

# A Survey on Different Edge Detection Techniques for Image Segmentation

K. Bala Krishnan\* , Shiva Prakash Ranga and Nageswara Guptha

Department of ISE, Sri Venkateshwara College of Engineering, Bengaluru - 562157, Karnataka, India;  
Bala.k.btech@gmail.com, shivaprakashranga@gmail.com, mnguptha@yahoo.com

## Abstract

**Objectives:** To compare different edge detection techniques for the effective performance results with respect to complex images. **Methods:** This paper compares and analyses the performance of diverse edge detection techniques for recent digital image processing like canny optimal edge detection, sobel operator, prewitt operator and LoG. **Findings:** Canny Optimal Edge detection (COE) technique provides better performance results under noise images or complex images when compared to other techniques. In COE the result of PSNR is high which helps to identify the effective edges.

**Keywords:** Canny edge detection, Classical Operator Edge detection, Edge detection algorithm, Image segmentation.

## 1. Introduction

In the current scenario, the digital images play vital role in different applications such as restorations and enhancements, digital cinema, image transmission and coding, color processing, remote sensing, robot vision, hybrid techniques, facsimile, pattern recognition, registration techniques, multi-dimensional image processing, video processing, high resolution display, high quality color representation and super high definition applications. Image segmentation process is important for these digital image processing applications because the raw images are captured by the digital camera or mobile camera with nonessential background scenes or noise. The elimination of background images and noise is quite important to get accurate results. This process is a low level image engineering process which converts the raw images into segments are pixels or objects. These pixels are converted into vectors and analyzed or tested with the any one of the image segmentation process. The removal of noise from the images is performed using de-noising techniques such as filtering, enhancement, detection and localization for identifying the edges. These edges are analyzed with the help of mid and high – level image engineering processing methods. The adequate

edges are identified by diverse edge detection techniques in several image processing applications such as object recognition, motion analysis, pattern recognition, computer- guided surgery, finger print recognition, automatic traffic controlling systems, anatomical structure and image processing. Detecting the edges from noisy images or corrupted images is difficult in nature. In the past two decades' several edge detection techniques or algorithms are proposed, based on that the effective edges are evaluated or analyzed. The ultimate reason behind in these methods to restrict the false detection in the edges, edge localization and computational time. In this, canny optimal detection algorithm aims to discover the optimal edge which reduces the probability of detecting false edges, and gives sharp edges.

The rest of this paper is organized as follows: Section 2 discusses the purpose of providing image segmentation. Section 3 focuses on diverse edge detection techniques or algorithms and comparison between detection methodologies. Section 4 presents the conclusion.

## 2. Image Segmentation

Image segmentation is the method to simplify the digital image into segments or pixels which are easier to analyze

\* Author for correspondence

and identify the effective edges in a complex image<sup>1</sup>. The image engineering processes are sub divided into low-level, mid-level, high level which is shown in Figure 1. In low-level engineering process, the raw image is taken as an input and the noises are eliminated. These raw images are transformed into pixels. The pixel is a collection of discrete cells in the particular image. The characteristics of color and texture are found as similarity in a pixel. Generally, the raw images are taken as a colored image which is needed to be modify into grayscale or black and white images, because the edges can be detected using the pixels. In the Mid-level, the output is presented in the form of attributes like edges, contours, and the identity of individual objects. In the High-level engineering process, involves making sense, of a recognized object in the image analysis and it performs the cognitive functions associated with computer vision.



**Figure 1.** Image engineering process.

The De-noising techniques available in low level image segmentation are<sup>2</sup>:

- Region Based.
- Edge Based.
- Threshold.
- Feature Based Clustering.
- Model Based.

In the Region based technique, pixels or segments are grouped together and the edges are identified. Each and every pixel in the regions are identified and considered to analyze the images. Finally, the pixels are transformed in the form of vectors. It is sometime called as Similarity based segmentation.

This paper focuses on edge detection based image segmentation. In this, edges are found between two equivalent regions in an image. The process or method to detect the adequate edges in an image is called the edge detection. The identified pixels are ready to process in edge detection algorithm.

The edge detection algorithm contains four steps namely:

- Filtering.
- Enhancement.
- Detection.
- Localization.

Filtering is an initial step or technique for transforms an image. It is used to highlight the features or remove some features in the image.

Image enhancement is the process of modify the digital images into pixels for analysis. The following operations are performed eliminate noise, sharpen an image, or brighten an image which is making it easier to identify key features of an image. It is used to measure the gradient magnitude with respect to pixels.

Detection is the process of identify the adequate edges from the pixel.

Localization which is used to analyze the resolution with respect to sub pixel.

These steps are followed one by one in order to predict the edges effectively. Commonly segmentation process uses the adaptive thresholding method<sup>3</sup>. But it is not suitable for the complex or noise images.

In Feature Based Clustering, the images are transformed into histogram and then clustering is performed. The pixels are identified using Histogram and detected by thresholding method. K-means algorithm is used for clustering the texture images. The color images or ordinary images of the pixels are clustered through Fuzzy C method. The complex images are taken to the fragmentation. The technique is successful proved for identifying the color region.

Model based segmentation is also known as Markov Random Field (MRF). A Built in feature is available in the MRF for the color images. MRF is used to identifying the edges accurately with respect to complex images.

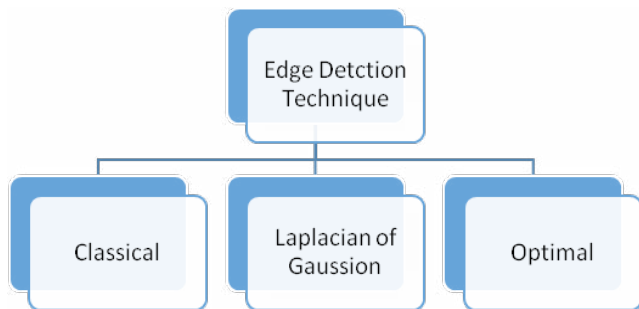
As shown in Table 1, edges are important feature in an image. Finding the effective edges with respect to noisy or corrupted images are difficult due to lighting conditions, background, luminance and geometrical features, noise volume, missing to detect existing edges and false edge. These techniques are classified with respect to pixels and sub pixels. Early edge deduction techniques like classical operators (Sobel operator, Prewitt, Robert's cross and Kirsch operators) all are used to compute the first derivative to detect the edges. Laplacian Gaussian operator is used to compute the second order gradients of an image. The Gaussian or optimal edge detection focuses on finding the sharp edges with fixed characteristics in all directions and sensitive to noise<sup>3</sup>.

**Table 1.** Uses of edge detection algorithm

S.No	Method	Uses
1	Filtering	Transform an image
2	Enhancement	Measures the gradient magnitude with respect to pixels.
3	Detection	Finds the exact edge point in the image
4	Localization	Analyzes the resolution with respect to sub pixel.

### 3. Edge Detection Techniques

The major classification of the edge detection technique or algorithm is given Figure 2.



**Figure 2.** Edge detection methods.

These techniques are analyzed and compared with respect to performances are given.

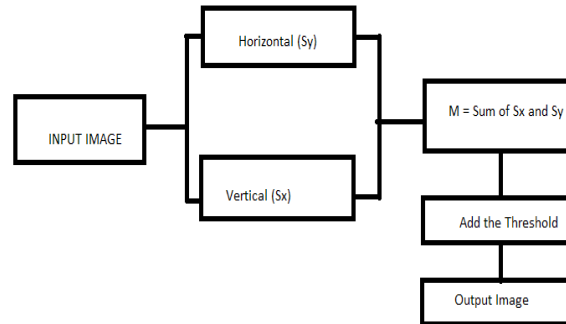
#### 3.1 Classical Operator

The evaluation of first order gradient edge detection techniques are well known as classical operator. Sobel operator, Prewitt operator, Robert’s cross operator and Kirsch edge detection are the classical edge detection techniques. These techniques are used to find the effective edges with respect to noisy images. It looks for minimum and maximum value of first level gradients. The advantage of classical operator is edge detection with respect to noise image is high and it is very simple to use. The main drawback of these techniques is inaccurate with complex images.

##### 3.1.1 Sobel Operator

In this, set of 3\*3 convolution kernels or mask are used<sup>1</sup>. Maximum of edges are identified with respect to perpendicular angle. In this, one kernel is allowed to rotate with 90 degrees and another one kernel is stay on this position.

The process of Sobel operator is shown in Figure 3. The raw image is taken as an input. It detects two types of edges like Horizontal edges (Sy) and Vertical Edges (Sx). The Magnitude (M) is calculated by adding the partial derivatives of Sx and Sy. The threshold is added with the magnitude to get the output image.



**Figure 3.** The process of sobel operator.

The Magnitude of the gradients (M) are calculated with Sx, Sy (Sx indicates the Vertical and Sy indicates the Horizontal position). The labeling of the neighborhood pixels are represented in Figure 4<sup>4</sup>.

The Sx, Sy partial derivations are given below<sup>5</sup>  
 $M = \sqrt{Sx^2 + Sy^2}$

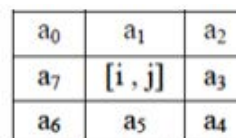
The angle of orientation of the edge is given by:

$$\theta = \arctan(Sy / Sx)$$

Where  $\theta$  is angle to find the direction.

$$Sx = (a2+Ca3+a4) - (a0+Ca7+a6)$$

$$Sy = (a0+Ca1+a2) - (a6+Ca5+a4)$$



**Figure 4.** Labeling of neighborhood pixels.

Where the threshold C=2,  $\theta$  is the angle and a0, a1, a2, a3, a4, a5, a6, a7 are masks.

Sx, Sy are implemented by using the following convolution mask which is shown in Figure 5 and 6. In image processing, a kernel or convolution matrix or mask is a small matrix.

-1	0	1
-2	0	2
-1	0	1

**Figure 5.** Sx convolution mask.

1	2	1
0	0	0
-1	-2	-1

Figure 6. Sy convolution mask.

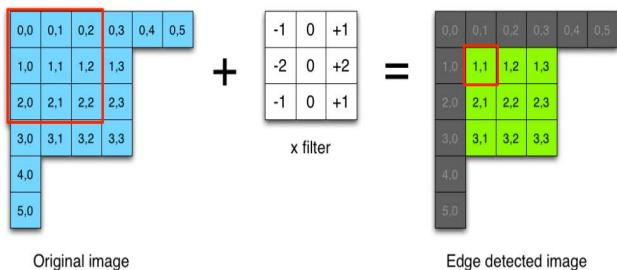


Figure 7. Conversion of original image into detected image.

An example of horizontal and vertical masks for edge detection is shown in Figure 7 and 8. The raw image is converted into masks and it is added with the Sx mask to find the deducted output. The output pixel for the convolution mask [1,1] is represented.

$$\text{output pixel } [1,1] = ([0,0] \times -1) + ([0,1] \times 0) + ([0,2] \times 1) + ([1,0] \times -2) + ([1,1] \times 0) + ([1,2] \times 2) + ([2,0] \times -1) + ([2,1] \times 0) + ([2,2] \times 1).$$

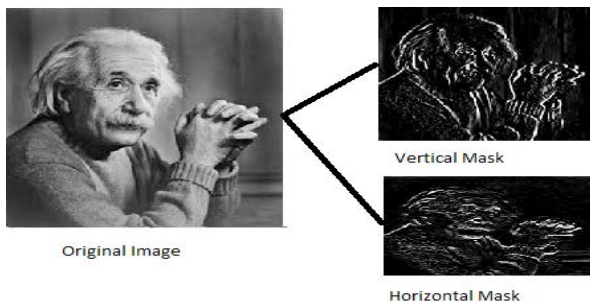


Figure 8. Horizontal and vertical mask.

### 3.1.2 Prewitt Operator

Prewitt operator is quite similar to the Sobel operator with the difference of the C value is 1. The convolution mask contains 1, -1, and 0 which is shown in Figure 9 and 10. The main advantage of this technique is to provide a better performance on horizontal and vertical edges in the images and higher responses for noisy images. The operator should have the following properties:

- Both negative and positive values should be in the all convolution masks.
- The final results should be zero when sum is obtained. It is widely used technique to evaluate the magnitude

(M) of the edges.

The Sx, Sy partial derivations are given<sup>6</sup>

$$M = \sqrt{S_x^2 + S_y^2}$$

$$S_x = (a_2 + Ca_3 + a_4) - (a_0 + Ca_7 + a_6)$$

$$S_y = (a_0 + Ca_1 + a_2) - (a_6 + Ca_5 + a_4)$$

Where the threshold value C=1 and a0, a1, a2, a3, a4, a5, a6, a7 are masks.

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

Figure 9. Sx Convolution mask.

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

Figure 10. Sy convolution mask.

The deducted edge is performed and magnitude is calculated using the example which is shown in the Figure 11. The original image pixel 12 is taken and the magnitude is calculated with Sx mask and the same procedures are followed to calculate the Sy mask. The final value of Sy mask is 20.

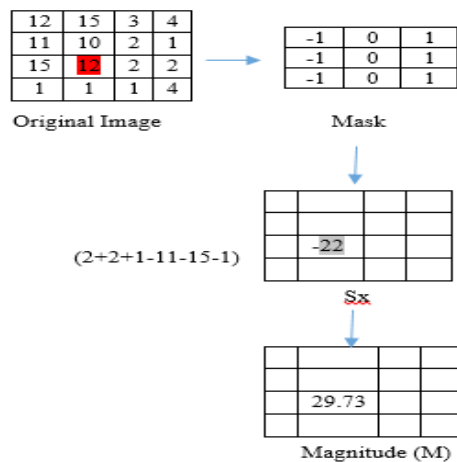
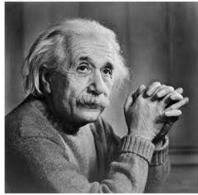


Figure 11. Example of calculating the magnitude with respect to Sx.

### 3.1.3 Kirsch Edge Detection

The Kirsch operator or Kirsch compass kernel is a linear edge that detects the maximum edge strength in a few predetermined directions. In this approach the single

mask is used to rotate with 45 degrees in the eight directions like North (N), North West (NW), West (W), South West (SW), South (S), South East (SE), East (E), and North East (NE) which is shown in the Figure 12. The main advantage of this approach is the maximum effective edges are found in each masks. The end mask is given



The Original Image



East (E) edges



North East (NE) edges



North (N) edges



North West (NW) edges



West (W) edges



South West (SW) edges



South (S) edges



South East (SE) edges

Figure 12. Kirsch edge detection masks.

$$E = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} \quad NE = \begin{pmatrix} -2 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

$$N = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & -1 \end{pmatrix} \quad NW = \begin{pmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{pmatrix}$$

$$W = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \quad SW = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{pmatrix}$$

$$S = \begin{pmatrix} 1 & 0 & 1 \\ 2 & 0 & -2 \\ -1 & 0 & -1 \end{pmatrix} \quad SE = \begin{pmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{pmatrix}$$

### 3.2 Laplacian of Gaussian

The laplacian Gaussians operator was introduced for finding the second level gradients for 2- D spatial derivate of an image. This operator normally deals with both input and output as gray scale images. In this approach, two operations are performed namely smooth the images and compute the laplacian.

#### Marr – Hildreth or LoG Edge Detection Algorithm<sup>7</sup>

- Smooth the raw image by Gaussian filter.
- Apply the laplacian(G) to the smoothed image.
- Smoothed image is done by the Equation (1)

$$S = g * I \tag{1}$$

Where S – smoothed image, g – Gaussian, I – image.

- Find the laplacianis done by the Equation (2)

$$\nabla^2 S = \frac{\partial^2 s}{\partial x^2} + \frac{\partial^2 s}{\partial y^2} \tag{2}$$

Where  $\nabla$  is second derivative gradient with respect to x and y coordinates and  $\Delta$  is laplacian.

The laplacian of Gaussian is derived by equation (3),

$$\Delta^2 S = \Delta^2 (g * I) = (\Delta^2 g) * I \tag{3}$$

Where  $\Delta^2 (g * I)$  indicates the process for input image and performs the Gaussian filter and  $(\Delta^2 g) * I$  indicates the process of takes the Gaussian and find the laplacian.

The first order and the second order derivation of Gaussian is calculated by the following Equation (4) and (5) and the mask is shown in Figure 13.

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-m)^2}{2\sigma^2}} \tag{4}$$

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x)^2+(y)^2}{2\sigma^2}} \tag{5}$$

$\sigma$  = Standard deviation, when the  $\sigma = 1$  the one dimensional equation  $G(x)$  is shown in the Figure 14.

When the  $\sigma = 2$  the one dimensional equation  $G(x, y)$  is shown in Figure 15.

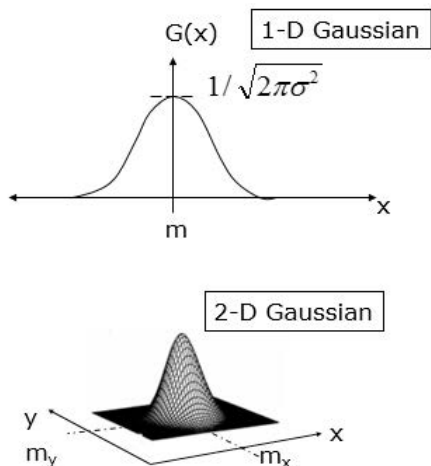


Figure 13. 1D and 2D Mask.

X	-3	-2	-1	0	1	2	3
G(x)	-.01	-.13	-.6	1	.6	.13	.011

Figure 14. One dimensional equation with respect to G(x).

0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	4	5	3	0	3	5	4	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	4	5	3	0	3	5	4	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

Figure 15. Second derivative detected output.

The advantage of the LoG edge detection method is used to evaluate the second order derivate edges of images. The laplacian operation is used to found the images either dark area or light area.

### 3.3 Gaussian

The Zero crossing operators performs the deduction of false zero crossing of an image and perform complex computation. Canny edge detection algorithm is widely used for optimal solutions. Robinson edge detection and Canny edge detection both techniques are the used to improve the optimized results on complex images.

#### Canny Edge Detection

This algorithm is focuses to separate the background noise from complex image and to find the effective edges for optimized solutions. It uses first derivative of an image. It is used to measures the mean square distance, error edge map and signals to noise ratio. The improved canny edge detection algorithm provides the better optimal solutions with respect to noisy images<sup>8</sup>.

Canny edge detection steps are given <sup>9</sup>

- Step1:** Smooth the image and gradient filter.
- Step 2:** Take the gradient of the image.
- Step 3:** Evaluate the edges using the gradient in the x and y coordinates.
- Step4:** Trace the edges with respect to x and y coordinates.
- Step5:** Non – maximum suppression.
- Step6:** Hysteresis.

#### Edge Detection Performance

The evaluation of detecting edge performance includes<sup>10</sup>

- Find the feasibility of false edges.
- Calculate the Possibility of missing edges.
- Mean and square distance of the edge estimate from the true edge.
- Tolerance to distorted edges and other features such as corners and junctions.

The first two criteria relate to edge detection, the second two to edge localization, and the last to tolerance to departures from the ideal edge model.

The performance is evaluated using the following parameters such as Performance Ratio (PR), Miss Count (MC), Peak Signal to Noise Ratio (PNSR), and Figure of Merit (FoM)<sup>11</sup>. Table 2 and Table 3 compare the effective performance of diverse edge detection algorithms<sup>3</sup>.

**Table 2.** Comparison of various edge detection techniques

S.no	Operators	Sensitivity	Operation	Advantage	Disadvantage
1	Sobel Operator	Medium	Maximum of edges are identified with respect to perpendicular angle	Simplicity Finding of smooth edges	In accurate Average results with respire to complex images.
2	Prewitt Operator		Provide a better performance on horizontal and vertical edges in the images		
3	Kirsch Edge Detection		The maximum effective edges are found in the each masks.		
4	LoG	Good	Considers the double edge images.	Finding the exact edges. Checks the pixels in wider area.	Few edges cannot be detected.
			Continuous edges can be detected using raw images		
5	Robert's cross operator		Find the 2 - D spatial gradient measurement on an image.		
6	Robinson Edge Detection		The maximum effective edges are found in the each masks.		
7	Canny Optimal Edge Detection	High	Used to eliminate the noise and to find the effective edges	Finding the error rate is high. Better detection Remove streaking problem. Adaptive in nature. Good localization	False Zero crossing.

**Table 3.** Performance of various edge detection techniques

Technique	PR	PSNR	FoM	MSE
Roberts	9.884	17.2154	0.083204	3934
Prewitt	9.5487	17.2104	0.094365	3922
Krisch	9.5627	17.2105	0.9536	3941
LoG	11.6908	17.2118	0.12455	3952
Canny	17.1933	17.2240	0.11672	3969

The PR and FoM is measured by following equations<sup>12</sup>

$$PR = \frac{\text{True edge pixels identified as Edge(S)}}{\text{False edge pixels identified as non - edge pixels}}$$

$$FoM = \frac{1}{\max(IA, I1)} \sum_{i=1}^{IA} \frac{1}{1 + d \alpha^2}$$

Where IA is the detected edges.

I1 is the Ideal edges.

d is the distance between actual and ideal edges.

$\alpha$  is the penalty factor for displaced edges.

To compare image compression quality MSE and PSNR is used.

The PSNR and MSE are measured by the following equations<sup>13</sup>

$$MSE = \frac{\sum |I1(m, n) - I2(m, n)|}{M * N}$$

Where M and N are rows and column of an image **I1** is considered as an original raw image and **I2** is a considered as a detected output image<sup>14</sup>.

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

R is the maximal variation in the input image data.

## 4. Conclusion

In this paper the various edge detection techniques for image segmentation are compared and their performances are analyzed with respect to various factors. From the survey, the canny optimal edge detection algorithm produces better result when compared to other techniques under sensitivity aspect. This paper will help the researchers in the area of digital image processing applications.

## 5. References

1. Beant K, Anil G. Mathematical morphological edge detection for remote sensing images. IEEE Transactions. 2011; 324-7.

2. Ireyuwa EI. Comparison of edge detection techniques in image processing techniques. *International Journal of Information Technology and Electrical Engineering*. 2013; 25–9.
3. Sharma A, Chaturvedi R, Chaturvedi U. Recent trends and techniques in image segmentation using practical swarm – Optimization – A survey. *International Journal of Scientific and Research Publications*. 2015; 1–6.
4. Khaire PA, Thakur NV. A fuzzy set approach for edge detection. *International Journal of Image Processing*. 2012; 403–12.
5. Kumar S, Singh I. Comparison between edge detection techniques. *International Journal of Computer Applications*. 2016; 15–8.
6. Lakshmi SJ, Prasanna. Image segmentation using various edge detection techniques. *International Journal of Emerging Technologies in Engineering Research*. 2016; 129–32.
7. Malik S, Kumar T. Comparative analysis of edge detection between gray scale and color image. *Communication on Applied Electronics*. 2016; 5(2):38–43.
8. Liu X, Dawn Z, Xu WWE. Image edge deduction algorithm based on improved wavelet transform. *International Journal of Signal Processing, Image Processing and Pattern Recognition*. 2016; 9(4):435–42.
9. Mohamed AZ, Reda RG, Sabah MA. Edge detection with a preprocessing approach. *Journal of Signal and Information Processing*. 2014; 5(4):123–34.
10. More MD, Andurker GK. Edge detection technique: A comparative approach. *World Journal of Science and Technology*. 2012.
11. Shuai MC. A new prediction method for edge detection based on human visual feature. *IEEE 24th Chinese Control and Detection Conference*; 2012. p. 1465–8.
12. Tobias OJ, Seara R. Image segmentation by histogram thresholding using fuzzy sets. *IEEE Transactions on Image Processing*. 2002; 11(12):1457–65.
13. Pradyna J, Usha J. Sobel edge detection implementation using FPGA. *International Journal of Science and Research*. 2015; 4(7):439–42.
14. Ramn M, Aggarwal H. Study and comparison of various image edge detection techniques. *International Journal of Image Processing*. 2009; 3(1):1–12.