Multilevel Image Thresholding using OTSU's Algorithm in Image Segmentation

Priya M.S
Research Scholar, Bharathiar University, Coimbatore.
Asso. Prof, Department of Computer Science
St.Anne's F.G.C, Bangalore, India
priwah@gmail.com

Dr. G.M. Kadhar Nawaz
Director, Department of Computer Application
Sona College of Technology,
Salem, India

Abstract— In any image processing research the main focus is the image which is expected to clean to analyze the expressions or features in it. Most of the images are disturbed with noise either because of natural phenomenon or by the data acquisition process. Pre-processing of images are used improve the quality of the image, which makes the subsequent process in image recognition easier. In this paper we discuss the thresholding algorithm for image pre-processing. Image segmentation is the fundamental approach of digital image processing. Among all the segmentation methods, Otsu method is one of the most successful methods for image thresholding because of its simple calculation. Otsu is an automatic threshold selection region based segmentation method. This paper studies various Otsu algorithms.

Index Terms—image processing, thresholding, local thresholding, global thresholding, OTSU, QIR, multilevel thresholding.

1. Introduction

The objective of digital image processing is extracting useful information from images without human assistance. The segmentation process for images with complicated structure is one of the most difficult problems in image processing and has been an active area of research for several decades. Segmentation divides an image into its constituent regions or objects. Segmentation of images is a difficult task in image processing. Segmentation allows extracting objects in images. Segmentation is unsupervised learning. First category is to partition an image based on abrupt changes in intensity, such as edges in an image. Second category is based on partitioning an image into regions that are similar according to predefined criteria [1]. This paper has taken the study of these second category threshold techniques. Survey of some of the methods found in Weszka [19], Sahoo et al. [18], and Lee et al. [29]. In many applications of image processing, the gray levels of pixels belonging to the object are quite different from the gray levels of the pixels belonging to the background. Thresholding becomes then a simple but effective tool to separate objects from the background. Examples of thresholding applications are document image analysis where the goal is to extract printed characters , logos, graphical content, musical scores, map processing where lines, legends, characters are to be found, scene processing where a target is to detected, quality inspection of materials. Other applications include cell images and knowledge representation, segmentation of various image modalities for non-destructive testing (NDT) applications, such as ultrasonic images in, eddy current images, thermal images, X-ray computed tomography (CAT), laser scanning confocal microscopy, extraction of edge field, image segmentation in general, spatio-temporal segmentation of video images etc. The output of the thresholding operation is a binary image whose gray level of 0 (black) will indicate a pixel belonging to a print, legend, drawing, or target and a gray level of 1 (white) will indicate the background. The main difficulties associated with thresholding such as in documents or NDT applications occur when the associated noise process is non-stationary, correlated and non-Gaussian. Other factors complicating thresholding operation are ambient illumination, variance of gray levels within the object and the background, inadequate contrast, object shape and size non-commensurate with the scene. Finally the lack of objective measures to assess the performance of thresholding algorithms is another handicap.

2. THRESHOLDING

Threshold is one of the widely methods used for image segmentation. It is useful in discriminating foreground from the background. By selecting an adequate threshold value T, the gray level image can be converted to binary image.

ISSN 2229-5518

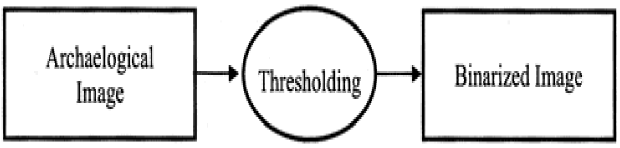

Fig. 1. The process of thresholding along with its inputs and outputs.

The binary image should contain all of the essential information about the position and shape of the objects of interest (foreground).

If g(x, y) is a threshold version of f(x, y) at some global threshold T it can be defined as [1],

$$g(x, y) = 1 \text{ if } f(x, y) \geq T = 0 \text{ otherwise}$$

Thresholding operation is defined as:

$$T = M\,[x, y, p(x, y), f(x, y)]$$

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). Thresholding techniques can be categorized into two classes: global threshold and local (adaptive) threshold. In the global threshold, a single threshold value is used in the whole image. In the local threshold, a threshold value is assigned to each pixel to determine whether it belongs to the foreground or the background pixel using local information around the pixel.

2.1 Global thresholding

When T depends only on f (x, y), only on gray-level values and the value of T solely relates to the character of pixels, this thresholding technique is called global thresholding.

2.2 Local thresholding

If threshold T depends on f(x, y) and p(x, y), this thresholding is called local thresholding. This method divides an original image into several sub regions [2], and chooses various thresholds T for each sub region reasonably [3].

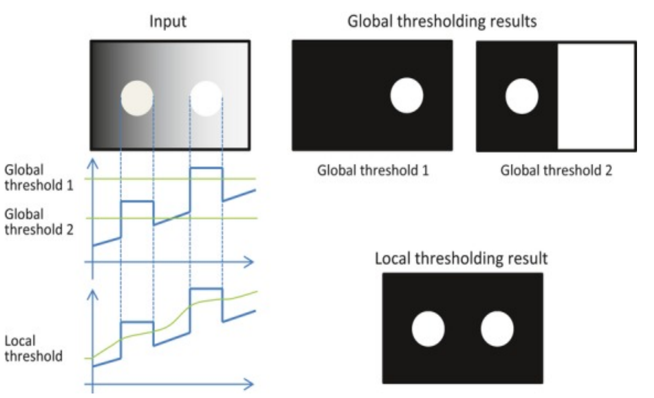

Fig.2 Binarization by Global and Local Thresholding.

3. Global Thresholding Algorithm

Global (single) thresholding method is used when there the intensity distribution between the objects of foreground and background are very distinct. When the differences between foreground and background objects are very distinct, a single value of threshold can simply be used to differentiate both objects apart. Thus, in this type of thresholding, the value of threshold T depends solely on the property of the pixel and the grey level value of the image. The Global thresholding algorithm works as:

1. Select an initial estimate for T.
2. Segment the image using T. This will produce two groups of pixels. G1 consisting of all pixels with gray level values >T and G2 consisting of pixels with values <=T.
3. Compute the average gray level values mean1 and mean2 for the pixels in regions G1 and G2.
4. Compute a new threshold value T=(1/2)(mean1 +mean2)
5. Repeat steps 2 through 4 until difference in T in successive iterations is smaller than a predefined parameter T0.

Some most commonly used global thresholding methods are Otsu method, Iterative thresholding and QIR [6]. Otsu's algorithm is a popular global thresholding technique. Moreover, there are many popular thresholding techniques such as Kittler and Illingworth, Kapur, Tsai, Huang, Yen and et al [9].

3.1 OTSU Algorithm

In image processing, segmentation is often the first step to pre-process images to extract objects of interest for further analysis. Segmentation techniques can be generally categorized into two frameworks, edge-based and region based approaches. As a segmentation technique, Otsu's method is widely used in pattern recognition, document binarization [7], and computer vision. The assumptions made in Otsu's model are:

- Histogram (and the image) is bimodal.
- There is no use of spatial coherence, nor any other notion of object structure.
- Assumes stationary statistics, but can be modified to be locally adaptive.
- Assumes uniform illumination (implicitly), so the bimodal brightness behavior arises from object appearance differences only.

In many cases Otsu's method is used as a pre-processing technique to segment an image for further processing such as feature analysis and quantification. Otsu's method searches for a threshold that minimizes the intra-class variances of the segmented image and can achieve good results when the histogram of the original image has two distinct peaks, one belongs to the background, and the other belongs to the foreground or the signal. The Otsu's threshold is found by searching across the whole range of the pixel values of the image until the intra-class variances [8] reach their minimum. As it is defined, the threshold determined by Otsu's method is more profoundly determined by the class that has the larger variance, be it the background or the foreground. As such, Otsu's method may create suboptimal results when the histogram of the image has more than two peaks or if one of the classes has a large variance.

The following formulas are used to calculate the total mean and variance.

The pixels are divided into 2 classes, C_1 with gray levels [1, t] and C_2 with gray levels [t+1, ... ,L].

The probability distribution for the two classes is:

$$C_1:\ p_1/w_1(t), \ldots, p_t/w_1(t) \text{ and}$$
$$C_2:\ p_{t+1}/w_2(t), \ldots, p_L/w_2(t)$$

$$\text{Where } w_1(t)=\sum_{i=1}^{t} p_i$$

$$\text{and} \quad w_2(t)=\sum_{i=t+1}^{L} p_i$$

Also, the means for the two classes are

$$\mu_1=\sum_{i=1}^{t} i p_i / w_1(t)$$

and

$$\mu_2=\sum_{i=t+1}^{L} i p_i / w_2(t)$$

Using Discriminant Analysis, Otsu defined the between-class variance of the threshold image as

$$\sigma_B^{\,2} = w_1(\mu_1-\mu_T)^2 + w_2(\mu_2-\mu_T)^2 .$$

For bi-level thresholding, Otsu verified that the optimal threshold t* is chosen so that the between-class variance $_B$ is maximized; that is,

$$t^* = \underset{1 \le t < L}{Arg\,Max} \{\sigma_B^{\,2}(t)\}$$

3.2 Multi Level Thresholding

A new iterative method is based on Otsu's method but differs from the standard application of the method in an important way. At the first iteration, we apply Otsu's method on an image to obtain the Otsu's threshold and the means of two classes [10] separated by the threshold as the standard application does. Then, instead of classifying the image into two classes separated by the Otsu's threshold, our method separates the image into three classes based on the two class means derived. The three classes are defined as the foreground with pixel values are greater than the larger mean, the background with pixel values are less than the smaller mean, and more importantly, a third class we call the "to-be-determined" (TBD) region with pixel values fall between the two class means. Then at the next iteration, the method keeps the previous foreground and background regions unchanged and re-applies Otsu's method on the TBD region[4] only to, again, separate it into three classes in the similar manner. When the iteration stops after meeting a preset criterion, the last TBD region is then separated into two classes, foreground and background, instead of three regions. The final foreground is the logical union of all the previously determined foreground regions and the final background is determined similarly. The new method is almost parameter free except for the stopping rule for the iterative process [11] and has minimal added computational load.

3.3 Quadratic Integral Ratio Algorithm

The QIR technique was found superior in thresholding handwriting images where the following tight requirements need to be met:

1. All the details of the handwriting are to be retained
2. The papers used may contain strong coloured or patterned background
3. The handwriting may be written by a wide variety of writing media such as a fountain pen, ballpoint pen, or pencil.QIR is a global two stage thresholding technique [5].

The first stage of the algorithm divides an image into three sub images: foreground, background, and a fuzzy sub image where it is hard to determine whether a pixel actually belongs to the foreground or the background.

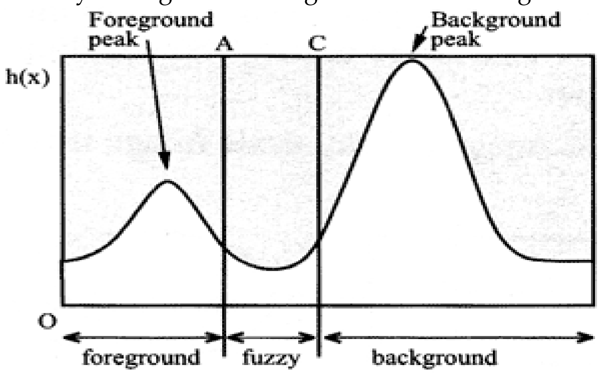

Fig.3 Three sub images of QIR method

Two important parameters that separate the sub images are A, which separates the foreground and the fuzzy sub image, and C, which separate the fuzzy and the background sub image. If a pixel's intensity is less than or

ISSN 2229-5518

equal to A, the pixel belongs to the foreground. If a pixel's intensity is greater than or equal to C, the pixel belongs to the background. If a pixel has an intensity value between A and C, it belongs to the fuzzy sub image and more information is needed from the image to decide whether it actually belongs to the foreground or the background.

4. RESULTS AND DISCUSSION

All the thresholding were implemented using MATLAB R2014a. In Otsu's method we exhaustively search for the threshold that minimizes the intra-class variance, defined as a weighted sum of variances of the two classes. OTSU Algorithm computes histogram and probabilities of each intensity level

1. Set up initial $\omega_i(0)$and $\mu_i(0)$
2. Step through all possible thresholds t=1,…maximum intensity
    1. Update ω_i and μ_i
    2. Compute $\sigma^2_b(t)$
3. Desired threshold corresponds to the maximum

The matlab function used for Global image threshold using OTSU's method is graythresh().

- level = graythresh(I) computes a global threshold (level) that can be used to convert an intensity image to a binary image with im2bw. level is a normalized intensity value that lies in the range [0, 1]. The graythresh function uses Otsu's method, which chooses the threshold to minimize the intraclass variance of the black and white pixels. Multidimensional arrays are converted automatically to 2-D arrays using reshape. The graythresh function ignores any nonzero imaginary part of I.
- [level EM] = graythresh(I) returns the effectiveness metric, EM, as the second output argument. The effectiveness metric is a value in the range [0 1] that indicates the effectiveness of the thresholding of the input image. The lower bound is attainable only by images having a single gray level, and the upper bound is attainable only by two-valued images.

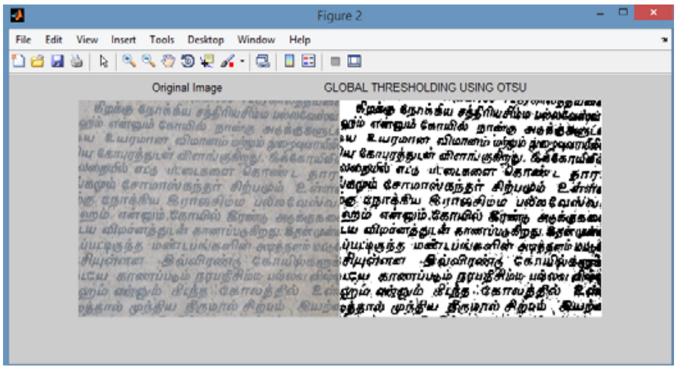

Fig. 4 Global image threshold using OTSU's method

Multilevel image thresholds using Otsu's method is done in matlab using multithresh().

- thresh = multithresh(A) returns the single threshold value thresh computed for image A using Otsu's method. thresh is an input argument to imquantize that converts A into a binary image.
- thresh = multithresh(A,N) returns thresh containing N threshold values using Otsu's method. thresh is a 1xN vector which can be used to convert image A into an image with N + 1 discrete levels using imquantize.
- [thresh,metric] = multithresh(___) returns metric, a measure of the effectiveness of the computed thresholds. metric is in the range [0 1] and a higher value indicates greater effectiveness of the thresholds in separating the input image into N + 1 classes based on Otsu's objective criterion.

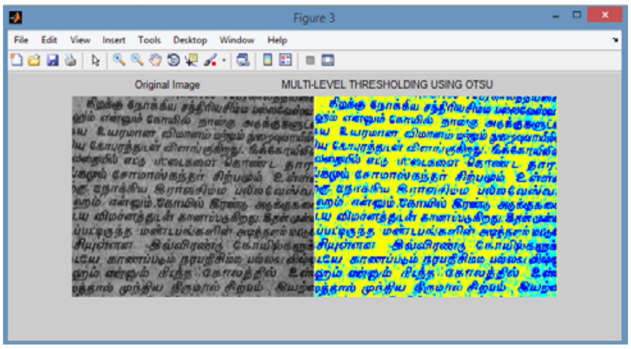

Fig. 5 Multilevel image thresholds using OTSU's method

From the experimental results, the performance of global thresholding techniques including Otsu's method is shown to be limited by the small object size, the small mean difference, the large variances of the object and the background intensities and the large amount of noise added.

5. CONCLUSION

In this paper we have used the different thresholding methods of OTSU's algorithm. OTSU's method exhibits a

ISSN 2229-5518
relatively good performance if the histogram has a bimodal distribution. But if the object area is small compared to the background area, then the histogram no longer exhibits bimodality. And if the variances of the object and the background intensities are large compared to the mean difference or the image is severely corrupted by additive noise, the sharp valley of the gray level histogram is degraded. Then the possibly incorrect threshold determined by OTSU's method results in the segmentation error. The limitations of Otsu's method can be solved effectively by two-dimensional Otsu's method. In this approach, the gray-level value of each pixel as well as the average value of its immediate neighborhood is studied so that the binarization results are greatly improved, especially for those images corrupted by noise.

REFERENCES

[1] Er. Nirpjeet kaur and Er Rajpreeet kaur, "A review on various method of image thresholding", International Journal on Computer Science and Engineering (IJCSE), Vol .3, No.10 October 2011.

[2] J. Gong, L. Li, and W. Chen, "Fast recursive algorithms for two-dimensional thresholding," Pattern Recognition,vol. 31, no. 3,pp. 295–300, 1998.

[3] J.S. Weszka: A Survey of Threshold Selection Techniques, Computer Graphics and Image Processing, Vol. 7, pp. 259-265, 1978.

[4] L. Dongju and Y. Jian, "Otsu method and k-means," in Hybrid Intelligent Systems, 2009. HIS '09. Ninth International Conference on, vol. 1, 2009, pp. 344–349.

[5] LIU Jian-zhuang, Li Wen-qing, "The Automatic threshold of gray level pictures via Two-dimentional Otsu Method",Acta Automatic Sinica,1993

[6] N. Ostu, A threshold selection method from gray-level histogram, IEEE Trans. Systems Man Cybemet. SMC-8, 1978, 62-66

[7] P.K. Sahoo, S. Soltani, A.K.C. Wong, Y.C. Chen, "Survey of Thresholding techniques, Computer Vision, Graphics, and Image Processing", Vol. 41, pp. 233-260, 1988.

[8] Priyanka G. Kumbhar, 2 Prof. Sushilkumar N. Holambe," A Review of Image Thresholding Techniques", International Journal of Advanced Research in Computer Science and Software Engineering , Volume 5, Issue 6, June 2015.

[9] R.C. Gonzalez, R.E. Woods (2008), Digital Image Processing, Prentice-Hall, Englewood Cliffs, Nj

[10] Rangachar Kasturi, Louis Lam, Seong - Whan Lee & Ching Y. Suen. "Document Image Analysis ".

[11] S.U. Lee, S.Y. Chung, R.H. Park,"A Comparative Performance Study of Several Global Thresholding Techniques for Segmentation, Computer Vision, Graphics, and Image Processing", Vol. 52, pp. 171-190, 1990.

[12] Van Solihin and C. G. Leedham ,"The Multi-stage Approach to Grey-Scale Image Thresholding for Specific Applications"

[13] W. X. Kang, Q. Q. Yang, R. R. Liang ,"The Comparative Research on Image Segmentation Algorithms", IEEE Conference on ETCS, pp. 703-707, 2009.

[14] Zhong Qu ,Li Hang ,"Research on image Segmentation Based on the Improved Otsu Algorithm", IEEE Conference on IHMSC, Vol. 2, pp. 228 – 231, 2010.

[15] Z. Ningbo, W. Gang, Y. Gaobo, and D. Weiming, "A fast 2d otsu thresholding algorithm based on improved histogram," in Pattern Recognition, 2009. CCPR 2009. Chinese Conference on, 2009, pp. 1 – 5.