Image Analysis Using Improved Otsu's Thresholding Method

Himanshu Makkar
(M.Tech Scholar)
Dept of Electronics & Communication
Jaipur National University
Jaipur, India
himanshu.makkar@gmail.com

Aditya Pundir
(Associate Professor)
Dept of Electronics & Communication
Jaipur National University
Jaipur, India
aditya.pundir1985@gmail.com

Abstract-- A fresh and new algorithm for retrieval of image is accessible using improved Otsu thresholding techniques in this chapter. The fundamental design work of the new algorithm is that the basic segments of the image are used to retrieve images within a digital library. the histogram derived from threshold. These methods include some knowledge of the distribution of an image, & will result in the less miss-classification. Isodata algorithm is the iterative process for finding a threshold value. It first segments the image into two regions according to a temporary threshold value chosen .It then calculates the mean value of an image corresponding to the two segmented regions. Calculate the new threshold value & repeat until the threshold value does not change any more. Finally, choose this value for the threshold segmentation. One of the advantages of the algorithm is that, for a given retrieval image, the user can select a query segment with which to perform retrieval, thus it can satisfy different needs from different users.

Keywords*: Otsu's thresholding, Distribution image, Histogram, Isodata algorithm, Segmentation,*

*****

I. INTRODUCTION

Various general-purpose algorithms & procedures have been build up for image segmentation. Since there is no universal explanation to the image segmentation dilemma, these methods frequently have to be merged with the domain knowledge so as to efficiently resolve an image segmentation problem for the problem domain. Various accepted techniques are used in the industry including the, '***Otsu's method'*** (maximum variance)

Thresholding is the easiest method of the image segmentation. From, thresholding converts any grayscale image to binary images . Throughout the thresholding process, each individual pixel in an image is marked as "***object***" pixel if it's value is greater than the some threshold value (assume an object to be intense than the background) & as "***background***" pixels or else. This principle is known as the '***threshold above'***. Alternatives comprise '***threshold below'***, opposite of the threshold above;

Threshold inside**:** Refers to a pixel tagged as "***object***" if its value lies between two thresholds; &
Threshold outside**:** It is the converse of the threshold inside.

Usually, an object pixel is assigned a value "***1***" while the background pixel is assigned a value "***0***." At last, a binary image is generated by coloring every pixel white or black, depending upon the pixel's labels.

II. THRESHOLD SELECTION

The chief constraint in the thresholding process is the selection of the threshold value (or the values, as mentioned previously). A variety of different methods for selecting a threshold persist; users can physically select a threshold value, or a particular thresholding algorithm can calculate a value mechanically, which is identified as automatic thresholding. A basic method is to select the mean or median value, the underlying principle being that if the object pixels are sharper than the background, then these should also be sharper than the average. In a noiseless image having consistent background & object values, the mean or median will perform well as the threshold, Nevertheless it is commonly the case. A more sophisticated approach is to create the histogram of the image pixel intensities & make use of the valley point as the threshold. The histogram approach supposes that there are a number of average values for equally the background & object pixels, but the real pixel values have some alteration about these average values.
On the other hand, it may be calculative costly, & image histograms may not have apparently characterized valley points; frequently the choice of an exact threshold is complicated. In such cases a Unimodal Threshold Selection Algorithm (UTSA) may be more suitable. One method which is comparatively easier, does not need an extra specific knowledge of the image, & is tough next to image noise. Here is the following ***iterative method:***

- An initial threshold ('***T'***) is selected; it can be done casually or according to any other technique as required.
- The image is then segmented in the object & the background pixels as explained above, generating two sets**:**
  - $G_1 = \{ f(m,n): f(m,n) > T \}$ (object pixels)
  - $G_2 = \{ f(m,n): f(m,n) \leq T \}$ (background pixels) (note, f(m,n) is the significance of the pixel traced in the m^{th} column, n^{th} row)
  - The average of each set is calculated.
    - ✓ M_1 = average value of G_1
    - ✓ M_2 = average value of G_1
  - A new threshold is generated ,i.e., the average of m_1 and m_2
    - ✓ $T' = (m_1 + m_2)/2$
- Go back to step two, now use the new threshold as calculated in step four, keep repeating till the new threshold resembles the one prior it (i.e. till convergence has been attained).

This iterative algorithm is a unique case of one-dimensional of the k-means clustering algorithm that has been

 ISSN: 2321-8169
2122 – 2126

demonstrated to congregate at a *local* minimum—indicating that a unlike initial threshold may give a dissimilar absolute result

III. OTSU THRESHOLDING

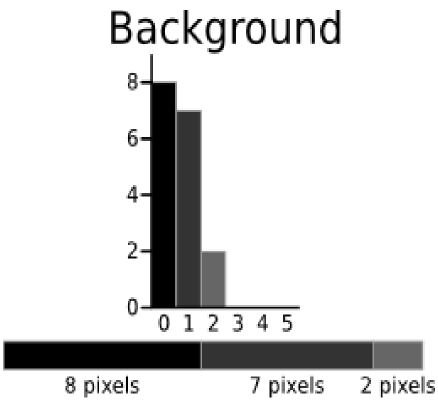
In the year 1979, Otsu proposed the highest class variance method(known as the ***Otsu method***). For the easy computation, constancy & effectiveness, this method has been extensively used. It is a well-performed mechanical threshold selection process, & the time it consumes is considerably less than the other thresholding algorithms.

Otsu's thresholding method entails iterating through all the probable threshold values & computing a gauge of spread for all the pixel levels at each side of the threshold, i.e. the pixels which either fall in the foreground or background. The main objective is to calculate the threshold value at places where the addition of foreground & background spreads is at its least possible value.

The algorithm will be revealed using the simple 6x6 image exposed below. The histogram for the image is also given away subsequently to it. To make simpler the description, only 6 grayscale levels are used.

The computations for finding the foreground & background variances (the gauge of spread) for a single threshold are currently revealed. In this particular case the threshold value is 3.

(A)



(B)

Figure 1: (a) A 6-Level Grayscale Image & (b) Its Histogram

Weight $W_b = \frac{8+7+2}{36} = 0.4722$

Mean $\mu_b = \frac{(0 \ X\ 8)+(1\ X\ 7)+(2\ X\ 2)}{17} = 0.6471$

Variance

$$\sigma^2{}_b = \frac{((0-0.6471)^2\ X\ 8)+((1-0.6471)^2\ X\ 7)+((2-0.6471)^2\ X\ 2)}{17}$$

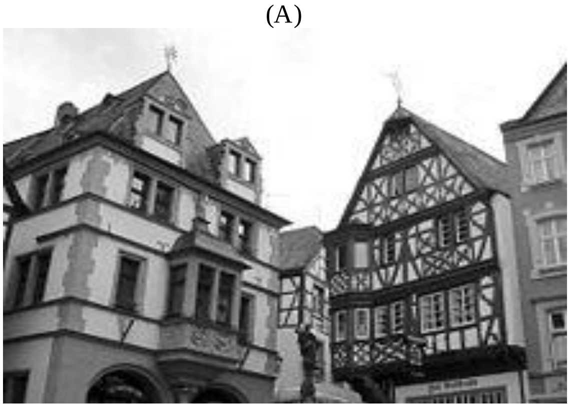
$$= \frac{(0.4187\ X\ 8)+(0.1246\ X\ 7)+(1.8304\ X\ 2)}{17}$$

$$= 0.4637$$

IV. OTSU's METHOD

In the computer vision & image processing, '***Otsu's method***' is used to mechanically execute histogram shape-based image thresholding, or, the decrease of a gray level image to a binary image. The algorithm take for granted that the image to be thresholded consists two groups of pixels or bi-modal histogram (for instance, foreground & background) and then evaluates the optimum threshold partitioning those two classes so that their joint spread (intra-class variance) is negligible. The expansion of the basic method to multi-level thresholding is addressed to as the Multi Otsu method. Otsu's method is named so after the Nobuyuki Otsu Method.

(A)



(B)

Figure 2: (a) Original Image (b) Image Thresholded Using Otsu's Algorithm

In Otsu's method we vigorously look for the threshold which reduces the intra-class variance (within the class variance), classified as the weighted sum of the variances of the two classes:

$$\sigma^2{}_\omega = \omega_1(t)\sigma^2{}_1(t) + \omega_2(t)\sigma^2{}_2(t)$$

where,
Weights ω_i *:* are the probabilities of two classes partitioned through a threshold t and σ_i^2 *:* are the discrepancies of these sets.
Otsu displays that by minimizing the intra-class variances is same as maximizing the inter-class variance as:

$$\sigma^2{}_b(t) = \sigma^2 - \sigma^2{}_\omega(t) = \omega_1(t)\,\omega_2(t)\,[\mu_1(t) - \mu_2(t)]^2$$

which is stated in terms of the class probabilities ω_i & the class means μ_i.

The class probability $\omega_1(t)$ will then be calculated from the histogram as t*:*

$$\omega_1(t) = \sum_0^t P\,(i)$$

Where as the class mean $\mu_1(t)$ is given as*:*

$$\mu_1(t) = \sum_0^t P(i)\,x(i)$$

Where, $x(i)$*:* refers to the value at the centre of i in the histogram bin.
Likewise, $\omega_2(t)$ & μ_t on the right-hand face, of the histogram for bins are bigger than t can be calculated. The class probabilities & the class means can be calculated iteratively. This scheme yields an efficient algorithm.

Algorithm:

- Calculate histogram & probabilities of each intensity level
- Place an initial $\omega_i(0)$ & $\mu_i(0)$
- Move throughout all probable thresholds $t = 1 \ldots$ maximum intensity
- Update ω_i & μ_i and calculate $\sigma_b^2(t)$
- Required threshold resembles to the maximum $\sigma_b^2(t)$

$$\textbf{Desired threshold} = \frac{\textit{threshold 1} + \textit{threshold 2}}{2}$$

where,
$\sigma_{b1}^2(t)$*:* is the greatest maximum &
$\sigma_{b2}^2(t)$*:* is the greater or equal maximum

OR

- Let *'N'* be the pixels of the segmenting image, if there exist *'L'* gray levels $(0,1,\ldots, L-1)$, & *'n_i'* denotes the pixels whose gray level is *'i'*, then

$$N = \sum_{i=0}^{L-1} n_i$$

& the probability density distribution is expressed with the form of histogram

$$p_i = n_i / N, \quad \sum_{i=0}^{L-1} n_i = 1, \quad p_i \geq 0$$

Suppose the image is separated into two categories C_0 & C_1 as target & back-goround respectively consisting of threshold *'t'*, then C_0 & C_1 correspondingly to the pixels whose grey level are $\{0,1,\ldots,t\}$ & $\{t+1, t+2, \ldots, L-1\}$.
Assume $\sigma^2{}_B(t)$ is the class variance when the threshold of the histogram is 't', then the optimal threshold can be obtained by computing the maximum of $\sigma^2{}_B(t)$, i.e.

$$\sigma^2{}_B(t^*) = \max_{0 \leq t \leq L-1} \{\sigma^2{}_B(t)\}$$

where, $\sigma^2{}_B(t) = \omega_0(t)[\mu_0(t)-\mu_T]^2 + \omega_1[\mu_1(t)-\mu_T]^2$

$$\sigma_T = \sum_{i=1}^{K} (i-\mu_T)^2 P_i, \qquad \eta = \max \sigma_B / \sigma_T$$

$$\mu_T = \sum_{i=0}^{L-1} iP_i, \qquad \omega_0(t) = \sum_{i=0}^{l} P_i, \qquad \omega_1(t) = 1-\omega_0(t),$$

$$\mu_T = \sum_{i=0}^{l} iP_i, \qquad \mu_0(t) = \mu(t)/\omega_0(t), \qquad \mu_1(t) = \mu_T - \mu_t / 1 - \omega_0(t)$$

OR
Otsu' Algorithm

$$(1)\ P_i \leftarrow \frac{n_i}{n}, \quad where\ \ n = \sum_{i=1}^{G} n_i$$

$$(2)\ \mu_r = \sum_{k=1}^{G} Kp_k$$

(3) Do for k = 1 to G

$$w(k) \leftarrow \textstyle\sum_{i=1}^{k} p_i$$

$$u(k) \leftarrow \sum_{i=1}^{k} ip_i$$

$$\sigma^2{}_g(k) \leftarrow \frac{[u_r\,w(k) - u(k)]^2}{w(k)\,[1 - w(k)]}$$

(4) Select K such that $\sigma^2{}_g(k)$ is maximized

Note that,

$$w_0 = w(k), u_0 = \sum_{i=1}^{k} ip\left(\frac{i}{c_0}\right) = \frac{u(k)}{w(k)}$$

$$w_1 = 1 - w(k), u_1 = \sum_{i=k=1}^{G} ip\left(\frac{i}{c_1}\right) = \frac{[u(t) - u(k)]}{[1 - w(k)]}$$

$$\sigma^2{}_w = w_0\sigma^2{}_0 + w_1\sigma^2{}_1$$

$$\sigma^2{}_g = w_0(u_0 - u_r)^2 + w_1(u_1 - u_1)^2$$

$$\sigma^2{}_r = \sum_{i=1}^{G} (i - u_r)^2\,p_i$$

$$\sigma^2{}_w + \sigma^2{}_g = \sigma^2{}_r$$

$$\epsilon = \frac{\sigma^2{}_g}{\sigma^2{}_r}, \quad k = \frac{\sigma^2{}_r}{\sigma^2{}_w}, \quad \lambda = \frac{\sigma^2{}_g}{\sigma^2{}_w}$$

V. DRAWBACKS OF OTSU ALGORITHM

- Dealing with the grey level images discontinuous the conventional Otsu algorithm could not find a good union of thresholds to the global optimum.
- Even though the Otsu algorithm does not make any hypothesis on the probability density function, & states the two objective & background probability density functions, thus it presume the two probability density functions & can be stated by making use of the two statistics.
- The Otsu algorithm fails when the global distributions of the target & the background vary extensively.
- When there are two classes in the image the Otsu algorithm is suitable.
- The method must be customized if there exists more than two classes in the image, in order to decide multi-threshold. This loom allows the largest among-class variance & the least in-class variance
- Otsu algorithm divides the image into two groups, yet no actual sense is made by the division. This method could not be applied straight in variable illumination conditions case.

VI. IMPROVED OTSU ALGORITHM

- Compute the grey level 'L' of an image also calculate the mean grey level value 'μ_T' of the image, & round it off from 'μ_T' to $\lfloor$ 'μ_T'$\rfloor$ i.e. $L = \lfloor \mu_T \rfloor$
- Allocate primary value for segmentation time $J = 1$.
  - The execution of the Otsu algorithm by calculating the pixel 'N' , threshold 'K' threshold Selection function 'η' , in-class variance σ_w .
  - Iterate, $N(J) = N$; $K(J) = K$ & ; $L = K$; $\eta(J) = \eta$; $\sigma(J) = \sigma w$; $J = J + 1$;
  - If $J \leq 1$ then go to the step (3).
  - Compute ε where, $\varepsilon = [N(J-1) - N(J)] / N(J-1) - [\sigma(J-1) - \sigma(J)] / \sigma(J-1)$.
  - If $\varepsilon \geq 0$ then go to the step (3).
  - Evaluate the threshold analogous to the largest '$\eta(J)$' ,as the best threshold value 'K' [**Appendix C_2,** pg.109].

VII. RESULTS OF SEGMENTATION

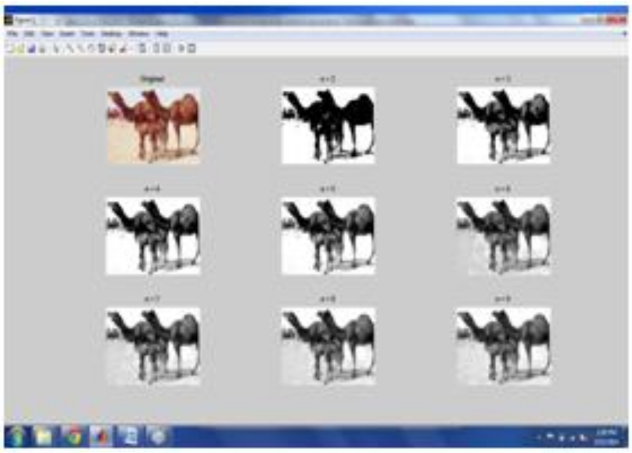


Figure 3: Segmented Image Using Improved Otsu's Method

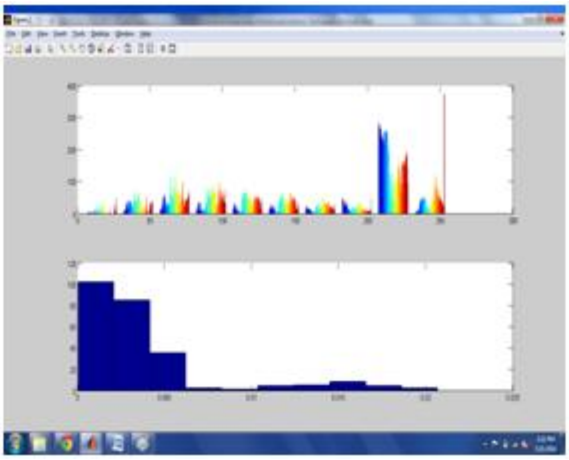


Figure 4: CAMEL: Histogram of Original Image & Segmented Image

VIII. CONCLUSION

The intention of image segmentation is to cluster the pixels of an image into the image regions. Segmentation can be used for image compression, object recognition, & image suppression processing. The main objective of the matter presented in the above chapters is to endow with a sense of perception about the beginnings of the digital image processing &, more significant, about the present & future areas of application of this technology.

This report re-examines the current segmentation techniques, & the key propensity of each method with their standard ideas, application field, benefits & shortcomings are considered. Each of the pixels in a region are analogous with respect to some features or calculated property, such as colour, intensity, or texture. Adjoining regions are appreciably different with respect to the identical characteristics.

We found that the results become poorer when we analyse our algorithm on the texture images with the same parallel measurement. And the key reason behind it is that the pixel value variance is actually very large in the texture region, which means that the pixels in the identical texture region may have minute similarities & are segmented into various groups.

REFRENCES

[1] Dzyubachyk O, Niessen W, Meijering E: Advanced Level - Set Based Multiple – Cell Segmentation and Tracking in Time – Lapse Fluorescence Microscopy Images. In IEEE International Symposium on Biomedical Imaging: From Nano to Macro Edited by: Olivo Marin JC, Bloch I, Laine A. IEEE, Piscataway, NJ; 2008:185-188.

[2] Allen Y. Yang, John Wright, Yi Ma, and Shankar Sastry. Unsupervised segmentation of natural images through miserable data concreteness. CVIU 2007.

[3] Jitendra Malik, Serge Belongie, & Thomas Leung, “Contour & texture study for image segmentation”, International Journal of Computer Vision, Kluwer Academic Publishers, vol.43(1), 2001, pp.7-27

ISSN: 2321-8169
2122 – 2126

[4] F. Sun, J. He," The Normalized Cuts Based Image Segmentation Techniques," 2009 2nd International Convention on Information and Computing Science

[5] F. Sun, J. He, "The Remote-Sensing Image Segmentation Using Textons In Normalized Cuts Framework," Proceedings of the 2009 IEEE International Convention on Mechatronics & Automation, August 9 - 12, Changchun, China.

[6] E. Sharon, M. Galun, D. Sharon, R. Basri, and A. Brandt, "Hierarchy & adaptivity in segmenting optical prospects," Nature, vol. 442, no. 7104, pp. 810–813, 2006.

[7] W. K. Pratt, "Image Segmentation," in *Digital image processing*, 4nd ed. Wiley, 2008, pp. 579-622.

[8] C. Pantofaru, M. Hebert, The assessment of image segmentation algorithms, Technical Report CMU-05-40, CMU (2005). 2, 14