A COMPARATIVE STUDY OF COMMON EDGE DETECTION OPERATORS IN DIGITAL IMAGE PROCESSING

¹Benjamin Kommey, ¹John Kwame Dunyo, ¹Eric Tutu Tchao and ¹Andrew Selasi Agbemenu

¹ Department of Computer Engineering, Kwame Nkrumah University of Science and Technology, KNUST-Kumasi, Ghana;
Email: {¹bkommey.coe@knust.edu.gh }

Abstract

Edge detection is a fundamental process in image processing that extracts information about the image and facilitates image segmentation and feature extraction. It has many applications in various fields of computer vision. Thus, it is very necessary to understand the performance of each of these edge detectors. This paper presents a comparative study of common edge detection operators in image processing using mean squared error (MSE), peak signal to noise ratio (PSNR), and Execution time (Et). The paper shows the Canny edge detector is computationally expensive but provides higher accuracy in edge detection with higher PSNR and lower MSE. The software tool used in the project is MATLAB SIMULINK R2020a.

Key Words – Digital image processing, Edge detection, Image segmentation, Mean squared error, Peak signal to noise ratio

1. Introduction

The field of digital image processing has seen unprecedented growth due to its significant social and economic impacts. Today, there is almost no field that has not been impacted either totally or partially by digital image processing. Digital image processing refers to processing digital images by means of a digital computer. Its main goal is to improve pictorial information for human interpretation [1].

Edge detection is a fundamental process in image processing that extracts information about the image and facilitates image segmentation and feature extraction. Edge detection is the process of identifying and locating sharp discontinuities in an image. These discontinuities are the abrupt changes in pixel intensity that characterize boundaries of objects in a scene. Edge detection significantly reduces the amount of data in an image while preserving important structural features of the image [2].

Effects such as refraction or poor focus can result in objects with boundaries defined by a steady change in intensity. So, there are problems of false edge detection, missing true edges, edge localization, high computational time and problem due to noise [4].

This paper, therefore, seeks to compare and analyze the performance of the common Gradient based and Laplacian based edge detection operators using Mean

Squared Error (MSE), Peak Signal to Noise Ratio (PSNR) and Execution time[Et] performance metrics.

This paper is organized as follows; Section 1 presents the introduction and Section 2 reviews related works done in edge detection. Section 3 shows the Material and Methods used in the project while section 4 provides the Test and Evaluation results from the experiment carried out in the project. Section 5 presents the conclusion.

2. Related Works

[4] have performed a comparative analysis between their proposed Haar wavelet-based Prewitt operator and the various classical edge detectors under both noise and blur conditions. The Haar wavelet operates on data by calculating the sums and differences of adjacent elements [15]. According to their experimental results, the Haar-based Prewitt method can perform better in blur conditions. However, they found out that the Canny edge detector is better in noisy and blur conditions but its results are highly dependent on adjustable parameters.

[2] have studied the most commonly used Gradient-based and Laplacian-based edge detection techniques. They provided a comprehensive analysis of both the advantages and disadvantages of these operators. They showed that the performance of the Canny algorithm depends heavily on the adjustable parameters, which are the standard deviation for the Gaussian filter and the threshold values. They went ahead to conclude that the

Canny's edge detector is computationally more expensive compared to Sobel, Prewitt and Robert's operator. However, the Canny's edge detector performs better than all these operators under almost all scenarios. [6] and [17] have also arrived at similar conclusions.

[5] have performed a comparative analysis on various edge detection operators based on discontinuity intensity levels. They showed that the Canny edge detector produces higher accuracy in the detection of objects with higher entropy, PSNR and MSE compared with the rest but at the expense of execution time.

[7] carried out an experimental analysis of the various edge detection algorithms under Salt & Pepper, Gaussian and Speckle noise conditions. According to their results, Log and Canny perform better than Sobel, Prewitt and Roberts under noiseless conditions. However, the Canny operator provides better results in the presence of noise. They concluded that the Canny operator is a better edge detector for detecting the edges for inner as well as outer lines of the object; and Sobel edge detector proves better for discovering better outer lines (continuous boundary) only of an object.

[14] analyzed the different edge detection operators used in image processing. They showed that Canny edge detector is very expensive in terms of computation but gives the best result with a low error rate, Sobel technique gives more accurate edges and Prewitt is quickly computed

3. Materials and Methods

3.1 Edge Detection Operator

In this paper, we used the five (5) common edge detection operators. They are Roberts, Prewitts, Sobel, Canny and LoG.

3.1.1. Roberts

Robert was invented by Robert Lawrence Gilman in 1965. It is very sensitive to noise [10]. It uses the discrete method to compute the gradient. It measures the 2D point very quickly in a spatial gradient [16]. The Roberts operator uses the following mask.

$$M_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad M_y = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

3.1.2. Prewitt

Prewitt detector was discovered in 1970 by Judith M. S. Prewitt. It is similar to Sobel and less sensitive. It detects the edges horizontally and vertically and very inexpensive in terms of computation [11]. It uses gray level contrast of adjacent pixels to detect when the

contrast reaches a peak at edges, removing some false edges [17]. The Prewitt edge detector uses the following mask to approximate digitally the first derivatives M_x and M_y .

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad M_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

3.1.3. Sobel

Sobel was invented by Sobel Irwin Edward in 1970. It is very fast to compute and detect thick edges [7]. Sobel operator uses the following masks. Sobel operator makes use of two 3*3 convolution masks. Second mask is 90-degree rotation of first mask [19]. It is calculated using the mask;

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad M_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

3.1.4. Canny

Canny edge detector was invented by Canny John F. in 1986 and was designed to be an optimal edge detector. It maximizes the conventional signal to noise ratio to detect a particular edge. The Canny edge detector is computationally more expensive as compared to other edge detectors. This detector finds edges by looking for local maxima of the gradient of $f(x, y)$ [12]. The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds to detect strong and weak edges and includes the weak edges in the output only if they are connected to strong edges [5] [18].

3.1.5. Canny

Canny edge detector was invented by Canny John F. in 1986 and was designed to be an optimal edge detector. It maximizes the conventional signal to noise ratio to detect a particular edge. The Canny edge detector is computationally more expensive as compared to other edge detectors. This detector finds edges by looking for local maxima of the gradient of $f(x, y)$ [12]. The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds to detect strong and weak edges and includes the weak edges in the output only if they are connected to strong edges [5] [18].

3.1.6. Laplacian of Gaussian (LOG)

Laplacian of Gaussian (LoG) was discovered by David Marr and Ellen C. Hildreth in 1982. It also known as Marr-Hildreth edge detector and it smoothens the images [11]. The LoG is based on the second order derivative and it detect edges by first smoothing the image with a

Gaussian filter to reduce noise sensitivity [13]. It is further convolved with the Laplacian filter to detect the true edges after which the edges are analyzed by looking at the zero crossings. The Laplacian $L(x,y)$ of an image with pixel intensity values $I(x,y)$ is given by:

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

3.2 Performance Measurement Metrics

In image processing applications, the performance evaluation plays main role to assess the image quality. The importance of performance evaluation is recognized in the computer vision community [20]. The metrics used in this project to measure the edge detector operators are MSE, PSNR and Et.

3.2.1. Mean Squared Error (MSE)

Robert MSE calculates the deviation between the pixel values of original image and edge detected image [9]. It represents the cumulative squared error between the original image and the edge detected image. The lower the value of MSE, the better the detector. It is calculated using equation (1) below:

$$MSE = \frac{1}{mn} \sum_{i=0}^m \sum_{j=0}^n ([g(i,j)] - [f(i,j)])^2 \quad (1)$$

Where $\mathbf{g}$ is the original image and $\mathbf{f}$ the edge detected image. The $\mathbf{m}$ and $\mathbf{n}$ are the pixels in the rows and the columns of each image.

3.2.2. Peak Signal to Noise Ratio (PSNR)

It determines the degree of similarity between reference and edge detected image. A bigger value shows good results [9]. This ratio is used as a quality measurement between the original and the edge detected image. The higher the PSNR, the better the quality of the edge detected image. It is calculated using equation (2) below:

$$PSNR = 10 \log_{10} \left(\frac{R^2}{MSE} \right) \quad (2)$$

Where $\mathbf{R}$ is the maximum fluctuation in the image data type or the number of gray levels in the input image. It is usually 256 [3].

3.2.3. Execution time (Et)

This is the total time taken by the edge detector to completely detect edges in the input image. It is measured in seconds.

3.3 Implementation

The main software used in the project is MATLAB

R2020A (MATrix LABoratory). MATLAB provides an interactive environment for numerical computations and graphics analysis using matrix computations.

Fig. 1 shows the flowchart of the project. First, a color image is entered into the MATLAB software and converted into a grayscale image. A grayscale image has its colors in shades of gray, which carries only intensity information. It is usually 8 bits, thus ensuring higher compression of image for faster processing with less noise. In the next step, different edge detectors are applied to the image to detect the edges. Finally, the MSE, PSNR and the Et are computed.

The MSE and PSNR are calculated using equations (1) and (2). The results are then tabulated for easier comparison. The edge detected images are also printed

together with the original image for visual inspection.

Fig 1. Project flowchart

4. Test and Evaluation

Figures 2 to 6 show the results of the experiment in which five (5) common edge detectors are applied to five (5) images to detect edges. The original image is in the first position followed by Roberts, Prewitts, Sobel, Canny and finally LoG. The MSE, PSNR and Et values for each edge detector applied on the image are computed and printed below each image.

The MSE, PSNR and Execution time for each edge detector applied to the image in the project is shown in table 1.

The edge detectors are also compared using their performance metrics for each of the image used. Fig. 7 shows the comparison of MSE for Roberts, Prewitt, Sobel, Canny and LoG operators on each of the image. Figures 8 and 9 provide similar comparison of PSNR and Execution

time respectively.

Fig 2. Original Vegetables image with different edge detectors and their performance metrics

Fig 3. Original John image with different edge detectors and their performance metrics

Fig 4. Original Kofi image with different edge detectors and their performance metrics

Fig 5. Original Goat image with different edge detectors and their performance metrics

Fig 6. Original Dog image with different edge detectors and their performance metrics

Fig7. Comparison of MSE values of the edge detectors on the 5 images

Fig 8. Comparison of PSNR values of the edge detectors on the 5 images

It can be seen from the experiment results that the Canny Edge Detector has the highest accuracy in edge detection with the highest PSNR, lowest MSE and highest execution time. Sobel detector has the lowest execution time but a high MSE. The LoG operator exhibits average performance with the MSE, PSNR and Execution time.

Fig 9. Comparison of Execution time values of the edge detectors on the 5 images

Table 1

Table showing MSE, PSNR and Et for the edge detectors applied on the 5 images

Image	Operator	Performance Metrics		
		MSE	PSNR	ET(s)
Vegetables	Roberts	9042.857	8.601743	0.099498
	Prewitt	9042.395	8.601965	0.004730
	Sobel	9042.382	8.601971	0.004942
	Canny	9032.775	8.606587	0.177244
	LoG	9037.686	8.604227	0.058694
John	Roberts	28766.81	3.575883	0.038094
	Prewitt	28766.65	3.575861	0.005065
	Sobel	28766.89	3.575870	0.005383
	Canny	28750.08	3.578408	0.087799
	LoG	28758.50	3.577128	0.046825
Kofi	Roberts	3201.948	3.110657	0.033244
	Prewitt	3201.977	3.110617	0.003171
	Sobel	3201.970	3.110627	0.003508
	Canny	3201.058	3.111863	0.109469
	LoG	3201.396	3.111406	0.041176
Goat	Roberts	52647.48	0.951024	0.054667
	Prewitt	52648.52	0.950938	0.012399
	Sobel	52648.10	0.950972	0.012804
	Canny	52635.17	0.952039	0.208287
	LoG	52639.37	0.951692	0.098152
Dog	Roberts	51417.32	1.053705	0.045847
	Prewitt	51418.01	1.053646	0.012056
	Sobel	51417.90	1.053656	0.010141
	Canny	51410.71	1.054263	0.107930
	LoG	51412.99	1.054071	0.076450

5. Conclusion

In this paper, a comparative study of the common edge detection operators is conducted using mean squared error (MSE), peak signal to noise ratio (PSNR) and execution time (Et) as performance metrics. The software used is MATLAB R2020a. According to the experimental observations, LoG and Canny operators produce good performance results. However, the Canny edge detector has the highest accuracy in edge detection with the highest PSNR and lowest MSE at the expense of execution time compared with Roberts, Prewitts, Sobel and LoG. Therefore, it is concluded that Canny edge detector is an optimal detector in possible scenarios of image processing.

References

- [1] R. C. Gonzalez & R.E. Woods, Digital Image processing (3rd Edition). Upper Saddle River, NJ, USA: Prentice Hall, Inc 2008.
- [2] R. Maini and H. Aggarwal, "Study and Comparison of Various Image Edge Detection Techniques", International Journal of Image Processing, Volume. 3, no. 1, pp. 1-12, 2009.
- [3] P. H. Pawar, R. P. Patil, "Image Edge Detection Techniques using MATLAB Simulink", International Journal of Engineering Research & Technology, Volume 3 Issue 3 June, 2014.
- [4] S. R. Joshi and R. Koju, "Study and Comparison of Edge Detection Algorithms", 2012 Third Asian Himalayas International Conference on Internet, 2012, pp. 1-5, doi: 10.1109/AHICI.2012.6408439.
- [5] P. P. Acharjya, R. Das and D. Ghoshal, "Study and Comparison of Different Edge Detectors for Image Segmentation", Global Journal of Computer Science and Technology, Volume 12, Issue 13, Version 1, 2012.
- [6] S. K. Katiyar and P.V. Arun, "Comparative analysis of common edge detection techniques in context of object extraction", IEEE TGRS, Volume 50, no.11b.
- [7] M. A. Ansari, D. Kurchaniya and M. Dixit, "A Comprehensive Analysis of Image Edge Detection Techniques", International Journal of Multimedia and Ubiquitous Engineering, Volume 12, No.11 (2017).
- [8] Jose, A., Dixon, K.D.M., Joseph, N., George, E.S. and Anjitha, V., 2014, July. Performance study of edge detection operators. In 2014 International Conference on Embedded Systems (ICES) (pp. 7-11). IEEE.
- [9] G. Padmavathi, P. Subashini, P. K. Lavanya, "Performance evaluation of the various edge detectors and filters for the noisy IR images", Sensors, Signals, Visualization, Imaging, Simulation and Materials.
- [10] Selvakumar, P. and Hariganesh, S., 2016, January. The performance analysis of edge detection algorithms for image processing. In 2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE'16) (pp. 1-5). IEEE.
- [11] Kaur, P. and Kant, R., 2014. A review on: Comparison and analysis of edge detection techniques. International Journal of Engineering Research and General Science, 2(3).
- [12] J. F. Canny, "A computational approach to edge detection", IEEE Transactions on Pattern Analysis Machine Intelligent, vol. PAMI-8, no. 6, pp. 679-697, 1986.
- [13] Dharampal and Mutneja, "Methods of Image Edge Detection: A Review", Journal of Electrical and Electronic Systems 2015, 4:2 DOI: 10.4172/2332-0796.1000150
- [14] R. Kumari, A. Kumar, H. L. Mandoria, B. K. Pandey, S. Prasad, "A Review on Comparative Study of Different Edge Detection Techniques", International Research Journal of Engineering and Technology (IRJET), Volume 07, Issue 10, 2020.
- [15] Chinu and A. Chhabra, "Overview and Comparative Analysis of Edge Detection Techniques in Digital Image Processing", International Journal of Information & Computation Technology (IJICT), Volume 4, Number 10 (2014), pp. 973-980.
- [16] G. M. H. Amer and A. M. Abushaala, "Edge detection methods," 2015 2nd World Symposium on Web Applications and Networking (WSWAN), 2015, pp. 1-7, doi: 10.1109/WSWAN.2015.7210349.
- [17] Z. Lu-Bin and Z. Wei, "Analysis and application of image segmentation and edge detection operator," 2015 10th International Conference on Computer Science & Education (ICCSE), 2015, pp. 720-724, doi: 10.1109/ICCSE.2015.7250339.
- [18] Shubhashree Savant, "A Review on Edge Detection Techniques for Image Segmentation", International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 5 (4), 2014, 5898-5900
- [19] A. Jose, K. Deepa Merlin Dixon, N. Joseph, E. S. George and V. Anjitha, "Performance study of edge detection operators," 2014 International Conference on Embedded Systems (ICES), 2014, pp. 7-11, doi: 10.1109/EmbeddedSys.2014.6953040
- [20] J. Mehena, "Medical Image Edge Detection Using Modified Morphological Edge Detection Approach," International Journal of Computer Sciences and Engineering, Vol.7, Issue.6, pp.523-528, 2019