Edge Detection Techniques- A Critical Study

Bharti Chourasia[1], Dr Sanjeev Gupta[2]
[1,2]AISECT University, Bhopal (M.P.) India.

ABSTRACT

Edge detection is method of image processing. Edges define the boundaries and helpful to segment the object, registration, feature extraction and identification of objects in a scene and reduces the amount of data and filters out useless information so that edge detection is significant field in image processing. In this paper different method for edge detection are discuss. Edge detection methods for edge detection are Canny' s method, Laplacian of Gaussian (LOG), prewitt, Robert, Sobel. The advantages and disadvantages are also studied in this paper. Application area of edge detection is very wide i.e. from astronomy to photography, medicine to war etc.

Keywords- Edge Detection, Canny, LOG, Prewitt, Robert, Sobel.

I INTRODUCTION

Edge detection methods aim to identifying points in a image at which the image intensity changes sharply or more properly, has discontinuities. The points at which image intensity changes sharply are typically planned into a set of curved line segments termed edges. The similar problem of finding discontinuities in 1D signal is recognized as step recognition and the problem of finding signal discontinuities over time is known as change detection[5]. Edge detection is a primary tool in image processing, machine vision and computer vision, mainly in the areas of feature detection and feature extraction.

Gradient based Edge Detection- The gradient method finds the edges by searching for the maximum and minimum in the first derivative of the image. Laplacian based Edge Detection- The laplasian method looks for for zero crossings in the second derivative of the image to find edges[5]. An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location.

II CANNY'S DETECTION

The traditional Canny edge detection method is broadly used in gray image processing. Although, this traditional algorithm is not capable to deal with color images and the parameters in the algorithm are difficult to be determined adaptively. It was developed by John F. Canny in 1986. Canny also produced a computational theory of edge detection. Canny's aim was to discover the optimal edge and Canny have following criteria during edge detection process.

(a) **Good detection-** It means algorithm should mark maximum edges in the image.

(b) **Good Localisation-** It means algorithm should be mark as close as possible edge in the real image.

(c) **Minimal Response-** A given edge in the image only be marked once.

Canny algorithm is a step-by-step process and the steps are shown in figure. 1. In this methods Gaussian filter is applied on original test image to minimize the noise effect. Noise is unwanted information which is associated with test [5,6]

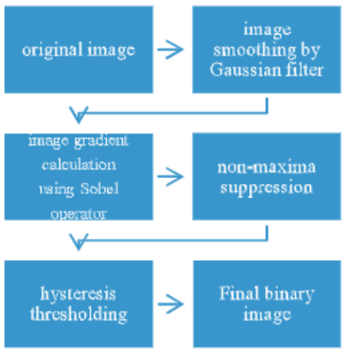


Fig. 1 Steps of Canny Algorithm

image during its transmission or others reasons. It degraded the quality of image and produce false edge of image. At second stage calculated the gradient values for finding the maximum value of pixel. After this level thresholding is used to mark the edge in image. Select the particular threshold value and compare others pixel's intensity value with respect to threshold value[1]. If the pixel value is higher than selected threshold value, than it is called edge of image otherwise that point is not consider as edge. After marking all edge point by comparing with threshold point finally get the edge of test image.

III NOISE REDUCTION

In digital image, pixel is the smallest element of any image or picture represented on the screen. These images have been passed across the Gaussian filter. Because this detector is more susceptible to noise present in original image data, it uses a filter based on a Gaussian, where the unprocessed image is processed with a gaussian filter[9]. Gaussian fitter is a filter whose impulse response is a Gaussian function. Gaussian filters have the properties of having no overshoot to a step function input while minimizing the rise and fall time.

(a) Finding the intensity gradient

An edge in an image may point in a range of directions, so the Canny algorithm uses four filters to detect horizontal, vertical and diagonal edges in the blurred image. The edge detection operator (Roberts, Prewitt, Sobel for example) returns a value for the first derivative in the horizontal direction (H) and the vertical direction (V). From this the edge gradient and direction can be determined:

$$G = \sqrt{(H^2 + V^2)}$$

$$\theta = a\tan 2(V, H)$$

Where atan2 is arctangent function with two arguments. The edge direction angle is rounded to one of four angles representing vertical, horizontal and the two diagonals[5].

Given estimates of the image gradients, a search is then carried out to determine if the gradient magnitude assumes a local maximum in the gradient direction. From this stage referred to as non-maximum suppression, a set of edge points, in the form of a binary image, is obtained. These are sometimes referred to as "thin edges".

(b) Tracing edges through the image and hysteresis thresholding

High intense and low intense gradient both have equal importance in edge detection. It is in most cases impossible to specify a threshold at which a given intensity gradient switches from corresponding to an edge into not doing so. Therefore Canny uses thresholding with hysteresis.

Thresholding with hysteresis requires two thresholds – high and low. Making the assumption that important edges should be along continuous curves in the image allows us to follow a faint section of a given line and to discard a few noisy pixels that do not constitute a line but have produced large gradients. Therefore we begin by applying a high threshold. This marks out the edges we can be fairly sure are genuine. Starting from these, using the directional information derived earlier, edges can be traced through the image. While tracing an edge, we apply the lower threshold, allowing us to trace faint sections of edges as long as we find a starting point. Once this process is complete we have a binary image where each pixel is marked as either an edge pixel or a non-edge pixel[5]. From complementary output from the edge tracing step, the binary edge map obtained in this way can also be treated as a set of edge curves, which after further processing can be represented as polygons in the image domain.

(c) Differential geometric formulation of the Canny edge detector

A more refined approach to obtain edges with sub-pixel accuracy is by using the approach of differential edge detection, where the requirement of non-maximum suppression is formulated in terms of second- and third-order derivatives computed from a scale space representation[6]. The Canny algorithm contains a number of adjustable parameters, which can affect the computation time and effectiveness of the algorithm.

IV LAPLACIAN OF GAUSIAN

The laplacian is a two- dimensional isotropic measure of the second special derivative ofan image. The laplacian of an image highlights regions of rapid intensity change and is therefore often used for edge detection. The laplacian is often applied to an image that has first been smoothed with something appxoximating a Gaussian smoothing filter in order to reduce its sencitivity to noise, and hence the two variants will be described together here. Since the input image is represented as a set of discrete pixels, we have to find a discrete convolution kernel that can approximate the second derivatives in the definition of the laplacian. Two commonly used small kernels are-

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

Using one of these kernels, the laplacian can calculated using standard convolution methods.

V PREWITT METHOD

The Prewitt operator is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function[7]. At each point in the image, the result of the Prewitt operator is either the corresponding gradient vector or the norm of this vector. The Prewitt operator is based on convolving the image with a small, separable, and[5] integer valued filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations. On the other hand, the gradient approximation which it produces is relatively crude, in particular for high frequency variations in the image. The Prewitt operator was developed by Judith M. S. Prewitt.

(a) **Simplified description-**In simple terms, the operator calculates the gradient of the image intensity at each point, giving the direction of the largest possible increase from light to dark and the rate of change in that direction[2]. The result therefore shows how "abruptly" or "smoothly" the image changes at that point, and therefore how likely it is that part of the image represents an edge, as well as how that edge is likely to be oriented. In practice, the magnitude (likelihood of an edge) calculation is more reliable and easier to interpret than the direction calculation.

Mathematically, the gradient of a two-variable function (here the image intensity function) is at each image point a 2D vector with the components given by the derivatives in the horizontal and vertical directions. At each image point, the gradient vector points in the direction of largest possible intensity increase, and the length of the gradient vector corresponds to the rate of change in that direction. This implies that the result of the Prewitt operator at an image point which is in a region of constant image intensity is a zero vector and at a point on an edge is a vector which points across the edge, from darker to brighter values.

(b) **Formulation-**Mathematically, the operator uses two 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical. If we define I as the source image, and H and V are two images which at each point contain the horizontal and vertical derivative approximations, the latter are computed as:

$$H=\begin{bmatrix}-1 & 0 & +1\\ -1 & 0 & +1\\ -1 & 0 & +1\end{bmatrix} * I \qquad V=\begin{bmatrix}+1 & +1 & +1\\ 0 & 0 & 0\\ -1 & -1 & -1\end{bmatrix} * I$$

where * here denotes the 2-dimensional convolution operation. Since the Prewitt kernels can be decomposed as the products of an averaging and a differentiation kernel[3], they compute the gradient with smoothing. For example, H can be written as

$$\begin{bmatrix}\mathbf{-1} & \mathbf{0} & \mathbf{+1}\\ \mathbf{-1} & \mathbf{0} & \mathbf{+1}\\ \mathbf{-1} & \mathbf{0} & \mathbf{+1}\end{bmatrix} = \begin{bmatrix}1\\1\\1\end{bmatrix}\begin{bmatrix}-1 & 0 & +1\end{bmatrix}$$

The *x*-coordinate is defined here as increasing in the "right"-direction, and the *y*-coordinate is defined as increasing in the "down"-direction. At each point in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using:

$$G=\sqrt{(H^2+V^2)}$$

Using this information, we can also calculate the gradient's direction:

$$\Theta = \operatorname{atan2}(V,H)$$

where, for example, Θ is 0 for a vertical edge which is darker on the right side.

VI ROBERT METHOD

The Roberts cross operator is used in image processing and computer vision for edge detection. It was one of the first edge detectors and was initially proposed by Lawrence Roberts in 1963. As a differential operator, the idea behind the Roberts cross operator is to approximate the gradient of an image through discrete differentiation which is achieved by computing the sum of the squares of the differences between diagonally adjacent pixels. Motivation-According to Roberts, an edge detector should have the following properties: the produced edges should be well-defined, the background should contribute as little noise as possible, and the intensity of edges should correspond as close as possible to what a human would perceive. With these criteria in mind and based on then prevailing psychophysical theory Roberts proposed the following equations:

$$Y_{i,j}=\sqrt{X_{i,j}}$$

$$Z_{i,j}=\sqrt{[(Y_{i,j}-Y_{i+1,j+1})^2+(Y_{i+1,j}-Y_{i,j+1})^2]}$$

where x is the initial intensity value in the image, z is the computed derivative and i,j represent the location in the image.

The results of this operation will highlight changes in intensity in a diagonal direction. One of the most appealing aspects of this operation is its simplicity; the kernel is small and contains only integers[11]. However with the speed of computers today this advantage is negligible and the Roberts cross suffers greatly from sensitivity to noise.

Formulation-In order to perform edge detection with the Roberts operator we first convolve the original image, with the following two kernels:

$$\begin{bmatrix}+1 & 0\\ 0 & -1\end{bmatrix} \quad \text{and} \quad \begin{bmatrix}0 & +1\\ -1 & 0\end{bmatrix}$$

Let G(x,y) be a point in the original image and H(x,y) be a point in an image formed by convolving with the first kernel and V(x,y) be a point in an image formed by convolving with the second kernel. The gradient can be defined as-

$$G = \sqrt{(H^2 + V^2)}$$

Gradient direction -

Θ = atan2(V,H)

VII SOBEL METHOD

This operator is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, it computing an approximation of the gradient of the image intensity function. At each point in the image, the result of the Sobel operator is either the corresponding gradient vector or the norm of this vector. The Sobel operator is based on convolving the image with a small, separable, and integer valued filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations[6]. On the other hand, the gradient approximation that it produces is relatively crude, in particular for high frequency variations in the image. The operator uses two 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical. If we define I as the source image, and H and V are two images which at each point contain the horizontal and vertical derivative approximations, the computations are as follows:

$$H=\begin{bmatrix}-1 & 0 & -1\\ +2 & 0 & -2\\ +1 & 0 & -1\end{bmatrix} * I \qquad V = \begin{bmatrix}+1 & +2 & +1\\ 0 & 0 & 0\\ -1 & -2 & -1\end{bmatrix} * I$$

where *here denotes the 2-dimensional convolution operation.

Since the Sobel kernels can be decomposed as the products of an averaging and a differentiation kernel, they compute the gradient with smoothing. For example, H can be written as

$$\begin{bmatrix}+1 & 0 & -1\\ +2 & 0 & -2\\ +1 & 0 & -1\end{bmatrix} = \begin{bmatrix}1\\ 2\\ 1\end{bmatrix}\begin{bmatrix}+1 & 0 & -1\end{bmatrix}$$

The x-coordinate is defined here as increasing in the "right"-direction, and the y-coordinate is defined as increasing in the "down"-direction. At each point in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using:

$$G = \sqrt{(H^2 + V^2)}$$

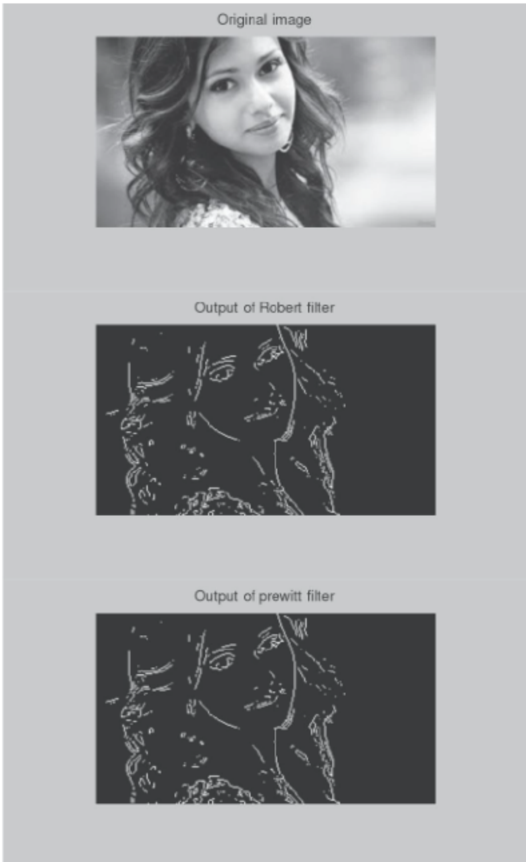
Using this information, we can also calculate the gradient's direction:

Θ(x,y) = atan2(H,V)

where, for example, Θ is 0 for a vertical edge which is darker on the right side.

VIII COMPARISON

In this paper, grayscale or a binary image takes as its input test image and returns a binary image BW of the same size as I (input image), with 1's where the function finds edges in I and 0's elsewhere. Sobel filter Robert filter and Prewitt filter returns edges at those points where the gradient of I (input image) is maximum. The canny 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. This method is therefore less likely than the others to be fooled by noise and more likely to detect true weak edges. The output image of proposed work is shows that canny methods gives better results as compare to mentioned methods.



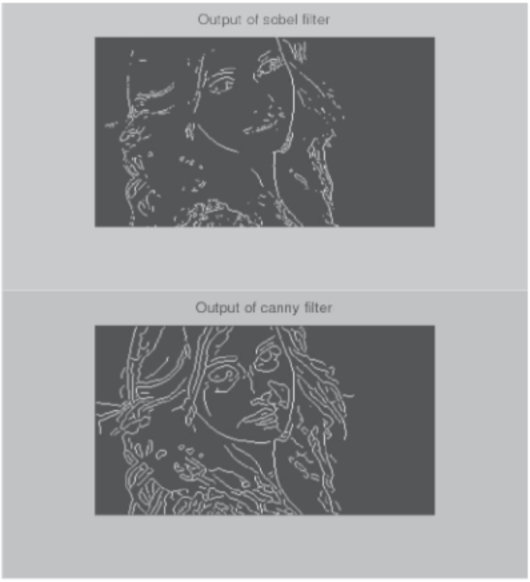


Fig. 1 Original image and output images of different operators

IX CONCLUSION

This paper provides diminutive outline of basic building blocks of edge detection methods. Edge detection methods improve the test image by eliminated the noise and enhance the quality of image and extract information of interest. It is necessary to examine merits and demerits of each edge detection filters at different situation so as to use best option according to situation or condition. Edge detection techniques such as Gradient-based and Laplacian based have been studied. Gradient-based algorithms have major drawbacks in sensitive to noise. The dimension of the kernel filter and its coefficients are static and it cannot be adapted to a given image. A novel edge-detection algorithm is essential to give an errorless solution that is adaptable to the different noise levels in images to help identifying the valid image contents. The performance of the Canny algorithm relies mainly on the changing parameters which are standard deviation for the Gaussian filter, and its threshold values. The size of the Gaussian filter is controlled by the greater value and the larger size. The larger size produces more noise, which is necessary for noisy images, as well as detecting larger edges. We have lesser accuracy of the localization of the edge then the larger scale of the Gaussian. For the smaller values we need a new algorithm to adjust these parameters. The user can modify the algorithm by changing these parameters to suit the different environments. Canny's edge detection algorithm is more costly in comparison to Sobel, Prewitt and Robert's operator. Even though, the Canny's edge detection algorithm has a better performance. The evaluation of the images shows that under the noisy conditions, Canny, LoG, Sobel, Prewitt, Roberts's exhibit better performance, respectively. When comparing Gradient and Laplacian transformation methodologies it seems that although Laplacian does the better for some features (i.e. the fins), it still suffers from mis-mapping some of the lines.

REFERENCES

[1] G. T. Shrivakshan, Dr.C. Chadrasekar (September 2012) IJCI Internetional Journal of Computer Science Issues, Vol. 9, issue 5, No 1, ISSn (online): 1694-0814.

[2] Raman Maini & Dr. Himanshu Aggarwal (2005) International Journal of Image Processing (IJIP), Volume (3): Issue (1).

[3] Shazia Akram, Dr. mehraj-Ud-Din Dar, Aasia Quyoum(november 2010) edge detection techniques International Journal of Computer application (0975-8887) Volume 10- No.5 [4] V.Torre and T.A. Poggio,(march,1986) On edge detection IEEE Transactions on Patern Analysis , Vol. 8, pp. 147-163..

[4] Rafael C. Gonzalez and Richard E. Woods: (2002)Digital Image Processing. Addison-Wesley

[5] Erdon, A., (1992). Edge detection methods in digital images. Msc. Thesis, ITU.,Istanbul,Turkey.

[6] A.K. Jain, R.P.W. Duin and J. Mao: Statistical (2000) pattern recognition: a review, IEEE Transactions on pattern analysis and machine Intelligence, vol. 22, no.1, 4-37.

[7] GZ Yang and DF Gillies (2007) Image Processing and Edge detection.

[8] U. G. Sefercik, O. E .Gülegen(2010) Edge detection in geologic formation extraction close range and remote sensing case studies.

[9] Erdon, A. (1992) "Edge detection methods in digital images". Msc. Thesis, ITU. Istanbul, Turkey.

[10] Cristian Munteanu and Agostinho Rosa(april 2004) Gray-Scale Image Enhancement as an Automatic Process Driven by Evolution, IEEE transactions on systems, man, and cybernetics—part b: cybernetics, vol. 34, no. 2.

[11] Zia-ur Rahmany, Daniel J. Jobsonz, Glenn A. Woodellz (2003)Retinex Processing for Automatic Image Enhancement.