JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

EDGE DETECTION WITH DIFFERENT PARAMETERS IN DIGITAL IMAGE PROCESSING USING GUI*SUBTITLE IF NEEDED*

Mr. Anupam Singh Dilwalia, Dr. A. P. Chaudhari, Prof. G. K. Mahajan

Student, Head of Department, Professor
Electrical Department
Dr. Babasaheb Ambedkar Technological university, lonere, INDIA

Abstract: Edge detection stands as a cornerstone in image processing, serving to delineate the boundaries of objects within an image. Its significance reverberates across various domains such as image processing, computer vision, and machine vision, facilitating tasks like image segmentation and data extraction. An edge signifies a shift in intensity within an image, while edge detection involves partitioning the image into distinct regions of discontinuity. In the realm of research, various operators like Sobel, Perwitt, Roberts, Log, and Canny are employed to identify edges in digital images. These operators are compared using metrics like Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Signal-to-Noise Ratio (SNR), revealing the superiority of the Canny operator for edge detection. Notably, the Canny operator's multi-stage approach enables simultaneous edge detection and noise reduction. The study also explores applications utilizing edge detection alongside a further analysis facilitated through Matlab GUI.

I. INTRODUCTION

Essentially, edges denote crucial local variations in an image's intensity, serving as distinct markers between objects and their backgrounds. In the domain of image processing, edge detection assumes a pivotal role. It involves dividing a digital image into segments to discern these edges.”

Identifying regions of discontinuity not only reduces the image size but also retains its properties without loss. Since object structure is often discerned through lines and edges, edge detection plays a pivotal role in processing and feature extraction. By isolating edges within an image, we can capture the variations in intensity levels caused by differing light intensities across different object surfaces [1]. The primary objective of edge detection is to extract edges from an image, enabling the measurement, detection, and localization of local grayscale changes within the image [2].

Edge detection is a fundamental step in image processing, accomplished through various operators such as Sobel, Perwitt, Roberts, Log, and Canny. Each operator operates on different digital images, yielding distinct outputs that necessitate comparison. Additionally, operators like the Gaussian and LoG serve to smooth the image, while the Laplacian operator identifies edge locations based on zero crossings within the image [1] [3].

The paper is structured as follows: Section II delves into the Literature Review, Section III presents the Proposed System Model, Section IV elaborates on Application using edge detector operator, Section V showcases the Experimental Results obtained from test images, and finally, Section VI encapsulates the Conclusion.

II. LITTERATURE REVIEW

Authors M. Khairudin and Dessy Irmawati have presented their research on comparing USG images using edge detection. Through their analysis, they demonstrated that the Sobel edge detector exhibits superior accuracy compared to various other operators. Consequently, their findings establish that the segmentation process on USG images achieves the lowest Mean Squared Error (MSE) when employing the Sobel operator [4].

Zhang Jin, Yu Chen Yan, and Huang Xiang introduced an enhanced algorithm focusing on thresholding through the Sobel edge detection operator. Their approach integrates Genetic Algorithms with an improved Sobel algorithm for image processing in thresholding applications. Their research concludes that this novel automatic thresholding algorithm surpasses the effectiveness of the conventional Sobel edge operator [5].

Authors Ashutosh Sharma, Mohd Dilshad Ansari, and Rajiv Kumar presented their research on edge detection. They compared various methods, including their proposed one, utilizing parameters like Mean Squared Error (MSE) and Peak Signal- to-Noise Ratio (PSNR). Their findings indicated that higher PSNR and lower MSE were achieved with their proposed method, ultimately establishing its superiority over the other three edge detection techniques [6].

Mrityunjay Kumar Ray, Deboleena Mitra, and Sankhadip Saha conducted research on edge detection in digital images using a novel method. According to their study, their approach provides a faster execution time while maintaining both smoothness and sharpness compared to conventional edge detection methods. They conducted comparisons between images processed with traditional edge detector operators and their proposed technique, demonstrating the superior efficiency of their method over other operators [7].

Nadia Ben Youssef, Aicha Bouchid, and Noureddine Ellouze introduced a color image detection method based on Multiscale utilizing Gaussian function combined with wavelet transform. Their method employs the first derivative of the bidimensional Gaussian function to enhance edges while simultaneously reducing noise. They conducted comparisons of all results using ROC analysis [8].

Dr. Samuel Manoharan and Professor Sathish have presented their research in the medical field concerning lung cancer. They employed the conservative graphical cut method, which, however, did not yield accurate results. Consequently, they introduced a moderate approach as an alternative. The algorithm they developed achieved high accuracy results in edge detection for lung cancer ”[9].

Seyed Yahya Nikouei and Yu Chen introduced their research on real-time applications employing a lightweight Convolutional Neural Network (LCNN). Their work focuses on detecting pedestrians with a reduced computational workload on edge nodes. They implemented their solution on a Raspberry Pi 3 platform [10]

Dong Xiaoheng, Li Minghang, and Miao Jiashu presented their research on edge detection utilizing a wavelet transform edge extraction algorithm. Their method effectively eliminates false edges, providing accurate edge detection results [11].

Liying Yuan and Xue Xu have introduced their research on edge detection utilizing the Canny operator. Their approach employs local edge detection, which utilizes a distance-weighted average method based on the K-average method using the Canny operator [12].

Chinu and Amit Chhabra introduced their research on edge detection. They conducted a comparison between sequential and parallel edge detection methods across various metrics. Their findings revealed a significant performance gain of 68% in the parallel approach [13]

Qianyu Zhang, Nattha Jindapetch, and Dujdow Buranapanichkit addressed the issue of flooding, which poses numerous challenges to human life and property. To mitigate this problem, they proposed using image edge detection techniques for monitoring and detecting flooded areas. Their research concluded that Canny edge detection yielded the most accurate results [14].

Shaoqing Mo, Haiyun Gan, and Rui Zhang introduced their research, which is based on the adaptive threshold method. They compared this approach with the traditional threshold method and observed a reduction in false results. Ultimately, the proposed method achieved the best results[15].

3.1PROPOSED SYSTEM

An edge constitutes the boundary or outer perimeter of an image. Edge detection methods are employed to detect these edges within an image. Essentially, edge detection involves partitioning an image into distinct regions of discontinuity, enabling the identification of various image features. This process not only reduces the data content within an image but also preserves its structural prop Two primary types of operators utilized for edge detection are the Gradient Operator and the Gaussian Operator. The Gradient operator can be categorized into three types: a) Sobel, b) Prewitt, and c) Robert d) log e) canny. These operators, falling under the gradient category, employ the first-order derivative method for edge detection.

3.2 Data and Sources of Data

For this study secondary data has been collected. From the website of KSE the monthly stock prices for the sample firms are obtained from Jan 2010 to Dec 2014. And from the website of SBP the data for the macroeconomic variables are collected for the period of five years. The time series monthly data is collected on stock prices for sample firmsand relative macroeconomic variables for the period of 5 years. The data collection period is ranging from January 2010 to Dec 2014. Monthly prices of KSE -100 Index is taken from yahoo finance.

A) SOBEL: This gradient operator employs the first derivative and is sensitive to noise. It identifies edges within an image along both horizontal and vertical directions, focusing on areas with a high gradient magnitude. Functioning as a discrete differentiation operator, it computes the image intensity function by utilizing masks in both vertical and horizontal directions. Below is the horizontal mask for the Sobel operator:

$$\begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix}$$

he vertical mask for the Sobel operator is depicted below:

$$\begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix}$$

From the masks shown above, it becomes apparent that they provide more information at the edges compared to the center, as evidenced by the presence of zeros at the center of the masks

B) PREWITT: The Prewitt operator resembles the Sobel operator, differing slightly in the values of their masks, which are simpler compared to Sobel. Both operators serve as discrete differentiation.

C)

operators. In this context, the masks possess certain properties: they must exhibit opposite signs, and the sum of the masks should equate to zero. By employing higher weights or values, more information about edges can be obtained, resulting in enhanced edge detection capabilities.

The vertical mask for the Prewitt operator is as follows:

$$\begin{matrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{matrix}$$

Horizontal mask of prewitt operator is as follows:

$$\begin{matrix} 1 & -1 & -1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{matrix}$$

D) ROBERTS: This operator is most suitable for binary images. However, it does not provide information about edge orientation; rather, it primarily emphasizes diagonal edge gradients and is sensitive to fluctuations. Functioning as a discrete differentiation operator, it calculates the sum of the squares of the differences between diagonally adjacent pixels.. The horizontal mask of Robert operator is as follows:

$$\begin{matrix} 1 & 0 \\ 0 & -1 \end{matrix}$$

The vertical mask of Robert operator is as follows:

$$\begin{matrix} 0 & 1 \\ -1 & 0 \end{matrix}$$

and the gradient magnitude of Robert operator is G= √Gx2+Gy2 Where Gx== Horizontal mask Gy == Vertical mask The Gaussian operator is divided into two types: a) LoG (Laplacian of Gaussian) and b) Canny operator. These operators, falling under the Gaussian category, employ a second-order derivative method.

E) LAPLACIAN OF GAUSSAIN (LOG): This operator serves the dual purpose of identifying edges and reducing noise in an image by smoothing it, thereby enhancing its resilience to noise. It applies the Laplacian to the second derivative of the image. Upon application to an image, it detects all zero crossings and applies a threshold to these crossings, thereby improving their effectiveness in edge detection. Essentially, locations where the second-order derivative crosses zero indicate areas with maximum intensity levels.

F) CANNY: This is a Gaussian-based edge detector that isolates edges within an image, detecting a broad spectrum of edges. Initially, it applies a Gaussian filter to the image to eliminate noise and smooth it. Then, it computes the gradient of the image by applying non- maximal suppression. Subsequently, edges are identified by employing hysteresis. The hysteresis thresholding involves factors for identification, such as low threshold value, high threshold value, or the width of the Gaussian, which determines the noise level of the image. In our paper, we employ various edge detection methods and implement them in realtime applications using a Graphical User Interface (GUI). We evaluate these methods using different parameters such as Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Signal-toNoise Ratio(SNR).

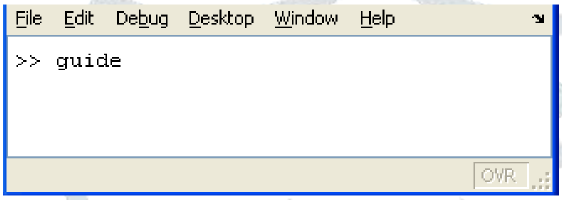
G) MSE: Mean Square Error (MSE) is used to measure errors in digital images, serving as one of the metrics for assessing image quality. A lower MSE indicates fewer errors and higher image quality, making it preferable to aim for images with low MSE.

H) PSNR: Peak Signal to Noise Ratio (PSNR) is the ratio between the signal and the noise, measured in logarithmic decibels. A high PSNR indicates lower noise levels, so a high PSNR value is always preferred. PSNR is used to assess image quality. Similarly, Signal to Noise Ratio (SNR) is the ratio between the power of the signal and the power of the noise. A high SNR is desirable for efficient results.

I) GRAPHICAL USER INTERFACE (GUI): In our paper, we employ various edge detection methods and apply them to real-time applications using a Graphical User Interface (GUI). A GUI is preferred because it is straightforward and makes the program easy for the end user to understand and interact with, allowing them to effectively utilize the application.

3.3 Data and Sources of Data

Step 1: Open MATLAB, navigate to the command window, and type `GUIDE`. The interface will appear as shown in the figure below.(1)



```
>> guide
```

Fig (1) Command window

Step 2: Select the first option, "Blank GUI (default)," as shown in the figure.(2)

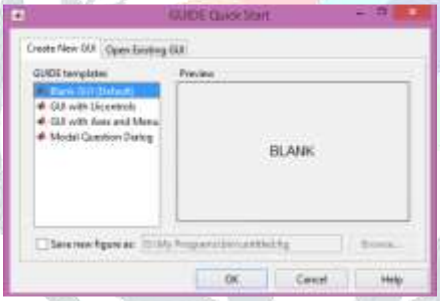


Fig (2) Blank GUI

Step 3: Now the screen will appear as shown in Fig (3). Here, we need to add components to this screen. This is where we will design our template according to our concept and determine how the GUI should look. Later, we can enhance the visual aspects of the GUI by selecting and adding various components

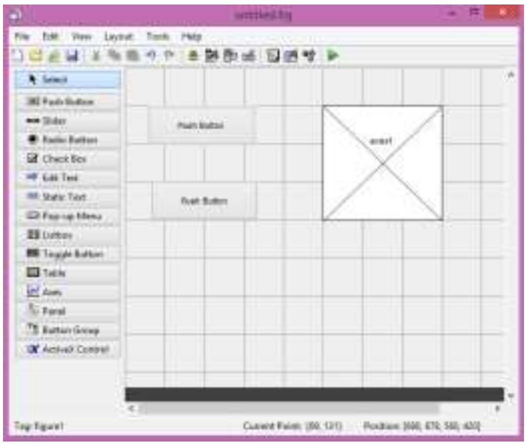
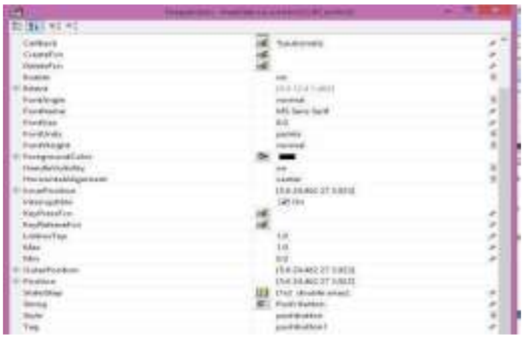


Fig (3) Basic screen of GUI with components

Step 4: To edit the properties of components, double-click on the component to open the Property Inspector, as shown in the figure.(4)



Callback %automatic
CreateFcn
DeleteFcn
Enable on
Extent
FontAngle normal
FontName MS Sans Serif
FontSize 8.0
FontUnits points
FontWeight normal
ForegroundColor
HandleVisibility on
HorizontalAlignment center
InnerPosition
Interruptible on
KeyPressFcn
KeyReleaseFcn
ListboxTop 1.0
Max 1.0
Min 0.0
OuterPosition
Position
SliderStep
String Push Button
Style pushbutton
Tag pushbutton1

Fig (4) Property inspector

Step 5: Now save the GUI with any file name. After saving, two files will be generated automatically: -` filename.fig`: This file contains the GUI layout. - `filename.m`: This file contains all the code for the GUI. Here, we need to write the code for all the components in the GUI, as highlighted in the red box in the figure. fig(5).

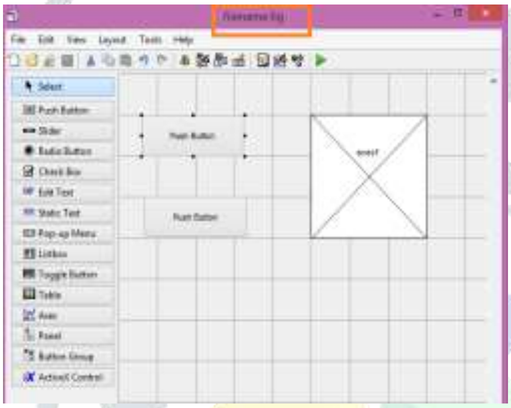


Fig (5) Saving file with name

Step 6: After writing code for GUI, now it needs to be to run in two ways-

First method: Click on therun button within the GUI editor.

Second method: Another method to open the GUI is by using the MATLAB command prompt. In the command prompt, type the name of the GUI file, and then open the file. In that interface, click on the run button, as demonstrated in the figure below (6).

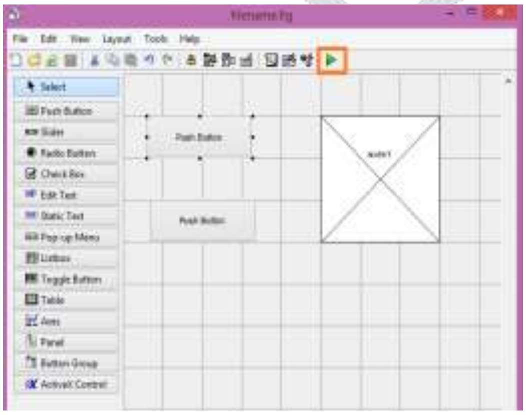


Fig (6) Execution of GUI

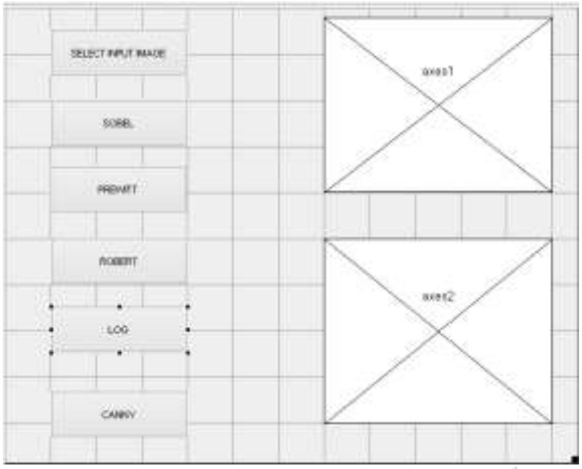


Fig (7) GUI structure of proposed system

3.4 Theoretical framework

Numerous applications utilizing edge detectors exist, yet in the current scenario, select real-time applications have been chosen and implemented using various edge detectoroperators via a Graphical User Interface (GUI).

A) FINGER PRINT: In real life, one of the major applications utilized by everyone is through mobile phones. People commonly secure their phones using fingerprint recognition. Here, we employ different edge detectors to accurately identify the edges of fingerprints, enhancing security and usability through a Graphical User Interface (GUI). Graphical User Interface (GUI). Edge detection methods play a pivotal role in enhancing image quality and facilitating image recognition

B) MEDICAL DIAGNOSIS: Another significant application of edge detection is in medical image diagnosis. Utilizing edge detectors, various human organs can be identified, aiding in the accurate delineation of anatomical structures. This method proves invaluable in 3D reconstruction during the pre-processing stage. Moreover, it facilitates the detection of various types of scans through the utilization of different edge detectors.

C) SATELLITE IMAGE: In this scenario, images captured by satellites from outer space, which depict the Earth's surface and other objects through remote sensing, are utilized. These images undergo edge detection, a crucial method in image analysis and the extraction of object boundaries within the images.

FLOWCHART

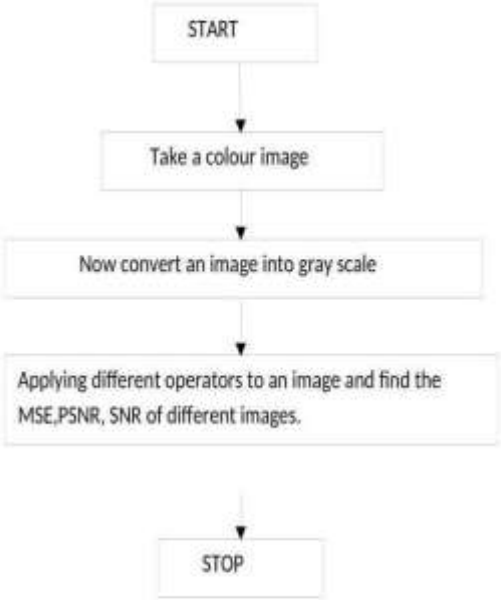

START
Take a colour image
Now convert an image into gray scale
Applying different operators to an image and find the MSE,PSNR, SNR of different images.
STOP

Fig (8) Flow chart of Edge detector

Calculation:

Calculating the magnitude of a gradient involves creating two gradient pictures, one for vertical changes (Sobel Y) and one for horizontal changes (Sobel X), after convolution. These pictures show how quickly the intensity shifts in each direction.

Combining Magnitudes: The Sobel Operator commonly combines the magnitude data from the vertical and horizontal gradients to produce a single gradient image

3.5 Comparison of the Models

The following figure displays the outcomes obtained fromvarious edge detector operators applied to different images. Additionally, these operators have been implemented in real-time applications for jpg & 1-JPG file format.

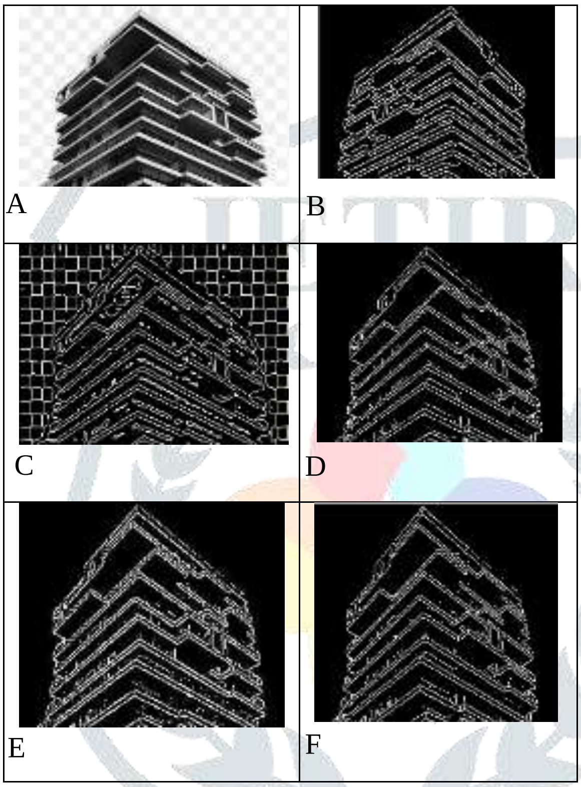
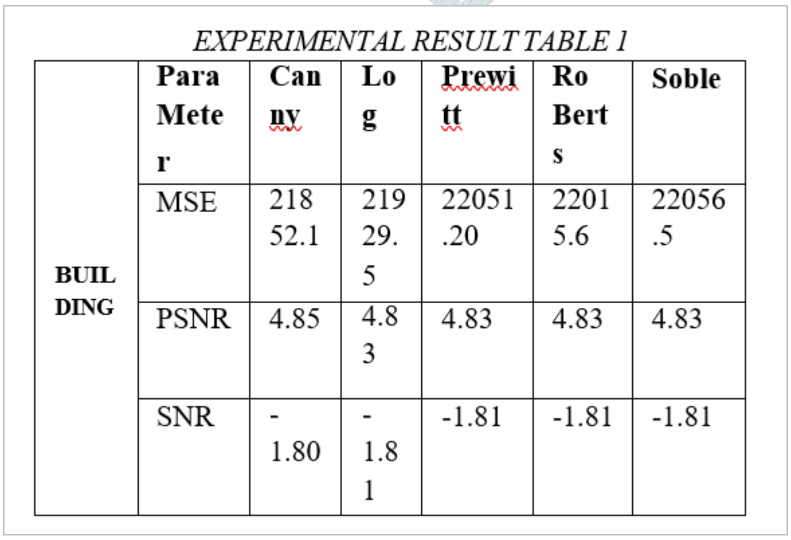


Table 1- Building (a) Original (b) Canny (c) log (d) Prewitt (e) Roberts (f) Soble



EXPERIMENTAL RESULT TABLE 1

	ParaMeter	**Canny**	**Log**	**Prewitt**	**RoBerts**	**Soble**
BUILDING	MSE	21852.1	21929.5	22051.20	22015.6	22056.5
	PSNR	4.85	4.83	4.83	4.83	4.83
	SNR	-1.80	-1.81	-1.81	-1.81	-1.81

EXAMPLE OF SOME APPLICATIONS OF EDGE DETECTOR

2-TIF file format

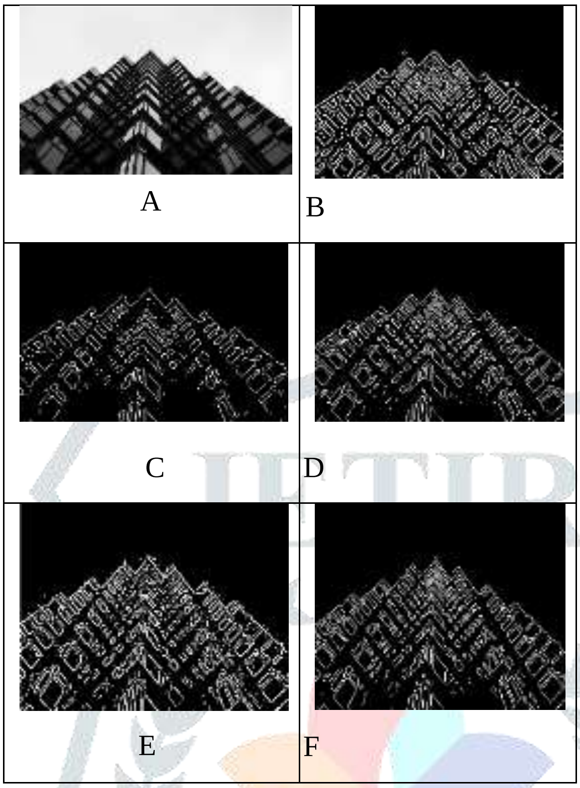


Table 2- Building (a) Original (b) Canny (c) log (d) Prewitt (e) Roberts (f) Soble

EXPERIMENTAL RESULT TABLE 2

	Parameter	**Canny**	**Log**	**Prewitt**	**Ro Berts**	**Soble**
BUILDING	MSE	21850.1	21925.5	22048.10	22014.5	22052.4
	PSNR	4.75	4.63	4.63	4.63	4.63
	SNR	-1.70	-1.61	-1.61	-1.61	-1.61

1- SATELLITE MAPJPG FORMAT

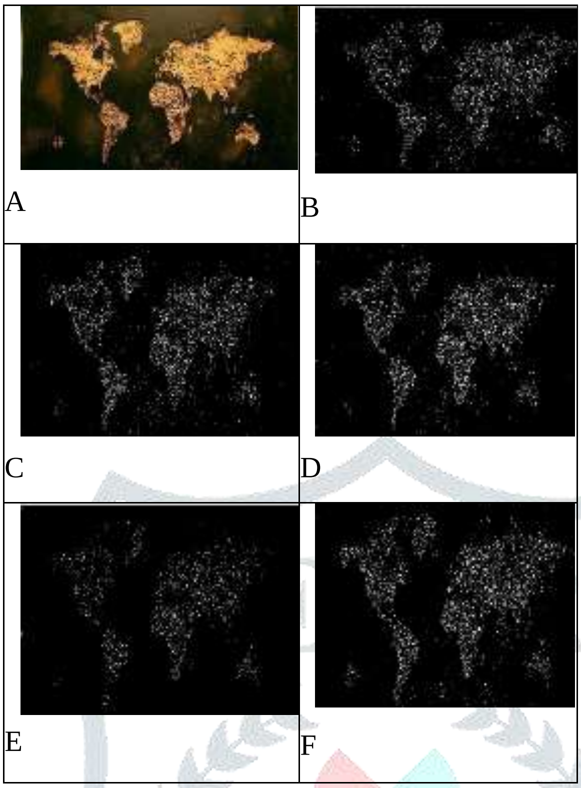


Table 3 -SATELLITE MAP (a) Original (b) Canny
(c) log (d) Prewitt (e) Roberts
(f) Soble

IV. RESULTS AND DISCUSSION

4.1 Results of Descriptive Statics of Study Variables

The proposed research work has extracted different edges from digital images using different edge detectors and it is implemented using Matlab GUI.

In our proposed work we have taken four different digital images like Building, Circuit board, Satellite using edge detector operators like Sobel, Prewitt, Roberts, LoG, Canny these are different operators have been implemented on these digital images and among all of these from the above output waveforms or pictures from Table 1 to Table 4, the difference in the output result can be observed. The Canny operator has low MSE and high PSNR and SNR is observed from table 1 to table 4, and finally it has concluded Canny operator as the best operator for detecting edges in an image.

Canny operator gives the best result when compared to other operators and this is implemented by using GUI in Matlab. Using this operators, some examples are considered in real time applications. In this real time application examples (Table 3 and Table 4) also by observing results, it is proved that the Canny operator is best among all other operators

V. ACKNOWLEDGMENT

I would like to express my deepest gratitude to everyone who contributed to the successful completion of this project on edge detection with different parameters in digital image processing using a graphical user interface (GUI).

First and foremost, I am immensely grateful to my advisor, Dr.A.P.Chaudhari and Prof. G.K.Mahajan for their invaluable guidance, support, and encouragement throughout the course of this project. Their profound knowledge and insightful feedback have been instrumental in shaping this work I also wish to extend my sincere thanks to my colleagues and friends who provided continuous support and motivation.

Their constructive criticism and suggestions greatly improved the quality of this project I am also grateful to Dr. Babasaheb Ambedkar Technological university, lonere ,for providing the necessary resources and a conducive environment for research and development. Last but not least, I would like to thank my family for their unwavering support and understanding during the entire period of this project.

REFERENCES

[1] Eswaran Perumal and Pramila Arulandhu, "Multi-Level Morphological Fuzzy Edge Detection for Color Images (MMFED)," presented at the International Conference on Electrical, Electronics, Communication, Computer, and OptimizationTechniques (ICEECCOT), 2017

[2] Dong Xiaoheng and Li Minghang, "Edge Detection Operator for Underwater Target Image," presented at the International Conference on Image Vision and Computing, IEEE, 2018.

[3] Lei Zhai and Honglian Ma, "Recent Methods and Applications in Image Edge Detection," presented at the InternationalWorkshop on Education Technology and Training, 2008.

[4] Dr. Samuel Manoharan and Prof. Sathish,"Improved Version of Graph-Cut Algorithm for CT Images of Lung Cancer with Clinical Property Condition," Journal of Artificial Intelligence andCapsule Networks, 2020.

[5] Seyed Yahya Nikouei and Yu Chen, "Real-TimeHuman Detection as an Edge Service Enabled by a Lightweight CNN," presented at the IEEE International Conference on Edge Computing,2018.

[6] Chinu and Amit Chhabra, "A Hybrid Approach for Color-Based Image Edge Detection," IEEE, 2014.

[7] Qianyu Zhang, Nattha Jindapetch, and Dujdow Buranapanichkit, "Investigation of Image Edge Detection Techniques for Real-Time Flood Monitoring," presented at the 16th ECTI-CON, 2019.

[8] Shaoqing Mo, Haiyun Gan, and Rui Zhang, "A Novel Edge Detection Method Based on Adaptive Threshold," presented at the 5th Information Technology and Mechatronics Engineering Conference (IOTEC), 2020.

[9] P.SRUJANA, J.PRIYANKA, V Y S S SUDIR PATNAIKUNI, NANCHARAIAH VEJENDLA, "Edge Detection with different parameters in Digital Image Processing using GUI", presented at the Fifth International Conference on Computing Methodologies and Communication, 2021