Adaptive Local Thresholding for Edge Detection

Gaur Sanjay B. C.
Dept. of Elec. & Comm. Engg.
JIET School of Engineering & Technology for Girls, Jodhpur

Sandip Mehta
Dept. of Elec. & Electro. Engg.
JIET School of Engineering & Technology for Girls, Jodhpur

Jayashri Vajpai
Dept. of Electrical Engineering,
M. B. M. Engg.College, Jodhpur

ABSTRACT
The Edge detection technique plays very important role in computer vision systems. Edges define the boundaries between different regions in an image, which helps in matching the pattern, segment, and recognize an object. In many applications the overall performance of the system depends on the proper detection of the edges such as Text Detection, Shape detection, Finger Print Recognition, Pattern Recognition etc. Hence edge detection is a fundamental aspect of low-level image processing. In this paper, a local threshold based method is proposed to detect the edge of an object. Experimental results suggest that this approach is more efficient in comparison with other traditional techniques like Prewitt, Sobel and Canny Edge detector. In order to test the performance of the proposed technique, twenty five test images have been considered. The experimental results show that the proposed method is better than the conventional techniques.

Keywords
Edge; Threshold value, binary image, image processing, gray-level image**.**

1. INTRODUCTION
Edge detection is a fundamental task in image processing which provides important information about the images. Edges are defined as a group of pixels where the intensity of neighboring pixels changes abruptly [1]. These edges form the outline of an object. Edge detection is one of the most commonly used operations in image analysis and is an essential preprocessing step in the image segmentation. It is also a part of feature detection, compression and image retrieval systems [2-5].However, edge detection is a challenging task, in computer vision to detect, segment and recognize an object in an image.

The basic steps for edge detection are smoothing, enhancement, detection and localization. This paper focuses on smoothing for noise reduction followed by detection and thinning to identify the primary edges.

The quality of edges degrades in the presence of noise asince suppression of the noise blurs the significant transitions of an image. However, some form of smoothing is necessary since edge detection depends on differentiating the image function and this amplifies all high frequency components. Low pass filters most widely used for smoothing filters [2].

To begin with, two simple classical approaches Sobel edge detection and Prewitt edge detection have been implemented but these two techniques suffer from drawbacks like edge displacement, vanishing edges and false edges [3]. To overcome these problems Canny edge detector [6] has proposed edge detection technique convolves a Gaussian filter with an image. In most of the applications Canny is most commonly used due to its better result as compared to the other traditional techniques. Canny is gradient based technique which uses Gaussian filter for smoothing an image. However, it gives unsatisfactory result on low contrast images and noisy images.

Thresholding is the fundamental requirement for gradient based edge detection. By selecting an adequate threshold value T, the gray level G (i, j) image can be converted to binary image f (i, j), given by equation (1).

$$\begin{aligned} G(i,j) &= 1 \text{ for } f(i,j) \geq T, \\ &= 0 \text{ for } f(i,j) < T, \end{aligned} \tag{1}$$

where T is threshold, G (i, j) =1 for image element of foreground , and G (i, j) = 0 for image element of the background [7].

The binary image should contain all of the essential information about the position and shape of the objects of interest (foreground). The advantage of obtaining first a binary image is that it reduces the complexity of the data and simplifies the process of recognition and classification. Otsu's method [7] is the most common way to convert a gray-level image to a binary image. It selects a single threshold value (T) and all the gray level values below this T will be classified as black (0), and those above T will be white (1). Hence proper selection of the value of threshold T is very important.

The global thresholding technique is based on the mean, standard deviation or variance of a complete image whereas local thresholding value is chosen for the local partition of the image.

The aim of this paper is to develop simple and fast edge detectors using local thresholding. This paper has been divided in to five sections. After introduction, section 2 presents the methodology for proposed technique. The detail of the experiment with illustrative examples is explained in section 3. The experimental results are shown in the section 4. The next describes the conclusion drawn from this paper on the basis of results obtained from the experiments.

2. METHODOLOGY
After an image is captured by the camera, it is passed through the preprocessing unit which processes the raw image captured by the camera. The main operation is to suppress the noise and enhance the image features, so that the proper edges can be detected. The major components of the proposed scheme are described in the following sections.

2.1 Pre-Processing
Preprocessing is necessary to detect the edges of an image. Preprocessing includes three basic operations: image resizing, grayscale conversion and removal of noise, so that the system can perform uniformly well for all still images. Further, the RGB color input image is converted to a 256 grayscale image using HSI model [1] given in Equation 2.

$$\text{Gray} = 0.299*\text{Red}+0.587*\text{Green}+0.114*\text{Blue} \tag{2}$$

Then a median filter with window size (5x5) is applied to the gray level image to remove the salt and paper noise, while preserving the sharpness of the image.

2.2 Local Thresholding

Although global thresholding techniques are computationally fast and their performance is good in gray-level images. But they cannot perform well in noisy images. The use of local thresholding is expected to solve this kind of problems. In this part the aim is to find proper local threshold value for the edge detection. Eight connected components p(i-1,j-1), p(i-1, j), p(i-1, j+1), p(i, j), p(i, j+1), p(i+1,j-1), p(i+1, j), and p(i+1, j+1) are used for the selection of threshold value for the centered pixel p(i,j).

p(i-1,j-1)	p(i-1, j)	p(i-1, j+1)
p(i,j-1)	p(i, j)	p(i, j+1)
p(i+1,j-1)	p(i+1, j)	p(i+1, j+1)

Figure 1: 3×3 Neighbourhood for local thresholding

The standard deviation of the pixels of 3X3 neighborhood is computed using equation 3 given below:

$$std(i,j)=\sqrt{\left(\frac{\sum(x(i,j)-\bar{x})^2}{3\times 3}\right)} \qquad (3)$$

where std(i, j) is the standard deviation for the window centered at pixel p(i,j) as shown in figure1. x(i, j) is the intensity of the centered pixel value p(i,j), $\bar{x}$ is the mean value of the 3X3 neighboring values.

The standard deviations evaluated for all the neighborhood positions of window are normalized in the range 0 to 1 using equation 4.

$$n(i,j)=\frac{std(i,j)}{\max(std(i,j))} \qquad (4)$$

Now the mean m(i,j) and standard deviation s(i,j) of n(i,j) values in the 3x3 window is calculated as follows.

$$m(i,j)=\frac{\sum_{i=1}^{3}\sum_{j=1}^{3}n(i,j)}{3\times 3} \qquad (5)$$

$$s(i,j)=\sqrt{\left(\frac{\sum(n(i,j)-\bar{n})^2}{3\times 3}\right)} \qquad (6)$$

The local threshold value depends on some local statistical features extracted by its neighborhood pixels, which can be expressed as equation 7.

$$T(i,j) = m(i,j) + k\, s(i,j) \qquad (7)$$

where m(i, j) is mean of n(i,j) for 3X3 neighborhood for centered pixel value p(i, j), s (i, j) is standard deviation and k is constant value between 0 to 1.

Above local threshold T(i,j) is now used to convert normalized n(i,j) into binary image b(i,j) as follows:

$$\begin{aligned} b(i,j) &= 1 \text{ for } n(i,j)\ge T(i,j),\\ &= 0 \text{ for } n(i,j) < T(i,j) \end{aligned} \qquad (8)$$

2.3 Thinning of the Binary Image

The output binary image b(i, j) is then processed using morphological operation for thinning process, which removes interior pixels. This option sets a pixel to 0 if all its 4-connected neighbors are 1 as shown in figure, otherwise leaves the value intact. Now final binary image is termed as edge image.

<table>
<tr><th>Output Binary Image</th><th>After thinning</th></tr>
<tr><td><table><tr><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td></tr></table>or<table><tr><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td></tr></table></td><td><table><tr><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td></tr></table></td></tr>
</table>

3. ILLUSTRATIVE EXAMPLE

The proposed approach has been implemented in Matlab-2010(a). Twenty Five images captured in day and night, with different cameras have been used to test and compare the results with Prewit, Sobel and Canny edge detection. The collection of image database includes texture, documents, faces, cars, roads, forests etc. The edge images, through above methods including the proposed method, (for three of them) are shown in figure 4-6.

In these figures, (a) is original input image, (b) shows output of Sobel edge detector (c) by Prewit edge detector (d) by Canny edge detector and finally (e) gives detected edge output by proposed edge detector.

(a) Original Lena Image

(b) Sobel Edge Detector

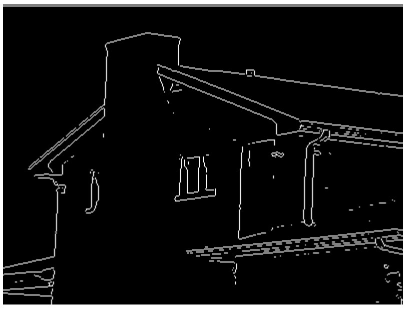
(c) Prewit Edge Detector

(d) Canny Edge Detector

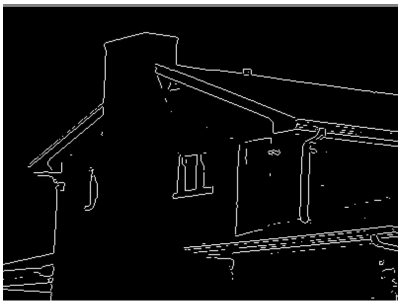
(e) Proposed Method (k=0.6)

Figure 4. Edge Detection for Lena Image

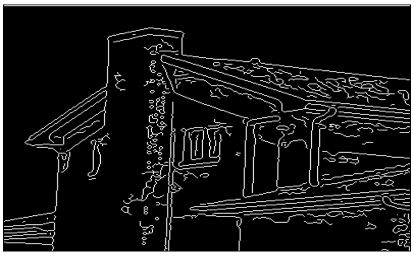
(a) Original Building Image



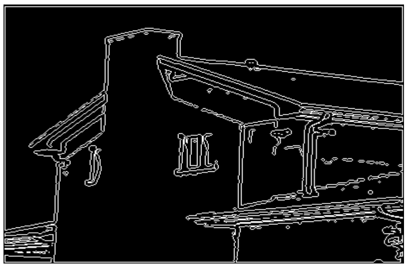
(b) Sobel Edge Detector



(c) Prewit Edge Detector



(d) Canny Edge Detector



(e) Proposed Method (k=0.6)

Figure 5. Edge Detection for a Building Image

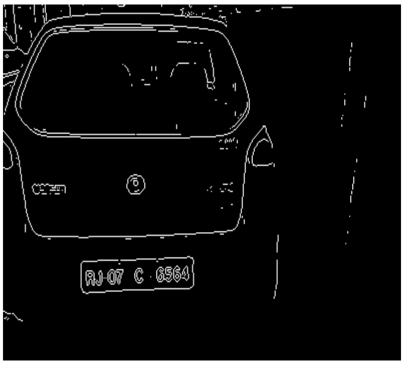
4. RESULTS ANALYSIS

Twenty five images have been tested to compare the result of the proposed technique with other classical techniques. Out of twenty five three images are shown here to discuss the results obtained by the above methods.

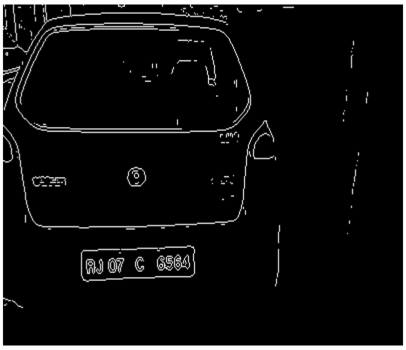
From visual perception, it is evident that figure 4(e)-6(e) presents better results than figure 4-6 (a to d) for all three input images.

It has been observed that Prewit, Sobel, and Canny edge detection technique eliminate the low intensity edges of an image. Whereas the proposed detector can detect such kind of edges by selecting the proper value of k. In this paper, the value of k is taken as = 0.6. The value of k decides the sensitivity of the detector which can be varying between 0 and 1 as per the requirement of application.

(a) Original Car Image



(b) Sobel Edge Detector



(c) Prewit Edge Detector

(d) Canny Edge Detector

(e) Proposed Method (k=0.6)

Figure 6. Edge Detection for a Car Image

5. CONCLUSION

Edge detection is an important task needed in most of the applications for analysis e.g. image retrieval, image segmentation, document analysis systems. The quality of the edge result affects the subsequent processing of system. In this paper, an adaptive local thesholding approach, based on the mean and standard deviation of its local intensity values, has been introduced to get proper binarization (Edges) for all kind of images.

The proposed algorithm has been applied over the twenty five textual, pictorial and synthetic test images. The algorithm showed robust behaviour in most of the images and performed well against the comparison techniques. The results show that the proposed technique is better for all kind of images compared to other classical techniques. Hence by selecting the proper value of k, the proposed technique can be used for any application e.g. text recognition, face recognition, image retrieval, tumor detection etc.

6. REFERENCES

[1] R. C. Gonzalez, and R. E. Woods, “Digital Image Processing”.3rd ed. Prentice Hall, 2013, pp.692 -794.

[2] Basu M., “Gaussian Based Edge Detection Methods-A Survey”, IEEE Transaction on System, Man and Cybernetics, Vol.32 No. 3.Aug. 2002, pp. 252-260.

[3] Sanjay B. C. Gaur, and Dr. Jayashri Vajpai, “Comparison of Edge Detection Techniques for Segmenting Car License Plates”, Special Issue of International Journal of Computer Applications (0975 – 8887) on Electronics, Information and Communication Engineering - ICEICE No.5, Dec 2011, pp. 8-12.

[4] M.B. Ahmad, and T. S. Choi , “Local Threshold and Boolean Function Based Edge Detection”, IEEE Transactions on Consumer Electronics, Vol. 45, No 3. August 1999.

[5] S. Belongie, J. Malik, and J. Puzicha, “Shape Matching and Object Recognition using Shape Contexts”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 24, No 24, 509 - 522, April 2002.

[6] J. F. Canny, “A computational approach to edge detection”, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679-697, 1986.

[7] N. Otsu, “A Threshold Selection Method from Gray-Level Histogram,” IEEE trans. Sys., Man and Cybernetics, vol. 9, pp 62-66, 1979.

[8] R. Medina-Carnicer, F. J. Madrid-Cuevas, N. L. Fernandez-Garcia and A. Carmona-Poyato, “Evaluation of Global Thresholding techniques in non-contextual edge detection”, pattern recognition letters 26, pp. 1423-1434, 2005.