

# Adaptive Thresholding Technique for Document Image Analysis

Naveed Bin Rais, M. Shehzad Hanif and Imtiaz A. Taj

Center for Advanced Studies in Engineering (CASE), Islamabad, Pakistan

naveedbinrais@yahoo.com, shehzadhanif\_178@hotmail.com, imtiaz.taj@gmail.com

## Abstract

*Binarization of document images is mostly used as a preprocessing step in document image analysis. Proper binarization is very important so as to save all or maximum subcomponents such as text, background and picture. The work of Niblack [2] is significant and has most acceptable results out of many techniques of thresholding [6], [7]. In this paper, we have presented a modified version of Niblack's Method of thresholding (image binarization) that takes care of the limitations of Niblack's technique. Our algorithm works very well and is independent of many challenging factors involved in thresholding of document images like ambient illumination, contrast stretch, shading effects. An optimal size of window for local binarization scheme is also proposed based upon experiments and their results. The results show that the method adapts and perform well in each case.*

## 1. Introduction

Image binarization or thresholding is an important tool in image processing and computer vision, to extract the object pixels in an image from the background pixels. Image binarization is central to many applications including document image analysis (printed characters, logos, graphical content, and musical scores are important as objects), map processing (lines, legends and characters need to be extracted), scene processing, quality inspection of materials, cell images, segmentation of various image modalities for non-destructive testing (NDT) applications (ultrasonic images, eddy current images, thermal images, X-ray computed tomography, laser scanning confocal microscopy, extraction of edge field and spatio-temporal segmentation of video images). A number of methods have already been proposed for image binarization but unfortunately, most of them are very much specific for a few applications. Thus, it can be said that a binarization (thresholding) method may

work well for one application but its performance can be unsatisfactory for another application.

Bi-level image is used as a pre-processing unit in several applications. The use of binary images decreases computational load for the overall application. These applications include document analysis, optical character recognition system, scene matching, quality inspection of materials etc. The binarization process computes the threshold value that differentiate object and background pixels. Under varying illumination and noise, the binarization can become a challenging job. A number of factors contribute to complicate the thresholding scheme including ambient illumination, variance of gray levels within the object and the background, inadequate contrast, object shape and size non-commensurate with the scene. A wrong selection of threshold value may misinterpret the background pixel and can classify it as object and vice versa, resulting in overall degradation of system performance. The determination of a threshold itself is application dependent since one threshold may work with one application and may not work with other one. In document analysis, binarization is sensitive to noise, surrounding illumination, gray level distribution, local shading effects, inadequate contrast, the presence of dense non-text components such as photographs, etc. while at the same time, the merges, fractures and other deformations in the character shapes affects the threshold value in OCR system. On the other hand, the simplification is needed that benefits the overall system processing characteristics (computational load, algorithm complexity, real-time requirements etc). So all these challenges and problems makes binarization a difficult task.

There are a number of important performance requirements that need to be considered while binarizing gray level images. These include:

- Loss of features after binarizing input image should be zero or minimum.
- The features (objects) with similar relative gray levels should have same binary values in the processed output image.
- The effect of noise or minor gray level variations should be eliminated.

With reference to the proposed algorithm, following sections describe all the related concepts briefly along with the related work done in the past. The algorithmic flow and system description will also be presented. Section 2 illustrates brief history of image binarization techniques presented in the past. It also explains the Niblack algorithm upon which our algorithm is based. Section 3 provides an overview about different supporting concepts relating to image thresholding, their advantages and disadvantages. Section 4 covers the overall system description, explaining the proposed algorithm in association with algorithmic block diagram. Section 5 provides results of our binarization technique and comparisons with previous methods. Finally, we conclude our paper on the conclusion presented in Section 6.

## 2. Previous Binarization Techniques

Several techniques for determination of threshold have been proposed in the literature. Most common thresholding techniques use statistical approach either on complete image i.e. determining a single threshold for complete image, called global threshold or on a small region window i.e. determining threshold for each window called local threshold.

*Global method is the most conventional approach.* The renowned Otsu's [1] method suggests minimizing the weighted sum of variances of the objects and background pixels to establish an optimum threshold. But it is clear that a fixed value of threshold value cannot give satisfactory results in case of badly illuminated images. Also in document image analysis, logical and semantic content conservation is needed, global threshold cannot conserve this kind of these contents.

Local method is an adaptive one in which a threshold value is determined over a small region. Local method performs better in case of badly illuminated images and document image analysis as threshold computation is dependant on region characteristics.

The well-known Niblack's [2] method uses mean and standard deviation to compute threshold over a small region ( $n \times n$  window). Bernsen [3] uses mean

and standard deviation along with contrast information to compute a threshold for a certain region. Sauvola [4] presents modified Niblack Method, which uses adaptive contribution of standard deviation in determining local threshold. Parker [5] uses a slightly different approach to solve bad illumination problem and computes local threshold value by classifying object and background pixels and then using region growing to produce a binarized image.

### 2.1. Ni-Black Method

Ni-Black is a local thresholding algorithm that adapts the threshold according to the local mean and the local standard deviation over a specific window size around each pixel location. The local threshold at any pixel  $(i, j)$  is calculated as:

$$T(i, j) = m(i, j) + k \cdot \sigma(i, j) \quad (1)$$

Where  $m(i, j)$  and  $\sigma(i, j)$  are the local sample mean and variance, respectively. The size of the local region (window) is dependent upon the application. The value of the weight 'k' is used to control and adjust the effect of standard deviation due to objects features. Ni-Black algorithm suggests the value of 'k' to be -0.2.

The document images binarized using the Ni-Black algorithm provides the most satisfactory results, out of all the other algorithms available at the time of writing. However, Ni-Black's algorithm suffers from the basic problem of local thresholding, i.e. providing unnecessary details in the binarized images that may not be required in the processing. Niblack fails to adapt large variation in illumination, especially in the document images. The local region analysis using Niblack does not provide any kind of information about the global attributes of the image that may be helpful in the binarization process of badly illuminated images. So, the gray level variations in the document images make it impossible to adapt threshold as will be shown in the results. Another problem it faces is the optimum selection of the weight 'k'. Ni-Black algorithm uses fix value of this weight. The fix given value of 'k' may work for document images but for gray-level images with a lot of variations of gray values, the value of the weight should not be fixed but to change from images to images depending upon their gray-level distributions, and therefore, the value of 'k' should be calculated at run-time.

### 3. Binarization Process

This section describes the details of a binarization scheme and the main components involved in thresholding the gray level images. The main components of binarization include statistical analysis of images, determination of a threshold value based upon the statistical results and applying the threshold value to gray-level images. Statistical analysis of gray-level images may include determination of mean, variance, standard deviation, contrast stretch, histogram etc. or it can be a combination of any of these. Determination of a threshold value is very much important and perhaps the most sensitive part of any image binarization scheme because a wrong value of threshold may result in losing some image information (an object can be considered as part of background and vice versa). Moreover, the value of threshold should be sensitive with the overall contrast stretch of the corresponding image. The threshold value is applied to the image to represent it in 1-bit after the proper determination of a threshold limit.

The simplest method to convert a gray scale image into a binary image is to compute mean of the image and set the value of mean as the threshold for binarization. But this approach has many shortcomings, as it may not take care about the features and objects in the image properly. This is due to the fact that the mean of an image may be disturbed drastically by the addition of noise pixel(s) or very few numbers of pixels having the intensity close to any of the boundaries of gray scale.

A solution to this limitation is to add the impact of standard deviation to some extent in selecting the threshold value [2]. But the influence of the value of standard deviation should neither be too small (as it does not make too much change to the value of mean and the problem will remain the same) nor too large (as the change of standard deviation will have too much affect on the threshold value and binarization will ultimately be affected). This paper basically focuses this aspect of thresholding - finding an optimal run-time value of the factor that controls the standard deviation.

Contrast stretch and histogram can also be helpful to find out threshold for binarization, as the variation in the gray levels of the image constitutes the object boundaries and histogram of the image elaborates the number of pixels of a certain gray level in the image.

### 4. Algorithm Description

This section describes the proposed algorithm in detail along with its advantages and improvements especially over Niblack's algorithm. It also illustrates the binarization algorithm block diagram.

#### 4.1. Proposed Binarization Scheme

Our algorithm is based on the Niblack's image thresholding method for image binarization. It offers great improvement over original Niblack's method. It does not entirely depend upon image's local statistical characteristics but also considers the global statistics. Our algorithm calculates "k" at runtime for each pixel and thresholding is done using Niblack method. In contrast, Niblack fixes this weight value to -0.2. Local mean (mean calculated over a small window) is the average illumination value in the small region, while global mean is overall illumination of the image. So, the normalized difference  $m_d(i, j)$  of global and local mean provides information about the illumination difference for each pixel window with respect to global illumination.

$$m_d(i, j) = \frac{(m_g(i, j) - m_l(i, j))}{\max(m_g(i, j), m_l(i, j))} \quad (2)$$

Where  $m_g(i, j)$  is the global mean of the entire image and  $m_l(i, j)$  is the local mean computed on each window respectively. Obviously, equation 2 provides a reasonable good 'k' factor for thresholding document images. But it fails to adapt changes in images with different contrast stretch i.e. same image with different contrast stretch values will result in different threshold values if equation 2 is used. The use of standard deviation with the above equation can solve this problem.

We have made use of the interrelation of global and local characteristics and set the threshold based on the relative change of local and global mean and standard deviation values. The impact of value of standard deviation remains almost the same on different kinds of images having different local illumination and contrast stretch histogram. The formula to compute the value of the weight 'k' is given as:

$$k = -0.3 * \frac{(m_g(i, j) * \sigma_g(i, j) - m_l(i, j) * \sigma_l(i, j))}{\max((m_g(i, j) * \sigma_g(i, j)), m_l(i, j) * \sigma_l(i, j))}$$

Where  $\sigma_g(i, j)$  is the global standard deviation of the entire image and  $\sigma_l(i, j)$  is the local standard deviation computed on each window respectively. The formula for 'k' is multiplied with -0.3 so as to keep the value of 'k' in the range of 0.3 and -0.3. This is done to minimize the effect of standard deviation in the Niblack's formula for computing threshold value. The algorithm process is shown in figure1.

The main algorithmic flow to compute the threshold value for binarization is as under:

- Find out the value of mean of the image (global mean - this is computed only once per image).
- Determine the standard deviation value of the image (global standard deviation - computed only once per image).
- Set the size of local region in terms of pixel area (generally known as window size).
- Select a local region of appropriate window size across every single pixel in the image one by one.
- For every local region, compute the mean of that region (local mean) and standard deviation of the corresponding region (local standard deviation).
- Determine the value of weight 'k' for each local window using (2). This is done using local mean and standard deviation and global mean and standard deviation.
- Apply (1) to determine the threshold value using local mean and standard deviation for every local region.
- Apply the threshold value to the single pixel across which the local window is selected.
- Increment window by one pixel and go to step 4 to compute threshold for next pixel.
- Finished when the threshold is computed and applied to all the image area. (The result after this step will be an image having two levels only - a binary image).

## 5. Results

A number of images have been binarized to evaluate the performance of our proposed algorithm. We have shown a few of them in this paper. We have also binarized the same images using Niblack's

algorithm in order to provide comparison. The badly illuminated document images taken for experimentation are scanned at a 300dpi. Each experiment is performed both of Niblack's algorithm and our algorithm by selecting different size of local regions around the pixels each time (different window sizes). Figure 2 shows our input test images. These images are scaled down after binarization so as to adjust them to be fit here. The results of Image 1 & 2 using Niblack's algorithm for different window sizes (31x31, 75x75 &

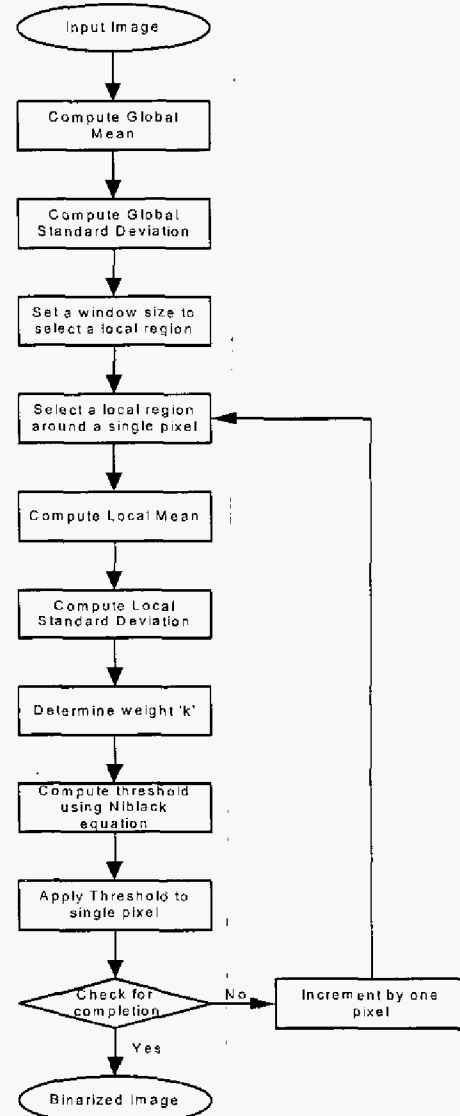


Figure 1 Binarization Algorithm Block Diagram

101x101) are shown in Figure 3 and Figure 4 respectively. Figure 5 and Figure 6 illustrate the results of binarizing Images 1 & 2 using our proposed algorithm and the improvement can be seen at the top left corner of the resulting images. The results of Image 2 using both algorithms are shown in Figure 4 (a)-(c) and 6 (a)-(c). One thing can be observed at the bottom right corner of resulting images in Figure 4 (a)-(c) and 6 (a)-(c). A white circle is common to all of these images. This is due to the fact that this portion is extremely suffered from bad illumination and has completely lost information even in the input image (Image 2). Our algorithm has recovered as much area as can be possible while not disturbing any surrounding area. This is again an improvement from the Niblack's results. By looking at results of different window sizes, we have observed that by increasing window sizes, we get the unnecessary black pixels eliminated from the image background in a better way while filling out characters and vice versa. In this way, we have found window size of 75x75 to be appropriate for this kind of images as can be observed from the resulting images as shown.

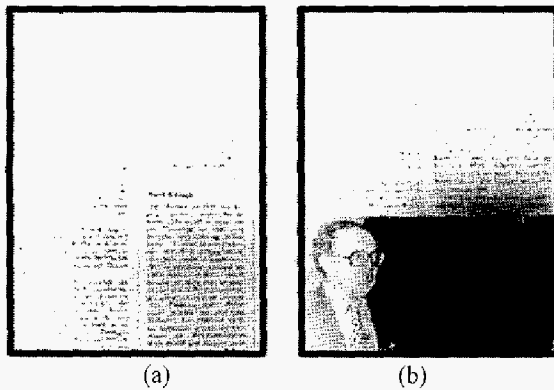


Figure 2 Document Test Images (a) Image 1 (b) Image 2

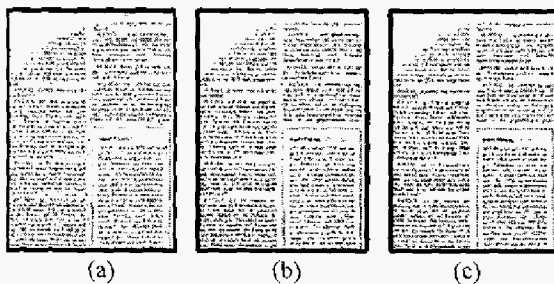


Figure 3 Niblack Algorithm's results for Image 1

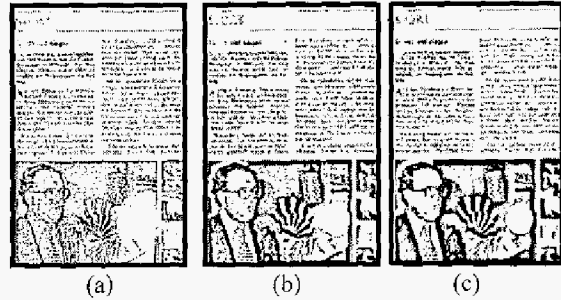


Figure 4 Niblack Algorithm's results for Image 2

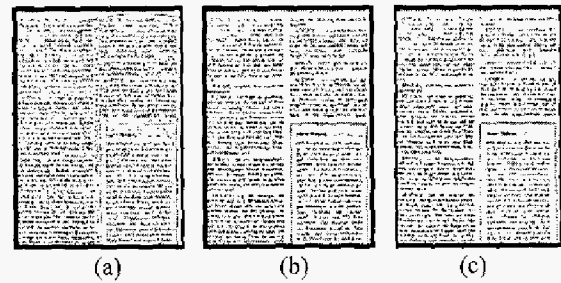


Figure 5 Proposed Algorithm's results for Image 1

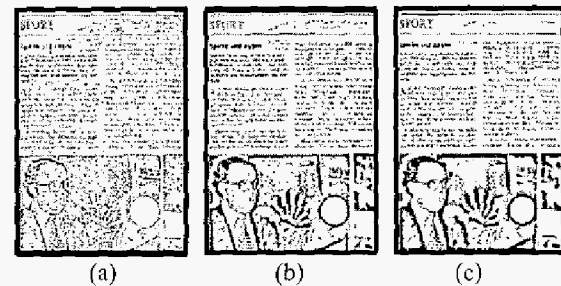


Figure 6 Proposed Algorithm's results for Image 2

## 6. Conclusion

In this paper, we have proposed an algorithm for binarization that works very well for document image analysis. Our technique is aimed at generic document type with cases of severe degradations. The results showed that it has improved document image binarization significantly over well-known Niblack's algorithm. We have also shown that a local window size of 75x75 is the right option for the thresholding scheme. Adaptive use of image's global and local statistics has made our algorithm very strong and robust. The proposed algorithm has been tested over

large database of document images and is found robust against bad illumination and other degradations occurred in document images. Though targeted here for document image analysis only, it will be a good candidate for other kinds of applications as well like scene processing and image segmentation.

## 7. References

- [1] N. Otsu, "A threshold selection method from grey level histograms." *IEEE Transactions on Systems, Man, and Cybernetics*, 9:62--66, 1979.
- [2] W. Niblack, "An Introduction to Digital Image Processing", pages 115-116, Englewood Cliffs, N.J.: Prentice Hall, 1986.
- [3] J. Bernsen, "Dynamic thresholding of grey-level images" *Proceedings of the 8th ICPR, Paris, Oct 1986*, 1251-1255.
- [4] J. Sauvola, T. Seppanen, H. Haapakoski and M. Pietikainen, "Adaptive Document Binarization", *International Conference on Document Analysis and Recognition*, volume 1, pages 147-152, 1997.
- [5] J. R. Parker, "Gray level thresholding in badly illuminated images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Volume 13, n.8, pages 813-819, August 1991.
- [6] B. Sankur and M. Sezgin. "Image thresholding techniques: A survey over categories", *Pattern Recognition*, 2001.
- [7] Øivind Due Trier , Torfinn Taxt, "Evaluation of Binarization Methods for Document Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v.17 n.3, p.312-315, March 1995.
- [8] Øivind Due Trier , Torfinn Taxt, "Improvement of Integrated Function Algorithm for binarization of document images", *Pattern Recognition Letters*, v.16 n.3, p.277-283, March 1995.
- [9] J. Sauvola, M. Pietikainen, "Adaptive Document Image Binarization", *Pattern Recognition*, vol. 33, no. 2, pp. 225- 236, 2000.