Article **ECORFAN Journal**
December 2015 Vol.1 No.1 32-35

Thresholding images

GARCÍA-MARTÍINEZ, Manuel Darío *†

Universidad de Guanajuato. División de Ingenierías Campus Irapuato-Salamanca.

Received July 29, 2015; Accepted October 29, 2015

Abstract

In this practice the implementation of a global thresholding algorithm and two local thresholding algorithms shown. In both groups of algorithms threshold values are dynamic, automatically calculated by the program in each iteration.

Thresholding, images.

Citation: GARCÍA-MARTÍINEZ, Manuel Darío. Thresholding images. ECORFAN Journal-Democratic Republic of Congo 2015, 1-1: 32-35

* Correspondence to Author (email: md.garciamartinez@ugto.mx)
† Researcher contributing first author.

Article **ECORFAN Journal**
December 2015 Vol.1 No.1 32-35

Introduction

Necessary to perform thresholding on images given functions were implemented. The implementation is in C, using the Open CV library, which provides a lot of useful functions for image manipulation. The first algorithm is global whose threshold is calculated using the cumulative histogram. The second algorithm is the local umbralizador Niblack [1], which uses a Local thresholding using square windows, and measures statistical used to calculate the threshold of each region. And finally, Bernsen algorithm [2] which was implemented as of Niblack uses the windows but this simply uses the maximum and minimum values of each window to determine a threshold in contrast.

Methodology

Three processing algorithms is performed on a converted image to grayscale, to this Equation 1 is used.

$$Y = 0{,}3\mathrm{R} + 0{,}59\mathrm{G} + 0{,}11\mathrm{B} \quad (1)$$

II-A. Global thresholding

The first algorithm is described below:

Get the histogram of the gray scale image.

Obtain the cumulative histogram.

Find the gray level value for which the cumulative histogram is 0.5 and assign the threshold *th*.

The pixels with higher intensities to paint *th* white and pixels with lower intensities are painted black.

Global multilevel thresholding

The global optimization algorithm was implemented for a multilevel thresholding, ie, more than a threshold. The procedure is quite similar to the previous, but now thi N values are calculated. The calculation of the threshold values given by the Equation n2, where i runs from 1 to N - 1.

$$thi = i/N \quad (2)$$

Local thresholding Niblack

For the second algorithm must travel the image pixel by pixel with a square window (see Figure 1). In the case of the algorithm implemented it has a resizable window. The threshold value th is calculated with Equation 3 where μ is the mean of the window, a parameter k is usually equal to 0.2 for clear images and - 0.2 for dark images, and σ is the standard deviation of the window. If the value of the center pixel is greater than TH, the pixel is painted white, otherwise it is painted black.

$$th = \mu + k\sigma \quad (3)$$

0	255	200
0	10	100
18	35	20

Figure 1 3x3 window. The center pixel is being evaluated, but using as a reference all the pixels around it.

There is a slight variation to the algorithm by adding another parameter c which suggested value is 1.3, so the formula is as in Equation 4.

$$th = \mu + k\sigma - c \quad (4)$$

ISSN-On line: 2414-4924

December 2015 Vol.1 No.1 32-35

Local Umbralizador of Bernsen

Bernstein algorithm also occupies a window, as for the previous case was implemented resizable. For this algorithm you have two values, a local contrast (see Equation 5) and a contrast threshold. The contrast threshold has a value of 15 suggested by the author, but can receive any nonzero value. Besides, it is necessary to calculate an average gray value (Equation 6).

Local Contrast = *max* - *min* (5)

half gray = (*max* - *min*) / 2 (6)

1) Algorithm:

If the Local contrast <Contrast threshold.

- If the average gray ≥ 128, the pixel is white.
- If the above condition is not met the pixel is black.

If the local contrast contrast threshold ≥

- If the average value of the center pixel gray ≥, the pixel is white.
- If the above condition is not met the pixel is black.

Results

The test images were obtained from the database provided in reference [3].

Global thresholding

The global thresholding results are shown in Table I. The tests for different levels of shrinkage are shown.

Thresholding Niblack

Different tests for different window sizes on Niblack thresholding algorithm is performed (see Table II). Tests were performed by changing the input para'metros but are not reported due to the large number of possible combinations, but the observations are discussed in the concluding section.

Thresholding Bernsen

Different tests for different window sizes on Bernsen thresholding algorithm (see Table III) were performed. Tests were performed changing the contrast threshold. The results reported to us, but the observations are discussed in the concluding section.

Table IV a visual comparison of the performance of the three algorithms analyzed in this report is shown.

Conclusions

Visually, to test cases the results are better with global thresholding. Local thresholding with small windows have a performance that even resembles edge detectors, but noisy in areas of almost constant color, this due to slight changes in color between pixels although the human eye color It is uniform. Using larger windows performance improves. One of the applications that would be interesting for thresholding pictures are very dark areas where objects are almost imperceptible to the human eye and yet no useful information in these regions. In local thresholding, the same object can be classified differently if your background is not constant, that is, if part of the is a dark area on either side of the on a clear area. This is a distinct disadvantage against global thresholding.

ISSN-On line: 2414-4924

Another drawback of the analyzed local thresholding is that they require a considerable number of parameters and that although the authors recommend parameters fixed, sometimes the best results using different parameters than those above. The fact that the images are taken for purposes of use in robotics and that the quality is not too high causes very noisy which can affect an algorithm with the characteristics of the local thresholding images. In terms of computational complexity of local thresholding require more operations, since each pixel performed in the case of Niblack 18 summations and 9 multiplications, in addition to the standard deviation calculation. In the case of Bernsen searches for minimum and maximum from 9 data. As for global thresholding only an amount per pixel for the histogram and a second sum and division to the cumulative histogram is performed. In the case of global thresholding, an image with a color irregular distribution could alter its functioning, so a good measure would apply a stretching algorithm in a pre-processing.

Referencias

[1] W. Niblack, An Introduction to Digital Image Processing. Prentice-Hall,
1986.

[2] J. Bersen, “Dynamic Thresholding of Grey-Level Images,” in Proc. of the 8th Int. Conf. on Pattern Recognition, 1986.

[3] http://www.cas.kth.se/IDOL/#Download.
[4] http://opencv.org/.

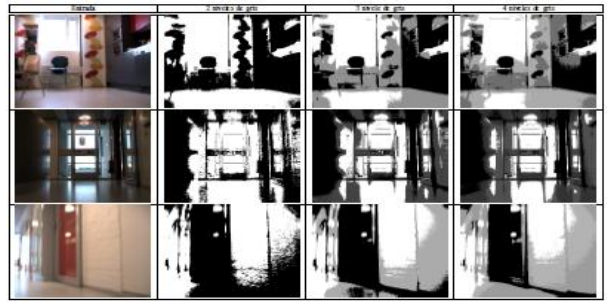

Table I Results from the global thresholding.

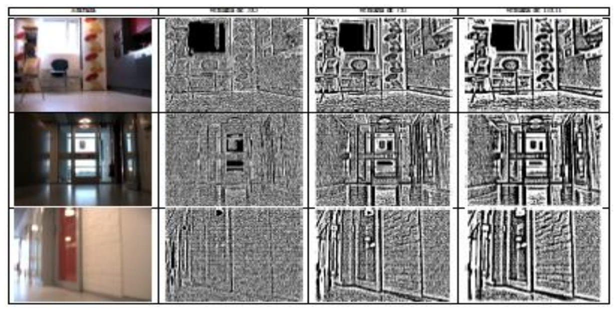

Table II Results obtained with thresholding algorithm Niblack.

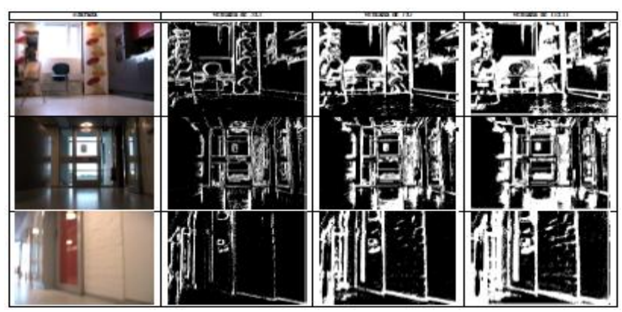

Table III Results obtained with thresholding algorithm Bernsen.

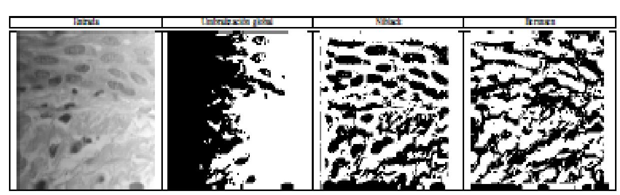

Table IV Comparison of the performance of different algorithms against the same image. In this case it is observed that the overall umbralizador not produce good results since the left side of the image is lightly shaded, which affects their performance. The algorithm with better results for this test is the Niblack.