

VOTING BASED IMAGE BINARIZATION

Costin-Anton Boiangiu¹

Mihai Simion²

Vlad Lionte³

Zaharescu Mihai⁴

ABSTRACT

In the literature there are a wide variety of algorithms for image binarization, the difference between them being the method that identifies the pixel threshold value. They can be split into two classes: algorithms that use a single threshold for the entire image (and tend to identify a few large objects) and algorithms that do the processing in localities (and tend to identify many small items). This paper aims at defining a method for image thresholding based on the results of several different algorithms. Knowing in advance the behavior of specific algorithms on different kinds of images, we can vote between their results. The end result of the proposed method is a mosaic of more binarization algorithms, hopefully better than any individual image.

Keywords: binarization, voting, black and white conversion, image interpretation

1. INTRODUCTION

The thresholding operation splits an image into areas of foreground and background image pixels. The usefulness of doing such a processing can be found in any image interpretation domain: one has to first separate the desired data from the background in order to analyze it. Some examples include: cell or particle objects counting in microscopy images [1], digitization of paper documents [2] and any kind of image masking.

This paper presents a method to combine multiple thresholding algorithms, to achieve an optimal result. A number of global and local algorithms are applied to the input image; the final binary image will have regions from each result, those where each individual algorithm performs best.

2. RELATED WORK

Greyscale image thresholding techniques can be grouped into two main categories: global, those that calculate statistics on the entire input image and calculate the threshold for each individual pixel and local, those that consider more than one pixel in the process of deciding if a region should be foreground or background. Global methods are very fast and give good results under ideal conditions (low noise, high contrast, uniform lighting). However, most of the interesting processing is done on images that are hardly ideal, and global methods fail (for example, they select an entire shady region as foreground).

¹ Associate Professor PhD Eng., costin.boiangiu@cs.pub.ro, "Politehnica" University of Bucharest, 060042 Bucharest, Romania

² Engineer, mihai.simion@cti.pub.ro, "Politehnica" University of Bucharest, 060042 Bucharest, Romania

³ Engineer, vlad.lionte@cti.pub.ro, "Politehnica" University of Bucharest, 060042 Bucharest, Romania

⁴ Engineer, mihai.zaharescu@cs.pub.ro, "Politehnica" University of Bucharest, 060042 Bucharest, Romania

Local thresholding methods (e.g. Sauvola) overcome this obstacle by computing the individual thresholds for each pixel based on the information from the pixels in the neighborhood. These methods achieve good results on those non-ideal images that are illuminated unevenly, but they cope worse with noise than global ones do, thus multiple objects can be falsely identified.

One paper that tries to combine the two approaches into one iteratively enlarges the local window until the standard deviation inside the window stops rising, meaning that no new information is added. The binarization threshold is then weighted with a fixed global binarization threshold. [4]

Voting based algorithms, which select regions from different algorithms and merge the obtained results into (presumably) superior ones, have been used in other domains also, like voting based image segmentation [5] and even voting based layout analysis [6]. These methods are very similar to what we are trying to achieve in this paper, but they are not knowledge-based (treat each algorithm significantly different) since they rely mostly on perform various voting schemes: Unanimous, Majority, etc.

In [3] is proposed a method based on extracting a set of features from the original image, and combining multiple algorithms on those features.

3. PROPOSED METHOD

Under these conditions, an optimal solution for image binarization is the selective combination of multiple algorithms, taking advantage of the strengths and weaknesses that every algorithm exposes. In the first phase we obtain multiple images using different binary algorithms. Then, by voting, considering the type of noise found in the images obtained, we replace the affected regions with the corresponding regions from other results where the algorithms were known to cope well in those situations. The results obtained are improved by filtering very small objects which may be considered noise.

The voting-based thresholding algorithm proposed in this paper relies on combining results obtained by sending the original image to a number of known algorithms. We chose the implementation of five algorithms (2 Global ones and 3 Local ones):

- Global Binarization Approaches:
 - Bimodal histogram thresholding
 - Otsu
- Local Binarization Approaches:
 - Niblack
 - Sauvola
 - Wolf

The following section briefly describes each method.

Bimodal histogram thresholding

This method generally works for greyscale images containing mostly objects of interest having similar gray levels (text documents); the histogram will contain two local maximum

points (modes), corresponding to either object or background. The choice of the threshold value in this case is carried out by identifying the point of local minimum between the two peaks found:

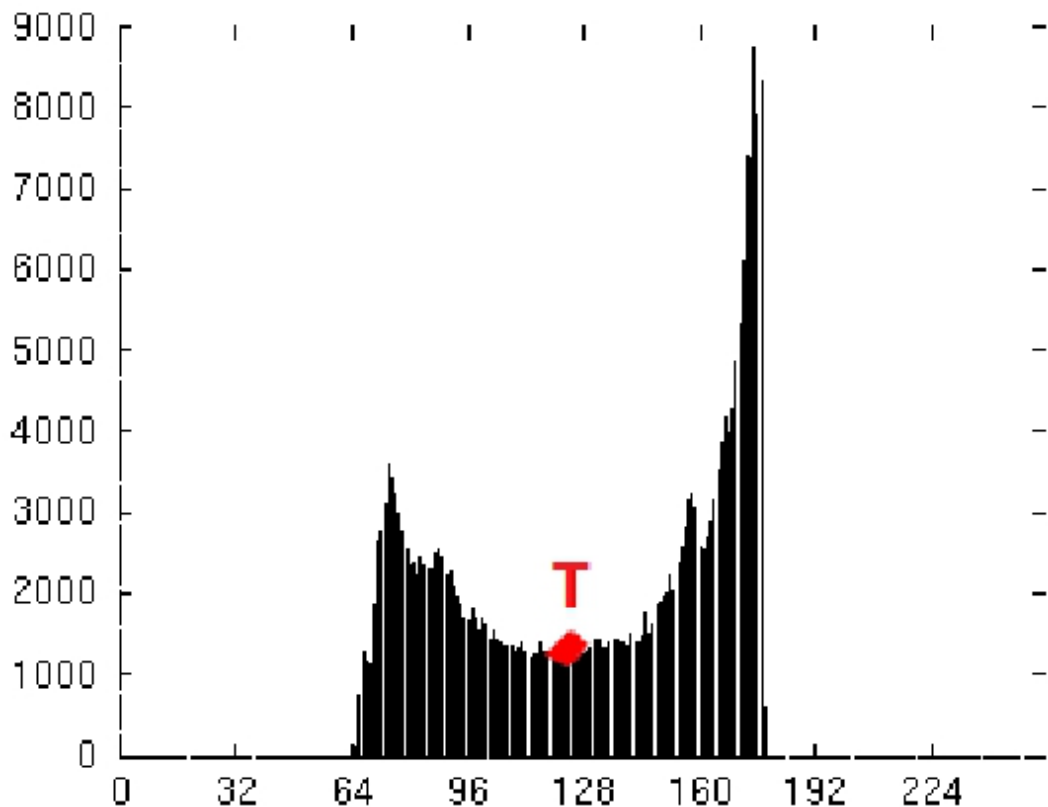


Figure 1. Selecting a threshold from a bimodal histogram (Picture taken from University of Iowa's Digital Image Processing Course [8])

As it is stated in [8] “histogram bimodality itself does not guarantee correct threshold segmentation”, thus the obvious need for validating and/or adjusting the threshold computed by the algorithm.

Otsu

Otsu method iterative searches in the pixel values (0-255) a threshold that minimizes intra-class variations (desired pixel classification into two classes: object and background that are as similar as possible, or in other words, the classes are as far apart as possible). We want to find the minimum value of the weighted sum of the standard deviation for the two classes:

$$\sigma_{\omega}^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

Niblack, Sauvola, Wolf

These methods of choice for local threshold value are based on calculating the average and mean square deviation in neighborhood windows of each pixel [7]. The results of these three binarization algorithms can be further controlled by a positive k parameter, with values between 0.2 and 0.5.

We are given:

- m - the average pixel values in the neighborhood window W
- s - mean squared pixel values deviation in the neighborhood window W
- R - maximum standard deviation on image (128 on 8BPP greyscale images)
- k - real parameter $\in [0.2, 0.5]$
- max_s - maximum standard deviation for neighborhood window
- min_I - minimum value of a pixel in the image

Local threshold values are calculated according to the following formula:

- Niblack: $t = m + ks$
- Sauvola: $t = m \left[1 + k \left(\frac{s}{R} - 1 \right) \right]$
- Wolf: $t = m + k \frac{m - \min_I}{\max_s - 1}$

Further, in order to obtain an optimal solution we must deduce how the algorithm behaves: the images obtained in the previous step are analyzed for the number of objects identified in each X*Y window. If this number is less than a specified minimum accepted (e.g. 10), it follows that for that window the algorithm that will provide better results falls in the local algorithms category. Otherwise (too many objects in the window), it is considered that the global algorithm shall be used in that window. These calculations are performed on each X*Y window pixels in the image.

In parallel, we keep in memory a matrix of thresholds. If in a particular window the algorithm identifies that the needed algorithm should be local, the respective values in the matrix will be incremented by one; otherwise, they will be decremented by one.

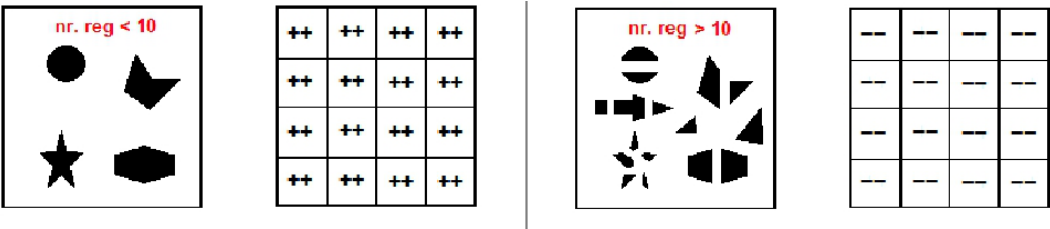


Figure 2. The values from the threshold matrix relative to the binarization result

Thus, after going through the entire image, the elements $P(i, j)$ from the matrix of thresholds will have a positive value if an algorithm needs to be local in more than half the pixels in the window. Similarly, a negative value is obtained for global algorithms.

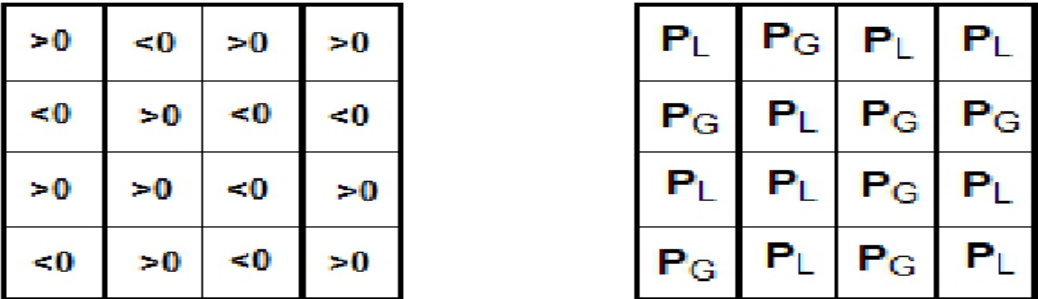


Figure 3. Regions from the image computed with different algorithms

Next, we will generate a new image with pixels selected either from the result of a local algorithm or from the one obtained by a global algorithm, depending on the value from the corresponding thresholds matrix.

4. PARTIAL RESULTS



Figure 4. Left to right and top to bottom: Original, Bimodal, Otsu, Niblack, Sauvola, Wolf, Normal Voting, Proposed

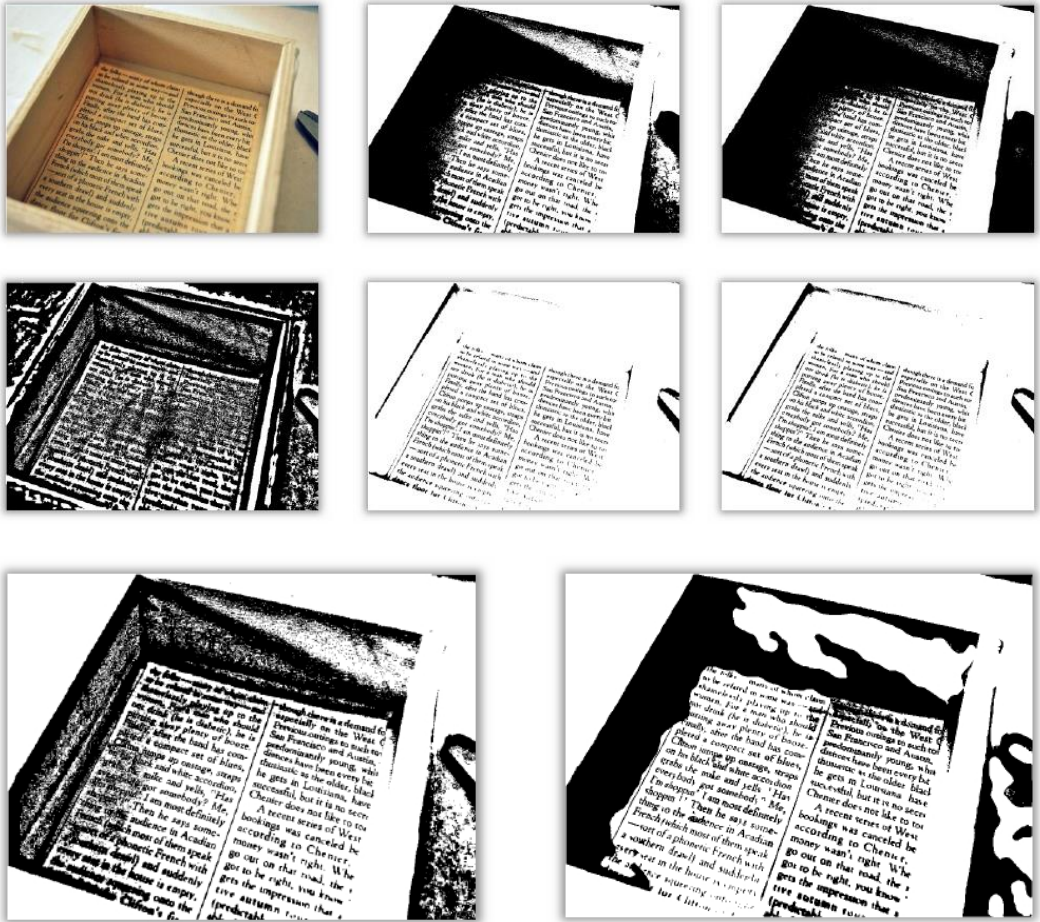


Figure 5. Left to right and top to bottom: Original, Bimodal, Otsu, Niblack, Sauvola, Wolf, Normal Voting, Proposed

During this research, we implemented an efficient method for separating objects of interest (ex: text characters) from background in a uniformly illuminated and noisy input image. The thresholding technique described in this article takes the results of other thresholding algorithms (local / global) and combines parts of their results, depending on the degree of trust each algorithm has under certain conditions. Finally a post-processing step is required: the elimination of noise occurred during processing of the resulted binary image by deleting small objects (<10 pixels). The obtained results prove a clearer delimitation between objects and background than any other algorithm used, taken independently.

5. IMPROVEMENTS

We also included a more adaptive local algorithm, one that has variable locality sizes. [4] A local algorithm with dynamic windows: the window calculation is based on the variance of pixel intensities. The window increases as long as the variance increases.

The results of the algorithms are combined to obtain the final binarization result, using algorithm specific heuristics:

Heuristic 1:

Each algorithm defines a confidence factor: confidence increases with the difference between the pixel and the threshold.

Each algorithm computes for each pixel in the image a confidence factor. These values are normalized in the range $[0, 1]$

Heuristic 2:

Global algorithms tend to generate large uniform regions. As an example, on a non-uniformly lit sheet of paper, the lit region will all be white and the shadowed region will all be black. The useful values are in between the two regions.

For global algorithms if all neighboring pixels are binarized to the same value (a large uniform region) then the binarization is uncertain in that pixel.

Heuristic 3:

Local algorithms tend to increase noise. If a locality falls on a uniform region of the image, the algorithm will still try to create the two regions of background and foreground. The result will thus be a random region of white and black (the binarization of the noise in the respective region).

For local algorithms, if most of the binarization values are different in neighboring pixels, the binarization is uncertain in that pixel.

Heuristic 4:

There are algorithms known to offer better results than others.

Each algorithm has a weight associated to its result.

6. RESULTS

Red pixels mark the areas that are uncertain, considering the heuristics mentioned in the previous chapter.



Figure 6. Top to bottom: Global Algorithm, 4x4 Local Algorithm, 10x10 Local Algorithm, Dynamic Window Local Algorithm, Proposed method

7. FUTURE WORK

The results from this research can be used in all the existing applications that need a binarized image as input. The upgrading from single binarization algorithm to the voting one is straightforward, and, depending on the needs, more or less sophisticated algorithms can be used. The results are better for multiple different simple algorithms than from a single time-consuming advanced one.

The approach with heuristics can also be applied in a trivial manner to image segmentation.

REFERENCES

- [1] U. Phromsuwan, C. Sirisathitkul, Y. Sirisathitkul, B. Uyyanonvara, P. Muneesawang, "Application of Image Processing to Determine Size Distribution of Magnetic Nanoparticles", *Journal of Magnetism*, Volume 18, Issue 3, 2013, pp.311-316, Publisher: The Korean Magnetism Society
- [2] Costin-Anton Boiangiu, Andrei Iulian Dvornic, Dan Cristian Cananau, „Binarization for Digitization Projects Using Hybrid Foreground-Reconstruction”, *Proceedings of the 2009 IEEE 5th International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Editor Ioan Alfred Letia, August 27-29, pp.141-144,
- [3] F. Shafait, D. Keysers, T. M. Breuel, "Efficient Implementation of Local Adaptive Thresholding Techniques Using Integral Images", *Image Understanding and Pattern Recognition (IUPR) Research Group*, Kaiserslautern, Germany
- [4] Costin-Anton Boiangiu, Alexandra Olteanu, Alexandru Victor Stefanescu, Daniel Rosner, Alexandru Ionut Egner (2010). „Local Thresholding Image Binarization using Variable-Window Standard Deviation Response” (2010), *Annals of DAAAM for 2010, Proceedings of the 21st International DAAAM Symposium*, 20-23 October 2010, Zadar, Croatia, pp. 133-134, Editor B. Katalinic, Published by DAAAM International, Vienna, Austria 2010
- [5] Costin-Anton Boiangiu, Radu Ioanitescu, "Voting-Based Image Segmentation", *The Proceedings of Journal ISOM Vol. 7 No. 2 / December 2013 (Journal of Information Systems, Operations Management – JISOM)*, pp. 211-220
- [6] Costin-Anton Boiangiu, Paul Boglis, Georgiana Simion, Radu Ioanitescu, "Voting-Based Layout Analysis", *The Proceedings of Journal ISOM Vol. 8 No. 1 / June 2014 (Journal of Information Systems, Operations Management – JISOM)*, pp. 39-47
- [7] J. Sauvola, M. Pietikainen, "Adaptive document image binarization", *Machine Vision and Media Processing Group*, Infotech Oulu, University of Oulu, Finland
- [8] Reinhard R. Beichel, Ph.D. Medical Computer Vision & Graphics Lab - The University of Iowa, Course 55:148 Digital Image Processing, Chapter 5, Part I, Segmentation: Thresholding, <http://user.engineering.uiowa.edu/~dip/lecture/Segmentation1.html>, Accessed on 15 November 2014.