International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 6, June - 2014

Image Edge Detection Techniques using MATLAB Simulink

Ms. Pawar P. H,
PG Student
E & TC Department
SKNCOE, Vadgaon Bk.
Pune, India

Prof. Patil R. P,
Asst. Professor
E & TC Department
SKNCOE, Vadgaon Bk.
Pune, India

Abstract **- Edge detection is one of the important part of image processing. Edge detection is the processes to detect the sharp changes in intensity value (pixel value) of the image. In past there are many edge detection techniques are there. The commonly used are Sobel, Robert, Prewitt, Laplacian, LoG (Laplacian of Guassian), Canny's edge detection technique and Beamlet transform edge detection technique. In this paper we deal with MATLAB/SIMULINK model for Canny's edge detection technique and Beamlet transform edge detection technique. Corresponding simulation results are compared using MSE (Mean Square Error), PSNR (Peak Signal to Noise Ratio) performance metrics.**

Keywords - Edge detection Algorithms, Canny edge detection, Beamlet transform edge detection, MATLAB/SIMULINK, MSE, PSNR.

I. INTRODUCTION

Edge detection refers to the process of identifying as well as locating sharp discontinuities from an image [4]. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. In classical methods of edge detection, convolving the image with an operator, which is constructed to be responsive to large gradients in the image while returning values of zero in even regions. There is an extremely large number of edge detection operators existing, each intended to be sensitive to certain types of edges. Edge detection is the process of finding quick contrasts in intensities in an image. This process significantly reduces the quantity of data in the image, while preserving the majority important structural features of that image [1]. Edge detection is a well-developed field within image processing in which region boundaries and edges are strongly related. There is often a sharp modification in intensity at the region boundaries.

Canny offered an optimal edge detector [2], mainly for two-dimensional image. Canny operator can give the edge information of intensity as well as direction. All methods mentioned above are reported superior results on selected domains of pixel-level edge discovery. It is still hard to extract linear features embedded in tremendously high noise or when the SNR (signal to noise) is so low that none of the pixel values is expected to yield significance, as there are so many methods. In this paper, a multi-scale algorithm based on beamlet transform is proposed to detect edges in image. Beamlets are generated by recursive dyadic partitioning then vertex marking and connection. Afterwards the beamlet transform is the gathering of all line integrals fashioned by viewing the image as a piecewise constant object. And integration along line segment in the beamlet dictionary, for the maximal beamlet coefficient existing the Canny criterion. Draw a line section depicting that beamlet, all these beamlets in dissimilar scales are merged to create an edge map at the image pixel point. The projected method detects lines with any location, orientation and length in dissimilar scales and a maximal beamlet coefficient is calculated to pass up individual setting. The new results give you an idea about that the method projected detect edges accurately even from high noise image and has a better presentation.

MATLAB is a high-performance language for technical measures as well as computations. It combines calculation, apparition, and programming in an easy-to-operate environment [6] where problems and solutions are uttered in recognizable mathematical details.

The objective lead to the use of a tool with a high-level graphical interface under the Matlab Simulink based blocks which makes it very easy to handle with respect to other software. A variety of applications where noise elimination, attractive edges and contours detection, blurring and so on. This paper presents the architecture of Canny Edge detection algorithm and Beamlet transform Edge detection algorithm with the help of Video and Image Processing block set.

II. EDGE DETECTION ALGORITHM

A. Canny Edge Detection Algorithm

The Canny edge detector [5] is an edge recognition operator that uses a multi-stage algorithm to identify a wide range of edges in images. Canny's aim was to find out the optimal edge detection algorithm. In this condition, an "optimal" edge detector means:

Good detection – the algorithm should mark real edges in the image as possible.

Good localization – Marked edges should be as close as possible to the edge in the real image.

Minimal response – a given edge in the image should only be noticeable once, and where possible, image noise should not generate false edges.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 6, June - 2014

To assure these necessities Canny used the calculus of variations – a method which finds the task which optimizes a given practical method. The optimal function in Canny's detector is described here uses median filtering for smoothing input image. As shown in Fig.1 algorithm runs in 5 separate steps [2] as:

a. Smoothing: Blurring of the image to eliminate noise by using median filtering.

b. Finding gradients: The edges should be marked where the gradients of the image has huge magnitudes.

c. Non-maximum suppression: Only local maxima should be considered as edges.

d. Double thresholding: Potential edges are determined by thresholding.

e. Edge tracking by hysteresis: Final edges are determined by suppressing all edges that are not connected to a very certain (strong) edge.

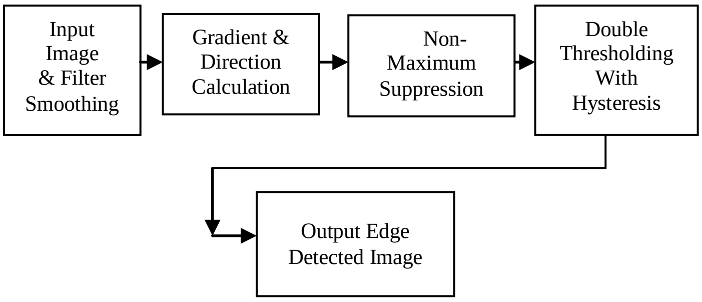


Fig.1 Canny Edge Detection

B. Beamlet transforms Edge Detection Algorithm

The main idea of beamlet representation is to approximate linear objects in two dimensions by multiscale/ location/orientation segments [3]. These are called beamlets. Manipulate the beamlet co-efficient the Beamlet framework involves 5 central components, Beamlet dictionary, Beamlet transform, beamlet pyramid, beamlet graph, beamlet algorithm extract data from the beamlet pyramid in a way driven by the structure of the beamlet graph.

Example: - Recursive Dyadic partitioning algorithms

The complete recursive dyadic partition can be described in fig. 2 a top-down fashion: a squared image is partitioned into 2×2 smaller d-squares, repeating the partitioning until the finest resolution of the image is reached.

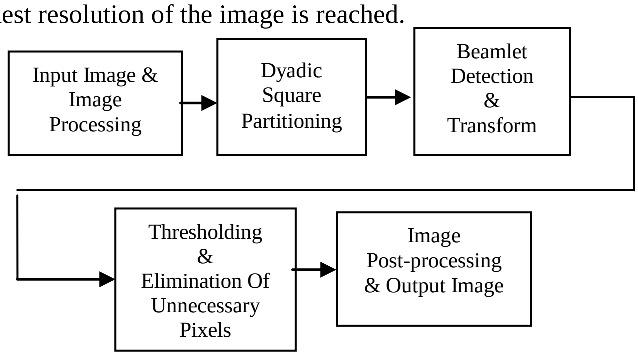


Fig. 2 Edge detection using Beamlet Transform

The beamlet transform is defined as the collection of line integrals along the beamlet. For a digital image, the beamlet transform is a measure of the line integral in a discrete domain. The beamlet transform for all the points along a beamlet b is defined as

$$T(b) = \sum_b fi1i2\ \varphi i1i2 \qquad (1)$$

Where, $fi1i2$ is the gray-level value of pixel $(i1, i2)$ and $\varphi i1i2$ is considered to be the weight function for each pixel shown as

$$\varphi i1i2 = \frac{ln}{L} \qquad (2)$$

Where L is the total length of the beam and ln is the length of segment in each square pixel on the beam shown as

$$L = \sum_n (ln) \qquad (3)$$

In image post-processing improvement in linear features such as discontinuities in an image different algorithms are used. Here line thinning is elaborated.

Example- Line thinning: - Calculate the orientation of the beamlet with its distance changes in the x and y directions respectively Δx and Δy . In fig.2 the value of the orientation is obtained as follows

$$\theta = \arctan(\frac{\Delta x}{\Delta y}) \qquad (4)$$

III. PERFORMANCE METRICS

A. Mean Square Error (MSE):

It calculates the deviation between the pixel values of reference image and edge detector image [7]. A lesser value shows good results.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m} \sum_{j=0}^{n} ([g(i,j) - f(i,j)])^2 \qquad (5)$$

Where M and N are pixels in the horizontal and vertical dimensions of image, g denotes input image and f denotes output image. The lowest mean square error represents best quality image.

B. Peak Signal to Noise Ratio (PSNR):

It determines the degree of similarity between reference and edge detected image. A bigger value shows good results [7].

$$PSNR = 20\ log_{10} \left[\frac{[L]^2}{\frac{1}{mn} \sum_{i=0}^{m} \sum_{j=0}^{n} ([g(i,j) - f(i,j)])^2} \right] \qquad (6)$$

Where, L denotes number of gray levels in image.

IV. IMPLEMENTATION

Canny edge detection algorithm and Beamlet transform edge detection algorithms are implemented using MATLAB Simulink models by using image processing block sets from library browser. MATLAB 7.10(2012a) is used for implementation.

A. Simulink model of Canny edge detection algorithm

Implementation of Simulink model for canny edge detection algorithm is shown in Fig.3. In this image from workspace block is used for input image. Canny 1 block is an embedded

function block in which MATLAB code for canny algorithm is written in which median filtering also applied [8].
Calculate parameters block is another embedded function block in which MATLAB code for calculation of MSE and PSNR is written. It takes two images for calculation one is input image another is edge detected image. Video viewer block is used to display both images.

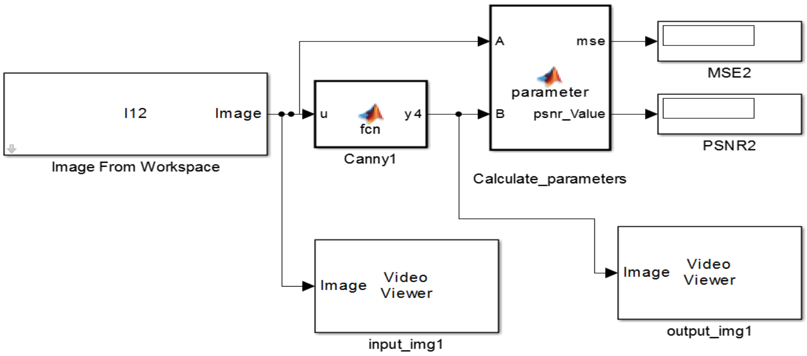


Fig.3 MATLAB Simulink model for Canny Edge detection

B. Simulink model of Beamlet Transform Edge detection Algorithm

Implementation of Simulink model for Beamlet transform edge detection algorithm is shown in Fig.4. In this input block is used for input image. Median filter block used to smooth input image and for removal of noise. In Beamlet block MATLAB code is written for beamlet transform algorithm. On which line thinning is applied. This uses embedded system block Line_thining as shown in fig.4.
Calculate parameters block is another embedded function block in which MATLAB code for calculation of MSE and PSNR is written. It takes two images for calculation one is input image another is edge detected image. Video viewer block is used to display original image, median filter output, beamlet output and final line thinning means edge image.

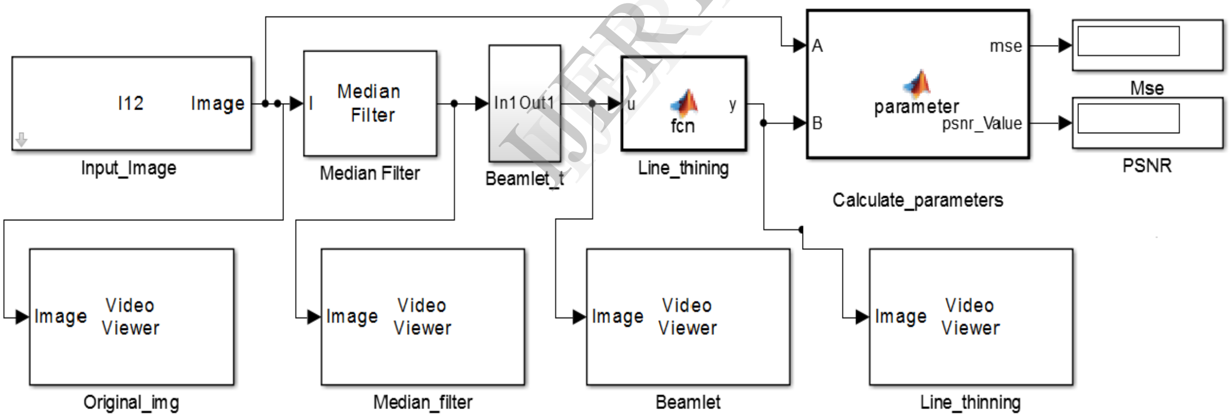
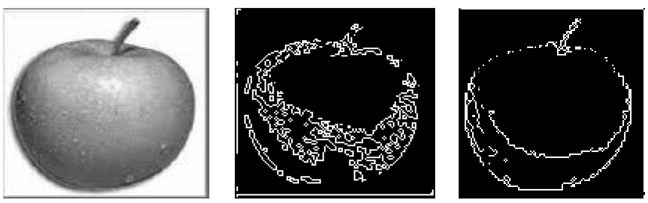


Fig.4 Simulink model for Beamlet transform edge detection algorithm

V. RESULTS AND DISCUSSIONS

Images of apple, seed, Lena, coco1, Flower, flower, rose1, rose2, rose3, apple2, person are taken as input test images. The test images are of any size converted to 128*128. From these images few results are shown below. The performance of the image edge detection process is evaluated on the basis of different set of qualitative measures. Table 1 shows the performance evaluation of the different edge detected images for various dataset. In Fig.5(a) Original image of "apple" is shown ,on which Canny and Beamlet algorithms are applied and results of simulink models for Canny and Beamlet transform is shown in Fig.5(b) and Fig.5(c) respectively.



(a) (b) (c)
Fig.5 (a) Original Image (b) Canny output (c) Beamlet output


(a) (b) (c)

Fig.6 (a) Original Image (b) Canny output (c) Beamlet output

In Fig.6(a) Original image of “seed” is shown ,on which Canny and Beamlet algorithms are applied and results of simulink models for Canny and Beamlet transform is shown in Fig.6(b) and Fig.6(c) respectively. MSE value tabulated in Table I for canny output image is 1.75E+04 and for beamlet is 0.5346 which clearly shows beamlet result is good than canny.


(a) (b) (c)

Fig.7 (a) Original Image (b) Canny output (c) Beamlet output

In Fig.7(a) Original image of “Lena” is shown ,on which Canny and Beamlet algorithms are applied and results of simulink models for Canny and Beamlet transform is shown in Fig.7(b) and Fig.7(c) respectively. For lena image PSNR values shown in Table I are 6.206 and 54 for canny and beamlet algorithm respectively. Beamlet PSNR is higher than Canny.

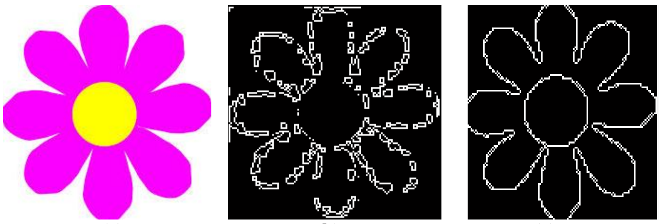

(a) (b) (c)

Fig.8 (a) Original Image (b) Canny output (c) Beamlet output

In Fig.8(a) Original image of “Flower” is shown ,on which Canny and Beamlet algorithms are applied and results of simulink models for Canny and Beamlet transform is shown in Fig.8(b) and Fig.8(c) respectively.


(a) (b) (c)

Fig.9 (a) Original Image (b) Canny output (c) Beamlet output

In Fig.9(a) Original image of “Person” captured by webcamera of laptop is shown ,on which Canny and Beamlet algorithms are applied and results of simulink models for Canny and Beamlet transform is shown in Fig.9(b) and Fig.9(c) respectively.

TABLE I: RESULTS OF QUANTITATIVE MEASURES FOR DIFFERENT IMAGE DATASETS

Sr. No.	Image Name	MSE (Canny)	MSE (Beamlet)	PSNR (Canny)	PSNR (Beamlet)
1	apple	3.54E+04	0.5346	2.678	50.88
2	Seed	1.75E+04	0.2785	5.736	53.72
3	Lena	1.57E+04	0.2612	6.206	54
4	coco1	3.47E+04	0.5279	2.761	50.94
5	flower1	3.45E+04	0.5136	2.789	51.06
6	flower2	4.38E+04	0.6554	1.747	50
7	rose1	5.09E+04	0.7556	1.097	49.38
8	rose2	4.33E+04	0.64	1.803	50.1
9	rose3	3.92E+04	0.5758	2.231	50.56
10	apple2	3.52E+04	0.5185	2.698	51.02

In Table I results of quantitative measures for different image datasets and comparative analysis is shown. MSE and PSNR are calculated for each image in tabular form. Calculations of MSE and PSNR with the help of equation(5) and equation(6). From Table I it clearely shows that MSE values for beamlet are much smaller than MSE values for Canny. Also PSNR values for beamlet are higher than PSNR values for canny. And it clearely shows from these results that Beamlet transform method gives better results than canny edge detection method.

VI.CONCLUSION

In this paper, implementation of Canny and Beamlet transform Image edge detection algorithms using MATLAB/ SIMULINK are described. Comparative analysis between two algorithms is done by calculating different performance metrics such as MSE (mean square error) and PSNR (Peak Signal to Noise Ratio). As explained in Table I MSE values are much smaller for beamlet transform method over canny edge detection method, which shows better performance of beamlet transform edge detection method. Also in case of PSNR values beamlet transform has much greater values than Canny edge detection method, this also describes superiority of Beamlet transform edge detection method over Canny edge detection method.

REFERENCES

[1] Raman Maini, Dr. Himanshu Aggarwal" Study and Comparison of Various Image Edge Detection Techniques" IJIP, Volume (3): Issue (1).

[2] Xiaoyang Li, Jie Jiang, Qiaoyun Fan" An Improved Real-time Hardware Architecture for Canny Edge Detection Based on FPGA"Third International Conference on Intelligent Control and Information Processing July 15-17, 2012 - Dalian, China.

[3] Dr.D.Selvathi, J.Dharani."Realization of Beamlet Transform Edge Detection Algorithm using FPGA " ICSIPRI 2013 R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev.

[4] R. Gonzalez and R.Woods, Digital Image Processing. Prentice Hall, 3rd edition, Jan. 2008.

[5] John Canny, "A computational approach to edge detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.8, no.6, pp 679-698, Nov.1986.

[6] M. Vinoth, B. Jayalakshmi, "Analysis of Edge Detection Algorithms and Denoising Filters on Digital X-Ray Images" The International Journal Of Science & Technoledge, Vol 2 Issue 2 February, 2014

[7] G.Padmavathi, P.Subashini, P.K.Lavanya, "Performance evaluation of the various edge detectors and filters for the noisy IR images", Sensors, Signals, Visualization, Imaging, Simulation and Materials.

[8] Mohammed Yousuf Khan, Masarath Nayeem Tayyaba, M.A.Raheem, Ayesha Siddiqua, Syed Sameena, "Image Enhancement and Hardware Implementation of Edge Detected Vascular Images using Simulink model", International Journal of Advanced Research in Computer and Communication EngineeringVol. 3, Issue 4, April 2014.