now readily available from diverse sources such as cameras and online platforms. Organizations manage these images to facilitate efficient applications in various domains [2]. Image processing encompasses several processes, one of which is image segmentation, a fundamental aspect of computer vision. This process divides an image into homogeneous, non-overlapping segments that represent image objects or other parts [3]. Image segmentation includes several processes, with edge detection being a critical step. The primary goal of edge detection is to identify details within an image and extract relevant information, such as the shape, position, and size of objects, as well as image sharpness and enhancement [4]. It plays a vital role in enhancing the understanding of object processing in computer vision [5]. Edge detection is a significant operation in computer vision and image processing, providing essential information for applications such as image recognition, image retrieval, face recognition, corner detection, road detection, medical imaging, and target tracking [6]. The benefits of edge detection are extensive, spanning multiple fields. For instance, in geology, edge detection is the optimal method for identifying and studying reservoir fractures, yielding substantial insights into geological features [7]. Infrastructure is another area where edge detection can be employed to gather information, such as by detecting cracks in roads. Based on the related research, edge detection is frequently utilized in digital image processing for various purposes, and its results provide information that can be applied later. This research has been conducted to test the potential of edge detection in clarifying road structures since it is commonly used to identify general image objects, serving as examples for comparison. Consequently, this research employs edge detection to assess the conditions of asphalt roads. Road images exhibit various conditions and circumstances, necessitating technology that can classify road conditions using edge detection techniques. This study applies image processing techniques, utilizing several edge detection algorithms to analyze images of asphalt roads in daylight. The objective is to compare these methods to identify the most effective algorithm for assessing road conditions, using asphalt road images as the dataset. The images are converted

to grayscale before processing. Each algorithm—Canny, Prewitt, Sobel, Roberts, and Laplacian of Gaussian—will process the road images, and the results will be compared to evaluate the accuracy of each algorithm. Accuracy will be measured using Mean Squared Error (MSE), Peak Signal to Noise Ratio (PSNR), and Structural Similarity Index Measure (SSIM). The results will be compared based on the dataset and the algorithms employed.

2. Related Work

A study on tracking leukemia cell patterns using edge detection [8] compared several algorithms in detecting and tracking immature cancer cells in blood samples. The algorithms employed included Sobel, Prewitt, Roberts, and Ant Colony Optimization. The results indicated that the Prewitt algorithm produced clearer edges of leukemia cells. Another research project predicted coder performance based on Advanced Discrete Cosine Transform (ADCT) values, MSE, and PSNR [9]. ADCT coders can achieve a favorable balance between compression ratio and image quality, although they require substantial resources for high-quality compressed data. This study discussed this limitation and developed a fast and accurate approach to predicting MSE or PSNR, which operated more efficiently than traditional compression methods, allowing for time and resource savings. It demonstrated that prediction correction is feasible and desirable when the image to be compressed is affected by noise. Another study classified roads from images containing road markings [10]. This research divided an image into three regions: the first region corresponded to the camera angle, while the second and third regions contained values used for analysis. The system identified a path by examining the grayscale values of the image, which served as parameters for segmenting the image, ultimately revealing the road markings and lines.

Additionally, research has focused on detecting damaged roads using a smartphone accelerometer [11]. Detecting cracks in roads is crucial for assessing road conditions, as it impacts driver safety. Hartono et al. developed an application that estimated the likelihood of road damage based on accelerometer data collected from a distance of approximately ten meters from potholes. According to research by Hagara and Kubinec, edge detection is a procedure utilized in digital image processing. This study compared selected methods using the Berkeley dataset segmentation. It explained each algorithm's process and compared the results. The research also introduced an edge detection algorithm based on fractional derivatives, demonstrating that this approach can yield comparable or superior results to other methods [12].

Research by Liu and Mao examined the conventional Canny algorithm and identified its limitations. The traditional Canny algorithm was enhanced to filter and threshold selection, utilizing statistical methods such as mean and variance instead of Gaussian filtering. Furthermore, improvements to the crossover operator and genetic operator enhanced the accuracy of the genetic algorithm used for optimal threshold selection. Simulation results indicated that the enhanced algorithm's edge detection results were closer to the original image and exhibited better denoising effects, confirming the feasibility and effectiveness of this enhancement [13].

Research conducted by Gandhi, Kamdar, and Shah highlighted a common issue in edge detection. When identifying object edges, only the object's edges should be emphasized, excluding background edges. This challenge was particularly relevant for teams working with aerial photography or unmanned aerial vehicles to locate objects in specific areas. This research proposed a comparison of algorithms and introduced a preprocessing method to be applied in edge detection, resulting in clearer images that eliminated background noise [14]. Another study discussed the comparison of edge detection optimized by metaheuristics for recognizing cracks in concrete walls. This research compared the performance of the Roberts, Prewitt, Canny, and Sobel algorithms [15]. The objective was to identify algorithms that demonstrated good performance after optimization for crack recognition. The findings indicated that the Roberts algorithm exhibited relatively strong classification performance in detecting cracks.

Han et al. conducted a study focused on searching for and matching the position of objects within images. The study identified objects and located them within the image. For example, if an image of a smartphone with a brand logo was provided, the system identified the logo and found its location within the smartphone image. The image was converted into lines, allowing the logo object to be located and overlaid onto the smartphone image [16]. Research by Chetia, Boruah, and Sahu improved the Sobel algorithm. This study enhanced the Sobel algorithm using Non-Maximum Suppression (NMS) and double threshold techniques, resulting in a new quantum representation method based on NEQR (Novel Enhanced Quantum Representation). The research analyzed the quantum circuit to realize edge detection, focusing on pixel count, simulation results, and algorithm circuit complexity. The results indicated significant improvements in the enhanced Sobel algorithm compared to the original [17].

Research by Tsai et al. developed a system that converts existing images into sketches of specific objects. For example, an image of two panda dolls sitting in front of a group of other dolls in the background was processed to produce a more detailed line drawing, which could then be further refined with user-applied shadow strokes, resulting in a sketch of the two panda dolls in the foreground [18]. Shafiabadi et al. conducted research to identify fractures in a given area. This study compared two algorithms to determine

which yields the most optimal results in fracture identification. The Canny and Sobel edge detection algorithms were employed to evaluate their effectiveness in identifying fractures in FMI logs. The findings revealed that while Sobel produced weak discontinuities and margins, the Canny algorithm effectively identified strong edges, even in noisy images. The Canny operator excelled at filling gaps between strong and weak edges, making it less susceptible to noise than other edge detection methods. The Sobel edge detection method separated the search for horizontal and vertical edges before simultaneously representing them. Consequently, the proposed method might work well for finding cracks in electrical image logs [7]. Research by Dipmala et al. created a customized Canny edge detection algorithm. This study presented a tailored edge detector that performed better in recognizing objects, yielding sharper object edges. Utilizing a webcam for real-time object recognition, this research compared the customized Canny algorithm with other operators, including Sobel, Roberts, and Laplacian of Gaussian, achieving an overall accuracy of 92.01%.

3. Methodology

3.1. Mean Square Error

MSE is calculated using the error value in a program. In this research, MSE assesses the accuracy of an algorithm in road images. A lower MSE value, closer to zero, indicates that the image produced by an algorithm is more accurate [19].

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M,N} \quad (1)$$

MSE is calculated using the following formula, where I1 represents the original image, I2 is the image resulting from the edge detection process, M is the height of the image, and N is the width of the image.

3.2. Peak Signal to Noise Ratio

PSNR compares the maximum value of the measured signal to the noise present in the image. It is typically used to evaluate image quality before and after processing. Before calculating PSNR, the MSE value must be determined. A higher PSNR value indicates better image quality [19].

$$PSNR = 10\, log_{10} \left(\frac{C^2_{Max}}{MSE}\right) \quad (2)$$

$C^{\,2}_{Max}$ is the maximal variation in the input image. This study uses 8-bit images, where the maximum value is 255.

3.3. Structural Similarity Index Measure

The SSIM algorithm assesses the similarity between two images and correlates strongly with the perceived quality of the Human Visual System (HVS). SSIM compares the structural features of images, providing a measure of image quality. A reference image is required before SSIM can be applied to evaluate image quality. The structural similarity between the experimental and reference images is then measured. Higher SSIM values indicate improved image quality, while lower values suggest diminished quality. The intensity distribution of light reflected from an object's surface constitutes an image, linking image pixels to surface illumination and reflection. The SSIM algorithm effectively separates structural information from the image, as the object's structure is independent of luminance. SSIM values range from -1 to 1, so the greater the value obtained, the better the image quality [19].

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

$$c_1 = (k_1 L)^2, c_2 = (k_2 L)^2$$

$$k\_1 = 0.01, k\_2 = 0.03 \quad (3)$$

μx represents the average value of x, while μy denotes the average of the y values. Then, σ_x^2 is the difference from the x value, while σ_y^2 is the difference from the y value. σ_{xy} is the covariance of x and y values. L denotes the pixel value. C_n is a division variable with a weak denominator.

4. Edge Detection

4.1. Canny Edge Detection

MSE is a Canny edge detection algorithm developed by John F. Canny in 1986. It involves multiple stages in its implementation, with each stage employing specific equations for the edge detection process [5].

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (4)$$

Initially, the image undergoes a Gaussian filter process, where x represents the x gradient direction, y is the y gradient direction, π is 3.14, and σ is 0.6. For Canny edge detection, a pair of 3x3 convolution masks are utilized in this study. One mask estimates the x gradient, while the other estimates the y gradient. The masks are illustrated in Figure 1.

$$\boldsymbol{G_x = G * D_x,\ \ G_y = G * D_y} \quad (5)$$

Using Equation 5, the x and y values are calculated with the masking operator, where Dx is the x mask, and Dy is the y mask, calculated through convolution.

-1	0	1
-1	0	1
-1	0	1

Gx

-1	-1	-1
0	0	0
1	1	1

Gy

Fig. 1 Canny edge convolution mask operator

$$G = |Gx| + |Gy| \quad (6)$$

The image gradient is computed using equation 6, where G represents the grayscale image, gx is the matrix for the x-axis, and gy is the matrix about the y-axis.

Subsequently, the gradient is transformed into axial coordinates with an angle between 0 to 180, referred to as the orientation of the gradient, as shown in Equation 7.

$$\theta = tan^{-1}\left(\frac{G_y}{G_x}\right) \quad (7)$$

The next step is NMS, which scans the image to remove unnecessary pixels and select maximum values to determine edges. NMS is essential for refining the edges of an image, effectively transforming thick edges into thin, sharp edges suitable for identification. NMS operates based on gradient magnitude, scanning the image along its edges and discarding pixel values deemed non-edges, resulting in a thin line.

The final stage is hysteresis thresholding, which identifies pixels in the image to determine the resulting edge area. This process involves two threshold values: T1 represents the high threshold, and T2 represents the low threshold. An edge area is formed when a pixel has a gray level exceeding T1, indicating a strong edge pixel. Conversely, if a pixel is a weak edge pixel with a gray level below T2, it results in a non-edge area. The outcome is influenced by adjacent pixels; if a pixel's gray level is between T1 and T2, it is classified accordingly.

4.2. Sobel Edge Detection

The Sobel algorithm, introduced in 1970, measures two-dimensional spatial gradients in images. This algorithm transforms a two-dimensional pixel array into a dataset that is not statistically correlated, eliminating redundant data. The Sobel edge detector employs a pair of 3x3 convolution masks—one for estimating the gradient in the x-direction and the other for the y-direction [20]. Sobel detectors are highly sensitive to noise in images, effectively highlighting noise as edges. Therefore, this operator is recommended for large data communications detected in data transfer [21].

Pixel values are entered into the Gx and Gy masks of the image using Figure 2, where Gx represents the gradient in the x direction, and Gy is the gradient in the y direction.

-1	0	1
-1	0	1
-1	0	1

Gx

1	1	1
0	0	0
-1	-1	-1

Gy

Fig. 2 Sobel edge convolution mask operator

$$G(x,y) = \sqrt{G_X^2 + G_Y^2} \quad (8)$$

$$\theta = arc\ tan\left(\frac{G_y}{G_x}\right) \quad (9)$$

Next, the magnitude of the edge value results is calculated based on the values obtained from the masks, as shown in Equation 8. The direction of the gradient is calculated using Equation 9.

4.3. Prewitt Edge Detection

The Prewitt algorithm calculates the image intensity gradient at each point, providing the most significant direction and rate of increase from light to dark in the image. Changes in frequency distribution indicate areas where sudden changes occur, signaling the presence of edges in an image [8]. Pixel values are entered into the Gx and Gy masks of the image using Figure 3, where Gx represents the gradient in the x direction, and Gy is the gradient in the y direction.

$$G(x,y) = \sqrt{G_X(x,y)^2 + G_y(x,y)^2} \quad (10)$$

$$\theta = tan^{-1}\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \quad (11)$$

The direction of the gradient is then calculated using Equation 11.

4.4. Roberts Edge Detection

In computer vision and image processing, edge detection is performed using the Roberts cross operator, proposed by Lawrence Roberts in 1963. This operator is a quick and straightforward method for determining the spatial gradient of a digital image.

The algorithm identifies locations with high spatial frequency, which often correspond to crack objects [15]. The Roberts cross operator approximates the image gradient through discrete differentiation by calculating the sum of the squares of the differences between diagonally adjacent pixels.

-1	0	1
-1	0	1
-1	0	1

Gx

1	1	1
0	0	0
-1	-1	-1

Gy

Fig. 3 Prewitt edge convolution mask operator

0	1
-1	0

Gx

1	0
0	-1

Gy

Fig. 4 Roberts edge convolution mask operator

The Roberts cross operator follows similar steps to the other algorithms but utilizes a 2x2 convolution mask for its mask operator. Pixel values are entered into the mask using Figure 4.

$$G_{(x,y)} = |Gx\ (x,y)| + |Gy\ (x,y)| \quad (12)$$

Next, the magnitude of the edge value results is calculated based on the following Equation 12.

$$\theta(x,y) = arctan(\frac{G_y(x,y)}{G_x(x,y)}) - \frac{3\pi}{4} \quad (13)$$

The direction of the gradient is calculated using Equation 11, where π is 3.14.

4.5. Laplacian of Gaussian

The Laplacian algorithm computes the second partial derivative of an image. It examines regions with rapid intensity changes, making it a common choice for edge detection [22]. To reduce an image's sensitivity to noise, Laplacians are typically applied after smoothing the image with a Gaussian-like filter. Consequently, both processes are discussed together in this section. The operator takes one grayscale image as input and produces another as output.

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

$$\frac{\partial^2 I}{\partial x^2} = i(x+1) + i(x-1)2i(x)$$

$$\frac{\partial^2 I}{\partial y^2} = i(y+1) + i(y-1) - 2i(y) \quad (14)$$

The initialization of the pixel value L (x, y) is defined using Equation 14, where represents the image, x denotes the x direction, and y denotes the y direction.

Pixel values are approximated using the convolution operator illustrated in Figure 5.

$$LoG = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right] e^{\frac{x^2+y^2}{2\sigma^2}} \quad (15)$$

Finally, the image is denoised using Equation 15, where x represents the x direction, y is the y direction, π is 3.14, and σ is 1.4.

0	0	0
1	-4	1
0	0	0

Fig. 5 Laplacian of Gaussian convolution mask operator

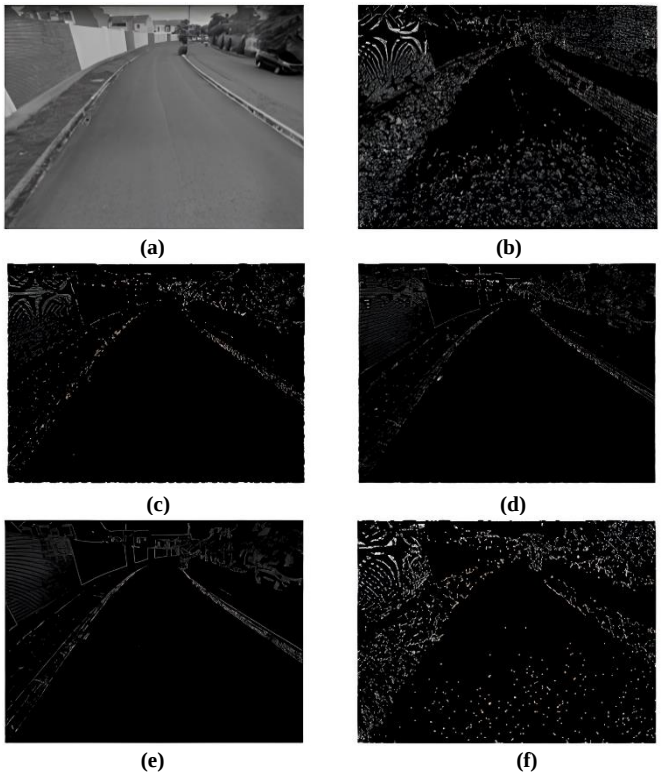


Fig. 6 Road Image 1: (a) Grayscale image (b) Canny (c) Sobel (d) Prewitt (e) Roberts (f) Laplacian of Gaussian

Table 1. Values from the comparison of Road Image 1 using edge detection algorithms

	Canny	Sobel	Prewitt	Roberts	Laplacian of Gaussian
MSE	0.199	0.1771	0.1770	0.176	0.183
PSNR	7.009	7.517	7.518	7.537	7.357
SSIM	0.006	0.003	0.003	0.001	0.016

5. Results and Discussion

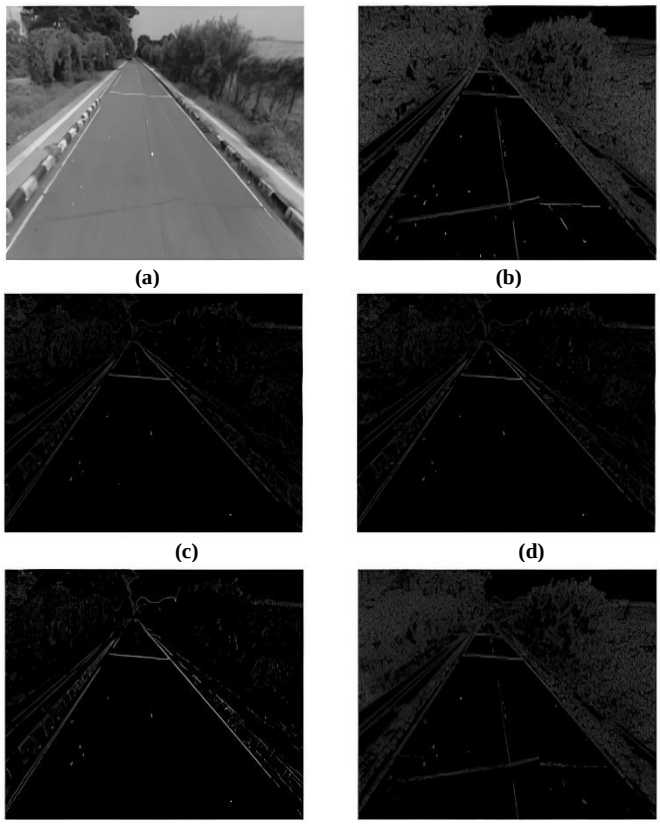
This study identifies the number of vehicles as a primary challenge, as the data used is sourced from Google Earth and Google Street View. The research requires images of roads with minimal vehicle presence. By selecting wide, paved roads with few vehicles, researchers must identify suitable data for this study. The results of the processed images depend significantly on the original image quality. Initially, the images are converted to grayscale to facilitate the edge detection process. An example of the images used is presented in Figure 6. Figure 6 illustrates the results of each edge detection algorithm. Table 1 presents the MSE, PSNR, and SSIM values for each algorithm based on the images in Figure 6. According to Table 1, the Canny algorithm exhibits the highest MSE value, followed by the Laplacian of Gaussian. The Sobel and Prewitt algorithms yield nearly identical values, while Roberts achieves the lowest recording of MSE. In terms of PSNR, Roberts produces the highest value, followed by Prewitt and Sobel, which are similar,

while Laplacian of Gaussian ranks next, and Canny has the lowest value. For SSIM, the highest score is attributed to the Laplacian of Gaussian, followed by Canny, with Sobel and Prewitt sharing the same value, while Roberts achieves the lowest score.

The next assessment focuses on processing time. As shown in Table 2, Prewitt demonstrates the fastest processing time compared to other algorithms, followed by Sobel, Roberts, and then Canny, with the Laplacian of Gaussian taking the longest processing time.

Table 2. Processing time for algorithms applied to Road Image 1

	Canny	Sobel	Prewitt	Roberts	Laplacian of Gaussian
Time	0.058s	0.008s	0.007s	0.017s	0.075s



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

Fig. 7 Road Image 2: (a) Grayscale image (b) Canny (c) Sobel (d) Prewitt (e) Roberts (f) Laplacian of Gaussian

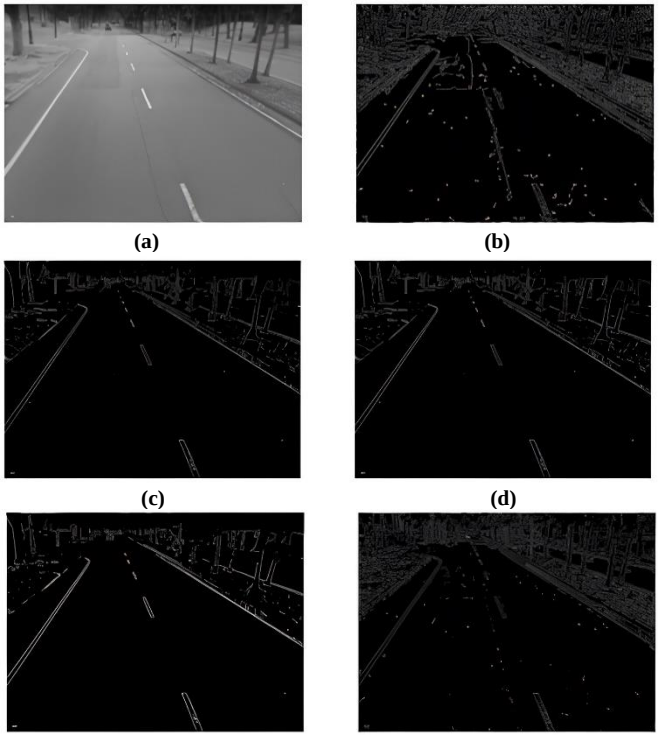
Table 3. Values from the comparison of Road Image 2 using edge detection algorithms

	Canny	Sobel	Prewitt	Roberts	Laplacian of Gaussian
MSE	0.240	0.218	0.218	0.216	0.228
PSNR	6.197	6.610	6.611	6.649	6.412
SSIM	0.006	0.004	0.004	0.003	0.017

The results for Road Image 2 are presented in Figure 7. Based on the images, Canny and Laplacian of Gaussian yield more detailed results than the other algorithms. Table 3 indicates that Canny produces the highest MSE, followed by Laplacian of Gaussian, while Sobel and Prewitt share the same value and the lowest MSE. In the PSNR row, Roberts achieves a significantly higher value than the other algorithms, while Sobel records the lowest value. For SSIM, the highest value is attributed to the Laplacian of Gaussian, while Roberts has the lowest value. In Figure 8, the Roberts algorithm produces edges that are significantly clearer and cleaner than the noise present in other algorithms. Prewitt and Sobel yield smoother and thinner edges, while Canny and Laplacian of Gaussian exhibit results with more noise compared to the other algorithms.

Table 4. Processing time for algorithms applied to Road Image 2

	Canny	Sobel	Prewitt	Roberts	Laplacian of Gaussian
Time	0.067s	0.011s	0.010s	0.015s	0.075s



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

Fig. 8 Road Image 3: (a) Grayscale image (b) Canny (c) Sobel (d) Prewitt (e) Roberts (f) Laplacian of Gaussian

Table 5. Values from the comparison of Road Image 3 using edge detection algorithms

	Canny	Sobel	Prewitt	Roberts	Laplacian of Gaussian
MSE	0.226	0.217	0.217	0.216	0.221
PSNR	6.451	6.632	6.632	6.654	6.549
SSIM	0.004	0.001	0.001	0.002	0.009

According to Table 5, the highest MSE value is produced by the Canny, Sobel, and Prewitt algorithms, which share the same value, and Roberts records the lowest MSE. In the PSNR row, the Roberts algorithm records the highest score, while Sobel and Prewitt have the lowest. In the SSIM line, the highest score is attributed to the Laplacian of Gaussian, while Sobel and Prewitt have the lowest scores. Table 6 indicates that the Sobel algorithm has the fastest processing time, while the Laplacian of Gaussian algorithm takes the longest.

Figure 9 illustrates the MSE values calculated for each algorithm. In Road Image 1, Roberts achieves the lowest score of 0.176, while Canny has the highest score of 0.199. In Road Image 2, Roberts maintains a lower value of 0.216, with Canny again recording the highest at 0.24. In Road Image 3, all algorithms exhibit similar values, but Roberts remains the lowest at 0.216, while Canny has the highest at 0.226.

Table 6. Processing time for algorithms applied to Road Image 3

	Canny	Sobel	Prewitt	Roberts	Laplacian of Gaussian
Time	0.027s	0.003s	0.004s	0.007s	0.032s

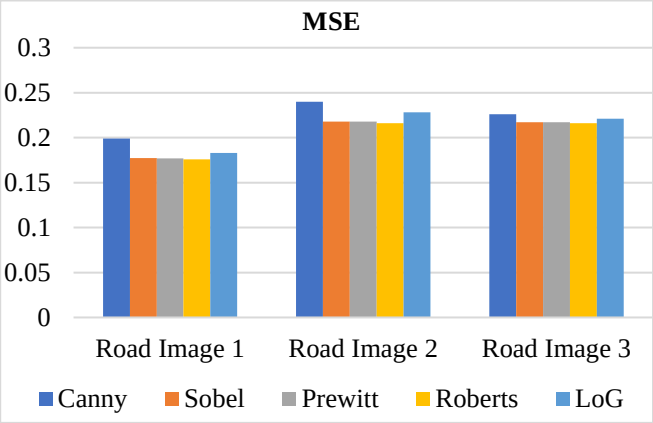


Fig. 9 Chart of MSE

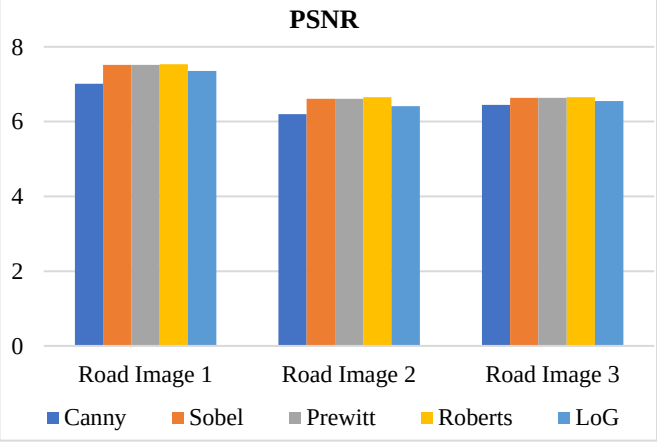


Fig. 10 Chart of PSNR

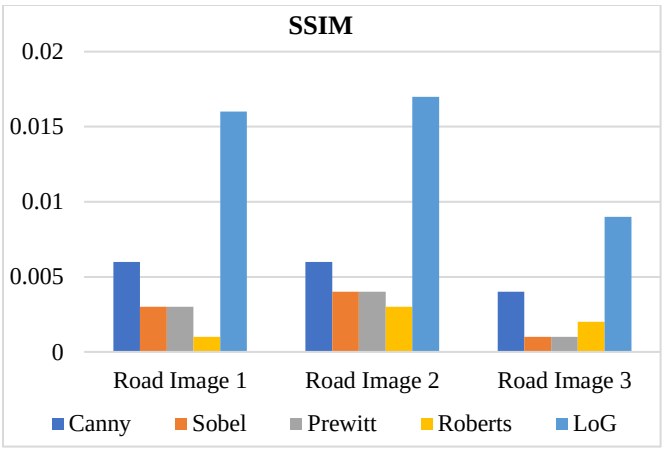


Fig. 11 Chart of SSIM

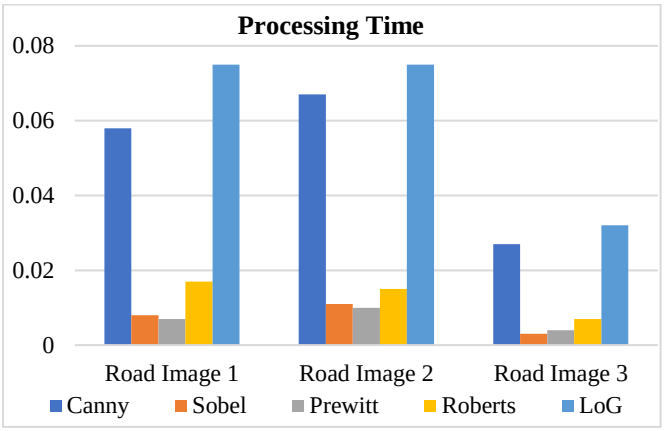


Fig. 12 Chart of processing time

Figure 10 presents the PSNR values for each algorithm. In Road Image 1, Roberts achieves the highest score of 7,537, while Canny has the lowest at 7,009. In Road Image 2, Roberts continues to hold the highest score at 6,649, and Canny has the lowest at 6,197. In Road Image 3, Roberts again records the highest score at 6,654, while Canny has the lowest at 6,451.

Figure 11 displays the SSIM values. In Road Image 1, the highest value is recorded by Laplacian of Gaussian at 0.016, while the lowest was obtained by Robert at 0.001. In Road Image 2, Laplacian of Gaussian remains the highest at 0.017, with Roberts at the lowest at 0.003. Finally, in Road Image 3, Laplacian of Gaussian again records the highest value at 0.009, while Sobel and Prewitt share the lowest value of 0.001.

Figure 12 illustrates the processing times for each algorithm. In Road Image 1, Prewitt has the fastest processing time at 0.007 seconds, while Laplacian of Gaussian takes the longest at 0.075 seconds. In Road Image 2, Prewitt again demonstrates the fastest time at 0.01 seconds, with Laplacian of Gaussian at 0.075 seconds. In Road Image 3, Sobel achieves the fastest time at 0.003

seconds, while Laplacian of Gaussian takes the longest at 0.032 seconds.

6. Conclusion

Based on the test results and discussions conducted in this research, it can be concluded that the Roberts algorithm outperforms others in the MSE assessment due to its low score, while the Laplacian of Gaussian algorithm exhibits the highest score. Similarly, in the PSNR assessment, the Roberts algorithm demonstrates superior performance compared to other algorithms, achieving the highest value.In the SSIM assessment, Laplacian of Gaussian is a good algorithm because it has a significantly higher value compared to other algorithms. Based on the processing time, the Prewitt algorithm is noted for its rapid processing capabilities.

References

[1] P. Vinista, and M. Milton Joe, "A Novel Modified Sobel Algorithm for Better Edge Detection of Various Images," *International Journal of Emerging Technologies in Engineering Research*, vol. 7, no.3, pp. 25-31, 2019. [Google Scholar] [Publisher Link]

[2] Rik Das, Sudeep Thepade, and Saurav Ghosh, "Novel Feature Extraction Technique for Content-Based Image Recognition with Query Classification," *International Journal of Computational Vision and Robotics*, vol. 7, no. 1-2, pp. 123-147, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[3] Asma Bellili, and Slimane Larabi, "An Image Analogy Approach for Multi-Scale Image Segmentation," *International Journal of Computational Vision and Robotics*, vol. 8, no. 6, pp. 639-657, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[4] Ehsan Akbari Sekehravani, Eduard Babulak, and Mehdi Masoodi, "Implementing Canny Edge Detection Algorithms for Noisy Image," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 4, pp. 1404-1410, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[5] Mamta Mittal et al., "An Efficient Edge Detection Approach to Provide Better Edge Connectivity for Image Analysis," *IEEE Access*, vol. 7, pp. 33240-33255, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[6] Junfeng Jing et al., "Recent Advances on Image Edge Detection: A Comprehensive Review," *Neurocomputing*, vol. 503, pp. 259-271, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Mina Shafiabadi et al., "Identification of Reservoir Fractures on FMI Image Logs using Canny and Sobel Edge Detection Algorithms," *Oil & Gas Science and Technology*, vol. 76, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[8] Moath Ali Alshorman et al., "Leukaemia's Cells Pattern Tracking via Multi-Phases Edge Detection Techniques," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 10, no. 1-15, pp. 33-37, 2018. [Google Scholar] [Publisher Link]

[9] Sergey Krivenko et al., "MSE and PSNR Prediction for ADCT Coder Applied to Lossy Image Compression," *2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies*, Kyiv, UKraine, pp. 613-618, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[10] Xuqin Yan, and Yanqiang Li, "A Method of Lane Edge Detection based on Canny Algorithm," *2017 Chinese Automation Congress*, Jinan, China, pp. 2120-214, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[11] Rudi Hartono, Yudi Wibisono, and Rosa Ariani Sukamto, "Damropa (Damage Roads Patrol): Damaged Road Detection Application Utilizing Accelerometer on Smartphone," *OSF Preprints*, pp. 1-6, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[12] Miroslav Hagara, and Peter Kubinec, "About Edge Detection in Digital Images," *Radioengineering*, vol. 27, no. 4, pp. 919-929, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[13] Ruiyuan Liu, and Jian Mao, "Research on Improved Canny Edge Detection Algorithm," *2018 2nd International Conference on Electronic Information Technology and Computer Engineering*, vol. 232, pp. 1-4, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[14] Meet Gandhi, Juhi Kamdar, and Manan Shah, "Preprocessing of Non-Symmetrical Images for Edge Detection," *Augmented Human Research*, vol. 5, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] Nhat-Duc Hoang, and Quoc-Lam Nguyen, "Metaheuristic Optimized Edge Detection for Recognition of Concrete Wall Cracks: A Comparative Study on the Performances of Roberts, Prewitt, Canny, and Sobel Algorithms," *Advances in Civil Engineering*, vol. 2018, no. 1, pp. 1-16, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[16] Chu Han et al., "TransHist: Occlusion-Robust Shape Detection in Cluttered Images," *Computational Visual Media*, vol. 4, pp. 161-172, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[17] R. Chetia, S.M.B. Boruah, and P.P. Sahu, "Quantum Image Edge Detection using Improved Sobel Mask Based on NEQR," *Quantum Information Processing*, vol. 20, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Hui-Chi Tsai et al., "User-Guided Line Abstraction using Coherence and Structure Analysis," *Computational Visual Media*, vol. 3, pp. 177-188, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[19] Umme Sara, Morium Akter, and Mohammad Shorif Uddin, "Image Quality Assessment through FSIM, SSIM, MSE and PSNR-A Comparative Study," *Journal of Computer and Communications*, vol. 7, no. 3, pp. 8-18, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[20] G. Ravivarma et al., "Implementation of Sobel Operator-Based Image Edge Detection on FPGA," *Materials Today: Proceedings*, vol. 45, no. 2, pp. 2401-2407, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] Ahmed Shihab Ahmed, "Comparative Study among Sobel, Prewitt and Canny Edge Detection Operators used in Image Processing," *Journal of Theoretical and Applied Information Technology*, vol. 19, no. 19, pp. 1-19, 2018. [Google Scholar] [Publisher Link]

[22] Tuan D. Pham, "Kriging-Weighted Laplacian Kernels for Grayscale Image Sharpening," *IEEE Access*, vol. 10, pp. 57094-57106, 2022. [CrossRef] [Google Scholar] [Publisher Link]