ISSN 2278 – 0211 (Online)

Edge Detection Techniques: A Survey

U. G. Nevagi
Professor, Department of Electronics &Telecommunication, S.R.I.E.I.T., Shiroda, Goa, India
Ashwini Shahapurkar
Faculty, Department of Electronics &Telecommunication, S.R.I.E.I.T., Shiroda, Goa, India
Sneha Nargundkar
Faculty, Department of Electronics &Telecommunication, S.R.I.E.I.T., Shiroda, Goa, India

Abstract:
An edge may be defined as a set of connected pixels that forms a boundary between two disjoints regions. Edge detection refers to the process of identifying and locating sharp discontinuities in an image. Edge detection process significantly reduces the amount of data and filters out useless information, while preserving the essential structural properties in an image. Since computer vision involves the recognition and classification of objects in an image, edge detections is a vital tool. In this paper, the main aim is to study edge detection process based on different techniques.

1.Introduction

Edge detection is a primary tool used in image processing, basically for feature detection and extraction, whose aim to identify points in a digital image where brightness of image changes sharply and find discontinuities. The purpose of edge detection is to significantly reduce the amount of data in an image and preserve the structural properties for further image processing. In a gray level image the edge is a local feature that, with in a neighborhood separates regions in each of which the gray level is more or less uniform with in different values on the two sides of the edge. For a noisy image it is difficult to detect edges as both edge and noise contains high frequency contents which results in blurred and distorted result.

The purpose of edge detection is to mark the points in a digital image at which the luminous intensity changes sharply. In Image analysis process to interpret an image, one first must be able to detect the edges of each object in the image. Edge representation of an image significantly reduces the amount of data to be processed, yet it retains useful information about the shapes of objects in the scene. The effectiveness of many image processing and computer vision tasks depends on the perfection of detecting meaningful edges. Edge-detection has been a challenging task in low level image processing .Various approaches are available for edge detection , some are based on error minimization, maximizing an object function, neural network, fuzzy logic, wavelet approach, Bayesian approach, morphology, genetic algorithms.

Classical methods of edge detection involve convolving the image with an operator (a 2-D filter), which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions. There are an extremely large number of edge detection operators available, each designed to be sensitive to certain types of edges. Variables involved in the selection of an edge detection operator include Edge orientation, Noise environment and Edge structure. The geometry of the operator determines a characteristic direction in which it is most sensitive to edges. Operators can be optimized to look for horizontal, vertical, or diagonal edges. Not all edges involve a step change in intensity. Effects such as refraction or poor focus can result in objects with boundaries defined by a gradual change in intensity. The operator needs to be chosen to be responsive to such a gradual change in those cases. So, there are problems of false edge detection, missing true edges, edge localization, high computational time and problems due to noise etc. Therefore, the objective is to do the comparison of various edge detection techniques and analyze the performance of the various techniques in different conditions.

There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories:

1.1. Gradient Based Edge Detection

The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image.

1.2. Laplacian Based Edge Detection

The Laplacian method searches for zero crossings in the second derivative of the image to find edges. An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location.

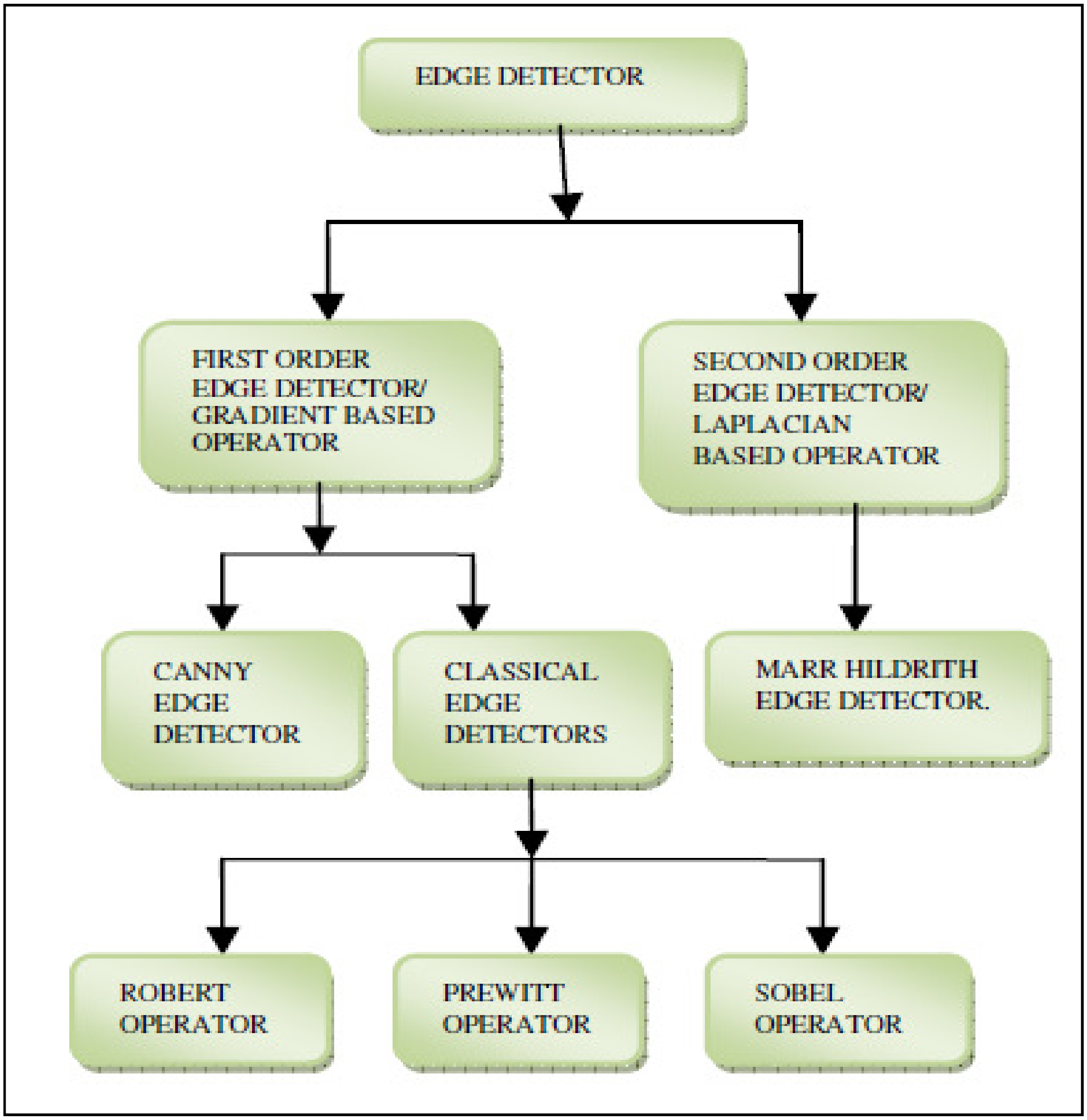

Figure 1: Different Edge detection methods

2. Edge-Detection Steps
The basic steps involve in the edge-detection process are:

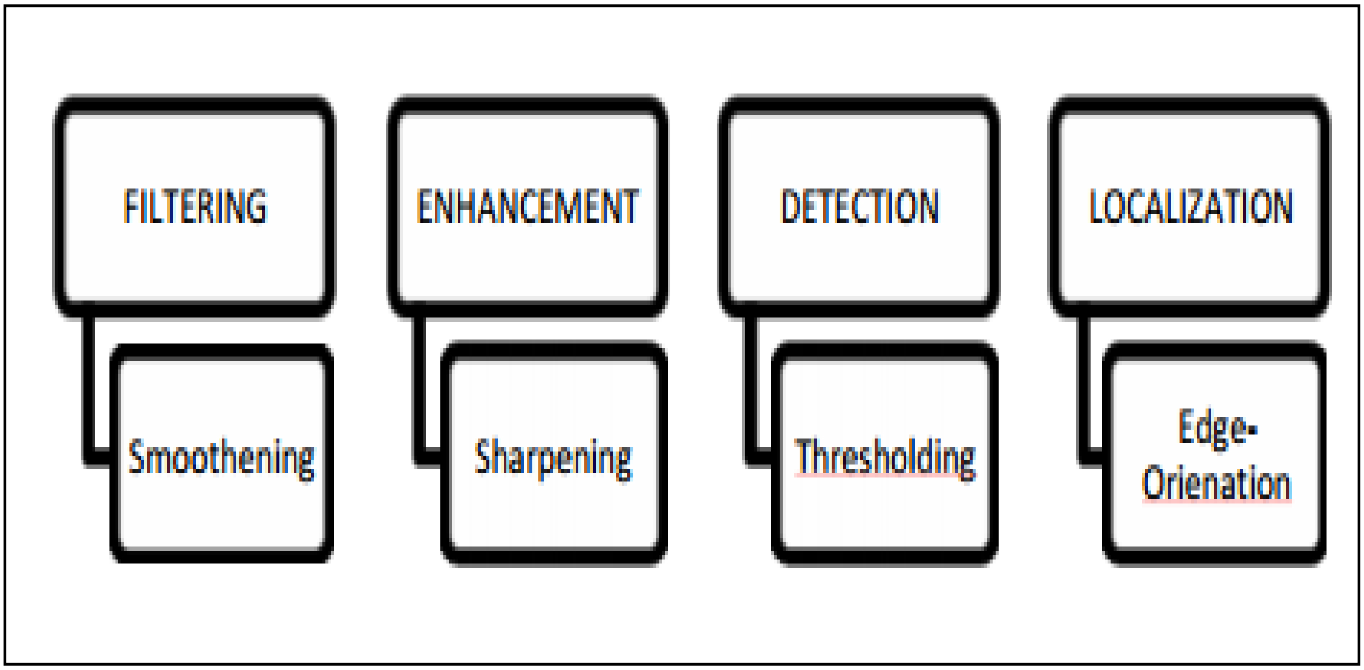

Figure 2: Edge-detection steps

A. Filtering: Filter image to enhance performance of the edge detector concerning noise. It includes suppressing the noise as much as possible, without destroying the true edges.
B. Enhancement/Sharpening: Give emphasis to pixels having considerable change in local intensity.
C. Detection: Decisive about which edge pixels should be superfluous as noise and which should be retained.
D. Localization: Determine the accurate locations of an edge .Edge thinning and linking are generally a requisite for edge localization.

3. Edge Detection Techniques
Classical operators Robert, Sobel, Prewitt are classified as classical operators which are easy to operate but highly sensitive to noise.

3.1. Robert Operator
It is gradient based operator. It firstly computes the sum of the squares of the difference between diagonally adjacent pixels through discrete differentiation and then calculate approximate gradient of the image. The input image is convolved with the default kernels of operator and gradient magnitude and directions are computed. It uses following 2x2 two kernels:

$$D_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} and D_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

The plus factor of this operator is its simplicity but having small kernel it is highly sensitive to noise not and not much compatible with today’s technology.

3.2. Sobel Operator
Sobel operator is a discrete differentiation operator used to compute an approximation of the gradient of image intensity function for edge detection. At each pixel of an image, sobel operator gives either the corresponding gradient vector or normal to the vector. It convolves the input image with kernel and computes the gradient magnitude and direction. It uses following 3x3 two kernels:

$$D_{\mathrm{x}} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} and D_{\mathrm{y}} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

As compared to Robert operator have slow computation ability but as it has large kernel so it is less sensitive to noise as compared to Robert operator. As having larger mask, errors due to effects of noise are reduced by local averaging within the neighborhood of the mask.

3.3. Prewitt Operator

The function of Prewitt edge detector is almost same as of sobel detector but have different kernels:

$$D_{\mathrm{x}} = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} and D_{\mathrm{y}} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Prewitt edge operator gives better performance than that of sobel operator.

3.4. Laplacian of Guassian (LoG) Operator

It was invented by Marr and Hildreth (1980). The Gaussian filtering is combined with Laplacian to break down the image where the intensity varies to detect the edges effectively. It uses linear interpolation to determine the sub pixel location of the edge .The digital implementation of the Laplacian function is made using the following mask.

$$Dx = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

The operator usually takes a single gray-level image as input and produces another gray-level image as output. The Laplacian L(a,b) of an image with pixel intensity values I(a.b) is given by:

$$L(a,b) = \frac{\partial^2 I}{\partial a^2} + \frac{\partial^2 I}{\partial b^2}$$

Since the input image is represented as a set of discrete pixels, we need to find a discrete convolution kernel that can approximate the second derivatives in the definition of the Laplacian.The disadvantage of LOG operator is that it cannot find orientation of edge because of laplician filter.

3.5. Canny Operator

It is a method to find edges by isolating noise from the image without affecting the features of the edges in the image and then applying the tendency to find the edges and the critical value for threshold. The canny edge detector first smoothens the image to eliminate noise. Then it finds the image gradient to highlight regions with high spatial derivatives. After that it perform tracking along these regions and suppresses any pixel that is not at the maximum. The gradient array at this moment can further be reduced by hysteresis which is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero. If the magnitude is above the high threshold, it is made an edge. Major application of canny edge detector is for remote sensing images which are inherently noisy.

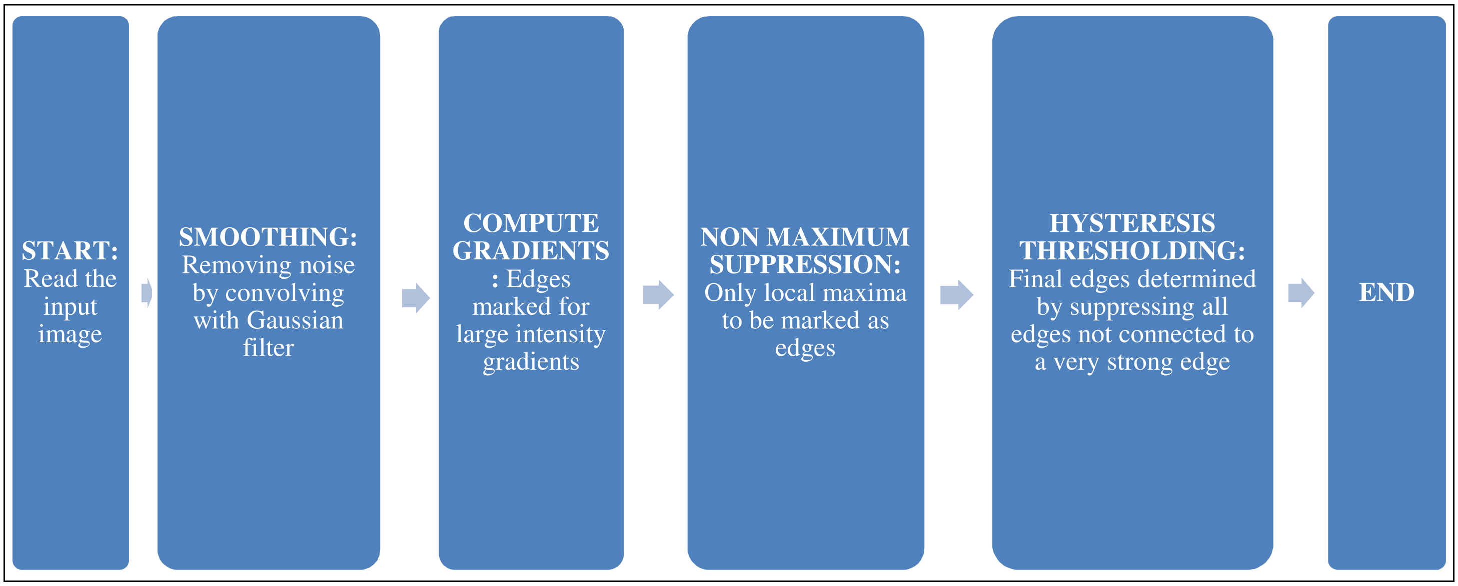

Figure 3: Flow Chart of Canny Edge Detection Algorithm

3.6. Canny Edge Detection Algorithm

3.6.1. STEP I: Noise reduction by smoothing

Noise contained in image is smoothed by convolving the input image I (i, j) with Gaussian filter G. Mathematically, the smooth resultant image is given by F (i, j) =G*I (i, j)

Prewitt operators are simpler to operator as compared to sobel operator but more sensitive to noise in comparison with sobel operator.

3.6.2. STEP II: Finding gradients

In this step we detect the edges where the change in grayscale intensity is maximum. Required areas are determined with the help of gradient of images. Sobel operator is used to determine the gradient at each pixel of smoothened image. Sobel operators in i and j directions are given as

$$D_{\mathrm{i}} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} and D_{\mathrm{j}} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

These sobel masks are convolved with smoothed image and giving gradients in i and j directions.

$$G_i = D_i * F(i, j) \quad \text{and} \quad G_j = D_j * F(i, j)$$

Therefore edge strength or magnitude of gradient of a pixel is given by

$$G = \sqrt{Gi^2 + Gj^2}$$

The direction of gradient is given by

$$\theta = arctan\left(\frac{Gj}{Gi}\right)$$

G_i and G_j are the gradients in the i- and j-directions respectively.

3.6.3. STEP III: Non maximum suppressions

Non maximum suppression is carried out to preserves all local maxima in the gradient image, and deleting everything else this results in thin edges. For a pixel M (i, j):

- Firstly round the gradient direction θ nearest45° ,then compare the gradient magnitude of the pixels in positive and negative gradient directions i.e. if gradient direction is east then compare with gradient of the pixels in east and west directions say E (i, j) and W (i, j) respectively.
- If the edge strength of pixel M (i, j) is largest than that of E (i, j) and W (i, j), then preserve the value of gradient and mark M (i, j) as edge pixel, if not then suppress or remove.

3.6.4. STEP IV: Hysteresis thresholding

The output of non-maxima suppression still contains the local maxima created by noise. Instead choosing a single threshold, for avoiding the problem of streaking two thresholds t_{high} and t_{low} are used.

For a pixel M (i, j) having gradient magnitude G following conditions exists to detect pixel as edge:

- If G <t_{low},discard the edge.
- If G > t_{high}, keep the edge.
- If t_{low}<G<t_{high} and any of its neighbors in a 3×3 region around it have gradient magnitudes greater than t_{high}, keep the edge.
- If none of pixel (x, y)'s neighbors have high gradient magnitudes but at least one falls between t_{low} and $t_{high,}$ search the 5×5 region to see if any of these pixels have a magnitude greater than t_{high}. If so, keep the edge. Else, discard the edge.

3.7. Advantages of Canny Edge Detection Algorithm.

On analyzing all these edge detection techniques , it is found that canny gives optimum edge detection .Following are the some points throwing light on the advantages of canny edge detector as compared to other detectors discussed in this paper:

1. Less Sensitive to noise: As compared to classical operators like Prewitt, Robert and Sobel canny edge detector is less sensitive to noise. Its uses Gaussian filter which removes noise at a great extent as compared to above filters. LoG operator is also highly sensitive to noise as differentiate twice in comparison to canny operator.
2. Remove streaking problem: The classical operators' like Robert uses single thresholding technique but it results into streaking. Streaking means, if the edge gradient just above and just below the set threshold limit it removes the useful part of connected edge, and leave the disconnected final edge. To overcome from this drawback canny detector uses 'hysteresis' technique which uses two threshold values t_{low} and t_{high}as discussed above in canny algorithm.
3. Adaptive in nature: Classical operator have fixed kernels so cannot be adapted to a given image. While the performance of canny algorithm depends on variable or adjustable parameters like E which is the standard deviation of Gaussian filter and threshold values t_{low} and t_{high}.Smaller the value of σ results smaller Gaussian filter in turns results in finer edges. So user can changes these parameters and can improve the result of canny algorithm.
4. Good localization: LoG operators cannot find edge orientation while canny operator provides edge gradient orientation which results into good localization.

4. Results & Conclusion

The edge detection is the primary step in identifying an image object, it is very essential to know the advantages and disadvantages of each edge detection filters. In this paper we dealt with study of various gradient-based and Laplacian based edge detection techniques.

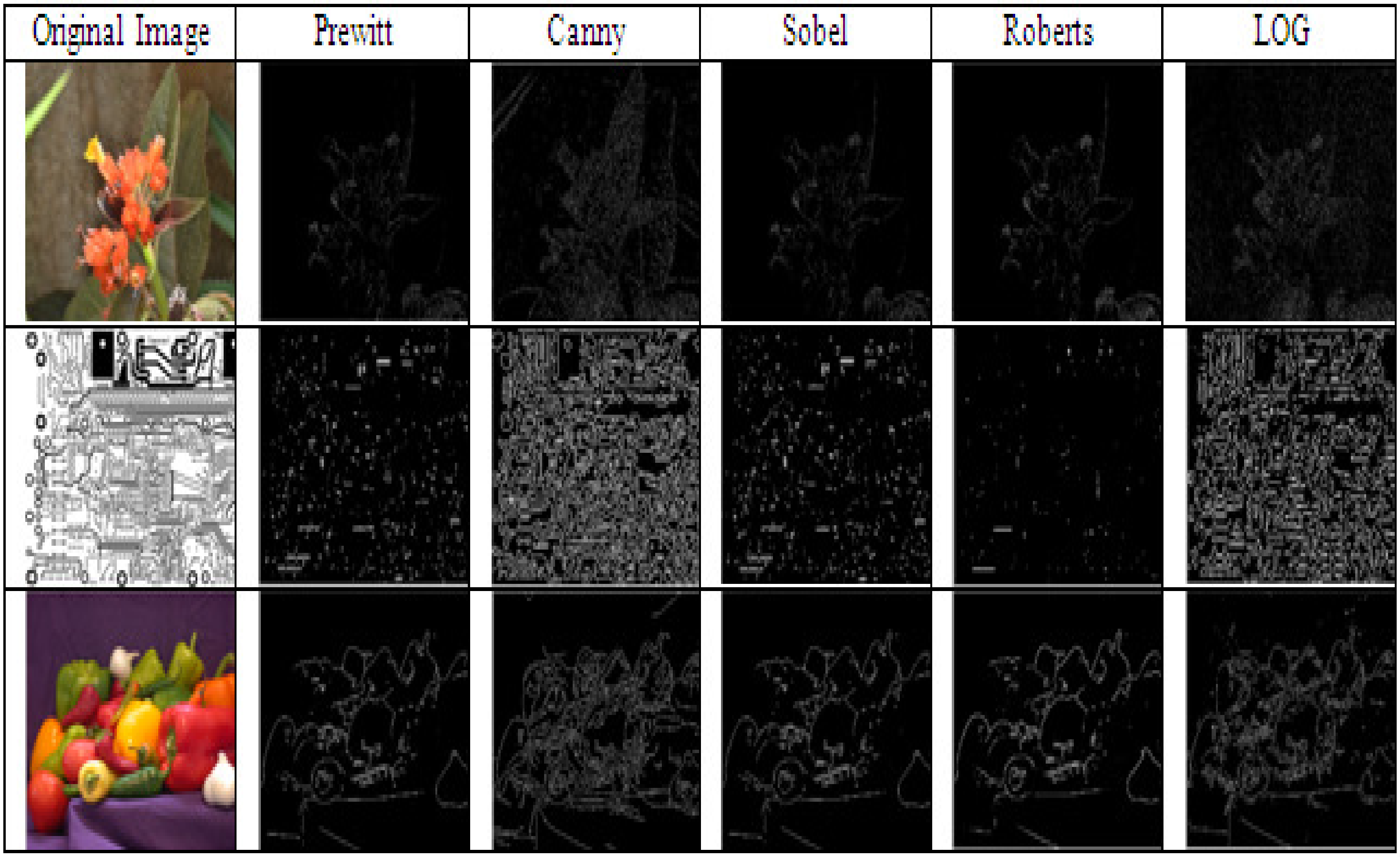

Figure 4

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 necessary to provide an errorless solution that is adaptable to the different noise levels of these images to help in identifying the valid image contents produced by noise. 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 comparing to Sobel, Prewitt and Robert's operator even though, the Canny's edge detection algorithm has a better performance.

5. References

i. Adnan Khashman , "Automatic Detection, Extraction And Recognition Of Moving Objects" ,International Journal of Systems Applications, Engineering and Development, Issue 1, Volume 2,2008.
ii. Kinjal A Joshi, Darshak G. Thakore ,"A Survey On Moving Object Detection And Tracking In Video Surveillance System", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July 2012.
iii. Y.Ramadevi, T.Sridevi, B.Poornima, B. Kalyani, "Segmentation And Object Recognition Using Edge Detection Techniques", International Journal of Computer Science & Information Technology (IJCSIT), Vol 2, No 6, December 2010.
iv. Kavitha Ganesan,Shanmugam Jalla,"Video Object Extraction Based On A Comparative Study of Efficient Edge Detection Techniques",The International Arab Journal of Information Technology, Vol. 6, No. 2, April 2009.
v. Beant Kaur, Mr.Anil Garg ," Comparative Study Of Different edge Detection Techniques", International Journal Of Engineering Science And Technology (IJEST).

vi. Bindu Bansal, Jasbir Singh Saini, Vipan Bansal, And Gurjit Kaur ,"Comparison Of Various Edge Detection Techniques", Journal Of Information And Operations Management ISSN: 0976–7754 & E-ISSN: 0976–7762 , Volume 3, Issue 1, pp-103-106, 2012.

vii. Canny, J., "A Computational Approach to Edge Detection", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 8:679-714, November 1986.

viii. N. Senthilkumaran, R. Rajesh, "Edge Detection Techniques for Image Segmentation and A Survey of Soft Computing Approaches", International Journal of Recent Trends in Engineering, Vol. 1, No. 2, PP.250-254, May 2009.

ix. T.G. Smith Jr., et.al. ; "Edge detection in images using Marr-Hildreth filtering techniques", Journal of Neuroscience Methods, Volume 26, Issue 1, pp. 75–81, November 1988.

x. R .Gonzalez and R. Woods, Digital Image Processing, Addison Wesley, 1992, pp 414 - 428.