Image Segmentation by Otsu Method

Daniela O. Albanez[1], Marcos N. Rabelo[1], Sérgio F. da Silva[1], Marcos A. Batista[1]

[1]Graduate Program in Modeling and Optimization, Federal University of Goiás
Av. Dr. Lamartine Pinto de Avelar, 1120, CEP 75704-020, Catalão, GO, Brasil

doalbanez@hotmail.com, rabelo@dmat.ufpe.br, sergio@ufg.br, marcos.batista@pq.cnpq.br

Abstract. *Image segmentation has become in indispensable task in many image applications. In order to analysis this issue, the use of thresholding technique is well-known. Because of its applicability, quite a number of thresholding methods have been proposed over the years. This paper attempts to undertake the study of image segmentation and as a result of this study was implemented OTSU algorithm that has the goal to convert images to black and white and so make them better identifiable.*

1. Introduction

Segmentation is usually first step in the analysis of an image, with the idea of segmenting an object by dividing the parts, or parts considered homogeneous according to some criterion uniformity, and ends when the objects interest has been separated. There are several segmentation techniques being used in literature. In generally this techniques is based on value number of each pixel in the texture analysis and image histogram [Gonzalez and Woods 2002].

The image segmentation consist in to subdvide the image into its constituent parts. The idea of image segmentation is to classify pixels of set of pixels. It is the first analysis stage image [Acharya and Ray 2005, Gonzalez and Woods 2002, Petrou and Bosdogianni 1999]. Segmentation helps identify differences between two or more objects, both among themselves and between each other and the background. The segmentation is based on the basic properties of intensity values in gray levels and may be a discontinuity or similarity. The discontinuity is due to the partition of the image by intensity values of gray. While the similarity is based on partitioning an image into regions that are similar according to a set of pre-defined criteria, thresholding and region growing are examples of methods in this category.

Thresholding is the identification of the best threshold in which object distinct from the background, several methods have been proposed [Otsu 1978, kapur and Wong 1985]. In this context, there are two types of thresholding algorithms:

- Global thresholding algorithms
- Local or adaptive thresholding algorithms

In global thresholding, a single threshold for all the image pixels is used. When the pixel values of the components and its background are fairly consistent in their respective values over the entire image, global thresholding could be used. In adaptive thresholding, different threshold values for different local areas are used. In this paper adaptive thresholding is used. In order to performs the analysis of images by thresholding technique, we will employee Otsu's method, which is considered one of the best method for adaptive thresholding images [Otsu 1978].

2. Theoretical Foundation

2.1. Histogram

The histogram is a graph that represents values of the shades of the pixels of an image with a particular brightness [Gonzalez and Woods 2002, Petrou and Bosdogianni 1999], in other words, is the representation of the distribution of colors in an image. The histogram shows detail in the shadows, midtones, highlights, but first of all shows the global image description. If the histogram is narrow, then the image is barely visible, it is because the image gray levels are low. On the other hand, an image with higher contrast and provides a uniform visibility of these gray level histogram [Acharya and Ray 2005].

According [Gonzalez and Woods 2002], the histogram is a discrete probability density function represented by:

$$p_x = \frac{n_x}{N} \tag{1}$$

where:

x is the gray level,
n_x is the number of the pixels in the image having gray level x and
N is the total number of the pixels in the image.

The $x \in \{1, 2, \ldots, L\}$, where L is the value of the pixels in the scanned image.

2.2. Thresholding

The thresholding is one of the simplest techniques of segmentation, it causes the image to grayscale becomes a binary image, separating regions of an image in two other regions - the white background determined by the value (1) and the black object determined by the value (0), thus having a black/white image. This is only possible by defining a threshold T that separates groups of image gray levels [Parker 2011, Jahne and Haubecker 2000, Gonzalez and Woods 2002].

Segmentation occurs by scanning the image, pixel by pixel, thereby distinguishing the background of the object, depending on the relationship between the pixel value and the threshold value. However, the thresholding is effective when the subject is very different gray levels, thus the gray levels of the object and background form distinctly two peaks and the thresholding is trivial.

Suppose that the gray level histogram corresponds to an image $f(x, y)$ composed of dark objects on the light background, in such a way that object and background pixels have gray levels grouped into two dominant modes. One obvious way to extract the objects from the background is to select a threshold T that separates these modes. Then any point (x, y) for which $f(x, y) > T$ is called an object point, otherwise, the point is called a background point.

Mathematically, thresholding may be defined as:

$$g(x,y) = \begin{cases} object & \text{if } f(x,y) > T \text{ ,} \\ background & \text{if } f(x,y) \leq T. \end{cases} \tag{2}$$

where $f(x, y)$ is the input image, T is the threshold and $g(x, y)$ is the output image (thresholding).

Usually it is very common to use the histogram to select a threshold for the thresholding and a threshold is obvious, when occurs at the low point between two peaks in the histogram. In this context, a histogram is a vector having the same number of dimensions as the image does gray levels, [Parker 2011, Petrou and Bosdogianni 1999].

2.3. Adaptive Thresholding

The adaptive thresholding technique which is a local aspects analyzes the image and determines a threshold for each region defining whether the pixel is considered black or white.

This technique has algorithms from simple solutions to the optimal solution of local thresholding. This technique is well known and there are several algorithms with different answers depending on the type of image [Ballard 1982, Sezgin 2004].

2.4. Otsu's Method

The Otsu method [Otsu 1978] is considered one of best methods for adaptive thresholding images, but its quality is directly coupled at the cost of complexity. The thresholding operation is considered to be the partitioning of the pixels of an image L gray levels into two classes, C_0 and C_1, which can represent the object and the background, or vise versa, and this partition will be in the gray level t. This is a non-parametric method and unsupervised, which seeks to automatically select the limits of gray levels that best separate or segment the elements of interest in an image of two classes. Thus we have:

$$\begin{array}{l} C_0 = 1, 2, \ldots, k \\ C_1 = k+1, \ldots, L \end{array} \tag{3}$$

A measure of region homogeneity is variance, regions with high homogeneity will have low variance. Otsu's method selects the threshold by maximizing the between-class variance (or, conversely, minimizes the within-class variance) of the two groups of pixels separated by the thresholding operator. It does not depend on modeling the probability density functions, however, it assumes a bimodal distribution of gray level values, if the image approximately fits this constraint, it will do a good job.

The normalized histogram is a probability distribution function.

$$p_x = \frac{n_x}{N} \tag{4}$$

$$\sum_{x=1}^{L} p_x = 1 \qquad p_x \geq 0$$

Then the probabilities of class occurrence and the class mean levels, respectively, are given by:

$$\omega_0 = Pr(C_0) = \sum_{x=1}^{k} p_x = \omega(k) \tag{5}$$

$$\omega_1 = Pr(C_1) = \sum_{x=k+1}^{L} p_x = 1 - \omega(k) \tag{6}$$

The mean gray level value of the background pixels and the object pixels respectively will be:

$$\mu_0 = \sum_{x=1}^{k} xPr(x|C_0) = \sum_{x=1}^{k} xp_x/\omega_0 = \mu(k)/\omega(k) \tag{7}$$

$$\mu_1 = \sum_{x=k+1}^{L} xPr(x|C_1) = \sum_{x=k+1}^{L} xp_x/\omega_1 = \frac{\mu_T - \mu(k)}{1 - \omega(k)} \tag{8}$$

where

$$\omega(k) = \sum_{x=1}^{k} p_x \tag{9}$$

and

$$\mu(k) = \sum_{x=1}^{k} xp_x \tag{10}$$

are the $zero_{th}$ and the first-order cumulative moments of the histogram up to the k_{th} level, respectively,

$$\mu_T = \mu(L) = \sum_{x=1}^{L} xp_x \tag{11}$$

is the total mean level of the original picture. The relation is true for any choice of k:

$$\begin{aligned} &\omega_0\mu_0 + \omega_1\mu_1 = \mu_T \\ &\omega_0 + \omega_1 = 1 \end{aligned} \tag{12}$$

The variance of the background and the object pixels will be:

$$\sigma_0^2 = \sum_{x=1}^{k} (x - \mu_0)^2 Pr(x|C_0) = \sum_{x=1}^{k} (x - \mu_0)^2 p_x/\omega_0 \tag{13}$$

$$\sigma_1^2 = \sum_{x=k+1}^{L} (x - \mu_1)^2 Pr(x|C_1) = \sum_{x=k+1}^{L} (x - \mu_1)^2 p_x/\omega_1 \tag{14}$$

The variance of the whole image is:

$$\sigma_T^2 = \sum_{x=1}^{L} (x - \mu_T)^2 p_x \tag{15}$$

It can be shown that the variance can be written as follows:

$$\begin{aligned} \sigma_T^2 &= \omega_0\sigma_0^2 + \omega_1\sigma_1^2 + \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 \\ &= \sigma_W^2 + \sigma_B^2 \end{aligned} \tag{16}$$

where σ_W^2 is defined to be the within-class variance and σ_B^2 is defined to be the between-class variance.

Since the total variance σ does not depend on T, the T minimizing σ_W^2 will be the T maximizing σ_B^2. Let's consider maximizing σ_B^2, we can rewrite σ_B^2 as follows:

$$\sigma_B^2(k) = \frac{[\mu_T\omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \tag{17}$$

Therefore, the optimal threshold $k*$ is:

$$\sigma_B^2(k*) = max_{1 \leq k < L}\sigma_B^2(k) \tag{18}$$

2.5. Algorithm

The Otsu's method is used to generate a black and white image by calculating the threshold automatically. He believes that the input image have two classes of pixels, the object and the background, seeking for the optimum threshold to segment the image with better quality. The Otsu algorithm follows these steps:

1. Get the frequency of each intensity $1 \leq x < L$ (histogram).

2. Divide each frequency by the total number of pixels , obtaining the probability (histogram normalization).

3. Get the partial sums accumulated ω and μ of each intensity through the odds.

4. For each intensity k , calculate $\sigma_B^2(k)$. If this is the highest value, update the optimal threshold $k*$ with k.

5. For each pixel, if the pixel intensity is less than or equal to $k*$, it is considered object; otherwise, background.

3. Experiments Verifications

The following text aims to present the results achieved in implementing the Otsu's method in targeting RGB images and also images of text. The algorithm assumes that the image to be threshold contains two classes of pixels (object and background) then calculates the optimum threshold separating those two classes so that their combined spread (within-class variance) is minimal.

In this section some experiments were performed to verify the quality of the results obtained from the implementation of the Otsu's method for some images. In Figure 1(a) we quantified the original image with 256 gray levels. After segmenting the image using Otsu's method, we obtain the optimum threshold T value to 0.3412 in $0 - 1$ scale, as shown in Figure 1(b).

In Figure 2(a) we quantified the original image. After segmenting the image using Otsu's method, we obtain the optimal threshold T value to 0.4275 in $0 - 1$ scale, as shown in Figure 2(b).

In Figure 3(a) we quantified the original image. After segmenting the image using Otsu's method, we obtain the optimum threshold value the 0.5608 in $0 - 1$ scale, as shown in Figure 3(b).

In Figure 4(a) we quantified the original image. After segmenting the image using Otsu's method, we obtain the optimum threshold value the 0.4902 in $0 - 1$ scale, as shown in Figure 4(b).

(a) (b)

Figure 1. (a) Original image. (b) Image threshoded with Otsu's method.

(a) (b)

Figure 2. (a) Original image. (b) Image threshoded with Otsu's method.

In Figure 5(a) we quantified the original image. After segmenting the image using Otsu's method, we obtain the optimum threshold value the 0.5765 in $0-1$ scale, as shown in Figure 5(b).

4. Conclusions

The use of computational tools to aid the image segmentation process is increasingly being discussed and studied by the scientific community. The objective of this work was to implement the Otsu's algorithm that aims thresholding an image to a black and white image. Only after the thresholding of the images is that they should be targeted to later find a optimum threshold.

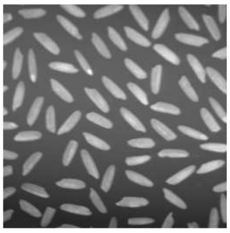
In this paper, we shown a method for automatically selecting an optimal threshold from a gray level histogram. The method was effective in improving job the visual quality of the images. Otsu's method is characterized by its nonparametric and unsupervised nature of threshold selection and may be recommended as the most simple and standard one for automatic threshold selection that can be applied to various practical problems.

One advantage of this method is that it is not restricted to the type of image histogram. That is, histograms can be applied to unimodal, bimodal or multimodal, but has

(a)

(b)

Figure 3. (a) Original image. (b) Image threshoded with Otsu's method.



(a)

(b)

Figure 4. (a) Original image. (b) Image threshoded with Otsu's method.

better performance in images with higher intensity variance [Otsu 1978].

Acknowledgments

The authors would like to thank the Coordination of Higher Education Personnel Training (Capes), Brazilian National Council for Scientific and Technological Development (CNPq) and Research Support Foundation of Goiás State (FAPEG) for the financial support.

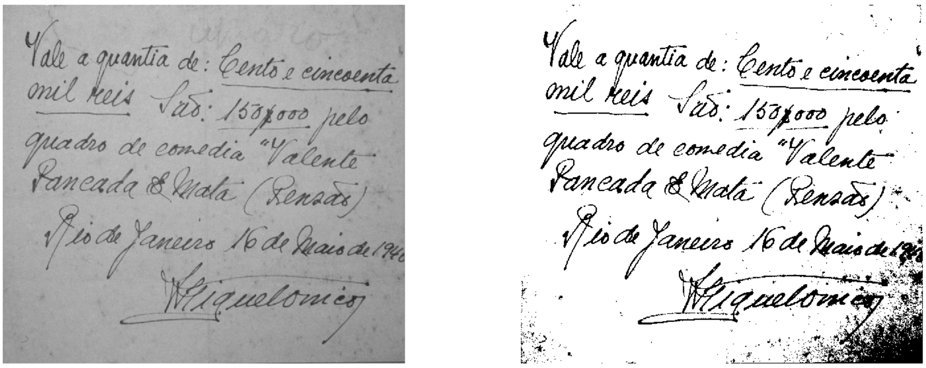
References

Acharya, T. and Ray, A. K. (2005). *Image Processing - Principles an applications*. John Willey & Sons.

Ballard, D. H., B. C. M. (1982). *Computer Vision*. Prentice-Hall.

Gonzalez, R. and Woods, R. (2002). *Digital Image Processing*. Prentice-Hall.

Jahne, B. and Haubecker, H. (2000). *Computer Vision and Applications: A Guide for Students and Practitioners*. Prentice-Hall.



(a) (b)

Figure 5. (a) Original image. (b) Image threshoded with Otsu's method.

kapur, J.N.; Sahoo, P. K. and Wong, A. K. C. (1985). *Computer Vision, Graphics and Image Processing*. Academia Press.

Otsu, N. (1978). *A threshold selection method from gray-level histogram.* IEEE Transactions on Systems, Man, and Cybernetics.

Parker, J. R. (2011). *Algorithms for Image Processing and Computer Vision.* Wiley Publishing, 2nd edition.

Petrou, M. and Bosdogianni, P. (1999). *Image Processing: The Fundamentals.* John Wiley & Sons.

Sezgin, M., S. B. (2004). *Survey over image thresholding techniques and quantitative performance evaluation*, volume 13. Journal of Electronic Imaging.