ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com

Performance Evaluation Of Edge Detection For Medical Images

[1]SP.Chokkalingam, [2]S.Mithra Devi and [3]M.Z.Rahima Sulthana

[1]*Associate Professor, Department of Computer Science and Engineering Saveetha School of Engineering, Saveetha University*
[2,3]*U.G Student Department of Computer Science and Engineering Saveetha School of Engineering, Saveetha University*

ARTICLE INFO

Article history:
Received 12 March 2015
Accepted 28 April 2015
Available online 1 June 2015

Keywords:
Image segmentation, Edge Detection, Prewitt, Roberts, Sobel and Canny

ABSTRACT

Image segmentation is the process of dividing an image into segments according to their characteristic e.g. Color and object present in the image. Edge detection filters the noise, frequencies and unwanted data and preserves the structural property of an image. Since edge detection is in the forefront of image processing for object detection, it is crucial to have a good understanding of edge detection methods. In this paper the comparative analysis of various Image Edge Detection methods is presented. Upon this evaluation, an edge detection method can be employed to characterize edges to represent the image for further analysis and implementation. The Performance of different edge detectors isanalyzed. It has been shown that the Canny's edge detection algorithm performs better than all these operators under medical scenarios.

© 2015 AENSI Publisher All rights reserved.

To Cite This Article: SP.Chokkalingam, S.Mithra Devi and M.Z.Rahima Sulthana., Performance Evaluation Of Edge Detection For Medical Images. ***Aust. J. Basic & Appl. Sci.,*** *9(11): 736-740, 2015*

INTRODUCTION

Edges are boundaries between different textures. Edge also can be defined as discontinuities in image intensity from one pixel to another. The edges for an image are always the important characteristics that offer an indication for a higher frequency. Detection of edges for an image may help for image segmentation, data compression, and also help for well matching, such as image reconstruction and so on.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. Edge detection is difficult in noisy images, since both the noise and the edges contain high-frequency content. Attempts to reduce the noise result in blurred and distorted edges.

Operators used on noisy images are typically larger in scope, so they can average enough data to discount localized noisy pixels. This results in less accurate localization of the detected 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.

II. *Edge detection:*

Edge detection is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision. The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. It can be shown that under rather general assumptions for an image formation model, discontinuities in image brightness are likely to correspond to:

- discontinuities in depth,
- discontinuities in surface orientation,
- changes in material properties and
- Variations in scene illumination.

In the ideal case, the result of applying an edge detector to an image may lead to a set of connected curves that indicate the boundaries of objects, the boundaries of surface markings as well as curves that correspond to discontinuities in surface orientation. Thus, applying an edge detection algorithm to an image may significantly reduce the amount of data to

Corresponding Author: SP.Chokkalingam, Associate Professor, Department of Computer Science and Engineering
Saveetha School of Engineering, Saveetha University
E-mail: 1cho_mas@yahoo.com.co

be processed and may therefore filter out information that may be regarded as less relevant, while preserving the important structural properties of an image. If the edge detection step is successful, the subsequent task of interpreting the information contents in the original image may therefore be substantially simplified. However, it is not always possible to obtain such ideal edges from real life images of moderate complexity.Edges extracted from non-trivial images are often hampered by *fragmentation*, meaning that the edge curves are not connected, missing edge segments as well as *false edges* not corresponding to interesting phenomena in the image – thus complicating the subsequent task of interpreting the image data.Common edge detection algorithms include Prewitt, Roberts, Sobel and Canny.

2.1. Prewitt operator:

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. 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 integer valued filter in horizontal and vertical directions 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.

2.2. Roberts operator:

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.

2.3. Sobel operator:

The Sobel operator, sometimes called Sobel Filter, is used in image processing and computer vision, particularly within edge detection algorithms, and creates an image which emphasizes edges and transitions. It is named after Irwin Sobel, who presented the idea of an "Isotropic 3x3 Image Gradient Operator" at a talk at the Stanford Artificial Intelligence Project (SAIP) in 1968.Technically, it is a discrete differentiation operator, 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. On the other hand, the gradient approximation that it produces is relatively crude, in particular for high frequency variations in the image. The Kayyali operator for edge detection is another operator generated from Sobel operator.

2.4. Canny operator:

The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. It was developed by John F. Canny in 1986. Canny also produced a *computational theory of edge detection* explaining why the technique works.

Process of Canny edge detection algorithm:

The Process of Canny edge detection algorithm can be broken down to 5 different steps:

1. Apply Gaussian filter to smooth the image in order to remove the noise
2. Find the intensity gradients of the image
3. Apply non-maximum suppression to get rid of spurious response to edge detection
4. Apply double threshold to determine potential edges
5. Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

Improvement on Canny Edge Detection:

While traditional canny edge detection provides relatively simple but precise methodology for edge detection problem, with the more demanding requirements on the accuracy and robustness on the detection, the traditional algorithm can no longer handle the challenging edge detection task. The main defects of the traditional algorithm can be summarized as following:

1. Gaussian filter is applied to smooth out the noise, but it will also smooth the edge, which is considered as the high frequency feature. This will increase the possibility to miss weak edges, and the appearance of isolated edges in the result.
2. For the gradient amplitude calculation, the old canny edge detection algorithm uses center in a small 2*2 neighborhoods window to calculate the finite difference mean value to represent the gradient amplitude. This method is sensitive to noise and can easily detect fake edges and lose real edges.
3. In traditional canny edge detection algorithm, there will be two fixed global threshold values to filter out the false edges. However, as the image gets complex, different local areas will need very different threshold values to accurately find the real edges. In addition, the global threshold values are

determined manually through experiments in the traditional method, which leads to complexity of calculation when large number of different images needs to be dealt with.

4. The result of the traditional detection cannot reach a satisfactory high accuracy of single response for each edge- multi-point responses will appear.

III. Proposed system:

There are a variety of edge detection techniques that have been used in the past. Simple techniques such as the Boolean edge detector converts a window of pixels into a binary pattern based on a local threshold, and then applies masks to determine if an edge exists at a certain point or not. In the Marr-Hildreth Edge Detector, we smooth then the image using a Gaussian and Laplacian function. This takes the second derivative of an image. If there is a step difference in the intensity of the image, it will be represented in the second derivative by a zero crossing.

The Sobel operator is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function. At each point in the image, the result of t h e Sobel operator is either the corresponding gradient vector or the norm of this vector. The operator consists of a pair of 3×3 convolution kernels designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The method finds edges using the sobel approximation to the derivative. It then returns edges at those points where the gradient of the image is the maximum. Prewitt operator is similar to the Sobel operator used for detecting vertical and horizontal edges in images. It is a fast method only suitable for well contrasted noiseless images.

The Canny edge detector is considered to be one of the most widely used edge detection algorithms in the industry. It works by first smoothing the image and finds the image gradient to highlight regions with high spatial derivatives. It then tracks along these regions to suppress any pixel that is not at the maximum. Finally, through hysteresis, it 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 and if the magnitude is between the two thresholds, it is set to zero unless there is a path from this pixel to a pixel with a gradient above the second threshold. That is to say that the two thresholds are used to detect strong and weak edges, and include the weak edges in the output only if they are connected to strong edges.All the edge detection techniques are applied to the original images and the result is noted. Features like area, mean, median, and variance are also note to evaluate the better performing edge detection operator.

VI. System design:

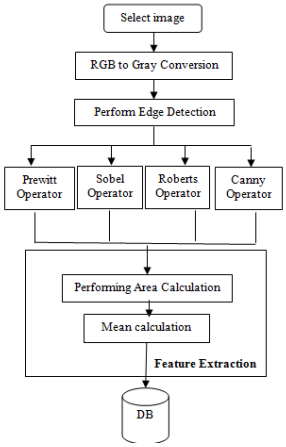

Fig. 1: System Architecture.

VII. Experimental analysis:

Edges are detected using the Sobel, Prewitt, and Roberts methods, by thresholding the gradient function. For the Laplacian of Gaussian method, thresholding is computed for the slope of the zero crossings after filtering the image with the LoG filter. For the Canny method, a threshold is applied to the gradient using the derivative of a Gaussian filter.

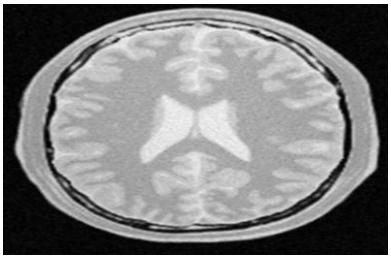

Fig. 2: Original MRI image.

The Prewitt method finds edges using the Prewitt approximation to the derivative. It returns edges at those points where the gradient of the image is maximum. Result of applying prewitt operator to snapshot 1 is displayed in snapshot.

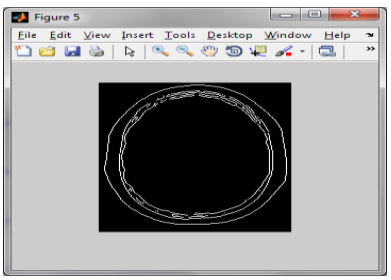

Fig. 3: Applying Prewitt.

Detection using Sobel filter as mentioned before, the Sobel method finds edges using the Sobel approximation to the derivative. It returns edges at

those points where the gradient of the image is maximum. Snapshot 3 is the result of applying sobel operator in snapshot.

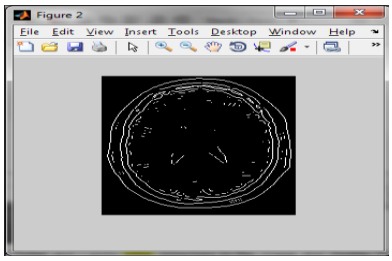

Fig. 4: Detection using Sobel.

Detection using Roberts The Roberts method finds edges using the Roberts approximation to the derivative. It returns edges at those points where the gradient of the image is maximum. Snapshot 4 is the result of applying Roberts operator in the snapshot.

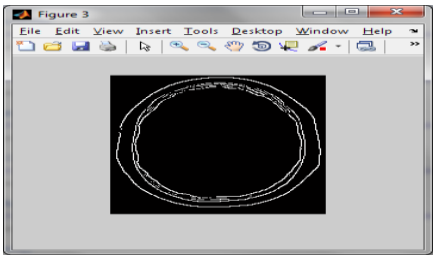

Fig. 5: Applying Roberts Operator.

Detection using Canny the Canny method finds edges by looking for local maxima of the gradient of the image. The gradient is calculated using the derivative of the Gaussian filter. The 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.

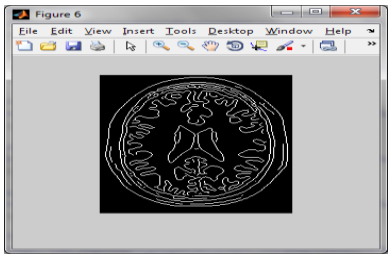

Fig. 6: Applying Canny Operator.

E. ***Area calculation:***

bwarea(BW) - 'bwarea' estimates the area of all of the on pixels in an image by summing the areas of each pixel in the image. The area of an individual pixel is determined by looking at its 2-by-2 neighborhood. There are six different patterns, each representing a different area:

- Patterns with zero on pixels (area = 0)
- Patterns with one on pixel (area = 1/4)
- Patterns with two adjacent on pixels (area = 1/2)
- Patterns with two diagonal on pixels (area = 3/4)
- Patterns with three on pixels (area = 7/8)
- Patterns with all four on pixels (area = 1)

F. ***Mean calculation:***

Val = mean(I) - The input image A can be numeric or logical. The output image B is a scalar of class double.

The input image gpuarrayA is a gpuArray whose underlying class is numeric or logical. The output image gpuarrayB is a gpuArray scalar with the underlying class double.

VIII. Conclusion:

In this work we have analyzed the behavior of Prewitt, Sobel, Roberts and Canny operator on the capability of edge detection for images. The methods are applied to the whole image. No specific texture or shape is specified. The objective is to investigate the effect of the various methods applied in finding a representation for the image under study. On visual perception, it can be shownclearly that the Sobel, Prewitt, and Roberts provide lowquality edge maps relative to the others. A representation of the image can be obtained through the canny method. Among the various methods investigated, the canny method is able to detect both strong and weak edges, and seems to be more suitable than the other edge detection methods.

REFERENCES

Huertas, A. and G. Medioni, 1986. "Detection of intensity changes withsub-pixel accuracy using Laplacian-Gaussianmasks" IEEE Transactionson Pattern Analysis and Machine Intelligence. PAMI, 8(5): 651-664.

Selvarajan, S. and W.C. Tat, 2001. " Extraction of man-made features fromremote sensing imageries by data fusion techniques" 22nd AsianConference on Remote Sensing, 5-9. Singapore.

Voorhees, H. and T. Poggio, 1987. " Detecting textons and texture boundries innatural images" ICCV, 87: 250-25.

Argyle, E., 1971. "*Techniques for edge detection*," Proc. IEEE, 59: 285-286.

Bergholm, F., 1986. "*Edge focusing*," in Proc. 8th Int. Conf. PatternRecognition, Paris, France, 597-600.

Matthews, J., "An introduction to edge detection: The sobel edgedetector," Available at

Gonzalez, R.C. and R.E. Woods, 2002. "*Digital Image Processing".* 2nd ed.Prentice Hall.

Torre, V. and T.A. Poggio, 1986."*On edge detection*".IEEE *Trans. PatternAnal. Machine Intell.*, PAMI-8(2): 187-163.

Frei, W. and C.C. Chen, 1977. "*Fast boundary detection: A generalizationand a new algorithm* ". lEEE Trans. Comput., C-26(10): 988-998.

Grimson, W.E. and E.C. Hildreth, 1985. "*Comments on Digital step edges from zero crossings of second directional derivatives*''.IEEE Trans.Pattern Anal. Machine Intell., PAMI-7(1): 121-129.

Haralick, R.M., 1984. "*Digital step edges from zero crossing of the seconddirectional derivatives*," IEEE Trans. Pattern Anal.MachineIntell., PAMI-6(1): 58-68.

Canny, J.F., 1986. "*A computational approach to edge detection*". IEEETrans. Pattern Anal. Machine Intell., PAMI-8(6): 679-697.

Canny, J., 1983. "*Finding edges and lines in image*". Master's thesis, MIT.

Hueckel, M.H., 1973. "A *local visual operator which recognizes edges andline*".*J. ACM,* 20(4): 634-647.

Yakimovsky, Y., 1976. "*Boundary and object detection in real world images*".JACM, 23(4): 598-619.

Marr, D. and E. Hildreth, 1980. "*Theory of Edge Detection*".Proceedings of theRoyal Society of London. Series B, Biological Sciences, 207(1167): 187-217.

Heath, M., S. Sarkar, T. Sanocki and K.W. Bowyer, 1997."*A RobustVisual Method for Assessing the Relative Performance of EdgeDetection Algorithms*". IEEE Trans. Pattern Analysis and MachineIntelligence, 19(12): 1338-1359.