Volume 4, Issue 5, May 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Condition Assessment of Metallic Objects Using Edge Detection

Rajiv Kumar Ranjan
M.Tech Final Year Student
Deptt. of ECE,MMEC, MMU, Mullana
Ambala, Haryana, India

Tarun Gulati
Associate professor
Deptt. of ECE,MMEC, MMU, Mullana
Ambala, Haryana, India

Abstract-Edge detection refers to the process of identifying and locating sharp discontinuities in an image. So, edge detection is a vital step in image analysis and it is the key of solving many complex problems. In this paper, the main aim is to study the theory of edge detection for image segmentation using various computing approaches based on different techniques applied on a corroded surface image In this paper the comparative analysis of various Image Edge Detection methods is presented .The evidence for the best detector type is judged by studying the edge maps relative to each other through statistical evaluation. Upon this evaluation, an edge detection method can be employed to characterize edges to represent the image for further analysis and implementation. It has been shown that the proposed edge detection algorithm performs better than all these operators under almost all scenarios.

Keywords: Dilation, Edge detection techniques, Threshold, Hole filling.

I. INTRODUCTION

Metallic surfaces/objects get corroded with age, usage and surrounding environment. Therefore, their condition assessment is necessary. The metallic surfaces/objects can vary in shape, size and usage, so their corrosion will vary and can be very difficult to assess. For example, larger boiler tanker, other metallic surface of larger ship body parts which is inside the water. There are many examples in present day life for the corroded metallic object that takes a lot of time to do measure how much part of the metallic surface is fine and how much part is corroded, Seeing this difficulties, we are willing to do a work on solving this problem. Therefore, the research to be done will focus on using image processing for identifying the corroded and non-corroded regions of metallic objects for the purpose of condition assessment. Small surfaces can be easily monitored visually; but for large surface it is very difficult and time consuming to do condition assessment visually. But it is easier to take the image of a large surface, so the research will be done to develop an image processing based algorithm for doing condition assessment of corroded metallic surfaces/objects. The algorithm will use the edge detection for detecting the corroded region and hole filling to determine the area for doing condition assessment.

II. EDGE DETECTION IN DIGITAL IMAGES

Edge is a part of an image that contains significant variation. Edges are those places in an image that correspond to object boundaries. Edges are pixels where image brightness changes. Physical edges are produced by variation in the reflectance, illumination, orientation, and depth of scene surfaces. The most common edge types are steps, lines and junctions. An Edge in an image is a significant local change in the image intensity, usually associated with a discontinuity in either the image intensity or the first derivative of the image intensity.[1] An edge is the boundary between an object and the background. The edge representation of an image significantly reduces the quantity of data to be processed, yet it retains essential information regarding the shapes of objects in the scene. The various types of edges of image are shown in the Fig.1. Edge detection refers to the process of identifying and locating sharp discontinuities in an image. So, edge detection is a vital step in image analysis and it is the key of solving many complex problems.

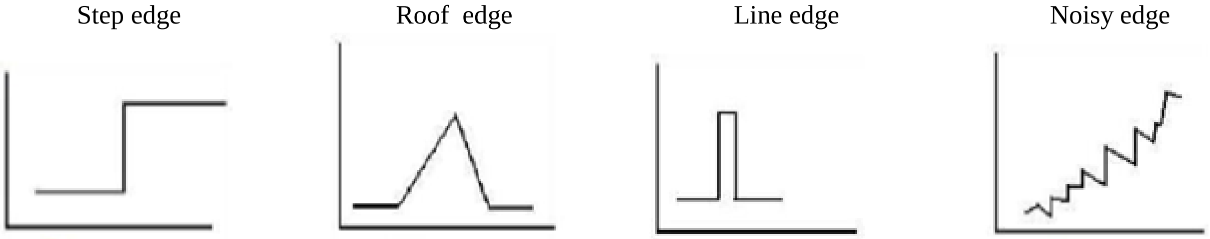


Fig 1:-Different types of edges.[4]

Edge detection is a fundamental tool used in most image processing applications to obtain information from the frames as a precursor step to feature extraction and object segmentation. The edge detection have been used by object recognition, target tracking, segmentation, data compression ,and also help for well matching, such as image reconstruction and so on.

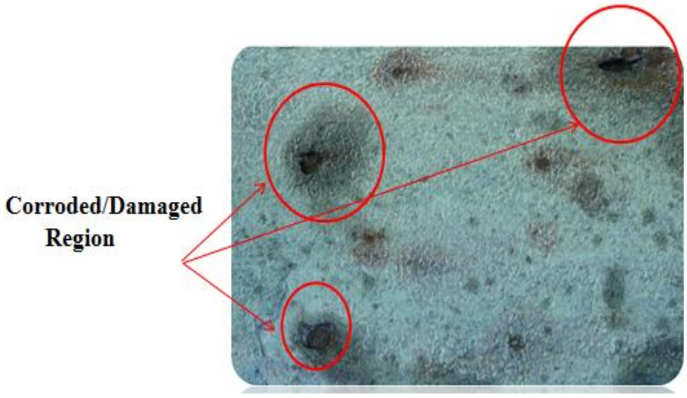
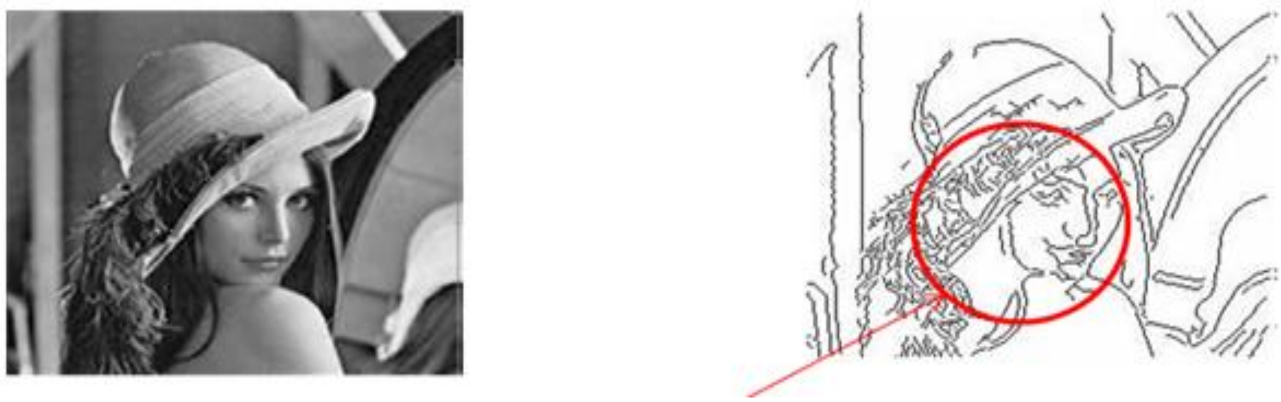


Fig 2:- Small corroded surfaces or damage region.

On applying edge detector to an image results in a set of connected curves that indicate the boundaries of objects, this boundaries on surface markings shows discontinuities in surface as shown in fig 3.



(a) Lena image (b) Edge detection using Canny edge detector

Fig 3:- The result of applying an edge detector to an image

III DIFFERENT EDGE DETECTION TECHNIQUES

3.1 Sobel operator:- The Sobel operator consists of a pair of 3×3 convolution masks shown below in Table 1 . One kernel is remain constant and the other is rotated by 90° shown in table 2. The kernels can be applied separately to the input image to produce separate gradient component measurement in each orientation (Gx and Gy). Magnitude is given by the Mathematical equation [9]. Gx is horizontal mask and Gy is vertical mask .

Table-1

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

Table-2

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

$$|G| = \sqrt[2]{Gx^2 + Gy^2} \qquad (1)$$

The Sobel edge detection method is introduced by Sobel in 1970. The Sobel technique performs a 2-D spatial gradient quantity on an image and so highlights regions of high spatial frequency that correspond to edges. This is very similar to the Roberts Cross operator.[3]

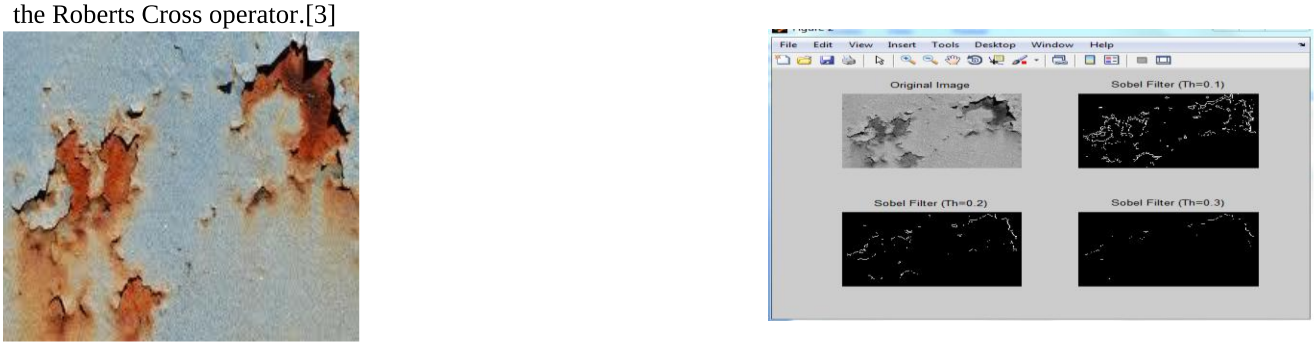


Fig 4:- The result of applying Sobel edge detector to an image

3.2 Perwitt operator:- Like Sobel operator , Perwitt edge detection technique consists of co a pair of 3×3 convolution kernels . One kernel is simply the other rotated by 90°. These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid. this operator uses two 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical. The Prewitt edge detection is proposed by Prewitt in 1970. Prewitt operator edge detection masks are the one of the oldest and best understood methods of detecting edges in images. The operator uses two 3x3 size masks which gives more information regarding the direction of the edges as they consider the nature of data obtain the approximations of derivatives for the horizontal and vertical edge changes, separately.

$$edge\ \ Magnitude = \sqrt{{P_1}^2 + {P_2}^2} \qquad (2)$$

$$Edge\ Direction = \tan^{-1}[\frac{\mathrm{P_1}}{\mathrm{P_2}}] \qquad (3)$$

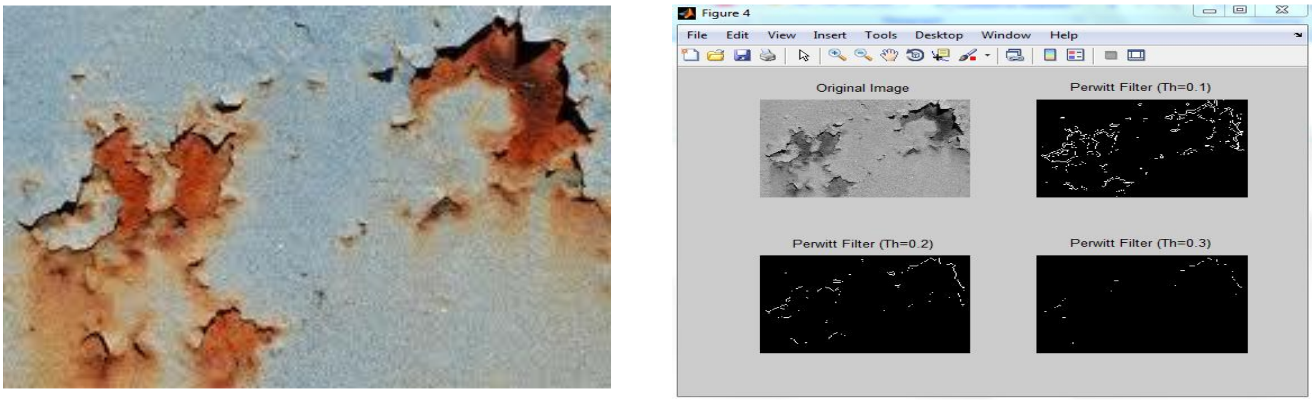


Fig. 5:- The result of applying Perwitt edge detector to an image

3.3 Robert operator:- The operator consists of a pair of 2×2 convolution kernels . One kernel is simply the other rotated by 90°. This is very similar to the Sobel operator. These kernels are designed to respond maximally to edges running at 45° to the pixel grid, one kernel for each of the two perpendicular orientations. The Roberts cross operator provide a simple approximation to the gradient magnitude[8].

$$G\big[f[i,j]\big] = |f[i,j] - f[i+1,j+1]| + |f[i+1,j] - f[i,j+1]| \qquad (4)$$

Using convolution masks, this becomes and Where Gx and Gy are calculated by using the following given equation.

$$\mathrm{G[f[i,j]]}=|Gx| + |Gy| \qquad (5)$$

$$\mathrm{Gx}=\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \qquad \mathrm{Gy}=\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad |G| = \sqrt{(Gx^2} + Gy^2) \qquad (6)$$

As 2 x 2 gradient operator, the differences are computed at the interpolated point [i + 1/2, j + 1/2]. The Roberts operator is an approximation to the continuous gradient at the interpolated point and not at the point [i, j] as it might be expected.

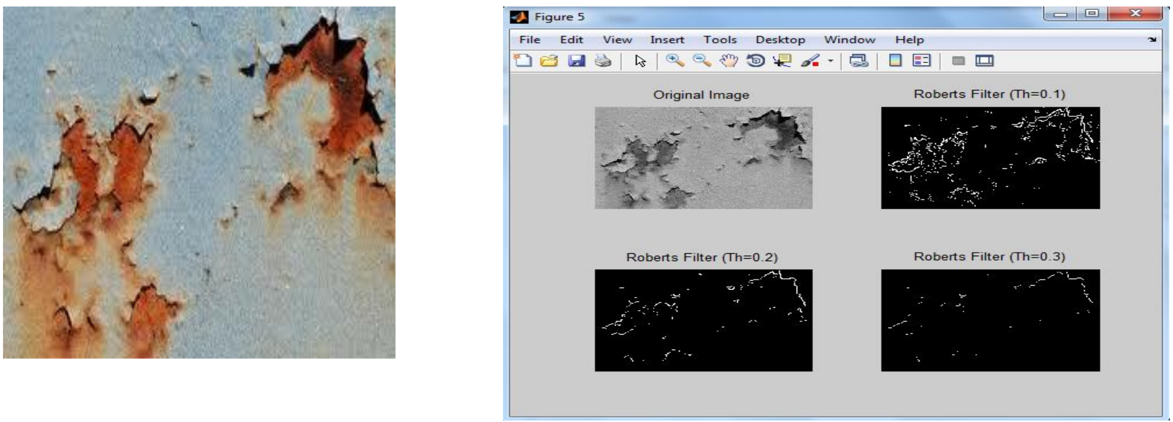


Fig. 6:- The result of applying Robert edge detector on an image

3.4 Canny filter:- This method was proposed by John F. Canny in 1986. Even though this method is quite old but is still used because of its precision in edge detection. The main advantage of this method is elimination of multiple responses to a single edge. It also have a good localization property, means the detected edges are much closer to the real edges. This method is not easily disturbed by noise and can keep the good balance between noise and edge detection. It can detect the true weak edge. good localization ,low error rate , uniqueness of the multiple responses. Steps in Canny edge detector:- The Canny Edge Detection Algorithm has the following Steps: Smooth the image with a Gaussian filter than compute the gradient magnitude and orientation using finite-difference approximations for the partial derivatives. Now apply non maxima suppression to the gradient magnitude, The Smoothing is computed as I[i,j] to denote the image. G[i,j,σ] has to be a Gaussian smoothing filter where σ is the spread of the Gaussian and controls the degree of smoothing[7]. The result of convolution of I[i,j] with G[i,j,σ] gives an array of smoothed data as:

$$S[i,j]=G[i,j,\sigma]*I[i,j] \qquad (7)$$

Firstly, the Gradient is calibrated for the smoothed array S[i,j] is used to produce the x and y partial derivatives P[i,j] and Q[i,j] respectively as[5]. The x and y partial derivatives are computed with averaging the finite differences over the 2x2 square. From the standard formulas for rectangular-to-polar conversion, the magnitude and orientation of gradient

$$P[i,j] \approx (S[i,j+1] - S[i,j] + S[i+1,j+1] - S[i+1,j])/2 \qquad (8)$$

$$Q[i,j] \approx (S[i,j] - S[i+1,j] + S[i,j+1] - S[i+1,j+1])/2 \qquad (9)$$

Fig 7:- The result of applying Canny edge detector to an image

$$M[i,j] = \sqrt{P[i,j]^2 + QP[i,j]^2} \qquad (10)$$

$$\theta[i,j] = \arctan(Q[i,j], P[i,j]) \qquad (11)$$

IV SYSTEM MODEL

Step.1 Read the input Image and convert it into grays scale image.

Step 2 Apply the different Edge Detection Techniques with threshold value lies in range of 0.1 - 1 on to the gray scale image to convert it into binary image .

Step 3 To convert the discontinuity in the edges into continuous edges and to make it connected dilation process is used.

Step 4.After the dilation process there are still holes in the interior of the selected region. To fill these holes the imfill function is used.

Step 5.Finally, in order to make the segmented object look natural, the corroded surface boundary is smoothened by eroding the image twice with a diamond structuring element. It is created using the strel function.

Step 6. An alternate method for displaying the segmented object would be to place an outline around the segmented cell. The outline is created using bwperim function.

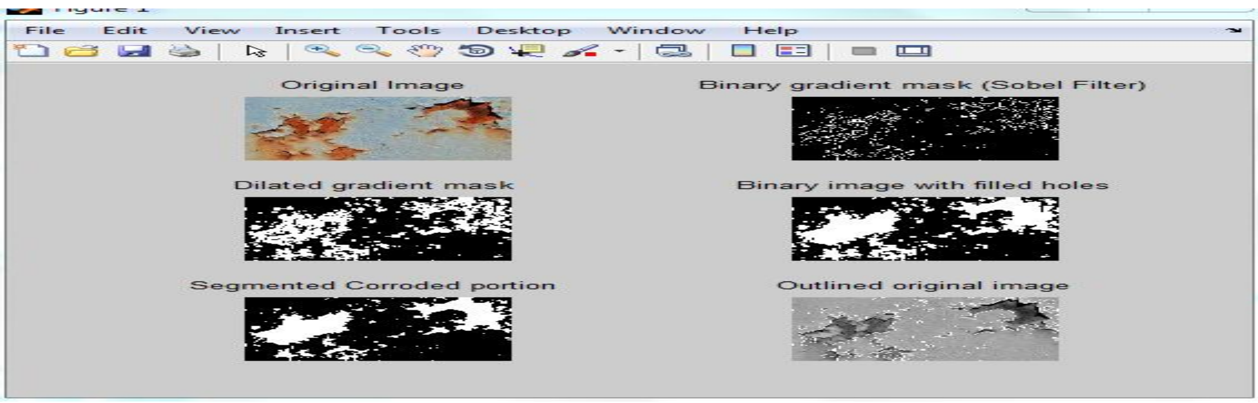


Fig 8:- Process of estimating corroded area of corroded metallic surfaces

V RESULTS

Image of corroded surface is taken as input image. MATLab is used for showing the comparison between different edge detection methods. Results are shown in Table 3. The Table 3 clearly reflects that the corroded area value of the proposed method is the most accurate as compared to other edge detections techniques used. The evaluations and comparisons made through visual observations.

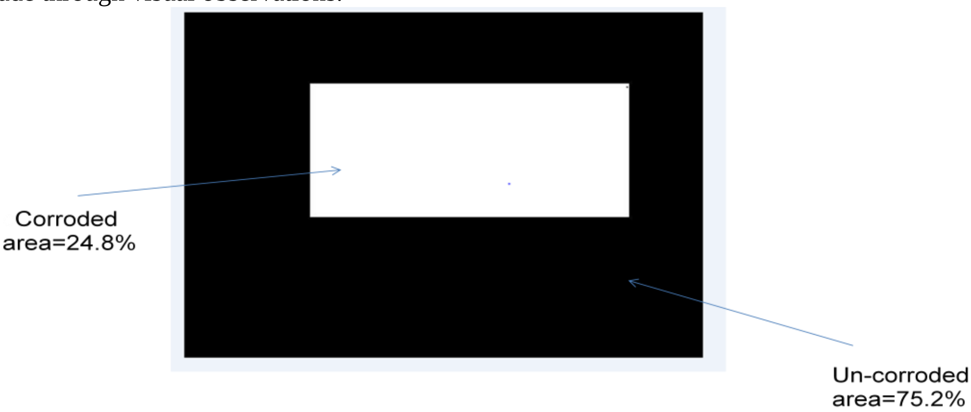


Fig. 9 Test image

Table-3 Comparison of detected corroded area using different edge detection techniques

Actual corroded area of test image	Corroded area detected by Sobel filter	Corroded area detected by Perwitt filter	Corroded area detected by Roberts filter	Corroded area detected by Canny filter	Corroded area detected by my algorithm
24.8%	23.8304%	23.8286%	24.038%	24.038%	24.2747%

VI CONCLUSION

From the results obtained it can be concluded that the sobel filter provides 96.09 % accuracy , Perwitt filter provides 96.08 % accuracy where as Robert and Canny modify filter provides 96.928 % accuracy and the results obtained by proposed modified filter is 97.88% accuracy .from this it is concluded that proposed filter proved to be very effective for calculating corroded area. It can be seen that the modify filter are the best in comparison to Sobel, Perwitt, Robert , canny filters .

REFERENCES

[1] S. Lakshmi and V.Sankaranarayanan (2010) "A Study of edge detection techniques for segmentation computing approaches", Computer Aided Soft Computing Techniques for Imaging and Biomedical Applications, 35-41.

[2] P. Thakare (2011) "A Study of Image Segmentation and Edge Detection Techniques", International Journal on Computer Science and Engineering, Vol 3, No.2, 899-904.

[3] N. Senthilkumaran and R. Rajesh (2009) "Edge Detection Techniques for Image Segmentation – A Survey of Soft Computing Approaches", International Journal of Recent Trends in Engineering, Vol. 1, No. 2, 250-254.

[4] D. Marr and E. Hildreth (1980), "Theory of edge detection", Proceedings of the Royal Society of London, Vol. 207, pp. 187–217.

[5] A. L. Yuille and T. A. Poggio (1986), "Scaling theorems for zero-crossings", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, pp. 15–25.

[6] A. P. Witkin (1983), "Scale-space filtering", in Proc. Int. Joint. Conf. Artificial Intelligence, vol. 2, pp. 1019–1022.

[7] M. Bennamoun, B. Boashash and J. Koo (1995), "Optimal parameters for edge detection", in Proc. IEEE Int. Conf. SMC, vol. 2, pp. 1482–1488.

[8] D. Heric, and D. Zazula (2007), "Combined edge detection using wavelet transform and signal registration", Elsevier Journal of Image and Vision Computing, Vol. 25, pp. 652–662.

[9] M. Y. Shih and D. C. Tseng (2005), "A wavelet based multi resolution edge detection and tracking", Elsevier Journal of Image and Vision Computing, Vol. 23, pp. 441–451.

[10] A. D. Santis and C. Sinisgalli (1999), "A Bayesian Approach to Edge Detection in Noisy Images", IEEE Transactions On Circuits And Systems—I: Fundamental Theory And Applications, Vol. 46, No. 6, pp 686-699.

[11] J. Wu, Z. Yin, and Y. Xiong (2007), "The Fast Multilevel Fuzzy Edge Detection of Blurry Images", IEEE Signal Processing Letters, Vol. 14, No. 5, pp 344-347.

[12] C. Kang, and W. Wang (2007), "A novel edge detection method based on the maximizing objective function", Elsevier Journal of Pattern Recognition, Vol. 40, pp. 609 – 618

[13] J. D. Scargle and M. K. Quweider (1991), "Edge Detection Using Dynamic Optimal Partitioning", Pattern Recognition, Vol. 24(6), pp. 465-478.

[14] R. . Haralick (1983), "Ridge and Valley on Digital Images", Computer Vision, Graphics and Image Processing, Vol. 22, pp. 28-38.

[15] K. Koundinya and B. Chanda (1994), "Detecting Lines in Gray Level Images Using Search Techniques", Signal Processing, Vol. 37, pp. 287-299.

