EDGE DETECTORS: SURVEY

By

Saurabh Allawadhi, Priti Gangania
USICT, GGSIPU, Sector 16C, Dwarka, India
CDAC, B-30, Block B, Industrial Area, Sector 62, Noida, Uttar Pradesh, India
saurabhallawadhi029@gmail.com, preetigangania@gmail.com

ABSTRACT
Image segmentation is the technique of detecting edges of the object within the picture that we want to identify. Image recognition is carried out in various steps: Step 1. RGB value of picture is extracted. Step 2. Gray image is acquired either using formulae 0.2989 * R + 0.5870 * G + 0.1140 * B or the usage of rgb2gray () function. Step 3. Binary image is acquired from gray image. Step 4. Now edge detection is implemented using various algorithms along with Sobel, Canny, Roberts, Prewitt, Laplacian of Gaussian, Zero cross detection. Edge detection is the most vital procedure as it symbolizes how nicely the object could be detected. This research papers gives comparison on existing methods such as Sobel operator, canny operator, Prewitt operator, Roberts operator, zero crossing method, Laplacian of Gaussian method using MATLAB R2016a.

General Terms
Pattern Recognition, MATLAB2016a.

Keywords
Canny, LoG, Zero cross, Prewiit, Sobel, Roberts

1. INTRODUCTION
Edges are the boundaries of picture or images which help in keeping the structural properties of image. On the edges there exist sharp contrast between the intensity of pixels consequently edge detection algorithms have a tendency to capture these variations and highlight these edges.

In this paper we focused on Sobel, Canny, Roberts, Prewitt, Gaussian, Zero cross. These have been compared on the basis of its outputs which is obtained from 3 input pictures and their entropy values. Here, entropy is used because it may target the difference between neighbor regions or a way of expressing the no. of states of a system. A system with higher states has a high entropy & a system with low states has a low entropy.

2 EDGE DETECTION ALGORITHMS

Each edge detection algorithm works upon the calculation of gradient vector at every point on the edge within the picture. Gradient symbolize change in orientation in the intensity at a particular point of picture. In image processing, term gradient vector is to be calculated which includes magnitude and direction. Direction of the vector is towards where there's max intensity change & magnitude symbolize the difference of the intensity at current point and from maximum intensity point.

2.1. First Order Derivatives

2.1.1. Prewitt

It is introduced by Judith M.S. Prewitt & it is used to detect an edge in the image. It detects 2 types of edges. First is Horizontal edges and second is Vertical edges

Calculation of gradient vector along horizontal and vertical axis (using 3*3 Matrix).

$$G_x = \begin{matrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{matrix} * \begin{matrix} x1 & x2 & x3 \\ y1 & y2 & y3 \\ z1 & z2 & z3 \end{matrix}$$

$$G_x = \begin{matrix} -x1+z1 & -x2+z2 & -x3+z3 \\ -x1+z1 & -x2+z2 & -x3+z3 \\ -x1+z1 & -x2+z2 & -x3+z3 \end{matrix}$$

$$G_y = \begin{matrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ +1 & +1 & +1 \end{matrix} * \begin{matrix} x1 & x2 & x3 \\ y1 & y2 & y3 \\ z1 & z2 & z3 \end{matrix}$$

$$G_y = \begin{matrix} -x1-y1-z1 & -x2-y2-z2 & -x3-y3-z3 \\ 0 & 0 & 0 \\ x1+y1+z1 & x2+y2+z2 & x3+y3+z3 \end{matrix}$$

$$\text{Magnitude}(G) = \sqrt{(G_x^2 + G_y^2)}$$

2.1.2. Sobel

Developed by Irwin Sobel and Gary Feldman

Im: original image
Calculation of gradient along horizontal Gx and vertical axis Gy using 3*3 matrix

$$G_x = \begin{matrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{matrix} * \begin{matrix} x1 & x2 & x3 \\ y1 & y2 & y3 \\ z1 & z2 & z3 \end{matrix}$$

$$G_x = \begin{matrix} x1-z1 & x2-z2 & x3-z3 \\ 2(x1-z1) & 2(x2-z2) & 2(x3-z3) \\ x1-z1 & x2-z2 & x3-z3 \end{matrix}$$

$$G_y = \begin{matrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix} * \begin{matrix} x1 & x2 & x3 \\ y1 & y2 & y3 \\ z1 & z2 & z3 \end{matrix}$$

$$G_y = \begin{matrix} x1+2y1+z1 & x2+2y2+z2 & x3+2y3+z3 \\ 0 & 0 & 0 \\ -1(x1+2y1+z1) & -1(x2+2y2+z2) & -1(x3+2y3+z3) \end{matrix}$$

Magnitude(G) = $\sqrt{(G_x^2 + G_y^2)}$

2.1.3. *Roberts*

Developed by Lawrence Robert
Im: original image
Calculation of gradient along horizontal Gx and vertical axis Gy using 2*2 matrix

$$G_x = \begin{matrix} +1 & 0 \\ 0 & -1 \end{matrix} * \begin{matrix} x1 & x2 \\ y1 & y2 \end{matrix}$$

$$G_x = \begin{matrix} x1 & x2 \\ -y1 & -y2 \end{matrix}$$

$$G_y = \begin{matrix} 0 & +1 \\ -1 & 0 \end{matrix} * \begin{matrix} x1 & x2 \\ y1 & y2 \end{matrix}$$

$$G_y = \begin{matrix} y1 & y2 \\ -x1 & -x2 \end{matrix}$$

Magnitude(G) = $\sqrt{(G_x^2 + G_y^2)}$

2.2. Second Order Derivatives

2.2.1. *Canny Operator*

It is introduced by JOHN F. CANNY in 1986. It is an edge detection operator which uses a multi stage algorithm to detect the edges in an image. This operator runs in 5 steps. 1. Smoothing: Blurring of the image to eliminate noise. 2. Finding Gradients: Edges should be marked where the gradient has higher magnitude. 3. Non-Maximum Suppression: Only local maximum (where gradient function is set to zero) should be marked as edges. 4. Double Threshold: Edges are determined. 5. Edge tracking by hysteresis: Final edges are obtained by suppressing the edges which are not strongly connected.

2.2.2. *Laplacian of Gaussian LOG*

It is introduced by SIMON DE LAPLACE
It is divergence of gradient obtained either from sobel, prewitt, roberts.
∇ (M) = d2(Px)/dx2 + d2(Py)/dy2 = dMx/dx + dMy/dy
Fall under 0 cross edge detection technique in which it determines those points on the picture which pass through zero value.
Blob detectors are based upon LOG. In this original picture, it is convuled with gaussian function onto this Laplacian operator is applied.
General mathematical gaussian function is

$$Gaussian = e^{\frac{-(x-midvalue)2}{2*(standarddeviation)2}}$$

General mathematical Laplacian operator is

$$\nabla \cdot \nabla P = \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} + \frac{\partial^2 P}{\partial z^2}$$

2.2.3. *Zero Cross*

It signifies complete technique around a point where the graph of the function used goes from above axis to below the axis or where the output changes from positive to negative or negative to positive.

3. FLOW CHART

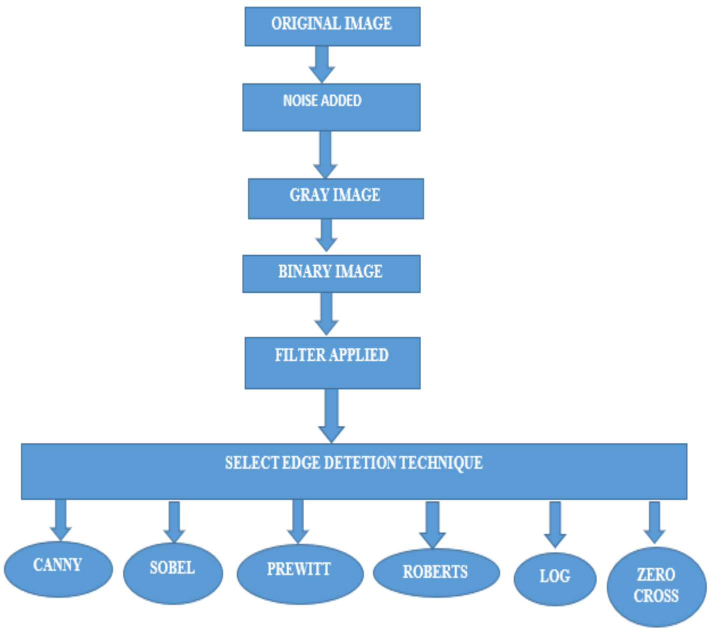

Fig1 Flow chart

4. EXPERIMENTAL RESULTS

Fig 2 original image

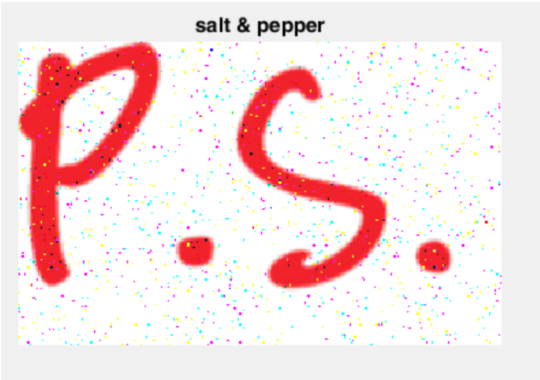

Fig 3 Image with Salt & Pepper noise

Fig 4 Image with Speckle noise

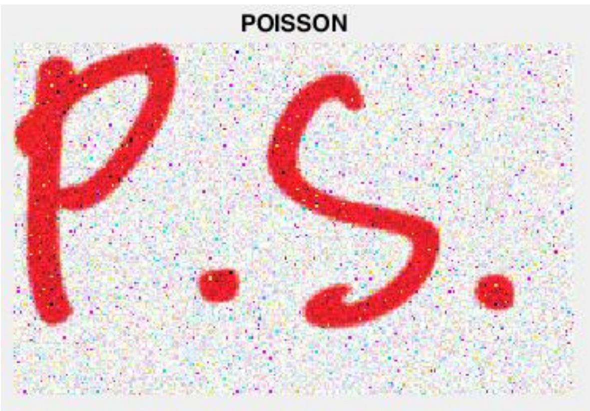

Fig 4 Image with Poisson noise

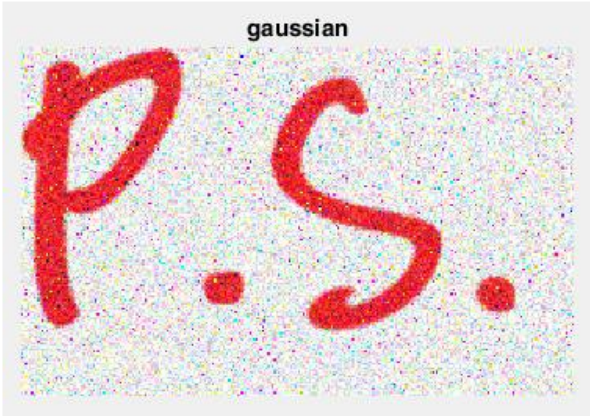

Fig 5 Image with Gaussian noise

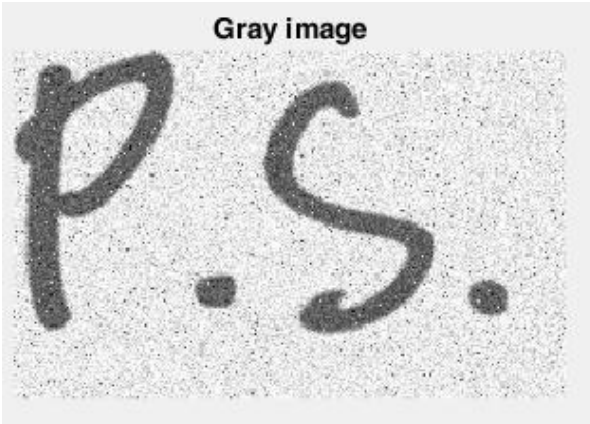

Fig 6 Gray image

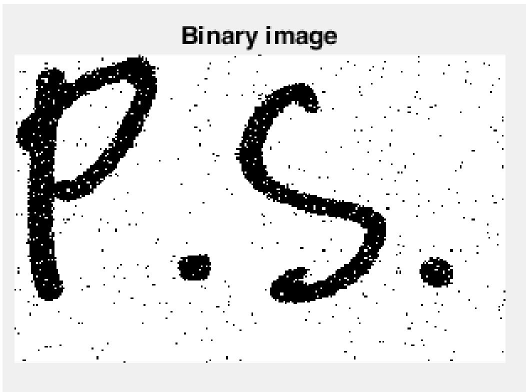

Fig 7 Binary image

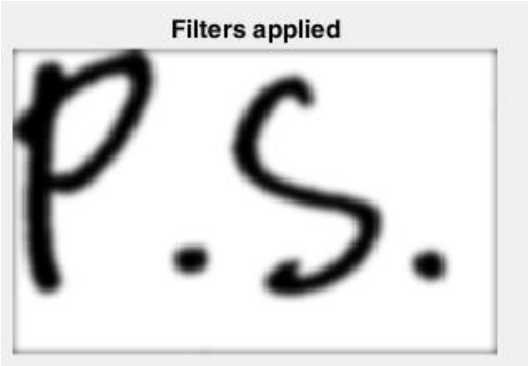

Fig 8 Filtered image with Poisson noise

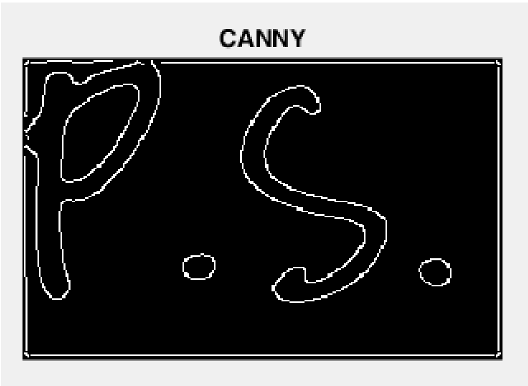

Fig 9 Image output of canny edge detector

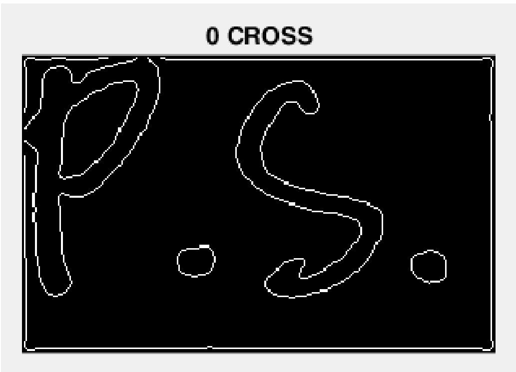

Fig 10 Image output of zero cross edge detector

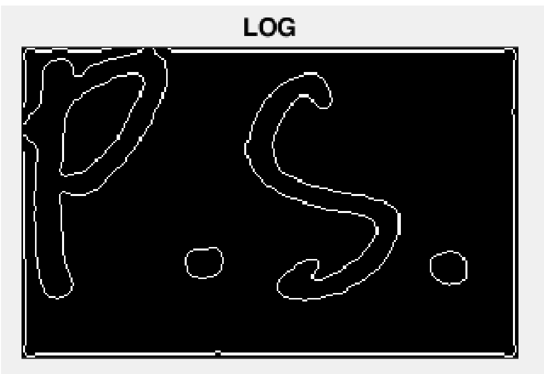

Fig 11 Image output of Laplacian of gaussian edge detector

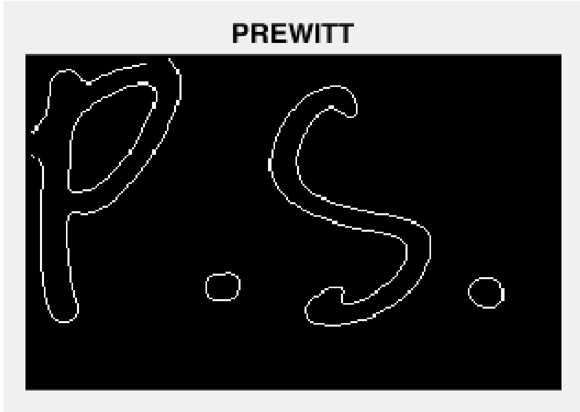

Fig 12 Image output of prewitt edge detector

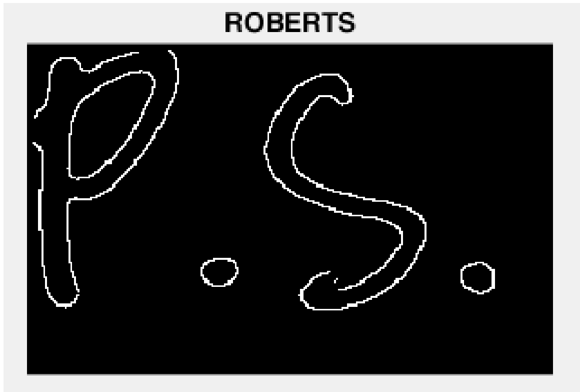

Fig 13 Image output of Roberts edge detector

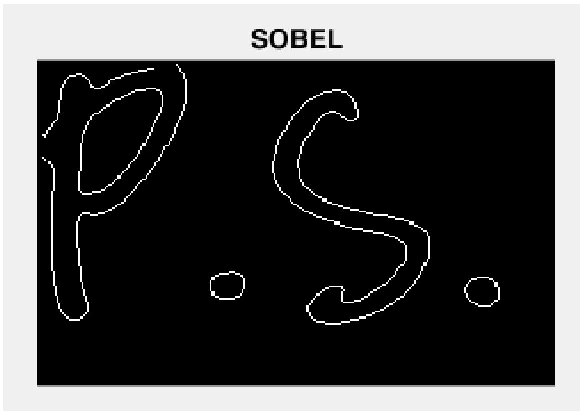

Fig 14 Image output of sobel edge detector

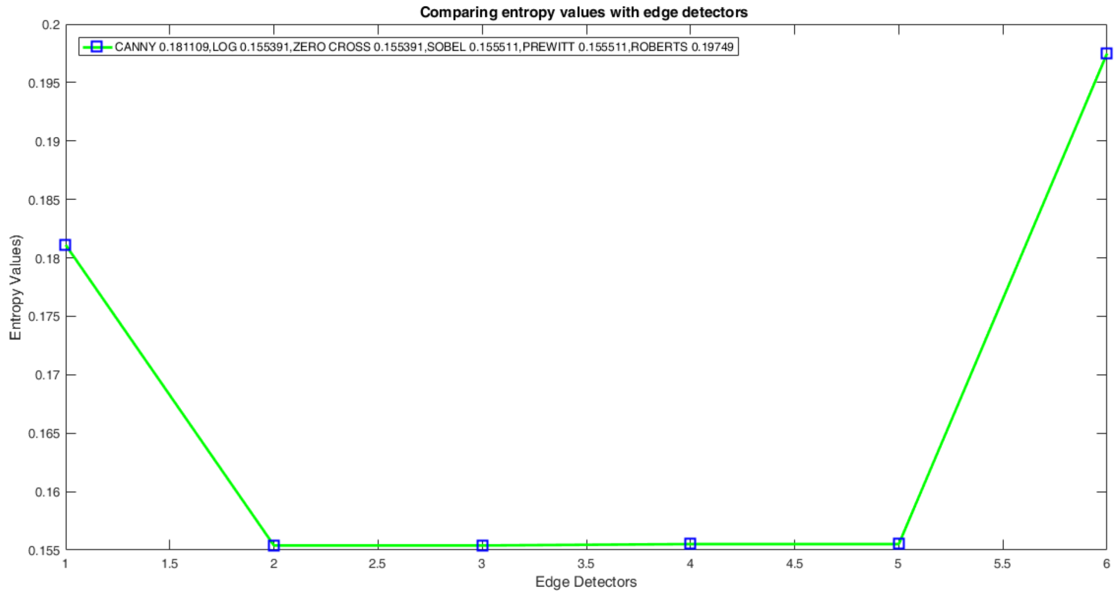

Fig 15 Plot to show entropy values comparison for image PS

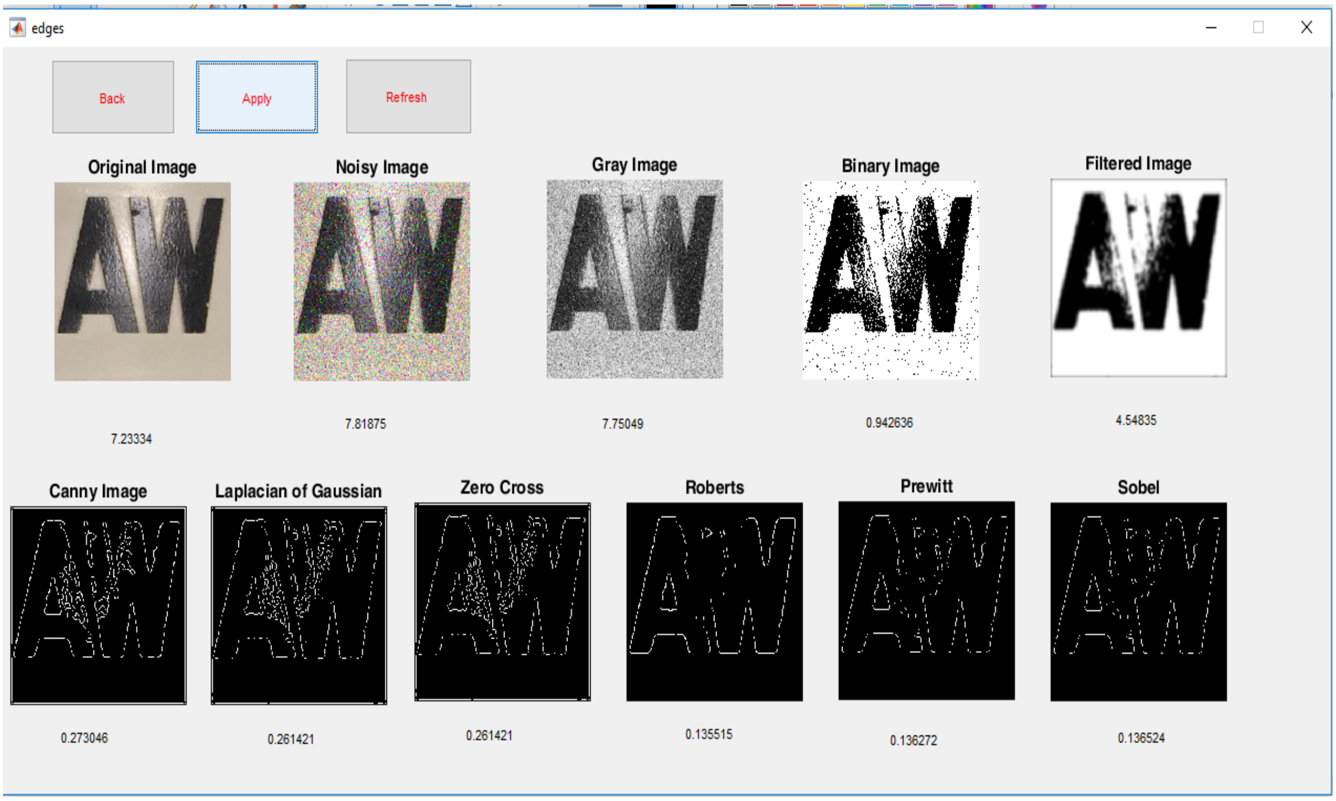

Fig 16 Plot to all edge detectors comparison in single GUI

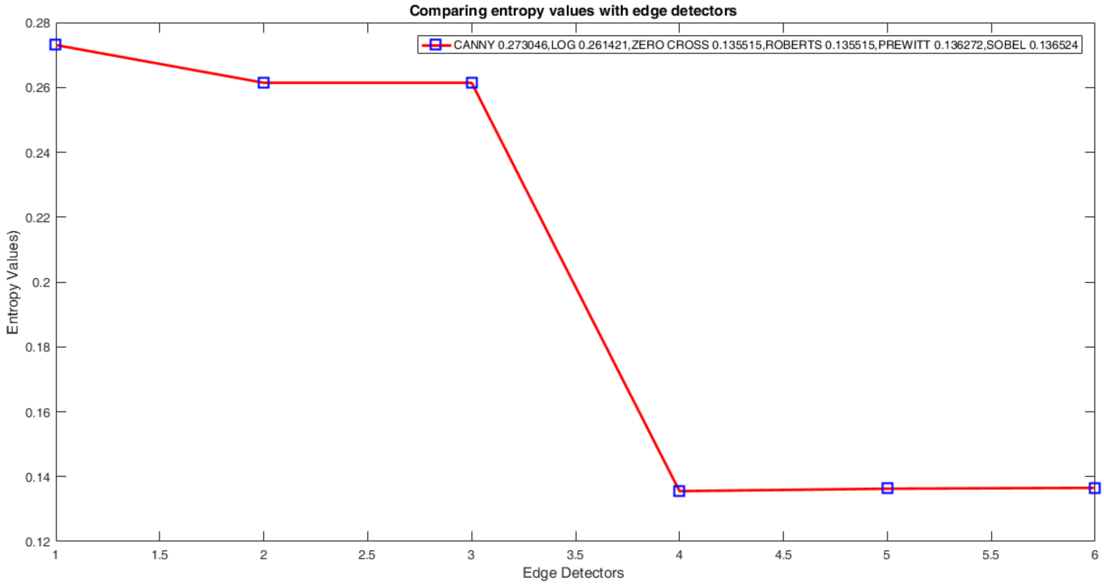

Fig 17 Plot to show entropy values comparison for image AW

5. CONCLUSION

To extract edges, we just need to select specific algorithm. So, to select the algorithm to detect edge entropy values and outputs are compared. In the paper entropy values of Original image, Noisy Image, Gray Image, Binary image, Filtered Image, image output from Canny edge detection, Image output from Prewitt edge detection, Image output from Sobel edge detection, Image output from Roberts edge detection, Image output from Laplacian of Gaussian edge detection, Image output from ZERO cross edge detection are measured. From the six algorithms used in the paper:

5.1. Based on the entropy values

5.1.1. Image1

Canny>Roberts>Prewitt=Sobel>LOG=Zero Cross

5.1.2. Image 2

Canny>LOG>Sobel>Prewitt>Zero Cross=Roberts

5.2. Based on output from matlab2016a

5.2.1. Image1

Canny>LOG>Zero cross>Prewitt>Sobel>Roberts

5.2.2. Image

Canny>LOG>Zero cross>Sobel>Prewitt>Roberts

So, from the above analogy we can say that canny produces optimum results and Roberts being produces worst results. So, we can conclude canny is the most suitable algorithm.

Also, letters and alphabets contain sharp edges which makes it difficult to completely detect these edges. Canny easily detects these sharp edges easily. On comparing entropy values, canny produces high value and Roberts & Sobel produces least entropy values. So, we conclude that more pixels are used to retain structure of the image which creates more number of closed region in the output.

LOG & Zero cross produce better results as they are calculated on the second derivates of pixel values by calculating rate of change of pixels in the image. Sobel, Prewitt, Roberts is based on single derivate. So, the gradient vectors magnitude & direction is calculated but the amount of change is not calculated.

So, we conclude that if the image contains significant number of sharp edges go for canny, log, zero cross but if the sharp edges are not in specific number then go for Sobel, Prewitt, Roberts. Also, calculations for Single derivates are less complex than double derivates so system scans the image faster in single derivates.

6. REFERENCES

[1] syed irfanullah, azeezullah ,danh anh huynh , "canny edge detection report", image processing 159.731.

[2] poonam kumari, sanjeev kumar gupta," sapog edge detection technique gui using matlab", issn(online): 2320-9801 issn (print): 2320-9798

[3] dharampal, vikram mutneja," methods of image edge detection: a review", electrical & electronic systems dharampal and mutneja, j electr electron syst 2015, 4:2 http://dx.doi.org/10.4172/2332-0796.1000150

[4] dr.s.vijayarani, mrs.m.vinupriya, "performance analysis of canny and sobel edge detection algorithms in image mining", issn(online): 2320-9801 issn (print): 2320-9798

[5] poonam dhankhar, neha sahu," a review and research of edge detection techniques for image segmentation" international journal of computer science and mobile computing issn 2320–088x ijcsmc, vol. 2, issue. 7, july 2013, pg.86 – 92

[6] g.t. shrivakshan,dr.c. chandrasekar," a comparison of various edge detection techniques used in image processing ", ijcsi international journal of computer science issues, vol. 9, issue 5, no 1, september 2012 issn (online): 1694-0814

[7] raman maini¸ dr. himanshu aggarwal," study and comparison of various image edge detection techniques", international journal of image processing (ijip), volume (3) : issue (1) raman maini & dr. himanshu aggarwal