Department of Electrical and Computer Engineering
North South University

CSE467: Digital Image Processing

Performance Evaluation of Edge Detection

by

Group 11

Submitted by	Jubayer Hossain Arnob
Email addresses	jubayer.arnob@northsouth.edu ;

Submitted to: Dr. Shafin Rahman
Fall 2022

Table of Contents

1.Introduction:

The aim of this project is to work on detecting edges using Gradient and Gaussian based operators such as Sobel, Canny and performance comparison. We will use these operators for extraction of features from distinct images and evaluate output images using evaluation metrics. We will use a fixed set of images composed of individual images with different visuals. The goal of this project is purely experimental and based on results.

2.Background and Related Work:

Author *Z Zainal Abidin, S A Asmai, Z Abal Abas, N A Zakaria and S N Ibrahim* in their paper *"Development of Edge Detection for Image Segmentation*" [1] shows the goals of their study are to provide a MATLAB interface for picture edge identification using derivatives and measure PSNR, SNR, and MSE values for analysis based on trials. That leads to a result which one is better in edge detection among Canny, Sobel, Prewitt, Robert and log method. In paper" Edge *Detection Enhancement Based on Filtering and Threshold Estimation"* [2] author *Khalid Alshalfan, Mohammed Zakariah* addressed that, The traditional edge detection approach has a number of drawbacks in this study. They improved the edge detection approach in this work by using filtering and detecting threshold values to distinguish between distinct picture contrasts. Another paper is "Performance Evaluation of Edge Detection Using Sobel, Homogeneity and Prewitt Algorithms" [3] done by Abdel Karim M. Baareh, Ahmad Al-Jarrah, Ahmad M. Smadi, and Ghazi H. Shakah. The relevance of this research stems from a study that compared and evaluated the impacts of three well-known edge detection strategies in a spatial domain, utilizing both subjective and objective methods to determine the optimal edge detection algorithm.

3.Problem Overview:

The purpose of this work is to provide a quick overview of several edge detection approaches, as well as their performance difficulties and limits. No edge detection method is perfect. All has pros and cons in it. Some major edge detecting issues are this –

- Noise Sensitivity
- Accuracy
- Edge loss or discontinuity in edges
- Loss in details of the image
- Complexity

Edge detecting methods that are good at handling these issues are better. This is a problem to know which method performs better in handling which issues or all of it. This paper will give the overview on this so that it helps people to know which method to choose in which sector or which one is the better option to use. For this we will be working

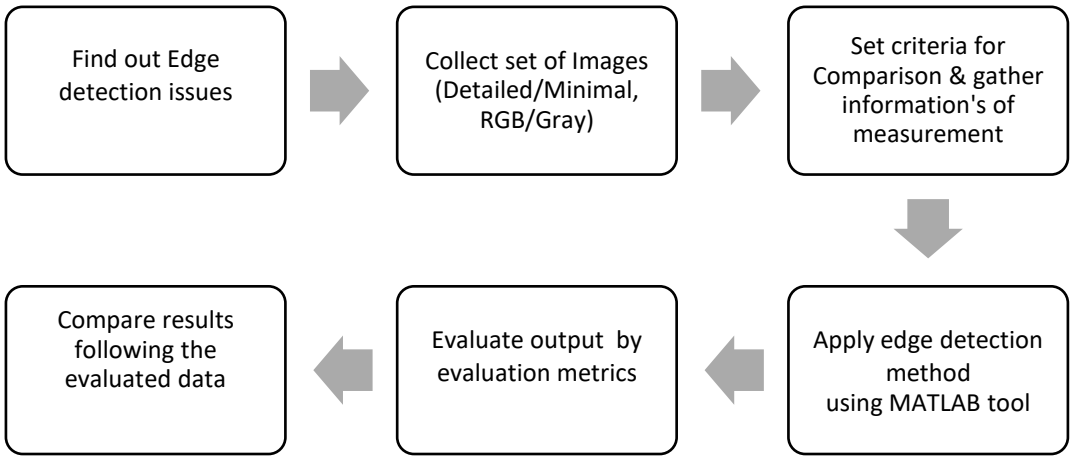
on few basic commonly used edge detection techniques. And if possible, this study will discuss how to make the techniques work better to overcome these issues further more.

4.Solution Strategy:

To analyze the performance of the commonly used edge detection techniques we need to experiment all by our own. For this we need a dataset or a set of images first on which we will experiment the methods. Dataset used for this work is The Berkeley Segmentation Dataset.[6]. Software tool we will be using is MATLABR2021b.[4] Next, we apply those edge detection methods on all the images and get the output. The performance evaluation work starts here. As we discussed in the upper section about possible issues with edge detection such as loss of details or noise issue etc. Here we will try to compare based on following criteria per image for all edge detection methods which will be compared with their corresponding ground truths –

- Noise in image
- Edge loss & object count
- Details in image

We will try to evaluation using evaluation metrics for image quality measurement. Future scope for this work is to get better outcome from these techniques of edge detection which will be experimental and based on results.



Apply edge detection method:

- Convert input image from RGB to Gray
- Smoothen Image to lessen noise
- Apply all methods at its default to get the general Idea of its performance. Sample input and outputs.

Evaluation metrics determine:

Approach of evaluation is to evaluate the generated image quality. The evaluation procedure is based on the following evaluation metrics of image quality-

- MSE (Mean Square Error) Score
- PSNR (Peak Signal-to-Noise Ratio) Score
- Number of Connected Components (8-CC)
- Information loss value = entropy (Original Image) – entropy (Output)
- PIQE (Perception based Image Quality Evaluator (no-reference)

Lower values are better: MSE score, PIQE score, No. of CC, Information loss value
Higher values are better: PSNR score, Entropy value

5.Experiment and Evaluation:

Image quality measure according to MSE and PSNR value

Here, to measure MSE and PSNR value we need 2 images, one is noise image which is our edge detected output and the other one is ground truth of edge detected image. Dataset we have used here for input image and reference ground truth is The Berkeley Segmentation Dataset.[6] The measurement evaluation idea here is – Lesser MSE score and Higher PSNR score refers to the better-quality image.

Mean Square Error (MSE) specifies the average difference of the pixels throughout the original ground truth image with edge detected image. The lesser MSE value is the better performing model considered. [7] The PSNR block computes the peak signal-to-noise ratio, in decibels, between two images. The higher the PSNR, the better the quality of the output image as it consists less noise. [8] In the following tables we can see in all the images Canny edge detection produced image has the lesser MSE value as well as the Higher PSNR value. So Canny edge detection image has less mean square error and definitely keeps lesser noise than any other method.

MSE SCORE DATASHEET

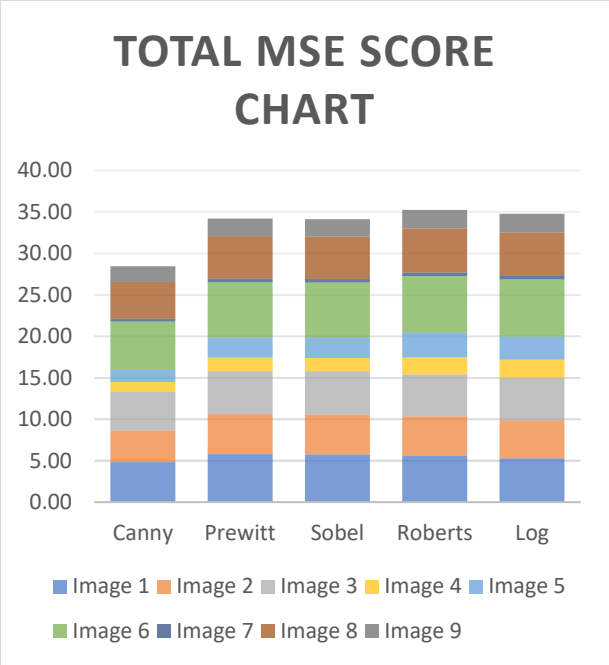
	Image 1	Image 2	Image 3	Image 4	Image 5	Image 6	Image 7	Image 8	Image 9
Canny	4.85	3.82	4.64	1.19	1.46	5.83	0.35	4.44	1.88
Prewitt	5.80	4.81	5.20	1.60	2.42	6.69	0.42	5.14	2.11
Sobel	5.78	4.79	5.20	1.60	2.42	6.69	0.41	5.13	2.10
Roberts	5.56	4.80	5.04	2.06	2.97	6.82	0.45	5.33	2.24
Log	5.29	4.60	5.14	2.15	2.77	6.90	0.52	5.12	2.31

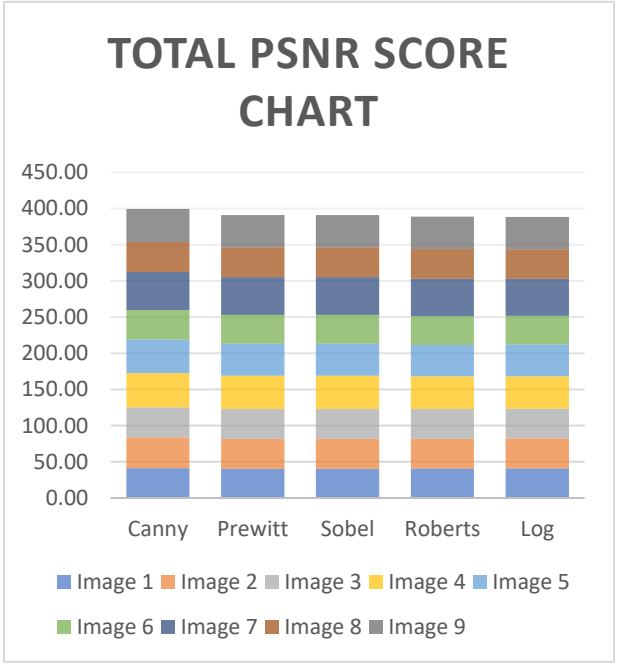
Table 1: MSE score

PSNR SCORE DATASHEET

	Image 1	Image 2	Image 3	Image 4	Image 5	Image 6	Image 7	Image 8	Image 9
Canny	41.31	42.35	41.50	47.40	46.52	40.50	52.78	41.69	45.41
Prewitt	40.53	41.34	41.01	46.11	44.32	39.91	51.97	41.06	44.92
Sobel	40.55	41.36	41.00	46.11	44.33	39.91	51.99	41.06	44.94
Roberts	40.72	41.35	41.14	45.03	43.43	39.83	51.67	40.89	44.67
Log	40.93	41.54	41.05	44.85	43.74	39.78	50.98	41.07	44.54

Table 2: PSNR score





Before this, we couldn't find any dataset with such ground truth image thus we had to take random images and couldn't use any ground truth data. We thought, As the purpose is to compare the edge detection methods using gray version of input image can help to measure that anyway. Though we couldn't get how good each method is performing. But we can definitely saw which one is doing better than others. Thought the conclusion of evaluation is still same, as it wasn't any well stablished or has no reliability, we had to search for a proper dataset which we found later.

Image quality measure according to PIQE value

To ensure the validity of what we observed so far and to get overall image quality comparison, we evaluated model outputs with a measurement metric that doesn't require multiple input parameter. Thus, we have measured all the outputs using PIQE metric. According to this, lesser the value, better the quality of image.[9] Although we aren't trying to reach best quality output like getting results less than 20 or 50. Our target is to find out which method is doing better than other method so that small difference is enough to do this comparison. Here Canny gets lower value for all images except 3 and 7. Which scores 7 out of 9 of the data. And those 2 lower values are taken by Log method. So Canny performed better.

PIQE SCORE DATASHEET

	Image 1	Image 2	Image 3	Image 4	Image 5	Image 6	Image 7	Image 8	Image 9
Canny	77.67	76.77	78.73	77.34	80.39	76.15	82.82	78.38	77.99
Prewitt	78.28	79.69	79.63	78.97	81.63	79.39	83.28	79.82	80.17
Sobel	78.75	79.60	79.14	79.23	81.37	79.21	83.59	79.54	79.76
Roberts	79.34	80.72	80.60	79.55	81.71	79.30	82.21	80.12	81.15
Log	79.05	77.32	78.35	79.48	80.85	76.81	81.15	78.63	80.89

Table 3: PIQE score

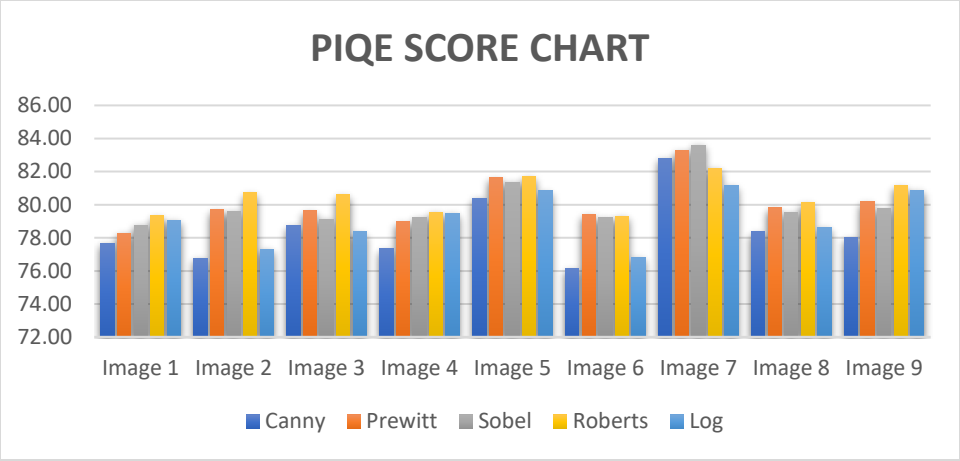


Image quality measure according loss in details

Entropy is used to measure the amount of data in a source. The more data you have the higher the entropy of the source is. [10] Entropy is a measure of image information content, it is used here to evaluate image details, the entropy value is used as it provides better comparison of the image details. What we have done here is to find the difference between original grayscale image entropy value and generated edge detected image entropy value. So, which method output has less difference has more details, in other word that generated image loses less information than other methods. [11]

So, analyzing below table we can say that Canny method loses less information of the original image or it produces more detailed edge detection image.

INFORMATION LOSS DATASHEET

	Image 1	Image 2	Image 3	Image 4	Image 5	Image 6	Image 7	Image 8	Image 9
Canny	7.56	6.89	7.28	7.34	5.78	6.68	2.48	6.80	5.83
Prewitt	7.74	7.18	7.48	7.51	5.95	6.86	2.59	7.00	6.06
Sobel	7.74	7.18	7.47	7.51	5.95	6.86	2.59	7.00	6.06
Roberts	7.75	7.16	7.46	7.50	5.93	6.86	2.57	7.07	6.01
Log	7.60	7.08	7.38	7.41	5.86	6.77	2.52	6.92	5.95

Table 4: Details loss using Entropy

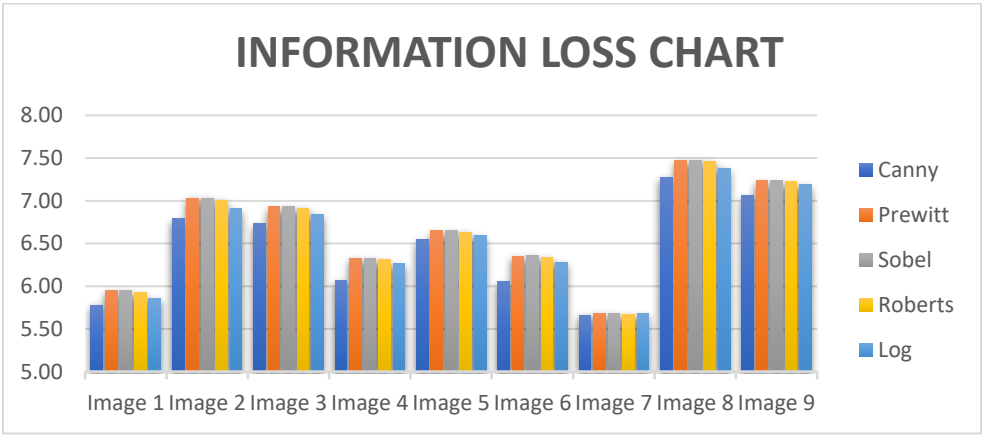


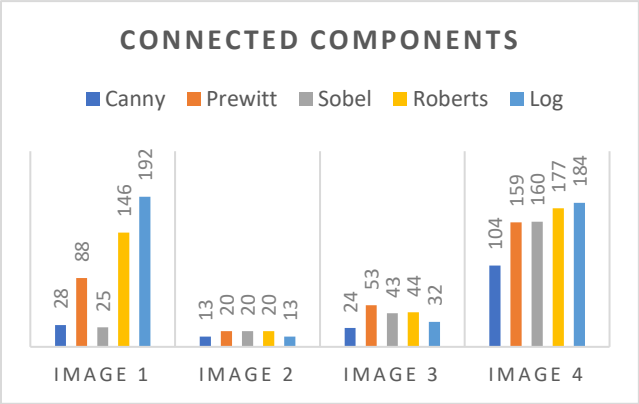
Image quality measure according Connected components

In terms of connected components, less object label means more edge continuity. The idea here is less edge loss connects more to label an object thus it has less discontinuity or break points in edges. And when edges have more break points it breaks down to more components which are not connected thus it increases the number of connected components. [13][14] We calculated the number of 8-connected components in this work Edge detection is an ill-posed issue, and various noise levels and detector settings can provide very different answers for the same scene, therefore any measure of the number of edges will be quite poor, especially in textured photos. [12] For this, we have used 4 simple and easy object detecting image to test these techniques on. Here, from below table and figure we can see Canny edge detection method has the closest accuracy in object detection in all 4 images than others. And it implies its less edge loss as well as better performance. For an easy example, here in image 1 from fig. 1, the image has actually 28 circles but only nearest result came from Canny edge detection as it detected all 28 objects. The others are nowhere close to it except Sobel which found out 25 as it misses few objects. And too many objects count means too many edges loss or break points. It implies canny method is actually good at not losing edges. It can handle edge discontinuity better than other methods.

CONNECTED COMPONENT DATASHEET & GRAPH CHART

	Image 1	Image 2	Image 3	Image 4
Canny	28	13	24	104
Prewitt	88	20	53	159
Sobel	25	20	43	160
Roberts	146	20	44	177
Log	192	13	32	184

Table 5: Connected components count



6.Output Results for the Evaluation

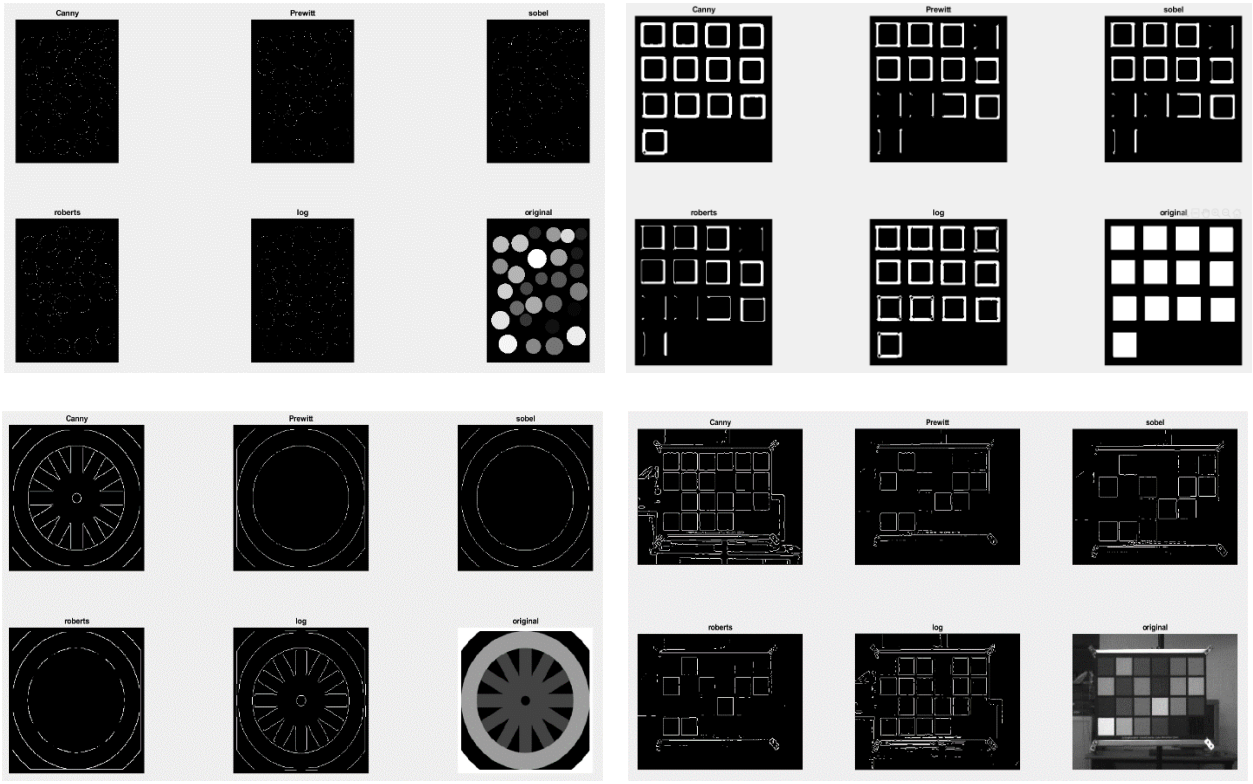


Fig 1: 4 image outputs for object count

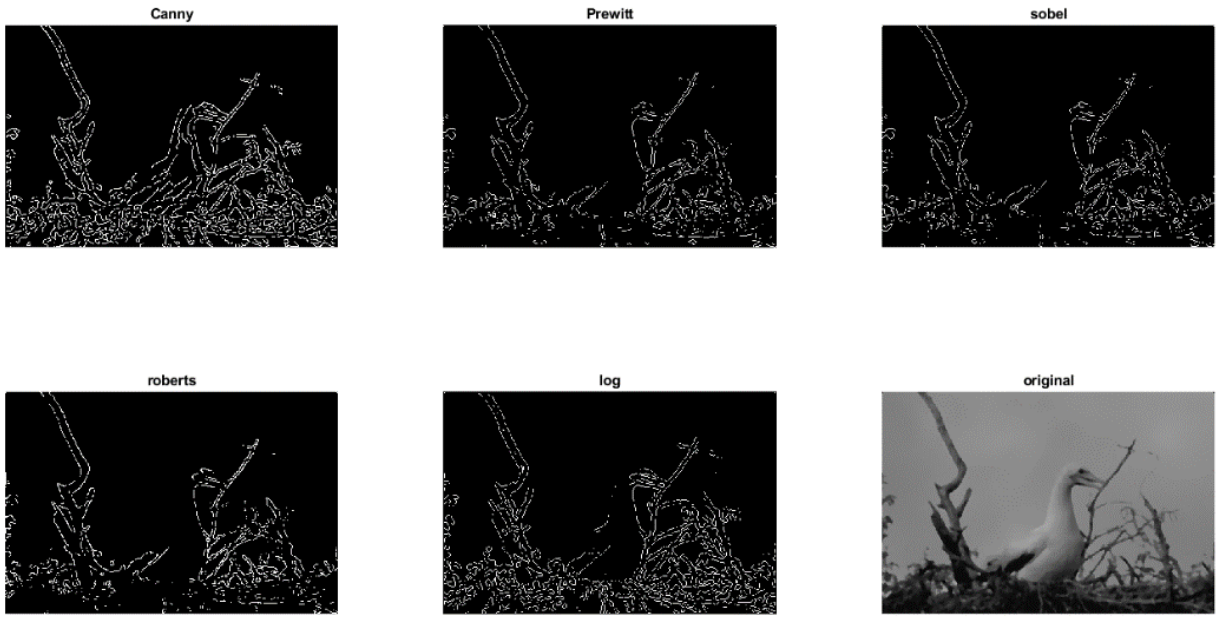


Fig 2: Image 1

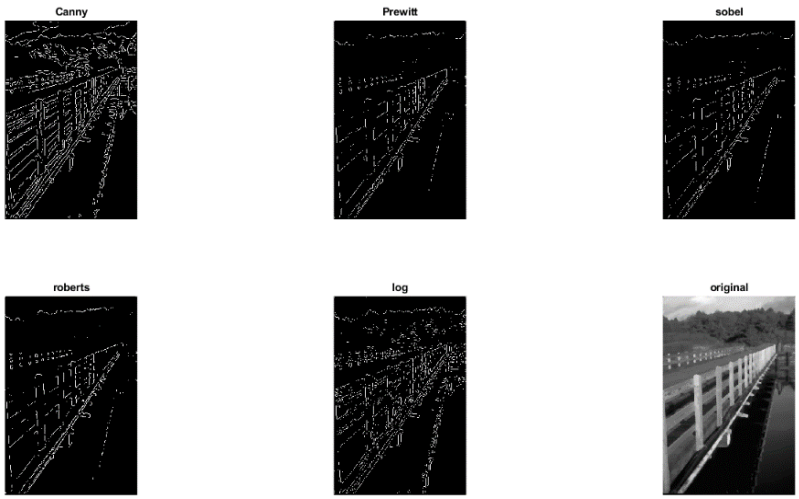


Fig 3: Image 2

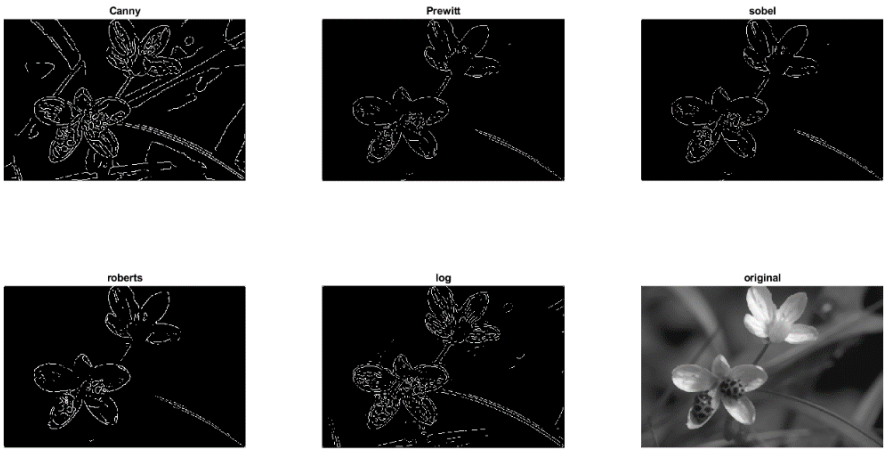


Fig 4: Image 3

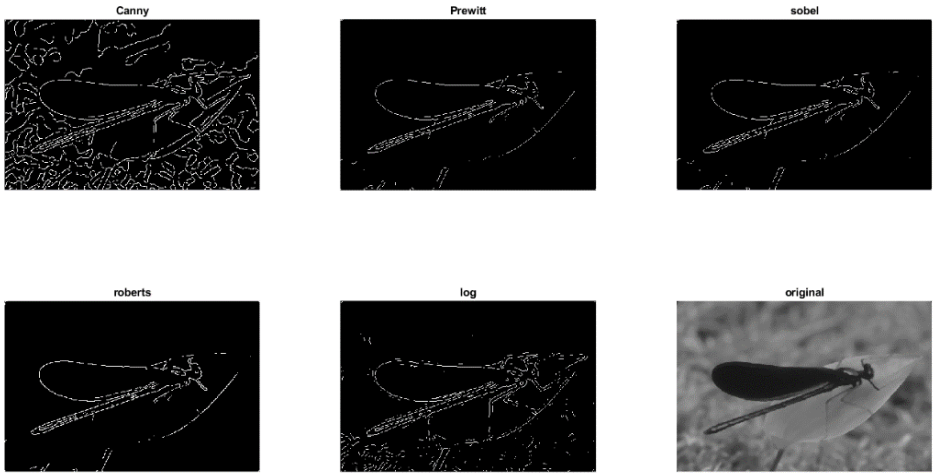


Fig 5: Image 4

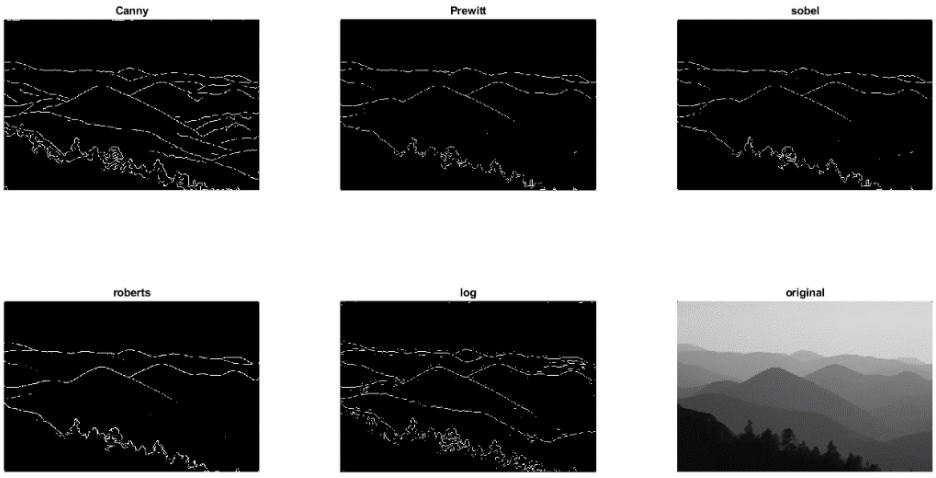


Fig 6: Image 5

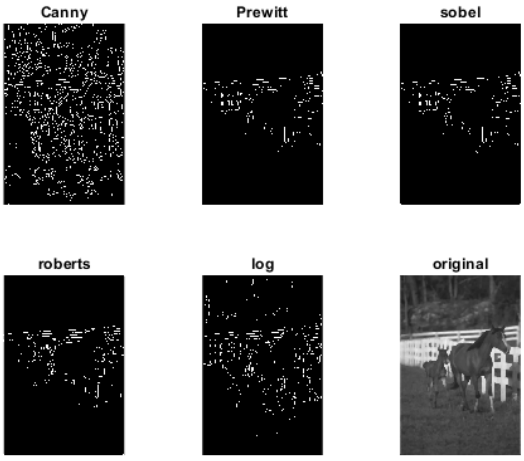


Fig 7: Image 6

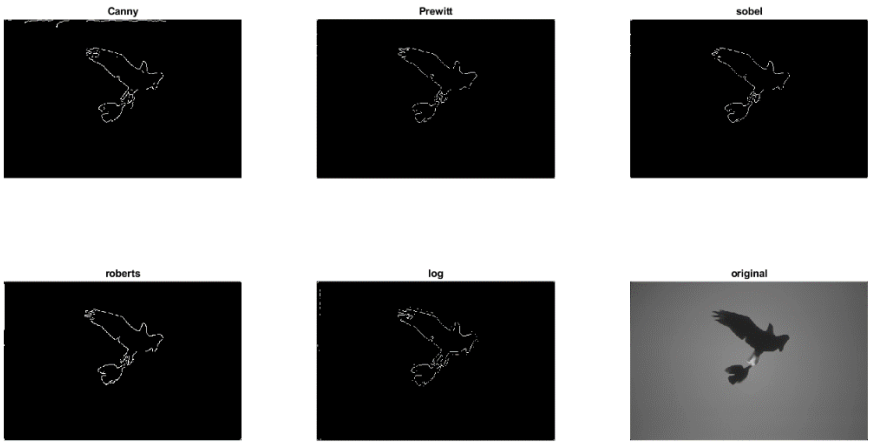


Fig 8: Image 7

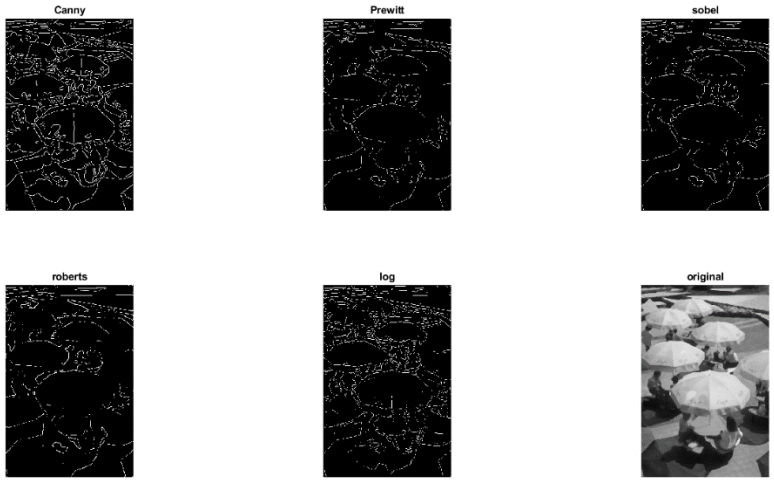


Fig 9: Image 8

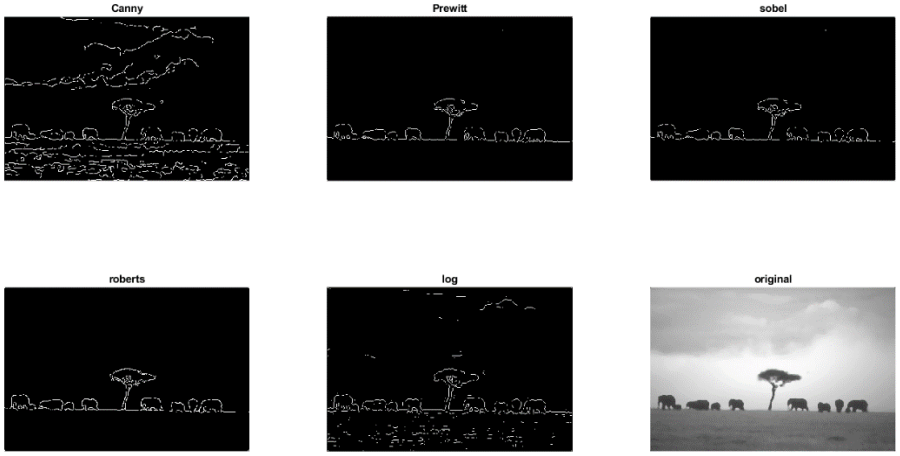


Fig 10: Image 9

7.Conclusion:

Edge detection is a fundamental principle on which lots of image processing work must be based. An attempt has been made in this work to offer a quick review of some of the most reliable, well-known, and effective edge detection approaches which concludes the performance comparison and why canny is better among these edge detection techniques. Future scope for this work is to find out the weak points for each technique and find a strategy to get better outcome from these techniques of edge detection which will be experimental and based on results.

8.References:

1. Microsoft Word - 059_ICONGETECH (iop.org)
2. E9957069520.pdf (ijeat.org)
3. Performance Evaluation of Edge Detection Using Sobel, Homogeneity and Prewitt Algorithms (scirp.org)
4. R2021b - Updates to the MATLAB and Simulink product families - MATLAB & Simulink (mathworks.com)
5. Question : How to calculate mean-square error(MSE) and PSNR values for edge detected images (titanwolf.org)
6. The Berkeley Segmentation Dataset.
7. Image Quality Assessment through FSIM, SSIM, MSE and … https://file.scirp.org › Html (google.com)
8. Compute peak signal-to-noise ratio (PSNR) between images – Simulink (mathworks.com)
9. Perception based Image Quality Evaluator (PIQE) no-reference image quality score - MATLAB piqe (mathworks.com)
10. Entropy (information theory) - Wikipedia
11. Entropy of grayscale image - MATLAB entropy (mathworks.com)
12. python - How can I count the number of edges after applying Canny edge detection? - Stack Overflow
13. How can we count the number of edges in an edge detected ... https://www.researchgate.net › post › How_can_we_count... (google.com)
14. How to count Image Edges with Different between space ? - (mathworks.com)