Performance analysis of multi level threshold based OTSU method

Ms. Bharti Chourasia[1], Dr Sanjeev Kumar Gupta[2], Anshuj Jain[3]

[1] Research Scholar, Electronics Engineering Department, AISECT University, M.P, India
[2] Dean Academic, AISECT University, M.P, India
[3] Research Scholar, Electronics Engineering Department, AISECT University, M.P, India

ABSTRACT

In this work an effort is made to study and analyses the performance of frequently used edge detection techniques for image segmentation and also the comparison of these techniques is carried out with an experiment by using image processing toolbox of MATLAB software. This work recommended multilevel thresholding for histogram-based image segmentation using OTSU algorithm. Image segmentation needs to segment the object from the background to read the image properly and identify the content of the image carefully, segmentation is necessary to interpretation of an image. For image segmentation Multilevel Thresholding method uses the Otsu's method to segment the image.

Keyword: - - *OTSU Method, Image processing, Segmentation.*

1. Introduction

In various computer vision and image processing applications, edge detectors are important tool for pulling out the feature of interest. The separation of a scene image into object and background, by mark out the edge between them, is an essential step in image interpretation. Therefore, faithful edge detection is necessary for several image investigation, assessment and recognition technique. In prior era, numerous studies and research work has been done in the area of image segmentation in various application using edge detection. The underlying idea of most edge detection techniques is by the computation of a local first or second derivative operator, followed by some regularization technique to diminish the effects of noise. Earlier edge detection methods, such as Sobel, Prewitt and Roberts' operator used local gradient method to detect edges along a specified direction. The lack of noise control resulted in their poor performance on blurred or noisy images. Canny use the Gaussian filter for filtering noise and double-threshold value to locate the image's edge points, and then get the image edge with only one pixel wide this is one of the reason that Canny Edge Detection methods gives improved results as compare to others method and it is also less sensitive to noise. The Canny edge detection operator was originally developed in 1986 by John F. Canny. Canny uses a multi-stage algorithm to detect a wide range of edges in images and produced a computational theory of edge detection. This method is also known as optimal edge detection method.

.2. OTSU Method

Multi-level OTSU thresholding method is used in this work. This method keeps the Canny good performance in fine detection, good edge localization and only one response to a single edge, improves the capability of preventive the fake edge [2]. In image processing and computer vision, OTSU's method is used to perform automatically clustering-based on image thresholding. It is universally acknowledged that the Otsu is the best method of choosing threshold value automatically which is proposed in 1979 [4]. Its basic principle is to split the image's pixels into two classes, and confirms the best threshold value through the variance maximum value between the two classes [8][9][10].

The gray-level histogram of an image is usually considered as efficient tools for development of image thresholding algorithms. Thresholding creates binary images from grey-level ones by turning all pixels below some threshold to zero and all pixels about that threshold to one. If g(x, y) is a threshold version of f(x, y) at some global threshold T, it can be defined as [1], g(x, y) = 1 if f(x, y) ≥ T = 0 otherwise. Thresholding operation is defined as:

T=M[x,y,p(x,y),f(x,y)]………………………..(1.1)

In this equation, T stands for the threshold; f (x, y) is the gray value of point (x, y) and p(x, y) denotes some local property of the point such as the average gray value of the neighborhood centered on point (x, y). A variety of thresholding approaches have been proposed for image edge detection but Otsu methods has advantages of finding sharp true edge in image; it is mostly used as compare to other methods .Brief details on OTSU method are given in earlier chapter. Canny algorithm can use the threshold value which is gained by the OTSU method automatically to obtain the image's contour line.

2.1 Algorithm

This algorithm is found very successful in achieving goal of edge detection. Ideas and methods used in this dissertation are based on multi-level thresholding. It is computational approach to Edge Detection. This followed a list of criteria to improve current methods of edge detection.

1. The first 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.
2. The second and most obvious is low error rate. It is important that edges occurring in images should not be missed and no responses to non-edges.
3. The 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 the noise. It then finds the image gradient to emphasize regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum [8]. The gradient array is now further reduced by hysteresis. Hysteresis is also 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 T1 it is set to zero. If the magnitude is above the high threshold T2 it is made an edge. If the magnitude is between the thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above T2[2, 3]. Proposed work find the best threshold level by using OTSU multilevel threshold method while keep the advantage of Canny edge detector to search the best threshold value to find the true edge of image. The OTSU algorithm can calculate the high threshold value which is significant to the Canny algorithm, and then this threshold value can be used in the Canny algorithm to detect the object's edge.

3. MODEL OF IMAGE DETECTION

The aim of Improved Canny algorithm was to have high-quality detection (minimum number of false edges), fine localization (closeness of the real edge and the detected edge) and minimal response (one edge should be detected only once). Steps involved in Improved Canny algorithm [4] are shown in Figure. 1. For edge detection, we take the original image & by using Gaussian filter remove the noise. Then by using the improved Canny calculate the gradient using the formula given below-

$$G = \sqrt{(G_x{}^2 + G_y{}^2} \quad \ldots\ldots\ldots\ldots.. \ (1.2)$$

Where G_x = Intensity along the x-axis, Gy = Intensity along the y axis

Then non-maxima suppression is followed, which differentiates between background and image boundary. OTSU thresholding is used for the automatic thresholding level. It is a self adoptive threshold method.

Thresholding level is selected on the basis of pixel of intensity of image. MATLAB is used for generating the output images.

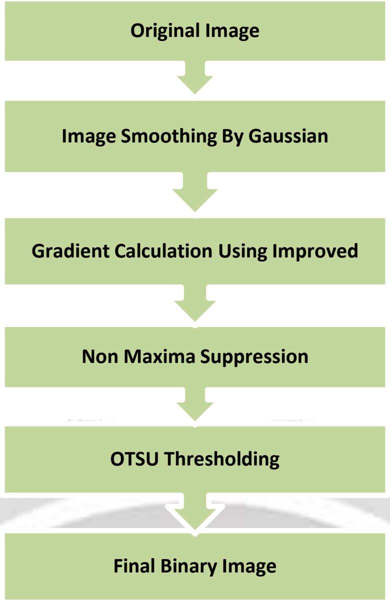

Figure.1 Model of Image Detection

4. MULTI-LEVEL THRESHOLDING

4.1.OTSU N, A THRESHOLD SELECTION METHOD FROM GRAY-LEVEL HISTOGRAMS

OTSU (I,N) segments the image I into N classes by means of multi-level OTSU thresholding method. Multi-level OTSU returns an array (A) containing the cluster indices (from 1 to N) of each point. Zero values are assigned to non-finite pixels.

A = OTSU (I) uses two classes (N=2, default value).

[A, sep] = OTSU (...) also returns the value (sep) of the separability criterion within the range [0 1]. Zero is obtained only with data having less than N values, whereas one is obtained only with N-valued arrays.

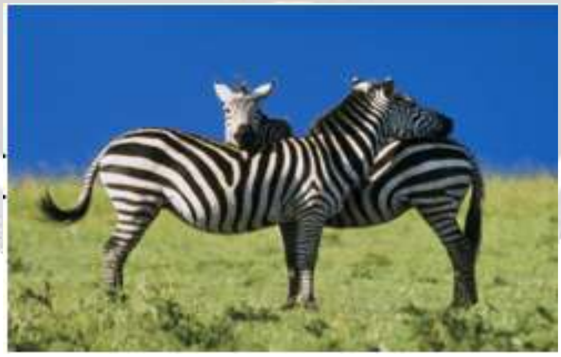

Figure.2 Input image (horse)

It should be noticed that the thresholds generally become less credible as the number of classes (N) to be separated increases. If I is an RGB image, then transform (color to binary/indexed conversion) is first performed on the three red, green, blue channels. The segmentation is then carried out on the image component that contains most of the energy.

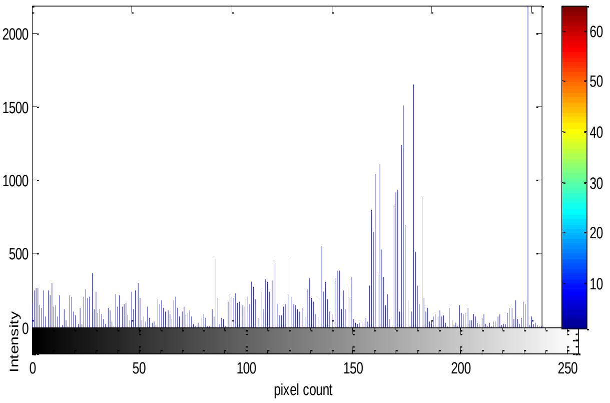

Figure 3 Histogram of input image (horse)

How It Works: - The input to a thresholding operation is typically a grayscale image or color image. In this work we used grayscale image. In the simplest Implementation, by using the segmentation we got output is a binary image. Variation on binary image is depending on the command of program. Let us take black pixels correspond to background and white pixels correspond to foreground (vice versa). In simple implementations, intensity threshold is used for the segmentation. In a single pass, each pixel in the image is compared with this (obtain threshold value) threshold. If the pixel's intensity is higher than the obtain threshold, the pixel is set to white in the output. If it is less than the threshold-level, it is set to black. In more sophisticated implementations, multiple thresholds level can be observed, so that a band of intensity values can be set to white while everything else part is set to black. For different colors or multi-spectral images, it may be possible to set different threshold-levels for each color channel, and select just those pixels within specified cuboids in Red, Green, Blue space[16]. Another common variant is to set to black all those pixels corresponding to background, but leave for ground pixels at their original intensity so that If it is possible to separate out the foreground of an image on the basis of pixel intensity, then intensity of pixels within foreground objects must be clearly different from the intensity of pixels within the background. In this case, we notice a distinct peak in the histogram corresponding to foreground objects such that thresholds can be chosen to isolate this peak accordingly. In this case, multi-level thresholding may be a better answer. Figure 4.5 shows suitable choices of threshold. We are using 4 &5 thresholding level for this work.

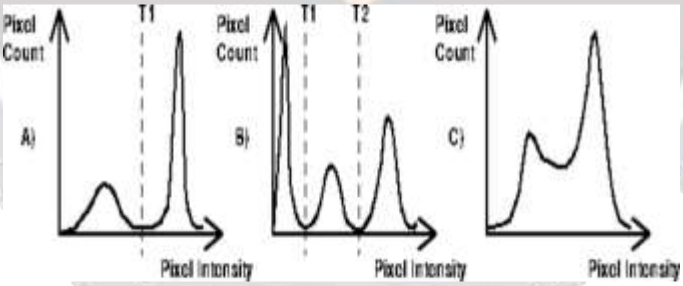

Figure 4 Thresholds for (A) Bi-modal (B) Two threshold (C) Two peak of Bi-modal

Image (A) Shows a classic bi-modal intensity distribution. This image can be successfully segmented using a single threshold T1. (**B)** is slightly more complicated. Here we suppose the central peak represents the objects we are interested in and so threshold segmentation requires two thresholds: T1 and T2. In (**C)**, the two peaks of a bi-modal distribution have run together and so it is almost certainly not possible to successfully segment this image using a single global threshold. In this case it provides good result for two threshold values are taken as separation parameters with intermediate data values in the image representation from the gray image.

The major problem with thresholding is that we consider only the intensity, not any relationships between the pixels. There is no guarantee that the pixels identified by the thresholding process are contiguous. The steps of the OTSU algorithm: For each potential threshold T, 1. Separate the pixels into two clusters according to the threshold. 2. Find the mean of each cluster. 3. Square the difference between the means. 4. Multiply by the number of pixels in one cluster times the number in the other.

4. RESULT & COMPARISON

Comparison between the proposed and standard edge detection methods

Sr. No	Image Name	Method	Variance	Mean	SNR	MSE	UIQI	PSNR
1	Knee	Previous Paper	0.0605	0.0647	0.2906	1.1464e-018	6.8767e-006	352.1354
		Proposed	0.4256	0.1247	0.3867	6.3062e-016	0.0029	406.9438
2	Football	Previous Paper	0.0858	0.0948	0.4325	1.9321e-016	3.3935e-006	306.2934
		Proposed	1.1560	0.2569	0.5902	1.2356e-013	0.0020	362.4104
4	Lena	Previous Paper	0.0731	0.0794	0.3592	1.2745e-005	4.7400e-006	92.0290
		Proposed	0.7309	0.1852	0.4909	0.0064	0.0024	146.0237

This paper presents a comparison between different edge detection models. For this purpose MATLAB 8.1 (R2010a) has been adopted to investigate the difference between edge detection models. Comparison between OTSU and multilevel thresholding edge detection methods are done on different images. Different parameters such as Variance, Mean, Signal to Noise Ratio, Mean square Error (MSE), UIQI and Peak Signal to Noise Ratio (PSNR) are used for analysis and compare the different methods. OTSU and multi-level thresholding are applied on more than 20 images and evaluate the performance of both methods. Out of those images six images shown here. After analysis of those parameter's value and output images, it has been found that the best model is multilevel threshold edge detection model.

Here we got the edges after applying OTSU method, Figure 7.1 represents the example to demonstrate the performance of OTSU method; in above output Figure window, six different images are shown. First part of image represents input image than gradient of image along x axis, gradient of image along y axis, overall gradient which is calculated by using equation 1.1. Then OTSU thresholding level gives the threshold image which shown in Figure and last image shows the outcomes of OTSU method. This method is effective as compare to canny method for weak objects to segment. Same method are applied on different images out of those images some are given below.

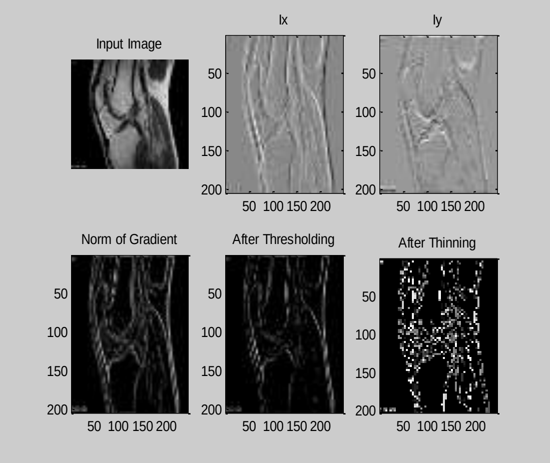

Figure5 Different output stages knee image

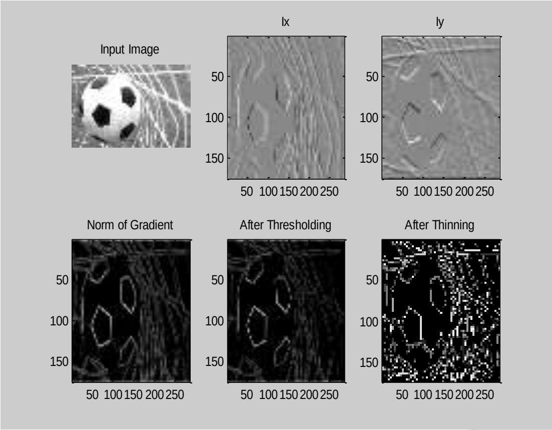

Figure 6 Different output stages of OTSU method for Football image

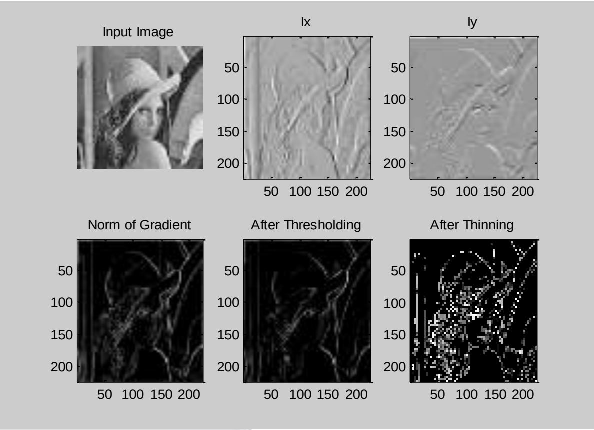

Figure 7 Different output stages of OTSU method for Lena image

5. CONCLUSION.

In multilevel thresholding method has been used to search-out false edge and get the fine and true edges of test image. The CANNY algorithm has difficulty in treating images which contain noise, and it does not have the adaptive ability in the variance of the Gaussian filtering than OTSU based on two level of threshold is used, but it has certain limitation for searching the best threshold value from given histogram and chance of selecting wrong threshold is increases definitely incorrect threshold gives the false edge. For this reason, a new algorithm is presented in this dissertation known as multilevel thresholding OTSU method for Edge Detection. This improved method can suppress noise on edge detection and preserve the edge information, detection with high precision. In this dissertation OTSU thresholding is used to get the correct threshold value from given intensity histogram. Different cluster are made for comparing the class variance between two classes and within class, it automatic decides the level of edges on the basis of intensity of pixel.

In this work we get the high value of SNR and less value of MSE which shows that reception of false edges are less. UIQI values goes towards to 1 gives better performance. Output images are also there in which we can seen the edges of test image on different level of edge detection process. From that we can say proposed method gives the better result for the edge detection rather than tradition methods.

Future Work: The proposed method mostly deals grayscale images, which can be extended to color images. The computational complexity of each algorithm can be studied and schemes need to be devised to implement them in parallel for better response time.

6. REFERENCES

[1] P. D. Sathya and R. Kayalvizhi, “Modified bacterial foraging algorithm based multilevel thresholding for image segmentation,” Engineering Applications of Artificial Intelligence, vol. 24, no. 4, pp. 595–615, 2011.

[2] S. U. Lee, S. Yoon Chung, and R. H. Park, “A comparative performance study of several global thresholding techniques for segmentation,”ComputerVision,Graphics and Image Processing, vol. 52, no. 2, pp. 171–190, 1990.

[3] N. R. Pal and S. K. Pal, “A review on image segmentation techniques,” Pattern Recognition, vol. 26, no. 9, pp. 1277–1294, 1993.

[4] J. Freixenet, X. Munoz, D. Raba, J. Marti, and X. Cufi, “Yet another survey on image segmentation: region and boundary information integration,” in Proceedings of the 7th European Conference on Computer Vision Copenhagen (ECCV ’02), vol. 2352 of Lecture Notes in Computer Science, pp. 408–422,Springer, 2002.

[5] M. Sezgin and B. Sankur, “Survey over image thresholding techniques and quantitative performance evaluation,” Journal of Electronic Imaging, vol. 13, no. 1, pp. 146–168, 2004.

[6] Y. Yang and B. Wu, “A new and fast multiphase image segmentation model for color images,” Mathematical Problems in Engineering, vol. 2012, Article ID 494761, 20 pages, 2012.

[7] B. Wu and Y. Yang, “Local- and global-statistics-based active contour model for image segmentation,” Mathematical Problems in Engineering, vol. 2012,Article ID 791958, 16 pages, 2012.

[8] G. K. Rohde, C. Chen, J. A. Ozolek, andW.Wang, “A general system for automatic biomedical image segmentation using intensity neighborhoods,” International Journal of Biomedical Imaging, vol. 2011, Article ID 606857, 12 pages, 2011.

[9] L.Wang and J. Bai, “Threshold selection by clustering gray levels of boundary,” Pattern Recognition Letters, vol. 24, no. 12, pp. 1983–1999, 2003.

[10] R. Adollah, M. Y. Mashor, H. Rosline, and N. H. Harun, “Multilevel thresholding as a simple segmentation technique in acute leukemia images,” Journal of Medical Imaging and Health Informatics, vol. 2, no. 3, pp. 285–288, 2012.

[11] B. Akay, “A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding,” Applied Soft Computing Journal, vol. 13, no. 6, pp. 3066–3091, 2013.

[12] S. Agrawal, R. Panda, S. Bhuyan, and B. K. Panigrahi, “Tsallis entropy based optimal multilevel thresholding using cuckoo

search algorithm,” Swarm and Evolutionary Computation, vol. 11, pp. 16–30, 2013.

[13] P.D. Sathya and R.Kayalvizhi, “Optimalmultilevel thresholding using bacterial foraging algorithm,” Expert Systems with Applications vol. 38, no. 12, pp. 15549–15564, 2011.

[14] P. D. Sathya and R. Kayalvizhi, “Comparison of intelligent techniques for multilevel thresholding problem,” International Journal of Signal and Imaging Systems Engineering, vol. 5, no. 1, pp. 43–57, 2012.