A novel high-speed image processing technique for detecting edges using abs-Laplacian kernel

A Ravi Kant
Department of Electrical & Electronics Engineering
Government Engineering College, Thrissur, India
ravikant0123@gmail.com

Abstract – Delineating patterns that are alike or in other words, detecting edges that separates them is the most critical step in image processing. There are various methods available e.g Sobel, Prewitt, Canny based edge detection etc. But most of them are either time consuming or allergic to noise levels and therefore tradeoff comes into the context where we need to choose between the methods that are less computationally expensive with those that provide clear edges. Techniques that allows finding edges as clear as possible without compromising with the computational load is always preferable. Here, I report a novel edge detection technique: abs-Laplacian which has reduced complexity of 7N that requires nearly half the amount of computation involved in Sobel and Prewit which has 15N and 13N complexity respectively. Further, quality of the edges does not show any significant difference with the either of them suggesting that the technique is better both in terms of speed and edge quality while the technique doesn't show up any aberrant fluctuations in the intensity plots and that the derivatives are bit lesser and wider in comparison to the peaks observable upon Sobel and Prewitt treatments, it further suggests that the method may hold some key potential in negating the noise levels to an extent, but subject to further analysis.

Keywords – Sobel, Prewitt, Laplacian kernel

I. INTRODUCTION

Edge detection is one of the important tasks involved in the image processing. The idea of edges detection can be understood in way humans interprets them in their daily lives. Consider a checkerboard containing boxes of white and black arranged alternatively along the two Cartesian coordinates. Let's say that white carries a value '1' and the black as '0'. When labeling with these values we therefore get a binary image, where some patches are zeros while some are ones. If we ponder on this image we observe that there are no variations within these patches or in other words what we see is almost the same. When we traverse from while to black boxes there is an abrupt change in the visual patterns and this is exactly how the humans would delineate a region of uniform pattern from the rest or its neighbors. Traditionally edge detection principle involves carrying out spatial derivatives in both coordinates. Regions that are almost alike would show lower derivative while the abrupt changes would appear as spikes and clearly marks a boundary between the two adjacent regions.

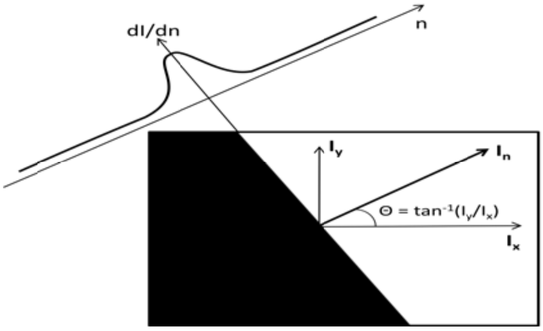

Figure1. Here is an example illustrating the idea of edge detection.

(Figure1) When, traversing along the normal direction, from the black to white regions as indicated by the vector 'n', we encounter change in intensity from low to high. If one were to take the 1st derivative along this direction he would observe that at the left end it is zero while on the right extreme also they vanish. But at the interface, when entering the white regions we see that gradients values go from zero to a higher values and reaches a peak which lies exactly at the middle. Thereafter the gradients reduce to zero. Remember that 1st derivative peaks may not be always small in width; a blurred image of this figure would have the effect of widening the distributions. 2nd derivatives results in 2 peaks, one peak resulting from the rising edge of the 1st derivative and 2nd peak from the falling edge. Zeros crossing in the 2nd derivates indicated an edge but then can we generalize this concept for noisy images? Further, higher order derivatives would result in peak pairs which would bear no physical meaning. It is difficult to get an ideal image in the real environments. A real image is flooded with lots of signals; objects, shadows, reflections, transmittance, Gaussian noise and others. Given the kind of complexity we are looking at, finding sharp peaks is not the kind of answers one should always seek; instead we should always let ourselves in finding out those subtle changes in the intensity that may be constitute an edge. Tracking out the intensity variations at regions of interest and correlating with the local pixel values would provide more meaningful insights in framing better edge detection techniques.

There are many techniques available to resolve the edges e.g. Sobel, Prewitt, Laplacian, Laplacian of the Gaussian, Gaussian of the Laplacian, Canny, Susan etc [1-8]. A priori knowledge on the kind of images being dealt is utmost necessary in designing robust edge detection algorithms. It is also important to understand that the edge detection techniques are highly sensitive to noise levels and therefore, as the images becomes more deteriorated by the various means it becomes more difficult to recognize the edges in the background of noise [9, 10]. Sobel and Prewitt, which are highly popular methods, are good enough to resolve edges from clear images that contain low noise level. As the noise levels go higher in intensities in comparison to the desired signal then there is high probability that wrongs edges are detected. Authors generally use pre-filter e.g. Gaussian or Diffusion or some smoothening techniques to eliminate the noise level so that edge detection technique can provide clear details of the edges. Nonlinear filtering are also being explored [11]. The theory of noise reduction still remains a challenging one. Speckle noises that arise in Ultrasound imaging is one of the highly studied subject. Authors say that they arise out of echoes arising due to the heterogeneous medium [12]. Highly computational methods that involve designing of complex pre-filters or noise elimination and others are also on pursuit, e.g. wavelet [13, 14], statistical [15], neural network [16], support vector machines [17] etc.

Noise modeling is an entire research area by itself, but before we can go on modeling noise it is important to understand more about the currently available edge detection methods and understand their potentials so that they can be tuned accordingly and made immune to noisy images. By saying that a methods is immune I mean that methodology is independent of the noise signals and whatever be the noise they are not going to alter the edge qualities. Of course it is very challenging to learn how and what the noise signals are and how can the the methods be made immune to it, but nevertheless when dealing with method we need to be aware of the noisy pixels in the images. Shadows are not noisy, reflection and speckles at times can be considered as undesirable but still they cannot be treated as the noisy signals. When we say, Sobel kernel is not the methods to be used for noisy images, then, what are we missing? Though, I haven't carried out noise analysis in the current context, it is important note that performance of edge detection techniques need to be correlated with the noise levels at the end.

In course of my research on the image processing I came across a novel technique (abs-Laplacian), to perform edge detection faster than Sobel and Prewitt. Being the first of its kinds I did a short analysis to figure out if the technique is good enough to predict the edges as close as more popular ones.

II. THEORY

A. *Image convolution with kernel*

Convolution is a mathematical operator that convolves two functions to give rise to another function. Convolution of an image is carried out on a pixel by pixel basis by multiplying the sub-image (I) with a kernel (K) and summing up the matrix elements (eq1.1 and eq1.2). Due to their filtering properties, they are used to smoothen, sharpen, intensity or enhance the image quality.

$$I_c = I \cdot K \tag{1.1}$$

$$I_c(i,j) = \sum_{k=1}^{m}\sum_{l=1}^{n} I(i+k-1,+l-1)\cdot K(k,l) \tag{1.2}$$

Some technique use 2 kernels for computing the convolution, one along the x-axis (G_x) and other along y-axis (G_y). In such cases where more than one kernel is employed, the final convoluted image is computed as the root of the sum of the squares of the convolution results of all axes. Some authors approximate the final convolution as a mere addition of the absolute of the convolution of along each direction rather than spending extra computation in finding the roots and squares [9, 18, 19**]**.

B. *Commonly used edge detection kernels (Sobel and Prewitt: 1st derivative operators)*

Sobel and Prewitt are the one among the most popular edge detection techniques. They involve two kernels one that performs convolution along x and the other along y direction. The final gradients are computed as the magnitude of the two components and at times they are approximated as a mere sum as discussed earlier. The direction of the normal to the edges or the orientation of the edge is given by inverse tangent of the 'x' component of the convolution upon the 'y' component.

Table1. Kernals for Laplacian 3x3 , Sobel and Prewitt

Kernel	Laplacian	Sobel	Prewitt
G_x	0 1 0 1 -4 1 0 1 0	-1 0 1 -2 0 2 -1 0 1	-1 0 1 -1 0 1 -1 0 1
G_y		-1 -2 -1 0 0 0 1 2 1	1 1 1 0 0 0 -1 -1 -1

C. *Absolute of the Laplacian (2^{nd} derivative operator)*

Absolute of the Laplacian is a modified version of the way Laplacian operators are used for edge detection. With the kernel containing 1's in the four orthogonal directions and a 4 at its base or centre point, the convolution is carried out with the input image to obtain a value at the corresponding (i, j)th location. Laplacian operators are highly sensitive to noise; moreover the edge quality appears blurred in comparison to the sobel and prewitt techniques. May be, one of the reasons could be that the Laplacian mask introduces negative values, which, rather than depicting the edges they are faded.

$$I_L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial x^2} \tag{1.3}$$

The novel technique takes advantage of the absolute operation to eliminate the fading away of the edges. Unlike the usual procedure of multiplying the local 3x3 sub matrix of the image with the Laplacian operator, we would compute the Abs-Laplacian in a bit different manner. Consider, an input image 'p1' of size (nx, ny). There are four basic operations to be performed which involves, taking the intensity differences of reference or the base pixel with its four neighbour pixels. It can be probably thought as taking a copy of the input image and shifting it in all the four directions by a small distance 'ns' in units of pixels. Then, we later perform a subtraction with the input image and take an absolute and take their sum to obtain output image 'p2'. Matlab routine for carrying out abs-laplacian of an image is shown below,

```
function [p2] = abs_laplacian_3 (p1, nx, ny, ns)
   q0 = p1 (ns:nx-1, ns:ny-1);
   q1 = p1 (1:nx-ns, 1:ny-ns);
   q2 = p1 (ns+1:nx, 1:ny-ns);
   q3 = p1 (1:nx-ns, ns+1:ny);
   q4 = p1 (ns+1:nx, ns+1:ny);
   p2 = abs (q1-q0) + abs (q2-q0) + abs (q3-q0) + abs (q4-q0);
end
```

III. RESULTS AND DISCUSSIONS

A. *Edge profiles of abs-Laplacian, Sobel and Prewitt*

Finding out differences among these methods has been in interesting topic to study. We do understand that there are many techniques available for edge detection but when it comes to the applications where real time edge detection is utmost necessary as in the case of robotic vision systems, choice of methods becomes very important. Good quality of the edges obtained also implies better detection of objects and good sensitivity. Kernels that are more than 3x3 in dimension, presuming that all are non-zero would involve matrix multiplications of more than 9 that would need more computational power. The question I am trying to understand is; whether is it possible to develop a newer method that can enable edge detection to acceptable limits and also take time lesser time than the currently available ones? Well, Sobel and Prewitt are the commonly used techniques to start my analysis. In these methods, we compute 1st derivatives of the intensity along the Cartesian coordinates that involves 3 pairs of adjacent neighbors for each kernel. While computing the spatial derivatives, I thought if we were to compute the derivates using the neighbor pixels then we should consider only sufficient number of neighbors that can delineate the edges to certain extent. In the mean time, I came across the idea of taking sum of the absolute differences of the base pixel from its neighbors. Of course to make the edges as clear as possible I resorted towards the concept of Laplacian. Laplacian kernel are nothing but an approximation of the 2nd derivative that considers the base pixel and 2 adjacent neighbors along each coordinate. The convolution with new kernel (abs-

Laplacian) was carried out for 200x200 pixel of an arbitrary image (Figure2). Edge detection using Sobel and Prewitt methods were also carried out to understand the quality of the images and the intensity levels obtained thereof against those edges resulting from abs-Laplacian operation.

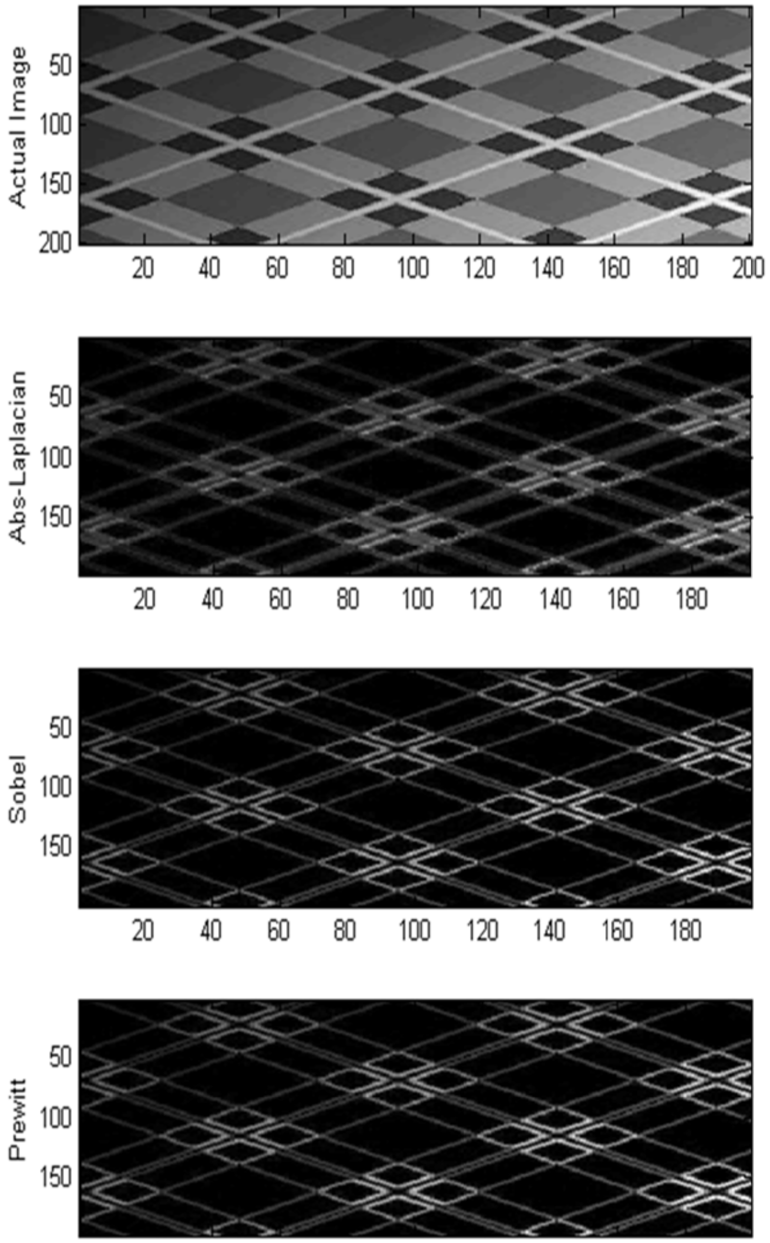

Figure2. (A) is a 200x200 size image and their edges as determined by Abs-Laplacian (B), Sobel (C) and Prewitt (D) are shown.

Quality of the images obtained using abs-Lapalacian were of good quality and matches with the intensity levels of Sobel and Prewitt. The parameter ‘ns’ tell us the amount of shift that we introduce from the base image or the actual image to obtain four images that are equivalent to the 1’s in the kernel (Table2). The higher the shift broader the edges would appear, while on the other hand, a shift of one would cause narrowing. Therefore, appropriate choice of the parameter ‘ns’ is also important in reliable detection of edges.

To have a clear vision of the quality of the edges, intensity plots were analyzed at the 100th rows as shown in figure3. Red, black and blue colored lines correspond to abs-Laplacian, Sobel and the Prewitt derivatives. It appears that the Sobel and Prewitt edges exactly coincide while the abs-Laplacian edges show little deviations. At places where the Sobel peaks are observed abs-Laplacian are little broader and also little lower in magnitude. Prominent edges are certainly captured with good accuracy. In general, the overall profile and visual inspection of the abs-Laplacian edges are comparable to the existing techniques suggesting this newer technique can be confidently used as a replacement to the existing ones.

Table2. Kernals for abs-Laplacian with shift = 2

Kernel	abs-Laplacian				
Gx	0	0	1	0	0
	0	0	0	0	0
	1	0	-4	0	1
	0	0	0	0	0
	0	0	1	0	0

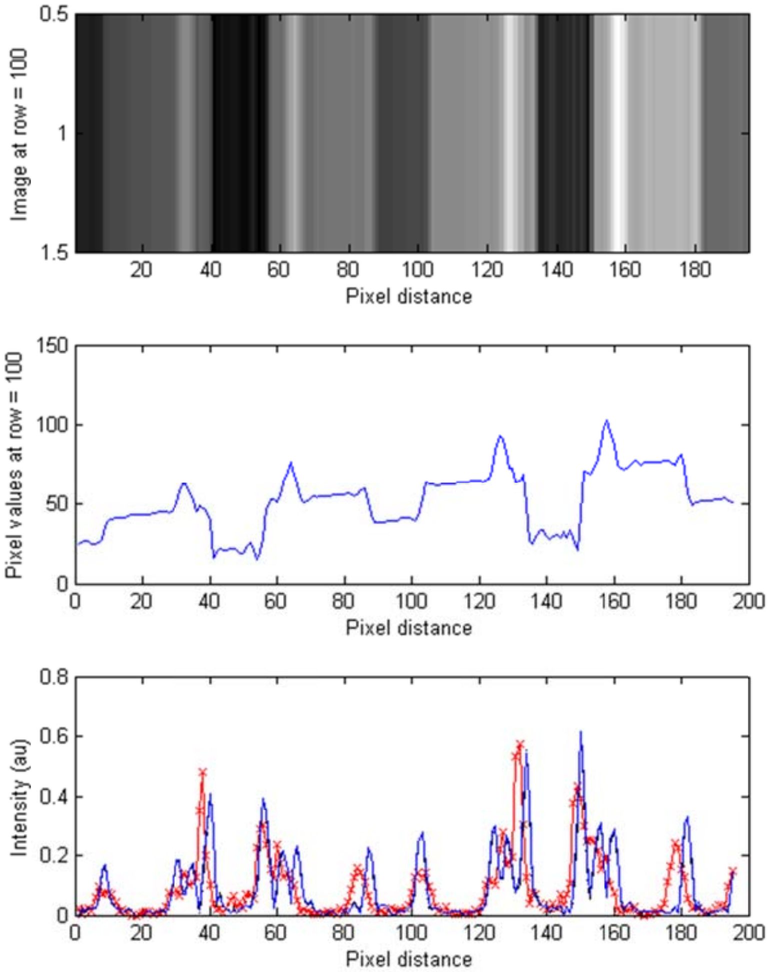

Figure3. Contour map of the image (A) at the 100th row is shown with the corresponding pixel values in (B). Notice the abrupt changes in the 1D signal at 7 positions along the x-axis. Sobel and Prewitt operations which are 1st derivative operators resulted in spikes at these regions as shown in (C) by red (Abs-Laplacian) and black (Sobel) respectively. Black line is not visible due to their near perfect alignment with blue line (Prewitt). From the red line it seems that they almost tend to follow the edges as detected by sobel and prewitt suggesting that Abs-Laplacian also can be used as alternative to these kernels.

B. *Kernel features*

Is that the abs-Laplacian is restricted only to fewer images? I wanted to know if the methods can predict other images as good as the other two methods. About 28 HD complex images were considered regardless of whether they are sharp images or noisy or blurred ones. Presuming that whatever be the kinds of images I am dealing with, abs-Laplacian should predict to a good extent what the Sobel and Prewitt does. Cumulative intensity was considered as a measure of the edge quality for the 28 samples and the results were plotted as shown in figure4. In contrast to the Sobel – Prewitt similarity of the edge intensities, I found that the abs-Laplacian almost coincides with Prewitt's predictions. Sobel predicts are higher than the rest and maintain an almost constant gap across the samples. The best things about these 3 techniques are that they are almost similar by nature and show same amount of cumulative intensities, no matter what kind of images we are dealing with. It is difficult to reason on why the curves nearly align with each other irrespective of the kind of images.

Probably, the kernel elements are very crucial what I mean is that abs-Laplacian and the Prewitt has only 1’s as the elements suggesting that they don’t keep biasing over one pixel like the Sobel kernels which has an element of value equal to 2.

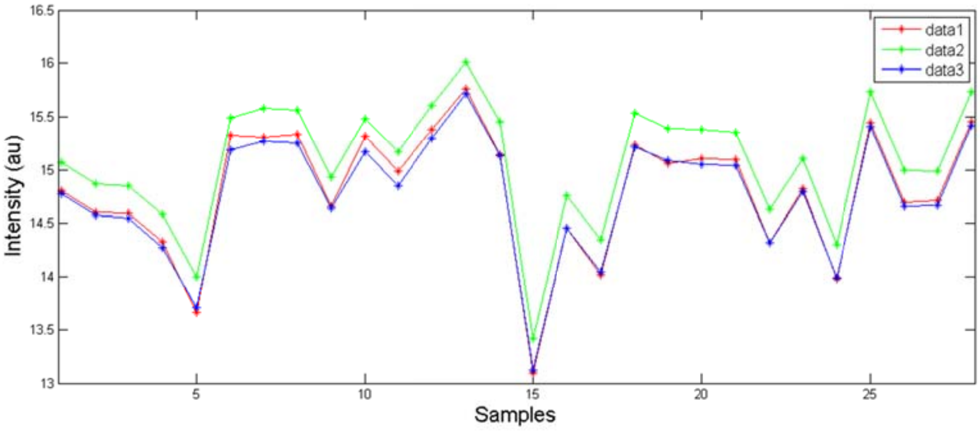

Figure4. Cumulative intensity of the edges computed for three operators: Abs-Laplacian (shown as red line), Sobel (green) and Prewitt (blue) are shown for 28 sample images.

In certain applications the amount of computation plays a crucial role e.g. robotic vision where by a edge detection tasks are carried out continuously, should be very simple and elegant enough to capture edges from the images quickly. Lower computational codes are always preferred but also they should be robust and reliable enough to identify the edges as best as possible.

The algorithm complexities were calculated in terms of the number of additions involved. Rather than finding complexity for the whole image I focused on finding out the computation involved for computing convolution at 1 pixel. The abs-Laplacian has 4 non-zero elements and therefore 4 multiplications are needed for matrix multiplication and 3 additions to obtain the final convolution sum. Alternatively, since the non-zero elements are three 1’s and one -4; the number 4 can go as 4 additions so that the final complexity comes to a mere sum of 8 elements that would need 7 successive additions to be performed. Sobel based technique involve 2 kernel each containing four 1’s and two 2’s which can be considered as convolution sum of 8 elements that would take 7 additions. Both the kernels it would take 14 additions plus 1 more addition to combine the x and y components. So, a total of 15 minimal additions are needed. Remember that I am trying to find the reduced algorithm complexity for the 3 methods and wherever possible I try to eliminate any extra computation required with the approximations authors have used in their study (refer to section IIA). Prewitt method required 6 additions for each kernel and so a total of 12 go for both the kernels plus 1 addition to obtain the final convolution sum. Therefore, 13 additions are required.

Also, the amount of time the Matlab actually takes to execute the algorithm was also measured using system clock command (cputime). We carried out the analysis for over 28 set of images and determined an average value. The abs-Laplacian code took only 0.28 seconds in comparison to the 16.44 and 16.41 by Sobel and Prewitt respectively. Remember that this is the time measured over Matlab platform and need not correlate with the complexity in proportionate manner. Since, there are many inherent .m function calls made by the program it actually delays the code execution time. However, what we learn is that the abs-Laplacian out performs in the context of edge detection and also comparable to the other methods in terms of image quality. Though more mathematical analysis is needed to understand more about the limits of the new method, as of now it appears to that this method has a great potential in high speed image processing applications.

Table3. Average time taken by the various methods for edge detection and the corresponding algorithm complexity are shown.

No.	Method	Time, sec	Complexity, (+)
1	abs-Laplacian	0.2808	7
2	Sobel	16.4425	15
3	Prewitt	16.4113	13

IV. CONCLUSION

Edge detection is the key step in the image processing and if one were to devise a robust ways of determining edges, one should know the inherent qualities of edge detection techniques. The newer technique: abs-Laplacian seems to be a promising tool in detecting edges taking less than half the amount of time taken by the popular edge detection methods without compromising edge qualities. It is difficult say anything about the noise immunity of the abs-Laplacian but further analysis needs to be carried out.

V. REFERENCES

[1] B. Horn Robot Vision, MIT Press, 1986.

[2] E. Davies Machine Vision: Theory, Algorithms and Practicalities, Academic Press, 1990.

[3] R. Gonzalez and R. Woods Digital Image Processing, Addison-Wesley Publishing Company, 1992.

[4] D. Marr Vision, Freeman, 1982.

[5] D. Vernon Machine Vision, Prentice-Hall, 1991.

[6] R. Haralick and L. Shapiro Computer and Robot Vision, Vol. 2, Addison-Wesley Publishing Company, 1993.

[7] D. Marr and E.C. Hildreth, “Theory of Edge Detection,” Proc. Royal Soc. London B, vol. 207, pp. 187-217, 1980.

[8] J. Canny, “A Computational Approach to Edge Detection,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 8, pp. 639-643, Nov. 1986.

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

[10] Adnan Khashman, Optimal Scale Edge Detection Utilizing Noise within Images, 46 SYSTEMICS, CYBERNETICS AND INFORMATICS VOLUME 1 - NUMBER 2

[11] Fabrizio Russo, Annarita Lazzari, Color Edge Detection in Presence of Gaussian Noise Using Nonlinear Prefiltering, 352 IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, VOL. 54, NO. 1, FEBRUARY 2005

[12] Ju Won Kwon and Yong Man Ro, Improvement of Speckle Noise Reduction Using Multi-resolutional Coherence Measurement in Ultrasound Image, 32nd Annual International Conference of the IEEE EMBS, Buenos Aires, Argentina, August 31 - September 4, 2010

[13] F. Faghih and M. Michael, “Combining Special and Scale-Space Techniques for Edge Detection to Provide a Spatially Adaptive Wavelet-Based Noise Filtering Algorithm,” IEEE Trans. Image Processing, vol. 11, pp. 1062-1071, Sept. 2002.

[14] Wenshuo Gao; Xiaoguang Zhang; Lei Yang; Huizhong Liu, "An improved Sobel edge detection," Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on , vol.5, no., pp.67,71, 9-11 July 2010

[15] S. Konishi, A.L. Yulle, J.M. Coughlan, and S.C. Zhu, “Statistical Edge Detection: Learning and Evaluating Edge Cues,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 25, no. 1, pp. 57-74, Jan. 2003.

[16] Johnson, Christopher M.; Page, Edward W.; Tagliarini, Gene A., Neural network approach to edge detection and noise reduction in low-contrast images. Proc. SPIE Vol. 2492, p. 156-164, Applications and Science of Artificial Neural Networks, Steven K. Rogers; Dennis W. Ruck; Eds.

[17] Hilario Gómez-Moreno, Saturnino Maldonado-Bascón, and Francisco López-Ferreras. 2001. Edge Detection in Noisy Images Using the Support Vector Machines. In *Proceedings of the 6th International Work-Conference on Artificial and Natural Neural Networks: Connectionist Models of Neurons, Learning Processes and Artificial Intelligence-Part I* (IWANN '01), José Mira and Alberto Prieto (Eds.). Springer-Verlag, London, UK, UK, 685-692.

[18] 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

[19] Shen-Chuan Tai; Shih-Ming Yang, "A fast method for image noise estimation using Laplacian operator and adaptive edge detection," Communications, Control and Signal Processing, 2008. ISCCSP 2008. 3rd International Symposium on , vol., no., pp.1077,1081, 12-14 March 2008