



## A Survey on Various Methods of Edge Detection

Ms. suman<sup>1</sup>  
M.TechStudent  
CSE Department

Mr. pawan<sup>2</sup>  
Assistant Professor  
CSE Department

<sup>1,2</sup>South point Institute of technology & mgmt. , DCRUST University

**Abstract:** An edge can be described as the boundary between an object and the background in an image, and it also indicates the boundary between overlapping objects in an image. Edge detection methods are a combination of image smoothing and image differentiation plus a post-processing for edge labelling. The image smoothing involves filters that reduce the noise, regularize the numerical computation, and provide a parametric representation of the image that works as a mathematical microscope to analyze it in different scales and increase the accuracy and reliability of edge detection. The image differentiation provides information of intensity transition in the image that is necessary to represent the position and strength of the edges and their orientation. The edge labelling calls for post-processing to suppress the false edges, link the disspread ones, and produce a uniform contour of objects. Edge detection discusses the process of identifying and locating sharp discontinuities in an image. In this paper, the main aim is to survey the theory of edge detection for image processing using different types of techniques.

**Keywords:** Entropy weighted method, cellular automata, precursor

### I. INTRODUCTION

Edge is a part of an image that contains significant variation. The edges provide important visual information since they correspond to major physical, photometrical or geometrical variations in scene object. Physical edges are produced by variation in the reflectance, illumination, orientation, and depth of scene surfaces. Since image intensity is often proportional to scene radiance, physical edges are represented by changes in the intensity function of an image [2].

The most common edge types are steps, lines and junctions. The step edges are mainly produced by a physical edge, an object hiding another or a shadow on a surface. It generally occurs between two regions having almost constant, but different, grey levels. The step edges are the points at which the grey level discontinuity occurs, and localized at the inflection points. They can be detected by using the gradient of intensity function of the image. Step edges are localized as positive maxima or negative minima of the first-order derivative or as zero-crossings of the second-order derivative (Figure 1).

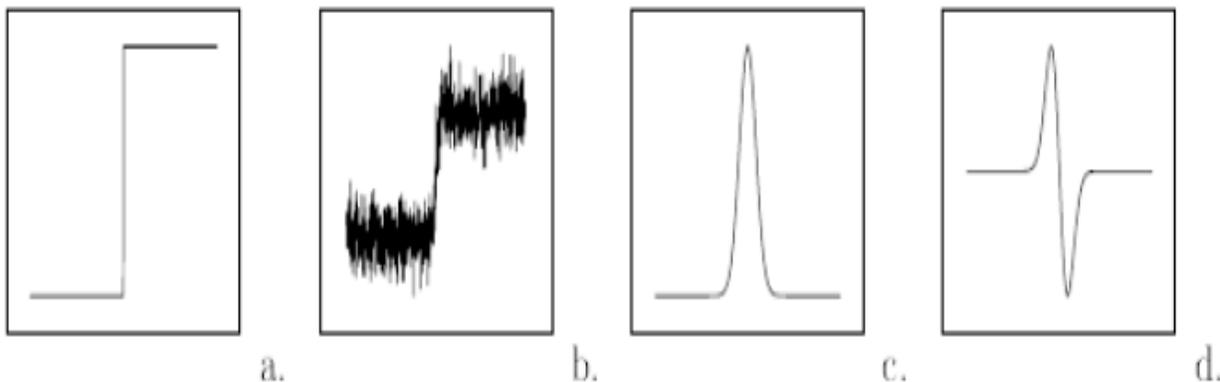


Figure 1 - profile of (a) ideal step edge (b) smoothed step edge corrupted by noise (c) first-order derivative (d) second-order derivative of the smoothed step edge corrupted by noise

It is more realistic to consider a step edge as a combination of several inflection points. The most commonly used edge model is the double step edge. There are two types of double edges: the pulse and the staircase (Figure 2).

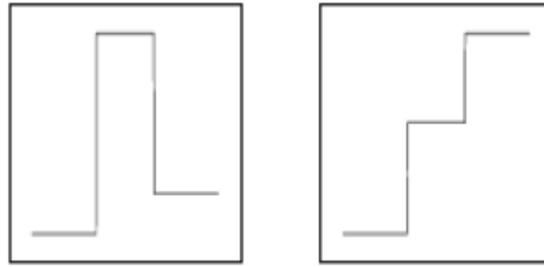


Figure 2 - profile of pulse (left) and staircase (right) step edges

Edge detection is a terminology in image processing that refers to algorithms which aim at identifying edges in an image. It is encountered in the areas of feature selection and feature extraction in Computer Vision. An edge detector accepts a digital image as input and produces an edge map as output. The edge map of some detectors includes explicit information about the position and strength of the edges and their orientation. The edge detection methods incorporate three operations: differentiation, smoothing and labelling. Differentiation consists in evaluating the desired derivatives of the image. Smoothing lies in reducing noise and regularizing the numerical differentiation. Labelling involves localizing edges and increasing the signal-to-noise ratio (SNR) of the detected edges by suppressing false edges. Labelling is often the last stage, but the order in which differentiation and smoothing are run depends on their properties. Smoothing and differentiation of an image are realized only by filtering the image with the differentiation of the smoothing filter.

## II. LITERATURE SURVEY

Shihu Zhu et. Al. [1] proposed the new method of edge detection based on multi-structure elements morphology and image fusion. Edges are detected using four different orientations SE (structure element) where direction angles of all the structure elements are  $0^\circ, 45^\circ, 90^\circ, 135^\circ$  and final edge result is got by image fusion using entropy weighted method. The proposed method not only can effectively eliminate the image noise, but also effectively maintain good edge information.

C.Naga Rajuet. Al. [2] propose an edge detection algorithm based on multi-structure elements morphology. The eight different edge detection results are obtained by using morphology gradient algorithm and final edge results are obtained by using synthetic weighted method. The proposed algorithm results are compared with the conventional mathematical morphological edge detection and differential edge detection operators such as Watershed method, Sobel operator and Canny operator and obtained the better edges over traditional methods.

Wenshuo Gao et. al. [3] proposed a method which combines Sobel edge detection operator and soft-threshold wavelet de-noising for edge detection. This method used on images which include White Gaussian noises. The widely used operators such as Sobel, Prewitt, Roberts and Laplacian are sensitive to noises and their anti-noise performances are poor. This paper proposes an edge detection method which combines soft-threshold wavelet de-noising and Sobel Operator, its anti-noise performance is very strong. Firstly soft-threshold wavelet used to remove noise, then Sobel edge detection used for edge detection on the image. The effect by using this method to do edge detection is very good and can remove the noise effectively.

Sabina Priya darshiniet. Al. [4] proposed a new technique of edge detection that employs simple additions and divisions and finds out fine edges. It makes use of a threshold that is computed automatically during the edge detection process and it's simple to compute the threshold value. It is based upon simple arithmetic and logic operations, consisting of three procedures: image binarization, image contraction and image subtraction. The proposed method is a computationally simpler and performs better than Sobel's method and require much lesser computation than Sobel's method.

Tapas Kumar et. Al. [5] proposed a new approach of edge detection based on cellular automata. The algorithm will correspond to edge detection for grayscale images. The proposed conception of cellular automata for gray levels of digital images is on the basis of bi-dimensional cellular automata. A result produced by Cellular Automata works satisfactorily for different gray level images and produce better edge detection effects as compared to Canny, Roberts, Prewitt and Sobel.

## III. EDGE DETECTION TECHNIQUES

### A. FIRST-ORDER DERIVATIVE EDGE DETECTION:

$$\nabla f = \begin{matrix} G_x \\ G_y \end{matrix} = \begin{matrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{matrix} \quad (1)$$

An important quantity in edge detection is the magnitude of this vector, denoted  $|\nabla f|$ , Where

$$|\nabla f| = \sqrt{G_x^2 + G_y^2} \quad (2)$$

Another important quantity is the direction of the gradient vector. That is,

$$\text{angle of } \nabla \mathbf{f} = \tan^{-1} \left( \frac{G_y}{G_x} \right) \quad (3)$$

Computation of the gradient of an image is based on obtaining the partial derivatives of  $\partial f/\partial x$  and  $\partial f/\partial y$  at every pixel location. Let the  $3 \times 3$  area shown in Fig. 3. represent the gray levels in a neighborhood of an image. One of the simplest ways to implement a first-order partial derivative at point  $z_5$  is to use the following Roberts cross-gradient operators:

$$G_x = (z_9 - z_5) \quad (4)$$

and

$$G_y = (z_8 - z_6) \quad (5)$$

These derivatives can be implemented for an entire image by using the masks shown in Fig. 4 with the procedure of convolution.

Another approach using masks of size  $3 \times 3$  shown in Fig. 5 which is given by

$$G_x = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3) \quad (6)$$

And

$$G_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7) \quad (7)$$

a slight variation of these two equations uses a weight of 2 in the center coefficient:

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \quad (8)$$

$$G_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7) \quad (9)$$

A weight value of 2 is used to achieve some smoothing by giving more importance to the center point. Fig.6, called the Sobel operators, is used to implement these two equations.

$z_1$	$z_2$	$z_3$
$z_4$	$z_5$	$z_6$
$z_7$	$z_8$	$z_9$

Fig. 3 A  $3 \times 3$  area of an image.

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

Fig. 4 The Roberts operators.

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

Fig. 5 The Prewitt operators.

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

Fig. 6 The Sobel operators.

IMPLEMENTATION OF FIRST-ORDER DERIVATIVE EDGE DETECTION:

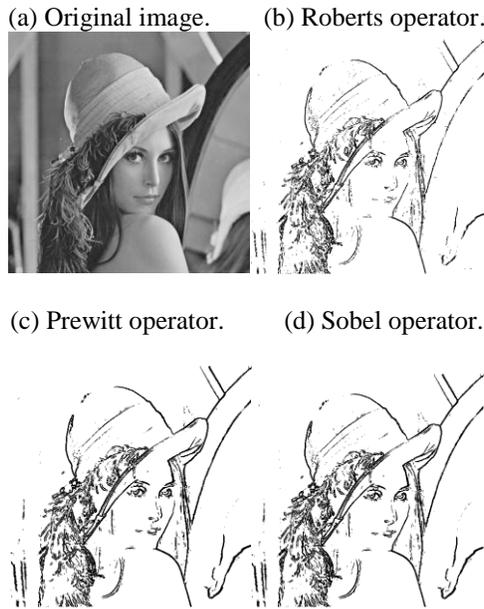


Fig.7 Examples of edge detection using three different operators.

Figure 7 shows Examples of edge detection using three different operators. Figure 7(a) shows original image of Lena. In figure 7 (b) Roberts operator is applied. Then in figure 7(c) Prewitt operator is applied on Lena image. And, in 7(d) Sobel operator is applied.

**B. SECOND-ORDER DERIVATIVE EDGE DETECTION**

The Laplacian of a 2-D function  $f(x, y)$  is a second-order derivative defined as:

$$\nabla^2 f = \partial^2 f / \partial x^2 + \partial^2 f / \partial y^2 \quad (10)$$

There are two digital approximations to the Laplacian for a  $3 \times 3$  region:

$$\nabla^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8) \quad (11)$$

$$\nabla^2 f = 8z_5 - (z_1 + z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9) \quad (12)$$

where the  $z$ 's are defined in Fig. 8. Masks for implementing these two equations are shown in Fig. 8.

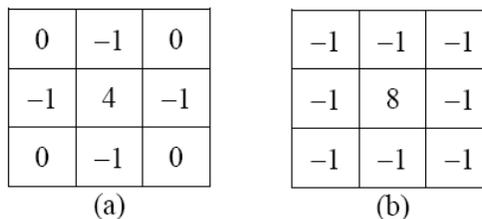


Fig. 8 Two kind of  $3 \times 3$  Laplacian mask.

The Laplacian is usually combined with smoothing as a precursor to finding edges via zero-crossings. The 2-D Gaussian function

$$h(x, y) = -e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (13)$$

where  $\sigma$  is the standard deviation, blurs the image with the degree of blurring being determined by the value of  $\sigma$ . The Laplacian of  $h$  is

$$\nabla^2 h(x, y) = -\left[ \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right] e^{-\frac{r^2}{2\sigma^2}} \quad (14)$$

This function is commonly referred to as the Laplacian of Gaussian (LOG).

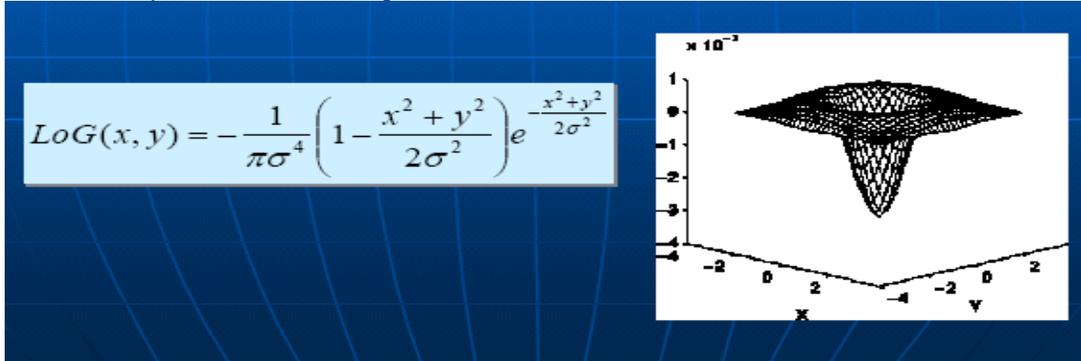


Fig. 9 3-dimension coordinate of Laplacian of Gaussian (LOG).

After calculating the two-dimensional second-order derivative of an image, we find the value of a point which is greater than a specified threshold and one of its neighbors is less than the negative of the threshold. The property of this point is called zero-crossing and we can denote it as an edge point. We note two additional properties of the second derivative around an edge: (1) It produces two values for every edge in an image (an undesirable feature); and (2) an imaginary straight line joining the extreme positive and negative values of the second derivative would cross zero near the midpoint of the edge. This zero-crossing property of the second derivative is quite useful for locating the centers of thick edges as shown in figure 10.

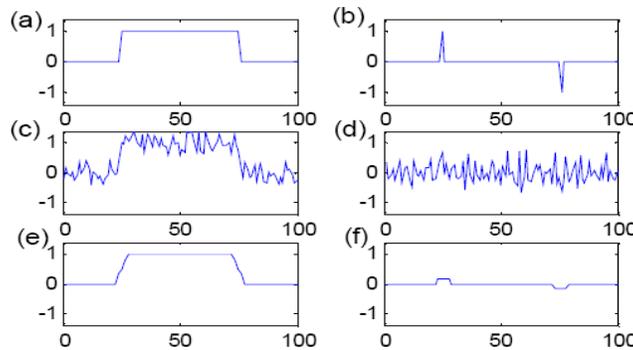


Fig. 10 Using differentiation to detect (a) the sharp edges, (c) the step edges with noise, and (e) the ramp edges. (b)(d)(e) are the results of differentiation of (a)(c)(e).

IMPLEMENTATION OF SECOND-ORDER DERIVATIVE EDGE DETECTION:

- (a) Original image. (b) Laplacian mask of **Error! Reference source not found.**(a)



- (c) Laplacian mask of **Error! Reference source not found.**(b) (d) LOG mask of (b)



Fig.11 Examples of edge detection using three different Laplacian masks.

Figure 11 shows Examples of edge detection using three different Laplacian masks. Figure 11(a) shows original image of Lena. In figure 11 (b) laplacian mask is applied. Then in figure 11(c) and (d) laplacian mask and LOG mask is applied on figure 11(b).

### C. Hilbert Transform for Edge Detection

There is another method for edge detection that uses the Hilbert transform (HLT) as shown in figure 12. The HLT is

$$g_H(\tau) = h(x) * g(x), \quad \text{where } h(x) = \frac{1}{\pi x} \quad (15)$$

and \* means convolution. Alternatively,

$$G_H(f) = H(f)G(f) \quad (16)$$

where  $G(f) = FT[g(x)]$  (FT means the Fourier transform),  $G_H(f) = FT[g_H(x)]$ , and

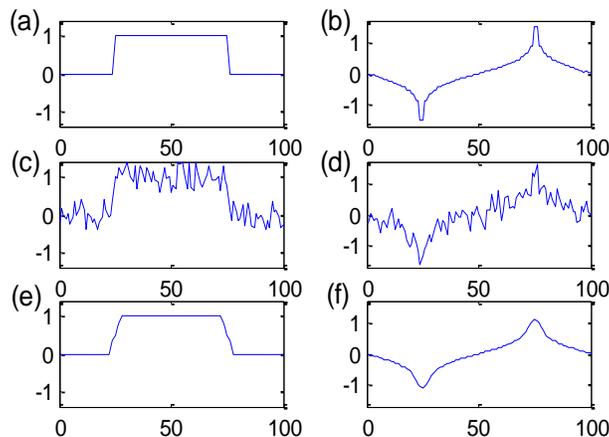
$$H(f) = -j \operatorname{sgn}(f), \quad (17)$$

where the sign function is defined as

$$\operatorname{sgn}(f) = 1 \text{ when } f > 0,$$

$$\operatorname{sgn}(f) = -1 \text{ when } f < 0,$$

$$\operatorname{sgn}(0) = 0 \quad (18)$$



**Fig. 12.** Using HLTs to detect (a) the sharp edges, (c) the step edges with noise, and (e) the ramp edges. (b)(d)(e) are the results of the HLTs of (a)(c)(e)

### D. Short Response Hilbert Transform for Edge Detection

Based on Canny's criterion, we develop the short response Hilbert transform (SRHLT), which is the intermediate of the original HLT and the differentiation operation. For edge detection, the SRHLT can compromise the advantages of the HLT and differentiation. It can well distinguish the edges from the non-edge regions and at the same time are robust to noise. We also find that there are many ways to define the SRHLT. If the constraints which come from Canny's criterion are satisfied, the resultant SRHLT will have good performance in edge detection.

The Definition of the SRHLT

We combine the HLT and differentiation to define the SRHLT. From the theorem of the Fourier Transform,

$$\operatorname{csch}(\pi x) \xrightarrow{FT} -j \tanh(\pi f) \quad (19)$$

where

$$\operatorname{csch}(x) = 2 / (e^x - e^{-x})$$

$$\tanh(x) = (e^x - e^{-x}) / (e^x + e^{-x}), \quad (20)$$

Therefore, from the scaling property of the FT:

$$g(bx) \xrightarrow{FT} |b|^{-1} G(f/b), \quad (21)$$

we obtain

$$|b| \operatorname{csch}(\pi b x) \xrightarrow{FT} -j \tanh(\pi f / b). \quad (22)$$

From 22, we can define the short response Hilbert transform (SRHLT) as:

$$\begin{aligned} g_H(\tau) &= h_b(x) * g(x), \quad \text{where } h_b(x) = |b| \operatorname{csch}(\pi b x) \\ G_H(f) &= H_b(f) G(f) \quad \text{where } G_H(f) = FT[g_H(\tau)], \\ G(f) &= FT[g(\tau)], \quad H_b(f) = -j \tanh(\pi f / b). \end{aligned} \quad (23)$$

(a) Original image      (b) Results of differentiation



(c) Results of the HLT      (d) Results of the SRHLT, b=8



**Fig.13** Experiments that use differentiation, the HLT, and the SRHLT (b=8) to do edge detection for Lena image.

Figure13shows the Experiments that use differentiation, the HLT, and the SRHLT (b=8) to do edge detection for Lena image.

#### E. Improved Harri's Algorithm for Corner and Edge Detections

Corner detection is important for feature extraction and pattern recognition. Harris and Stephens proposed a corner detection algorithm. First, they used a quadratic polynomial to approximate the variation around  $[m, n]$ :

$$\begin{aligned} |L[m+x, n+y] - L[m, n]|^2 &= \\ A_{m,n} x^2 + 2C_{m,n} xy + B_{m,n} y^2 + \text{remained terms} \end{aligned} \quad (24)$$

Where  $A_{m,n}$ ,  $B_{m,n}$ , and  $C_{m,n}$  were calculated from the correlations between the variations and a window function:

$$\begin{aligned} A_{m,n} &= X^2 \otimes w, \quad B_{m,n} = Y^2 \otimes w, \quad C_{m,n} = XY \otimes w, \\ X &= L[m, n] \otimes [-1, 0, 1], \quad Y = L[m, n] \otimes [-1, 0, 1]^T, \\ w_{x,y} &= \exp[-(x^2 + y^2) / 2\sigma^2]. \end{aligned} \quad (25)$$

Then the variations along the principal axes can be calculated from the eigen values of the following 2x2 matrix:

$$H_{m,n} = \begin{bmatrix} A_{m,n} & C_{m,n} \\ C_{m,n} & B_{m,n} \end{bmatrix} \quad (26)$$

If both the two eigen values of  $H_{m,n}$  are large, then we recognize the pixel  $[m, n]$  as a corner.

#### IV. CONCLUSIONS and FUTURE WORK

In this research, we have surveyed many methods of edge detections, such as first-order derivative edge detection, second-order derivative edge detection, HLT and SRHLT. SRHLT has higher robustness for noise than HLT and can successfully detect ramp edges. The SRHLT can also avoid the pixels that near to an edge be recognized as an edge pixel, which is usually an important problem when using the HLT for edge detection. We also studied that the SRHLT can successfully detect the edges of a complicated image. Moreover, directional edge detection (i.e., detect the edges with certain direction) are also the possible applications of the SRHLT. Although, improved Harris' algorithm can remove the drawbacks of SRHLT and is also helpful for increasing the robustness. So, it can be concluded that different methods that has been described, but they are not better in terms of edges detection as they are too long and complex for detection of edges from coloured image. Also they take much longer time to detect edges. In future, there is a need to propose a method that can first smooth the coloured image along with the boundaries and then identify the corners of each and every objects of that image. A synthetic method to detect

the edges of the images in RGB colour space is defined using Kuwahara filter to smoothen the image and then we will use Sobel operator to detect the edge. A new automatic threshold detection mechanism based on histogram data can be used.

#### **REFERENCES**

- [1] Shihu Zhu, "Edge Detection Based on Multi-structure Elements Morphology and Image Fusion", IEEE, 978-1-4244-9600-vol-6, 2011.
- [2] C.Naga Raju, S.Naga Mani, G.rakesh Prasad, S.Sunitha, "Morphological Edge Detection Algorithm Based on Multi-Structure Elements of Different Directions", International Journal of Information and Communication Technology Research Volume 1 No. 1, May 2011.
- [3] Wenshuo Gao, Lei Yang, Xiaoguang Zhang, Huizhong Liu, "An Improved Sobel Edge Detection", IEEE, 978-1-4244-5540-vol-9,2010.
- [4] Sabina Priyadarshini, Gadadhar Sahoo, "A New Edge Detection Method based on Additions and Divisions", International Journal of Computer Applications (0975 – 8887) Volume 9– No.10, November 2010.
- [5] Tapas Kumar, G. Sahoo, "A Novel Method of Edge Detection using Cellular Automata", International Journal of Computer Applications (0975 – 8887) Volume 9– No.4, November 2010.
- [6] Soo-Chang Pei, Jian-Jiun Ding, "Imporved Harris' Algorithm For Coner And Edge Detections", vol 1, 2005.
- [7] William K. Pratt , "Digital Image Processing \_William K. Pratt 3<sup>rd</sup>", chapter 15.
- [8] Jiun De Huang, "Image Compression by Segmentation and Boundary Description", chapter 2.
- [9] D. Ziou& S. Tabbone, "Edge Detection Techniques An Overview", 1998.