International Journal of Advances in Engineering Research http://www.ijaer.com/

(IJAER) 2011, Vol. No. 2, Issue No. VI, December **ISSN: 2231-5152**

COMPARATIVE STUDY OF EDGE DETECTION ALGORITHMS ON THE REMOTE SENSING IMAGES USING MATLAB

Harshlata Vishwakarma[1],S.K.Katiyar[2]

[1] GIS Developer, Flairsoft Consulting Group Bhopal 462003 (M.P)

[2] Professor, Geoinformatics Division,Civil Deptt,MANIT Bhopal462016(M.P)

ABSTRACT

In this paper, classified and comparative study of edge detection algorithms are presented. Experimental results prove that Canny operator is better than Prewitt and Sobel for the selected image. Subjective and objective methods are used to evaluate the different edge operators. This paper evaluates the performance of Canny, Sobel and Prewitt Edge Detector for detection of edges in digital images. Further, the various images are examined to validate our results. The software is developed using MATLAB 7.0. It has been observed that Canny operator gives the best results in comparison to the Sobel and Prewitt operator. With the help of remote sensing images and applying edge detection techniques we obtained the roads, vegetation, buildings, vehicle edgeinformation.

Keywords*: Edge Detector, Digital Image Processing, performance comparison.*

INTRODUCTION

Remote Sensing is the science and art of acquiring information (spectral, spatial, and temporal) about material objects, area, or phenomenon, without coming into physical contact with the objects, or area, or phenomenon under investigation. Without direct contact, some means of transferring information through space must be utilized. In remote sensing, information transfer is accomplished by use of electromagnetic radiation (EMR). EMR is a form of energy that reveals its presence by the observable effects it produces when it strikes the matter.

Digital image processing is the use of computer algorithms to perform image processing on digital images. Digital image processing has the same advantages (over analog image processing) as digital signal processing has (over analog signal processing) it allows a much wider range of algorithms to be applied to the input data, and can avoid problems such as the build-up of noise and signal distortion during processing. Digital image processing allows the use of much more complex algorithms for image processing.

Edge detection is a fundamental problem of computer vision and has been widely investigated. The goal of edge detection is to mark the points in a digital image at which the luminous intensity changes sharply. For Computer vision and Image processing Systems to Interpret an Image, they first must be able to detect the edges of each object in the image[4].Edge representation of an image drastically reduces the amount of data to be processed, yet it retains important information about the shapes of objects in the scene. This descriptionofanImageiseasytointegrateintoalargenoofobjectrecognitionalgorithmsusedin computer vision and other image processing applications. Edge detection produces an edge map that contains important information about the image.

Edge detection is difficult in noisy images, since both the noise and the edges have high frequency. Noisy images are typically larger in scope, so they average enough data to discount localized noisy pixels. This result is 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 [1]. 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 compare various edge detection techniques and analyze their performance in different conditions. There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories:

Gradient based Edge Detection

The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image.

Laplacian based Edge Detection:

The Laplacian method searches for zero crossings in the second derivative of the image to find edges. An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location.

Suppose we have the following signal, with an edge shown by the jump in intensity below in figure 1.

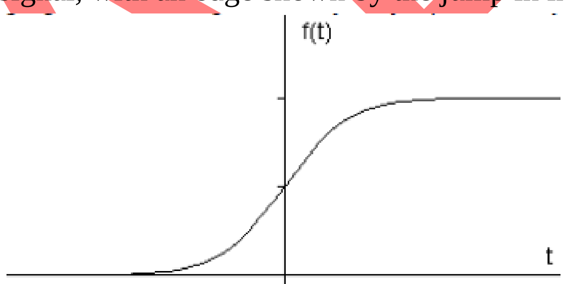


Figure 1

If we take the gradient of this signal (which, in one dimension, is just the first derivative with respect to t) we get the following:

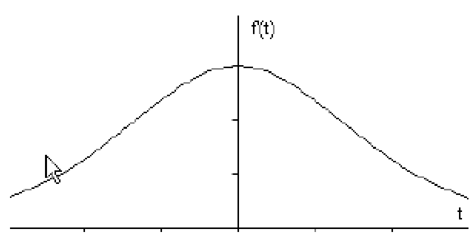


Figure 2

Clearly, the derivative shows a maximum located at the center of the edge in the original signal. This method of locating an edge is characteristic of the "gradient filter" family of edge detection filters and includes the Sobel method. A pixel location is declared an edge location if the value of the gradient exceeds some threshold. As mentioned before, edges will have higher pixel intensity values than those surrounding it. So once a threshold is set, you can compare the gradient value to the threshold value and detect an edge whenever the threshold is exceeded [2]. Furthermore, when the first derivative is at a maximum, the second derivative is zero. As a result, another alternative to finding the location of an edge is to locate the zeros in the second derivative. This method is known as the Laplacian and the second derivative of the signal is shown below:

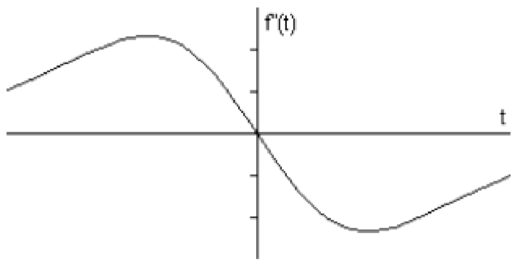


Figure 3

In this paper we analyzed and did the visual comparison of the most commonly used Gradient and Laplacian based Edge Detection techniques.

PROBLEM FORMULATION

In this work the problems of false edge detection, missing true edges, producing thin or thick lines and problems due to noise etc are analyzed and discussed. In this paper we analyzed and did the visual comparison of the most commonly used Gradient and Laplacian based Edge Detection techniques for problems of inaccurate edge detection, missing true edges, producing thin or thick lines and problems due to noise etc. The software is developed using MATLAB 7.0

EDGE DETECTION TECHNIQUES

Sobel Operator

The operator consists of a pair of 3×3 convolution kernels as shown in Figure 4. One kernel is simply the other rotated by 90°.

-1	0	1
-2	0	2
-1	0	1

G_x

+1	+2	+1
0	0	0
-1	-2	-1

G_y

Figure 4: Masks used by Sobel Operator

These kernels are 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 kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation call these (*Gx* and *Gy*). These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient [2]. The gradient magnitude is given by:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

Typically, an approximate magnitude is computed using:

$$|G| = |Gx| + |Gy|$$

which is much faster to compute.

The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by:

$$Theta = tan^{-1}(Gy/Gx)$$

Prewitt's operator

Prewitt operator is similar to the Sobel operator and is used for detecting vertical and horizontal edges in images.

-1	0	+1
-1	0	+1
-1	0	+1

Gx

+1	+1	+1
0	0	0
-1	-1	-1

Gy

Figure 4.2: Masks for the Prewitt gradient edge detector.

Canny's operator

The Canny edge detection algorithm is known to many as the optimal edge detector. Canny's intentions were toenhance the many edge detectors already out at the time hestarted his work. He was very successful in achieving his goal and his ideas and methods can be found in his paper, "A Computational Approach to Edge Detection"[5]. In his paper, he followed a list of criteria to improve current methods of edge detection. The first and most obvious is low error rate. It is important that edges occurring in images should not be missed and that there be no responses to non-edges. The second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum. A third criterion is to have only one response to a single edge. This was implemented because the first two were not substantial enough to completely eliminate the possibility of multiple responses to an edge. Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (non maximum suppression). The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a non edge). If the magnitude is above the high threshold, it is made an edge. And if the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient aboveT2.

Step 1:-

In order to implement the canny edge detector algorithm, a series of steps must be followed. The first step is to filter out any noise in the original image before trying to locate and detect any edges. And because the Gaussian filter can be computed using a simple mask, it is used exclusively in the Canny algorithm. Once a suitable mask has been calculated, the Gaussian smoothing can be performed using standard convolution methods. A convolution mask is usually much smaller than the actual image. As a result, the mask is slid over the image, manipulating a square of pixels at a time. The larger the width of the Gaussian mask, the lower is the detector's sensitivity to noise. The localization error in the detected edges also increases slightly as the Gaussian width isincreased.

Step 2:-

After smoothing the image and eliminating the noise, the next step is to find the edge strength by taking the gradient of the image. The Sobel operator performs a 2-D spatial gradient measurement on an image. Then, the approximate absolute gradient magnitude (edge strength) at each point can be found. The Sobel operator [3] uses a pair of 3x3 convolution masks, one estimating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows). They are shownbelow:

$$\begin{matrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{matrix} \qquad \begin{matrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix}$$

Gx Gy

The magnitude, or edge strength, of the gradient is then approximated using the formula:

$$|\boldsymbol{G}| = |\boldsymbol{Gx}| + |\boldsymbol{Gy}|$$

Step 3:-

The direction of the edge is computed using the gradient in the x and y directions. However, an error will be generated when sum x is equal to zero. So in the code there has to be a restriction set whenever this takes place. Whenever the gradient in the x direction is equal to zero, the edge direction has to be equal to 90 degrees or 0 degrees, depending on what the value of the gradient in the y-direction is equal to. If Gy has a value of zero, the edge direction will equal 0 degrees. Otherwise the edge direction will equal 90 degrees. The method for finding the edge direction is just:

$$\mathbf{Theta} = \mathbf{tan}^{-1}(\mathbf{Gy/Gx})$$

Step 4:-

Once the edge direction is known, the next step is to relate the edge direction to a direction that can be traced in an image. So if the pixels of a 5x5 image are aligned as follows:

x x x x x
x x x x x
x x a x x
x x x x x
x x x x x

Then, it can be seen by looking at pixel "a", there are only four possible directions when describing the surrounding pixels - 0 degrees (in the horizontal direction), 45 degrees (along the positive diagonal), 90 degrees (in the vertical direction), or 135 degrees (along the negative diagonal). So now the edge orientation has to be resolved into one of these four directions depending on which direction it is closest to (e.g. if the orientation angle is found to be 3 degrees, make it zero degrees). Think of this as taking a semicircle and dividing it into 5 regions. Therefore, any edge direction falling within the yellow range (0 to22.5&157.5to180degrees)issetto0degrees.Anyedgedirectionfallinginthegreenrange(22.5to 67.5 degrees) is set to 45 degrees. Any edge direction falling in the blue range (67.5 to 112.5 degrees) is set to 90 degrees. And finally, any edge direction falling within the red range (112.5 to 157.5 degrees) is set to 135 degrees.

Step 5:-

After the edge directions are known, non-maximum suppression now has to be applied. Non-maximum suppression is used to trace along the edge in the edge direction and suppress any pixel value (sets it equal to 0) that is not considered to be an edge. This will give a thin line in the output image.

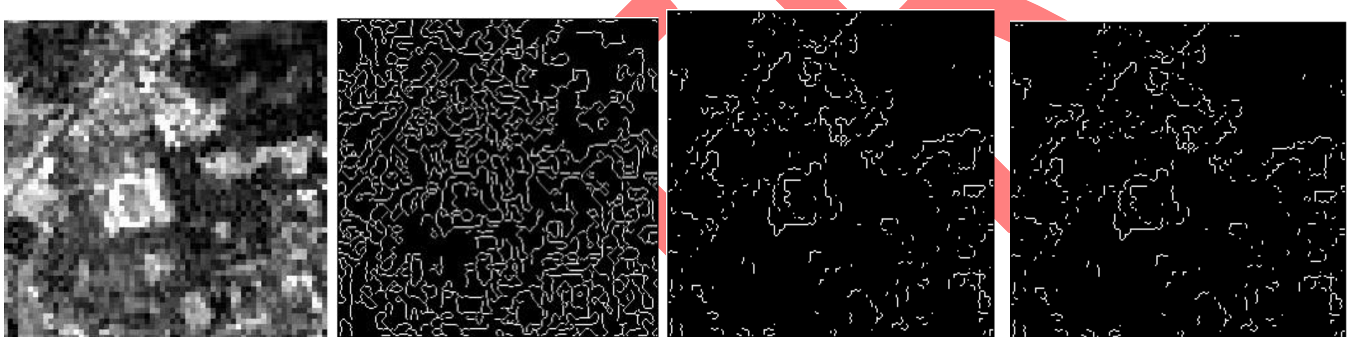
Step 6:-

Finally, hysteresis [12] is used as a means of eliminating streaking. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold. If a single threshold, T1 is applied to an image, and an edge has an average strength equal to T1, then due to noise, there will be instances where the edge dips below the threshold. Equally it will also extend above the threshold making an edge look like a dashed line. To avoid this, hysteresis uses 2 thresholds, a high and a low. Any pixel in the image that has a value greater than T1 is presumed to be an edge pixel, and is marked as such immediately. Then, any pixels that are connected to this edge pixel and that have a value greater than T2 are also selected as edge pixels. If you think of following an edge, you need a gradient of T2 to start but you don't stop till you hit a gradient below T1.

STUDY AREA AND DATA RESOURCES

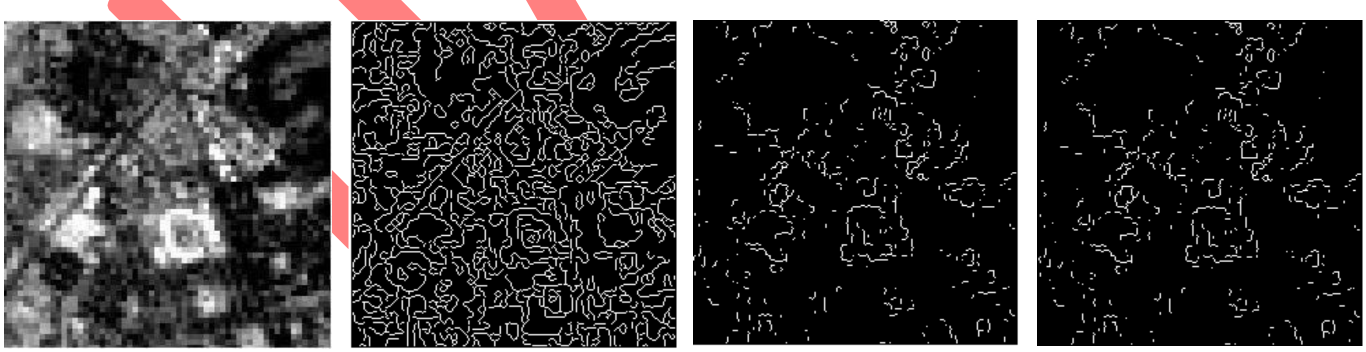
The different types of areas have been selected in Bhopal city so that the Land use feature combination is analyzed. In the selected area various type of feature boundaries were identified in such a way that some are strong and some weak edges. Image data for analysis has been collected from following sources: LISS-III, PAN, CARTOSET-1 (IRS-P5). Some PAN sensor images of poor radiometry i.e. Salt & Pepper noise and blurry image were also selected.

IMAGE COMPARISON OF VARIOUS EDGE DETECTION ALGORITHMS



(a) (b) (c) (d)

Figure 1: Comparison of Edge Detection Techniques on LISS-III band1 (b) Canny (c) Sobel (d)Prewitt

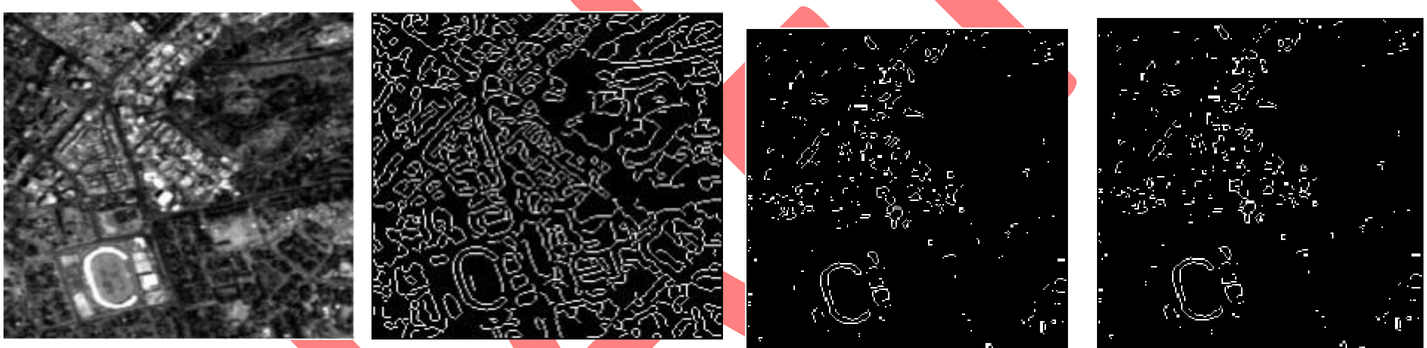


(a) (b) (c) (d)

Figure 2: Comparison of Edge Detection Techniques on LISS-III band2 (b) Canny (c) Sobel (d)Prewitt

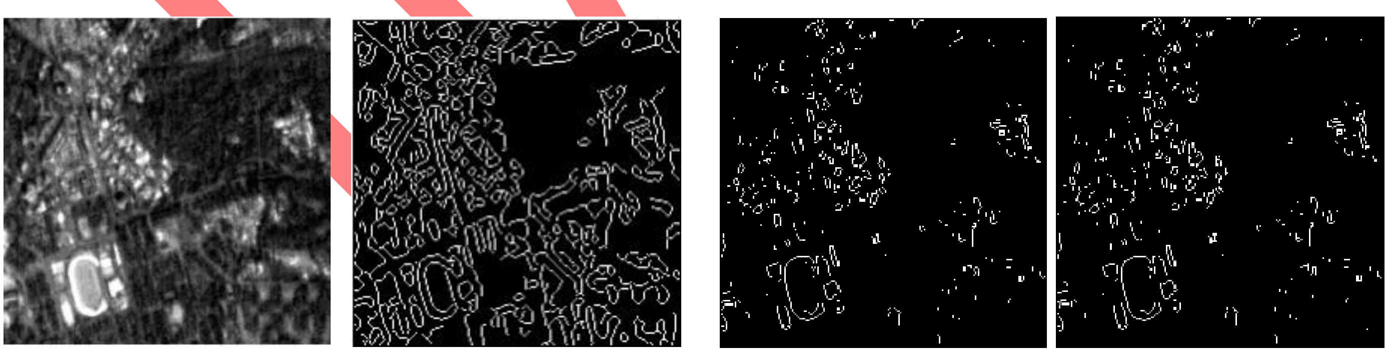
(a) (b) (c) (d)

Figure 3: Comparison of Edge Detection Techniques on LISS-III band3 (b) Canny (c) Sobel (d)Prewitt



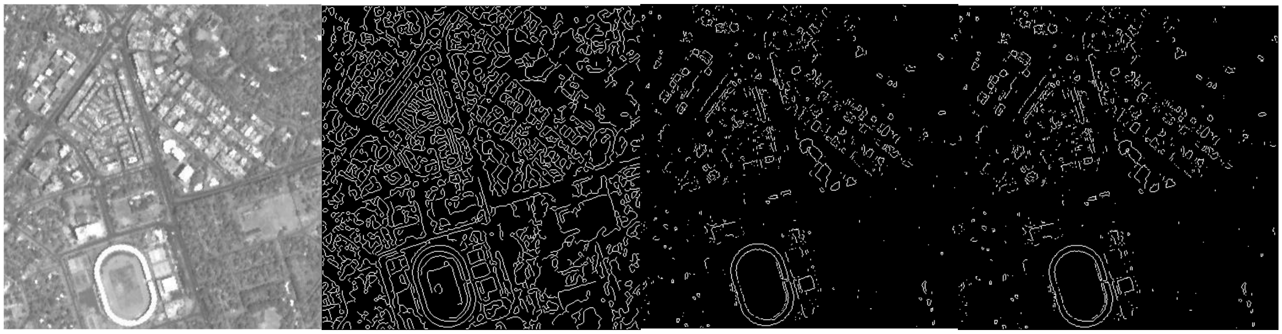
(a) (b) (c) (d)

Figure 4: Comparison of Edge Detection Techniques on PAN Image (b) Canny (c) Sobel (d) Prewitt



(a) (b) (c) (d)

Figure 5: Comparison of Edge Detection Techniques on PAN Blurred Image (b) Canny (c) Sobel (d)Prewitt

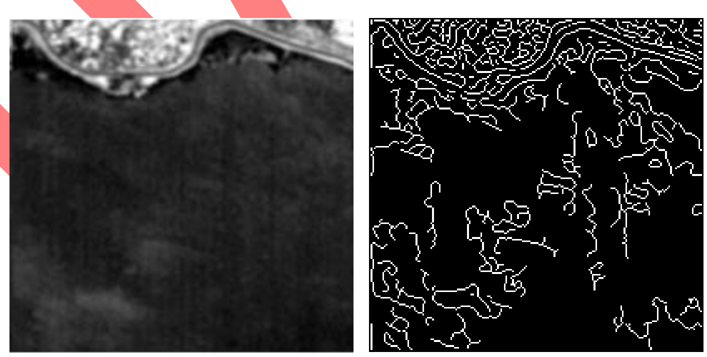


(a) (b) (c) (d)

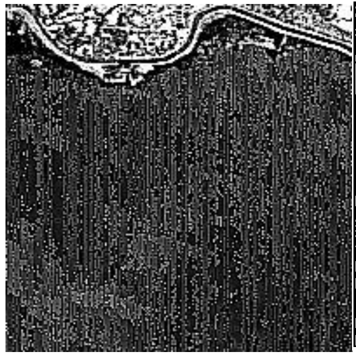
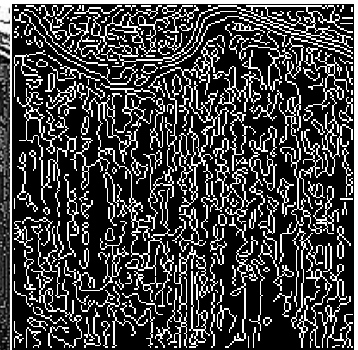
Figure 6: Comparison of Edge Detection Techniques on CATOSAT-1(IRS-P5) (b) Canny (c) Sobel (d) Prewitt

DIFFERENT SPATIAL FILTERING APPLY ON PAN SALT & PEPPER NOISY IMAGE

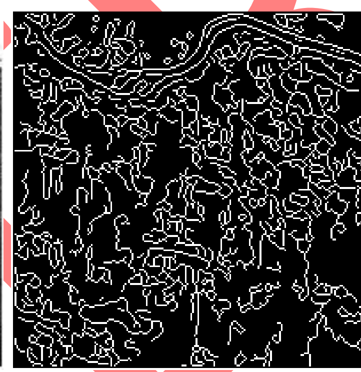
(a) (b)



(c) (d)

(e) (f)

(g) (h)

Figure 7. Different spatial filterers apply on PAN Salt & Pepper noise image and effects of canny operator on that filtered images, (a) Median filter image, (b) corresponding canny image, (c) Low-pass filter image, (d) corresponding canny image, (e) High-pass filter image, (f) corresponding canny image, (g) High-Boost Filtering (h) corresponding edge images detected by canny operator

CONCLUSIONS

Since edge detection is the early step in object recognition, it is important to know the differences between edge detection techniques. In this paper the most commonly used edge detection techniques of Gradient-based and Laplacian based Edge Detection has been used. The software is developed using MATLAB 7.0. Canny operator has given best results in the edge detection as compared to other two edge detectors i.e. Sobel & Prewitt. Even in the high resolution Cartosat-1 satellite images the other two edge detectors could not perform well but canny operator has shown considerable improvements. In the LISS-III images the band 3 (near infrared) has given more prominent edges as compared to other bands. In the mixed land use, there are chances of false edges and this is more in the coarse resolution images (LISS-III sensor) as compared to high resolution (Cartosat-1 sensor image).

The radiometric quality of the input image is very important in the edge detection and poor radiometry images result distorted, broken and shifted edges. Canny edge detector is very much sensitive to radiometric noise on the image. The salt peeper type noise of image can generate arbitrary false edges even on the homogeneous earth surfaces like water bodies. This problem can be resolved by low pass spatial filtering of the images before detection of the edges. The edges detected from Cartosat-1 image have shown good geometric integrity as compared to Google earth image almost at same level of spatial resolution. This may be due to good radiometry of the images from Cartosat-1satellite.

In the analysis for images at various specific thresholds, it has been observed that at less threshold values edge detector detects many edges which results in dense edges or mixed edges. Whereas at high threshold values the edge detection is negligible for Canny, Sobel, and Prewitt respectively. With this it is fond that optimum threshold values for Google earth images is **Th=0.2,** and the optimum threshold values for PAN image is **Th=0.08,** and the optimum threshold value for Cartosat-1 image is **Th=0.06.** Based on the various types of analysis on the three edge detectors, it has been observed that Canny's edge detection algorithm is computationally more time consuming as compared to Sobel, Prewitt operator. However, the Canny's edge detection algorithm performs better than all these operators under allconditions.

REFERENCES

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

2. J. Matthews **"An introduction to edge detection: The sobel edge detector"** Available at http://www.generation5.org/content/2002/im01.asp,2002.

3. **Digital image processing** by R.C.Gonzalez & R.E.Woods,Image Enhancement page 189, Addison-Wesley LongmanPublication,1999.

4. J. Koplowitz "**On the Edge Location Error for Local Maximum and Zero-Crossing Edge Detectors**", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.16, pg-12,Dec.1994.

5. J. F. Canny "**A computational approach to edge detection**". IEEE Trans. Pattern Anal Machine Intell, vol.8, pg 679-697,1986.

6. Barbara Zitova,Jan Flusser ,"**Image registration methods: a survey**" ,Department of Image Processing, Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic June 2003 in Image and Vision Computing 21 (2003). Pg-977–1000.

7. Mohamed Roushdy, "**Comparative Study of Edge Detection Algorithms Applying on the Grayscale Noisy Image Using Morphological Filter"**, C.S. Department, Ain Shams University, Abbassia, Cairo, Egypt published in GVIP Journal, Volume 6, Issue-4Dec,2006.

8. Chuan XU et al"**A New Urban Road detection method in high-resolution images based on Bayesian Network**", Wuhan University,China.

9. Lourdes A.Funtanil, "**GIS Pattern Recognition and Rejection Analysis Using MATLAB**", MSComputer Science, Texas A&MUniversity.

10. Gang Hong and Yun Zhang, **"The Image Registration Technique for High Resolution Remote Sensing image in hilly area**", Department of Geodesy and Geomatics Engineering University of New Brunswick,Canada.

11. O.R.Vincent, O.Folorunso**, "A Descriptive Algorithm for Sobel Image Edge Detection"**, Clausthal University of Technology,Germany.

12. Maini Raman and Agrawal Himanshu**, "Study and Comparison of Various Image Edge Detection Techniques"**, Punjabi University, Patiala-147002(Punjab),India.

13. Senthilkumaran.N and R.Rajesh, "**Edge Detection Techniques for Image Segmentation – A Survey of Soft Computing Approaches**", School of Computer Science and Engineering, Bharathiar University, Coimbatore -641 046,India.

14. Wilkie Kiefer**, "Real-Time Image Registration and Noise Reduction"**, Professor Szymon University.

15. Moyan Xiao and Yonghong Jia, **"Edge Detection of Riverway in Remote Sensing Images Based on Curvelet Transform and GVF Snake" ,** School of Remote Sensing and Information Engineering, Wuhan University,129 Luoyu Road, Wuhan 430079, P.R.China.

16. Shinji Hayashi and Osamu Hasegawa**, "Robust Face Detection for Low- ResolutionImages"** Tokyo Institute of Technology.

17. R.Gecena and G.Sarpb**, "Road Detection from High and Low Resolution Satellite Images"** Department of Geodetic and Geographic Information Technologies, Middle East Technical University.

18. J.Anthony Parker et al**, "Comparison of Interpolating Methods for Image Resampling",**.,IEEE Transactions on Medical Imaging,Vol. MI-2, No. 1, March1983.

19. Gurjar S.B and Padamanabhan.N **,"Study of various Resampling Techniques for high resolution remote sensing imagery"** ,Journal of the Indian Society Of Remote Sensing Vol.33, No.1, 2005,Space Applications center, Indian Space Research OrganizationAhmedabad-380015,India.

20. Foster M.P, **"Writing MEX Files for Image Processing"**, January 16,2009.

21 .R.Maini and Sohal.S ,"**Performance Evaluation of Prewitt Edge Detector for Noisy images"**, GVIP Journal, Volume 6, Issue 3, December, 2006,U.C.E. Punjabi University, Patiala-147002India.

22. Katiyar S.K et al, **"GPS for geometric correction of remotely sensed imagery: possibilities after termination of** SA", 24 - 25 October 2002, India International Centre, New Delhi,India.

23. Katiyar S.K**, "Surveying to Geomatics: Emergence of modern mapping techniques"**

24. Armin Gruen et al **, "Linear feature Extraction with Dynamic Programming and Globally Enforced Least Squares Matching",** Institute of Geodesy & Photogrammetry,Switzerland**,** *Automatic Extraction of Man-Made Objects from Aerial and Space Images,1995, Birkhauser Verlag.*