Degraded document image binarization based on combination of two complementary algorithms

M. Valizadeh, M. Komeili, N. Armanfard, E. Kabir

Abstract—In this paper we combine two binarization algorithms that are complementary to each other. The main idea is to select the better algorithm in each part of document image. There are algorithms that properly distinguish the text from the background in the regions close to the text, but get wrong in the regions far from the text and introduce some part of background as text. We propose a new binarization algorithm that effectively eliminates background and reliably extracts some parts of each character. Then according to the distance of each pixel form the text, the appropriate algorithm is selected to binarize that pixel. Proposed method is applicable for various types of degraded document images. After extensive experiment, the proposed binarization algorithm demonstrate superior performance against four well-know binarization algorithms on a set of degraded document images captured with camera.

I. INTRODUCTION

DOCUMENT image analysis is an important field of image processing and pattern recognition. It consists of image capturing, binarization, segmentation and recognition. Image binarization aims to convert the gray-scale image into binary image and its quality affects the overall performance of document analysis systems. Although many researches have been carried out in the field of document image binarization and various thresholding algorithms have been developed, binarization of document images with poor and variable contrast, shadow, smudge, variable foreground and background intensity are still a challenging problem.

The binarization methods reported in the literatures are generally global or local. Global methods select a single threshold for the image by using its gray levels. These methods compare the gray level of each pixel with the threshold value and classify it as text or background. Threshold selection based on clustering[1], entropy minimization[2], valley seeking in the histogram [3], feature based method [4] and model based method [5] are some popular global binarization algorithms. Global methods have good performance in the case of good separation between foreground and background gray levels but when the histogram of the foreground overlaps with that of the background, these methods fail to operate properly.

To overcome the disadvantages of global methods, various local binarization algorithms have been developed. These methods use local area information to classify each pixel as text or background. Some local methods use only the gray levels of each pixel and its neighborhoods to classify it [6-9]. Whereas some methods in addition to gray level, use the structural features of text such as stroke width [10, 11]and double edge [12, 13] to improve the binarization quality. Topographic analysis using water flow model introduced by Kim et al [14] is a promising idea for document image binarization. This algorithm uses this fact that the text pixel is darker than its neighboring background pixels. Therefore, the problem of binarization is reduced to finding local minima. In this algorithm, water flow model is utilized to find local minima and label them as text. However, this algorithm has some restrictions that reduce its applicability.

In this paper, we incorporate the stroke width and double edge features of text into the water flow model to solve its problems and make it a contrast independent algorithm for document image binarization. The proposed algorithm effectively eliminates the non-text regions and extracts texts. However, it produces broken characters. Therefore, we combine it with the Niblack's local binarization algorithm, which is complementary to our algorithm, and get a high quality binary image.

The rest of this paper is organized as follows: section II briefly reviews related works on document image binarization. Section III describes our methodology for degraded document image binarization. Experimental result and comparison with other binarization algorithms are shown in section IV and conclusions are given in section V.

II. RELATED WORK

Local binarization algorithms have good performance in extracting text from degraded document images. We briefly review three related local binarization algorithms which will be evaluated and compared with our binarization method.

Niblack [9] proposed an algorithm which calculate separate threshold for each pixel by shifting a rectangular window across the image. The threshold *T(x,y)* for the center of window is computed using local information.

$$T(x,y) = m(x,y) + ks(x,y) \quad (1)$$

Where *m(x,y)* and *s(x,y)* are local mean and standard deviation of the gray level in the local window centered on pixel *(x,y)*. The window size and *k* are the predetermined parameters of this algorithm. The value of *k* determines the amount of text region inside the local window and set to -0.2. This method can separate text from background in the area close to text region. However, in the regions far from the text, some parts of background are regarded as text and

Manuscript received june 2, 2009.
M. valizadeh (e-mail: valizadeh@ modares.ac.ir).
M. komaili (e-mail: komaili@modares.ac.ir)
N. armanfard(e-mail: armanfard@modares.ac.ir)
E. kabir(e-mail: kabir@modares.ac.ir)

background noise is magnified.

Sauvola [7] used the knowledge about the intensity of foreground and background to solve the problems of Niblack's method. He proposed a threshold criterion as follows.

$$T(x,y) = m(x,y) \times (1 - k(1 - s(x,y)/R)) \qquad (2)$$

Where R is constant and for the image with 256 gray levels, set to 128 and k set to 0.5. This procedure gives satisfactory binary image in the case of high contrast between foreground and background. However, in the case of poor contrast images, the text regions are eliminated.

In image binarization based on water flow model [14], the image surface is regarded as three-dimensional terrain that is composed of mountains and valleys. Where the mountains represent the background and valleys represent the text regions. In this method, the drops of water fall on all pixels of the image and water travels onto the terrain and fills the valleys that are local minima in the terrain. If a drop of water fills a local minimum, its height increases by the rate of rainfall process. In, the rate of rainfall process is set to one. This procedure is applied on the image w times. Where w represents the amount of rainfall process, which is a predetermined parameter. After the rainfall process stops, the ponds are extracted and the average water level of each pond is assigned to the entire points of the related pond. The difference between original terrain and water level after rainfall named water amount is thresholded and produces the binary image. Since this algorithm uses a global binarization algorithm to extract text, the low contrast text related to shallow pounds are lost.

III. OUR METHODOLOGY

There are algorithms that efficiently distinguish the text from the background in the area close to the text. However, in the area far from the text, they label some parts of background as text. Niblack's method has such characteristic. In this paper, we present a contrast independent binarization algorithm that complements the Niblack's method. It effectively eliminates the background but produces broken characters. Therefore, we combine these two complementary binarization algorithms to get a reliable binarization algorithm.

A. Proposed contrast independent binarization method

Most of binarization algorithms are sensitive to the contrast between the text and background and eliminate the low contrast text. We apply some modifications to the image binarization algorithm based on water flow model to solve its problems and provide a contrast independent binarization algorithm. The main differences between the original algorithm and our modified method can be summarized as follow:

1- We pour the water only onto the edge pixels

2- We use the stroke width and double edge features of text to separate text pounds.

In our modified water flow model, we need to extract some useful information from gray level image. These information contain the edge pixels and the stroke width, SW, of characters. Where the edge pixels determine the regions in which the drops of water fall down and SW help us to set the rate of rainfall process so that the average water amount of text pounds increase by the known value in each iteration of rainfall process.

1) Edge pixels extraction and SW measurement

Since we pour the water only onto the edge pixels, the lost of edge pixels causes the elimination of characters. Therefore, it is essential to use a reliable edge detection algorithm to avoid the lost of weak edges. Canny edge detector efficiently extracts weak edges, so we select it for this application.

Stroke width, SW, is a useful structural feature of text. Appropriate measurement and usage of this feature can help us to improve the binarization quality. In [10] the run length histogram analysis is utilized to measure SW. Since run length histogram is calculated by using binary image, they first find the sub-images, which their histograms are bimodal and binarize them. Then they calculate the run length histogram from binary sub-images. This process is time consuming. In our work the edge information that are previously extracted are utilized to measure SW. The distances between two successive edge pixels in horizontal scan line determine the *SW*. Therefore, we compute the histogram of the distances between two successive edges pixel in horizontal scan line. This distance histogram is denoted as a one dimensional array $h(d)$, $d\{2,\ldots..L\}$, where L is the maximum distance to be counted. $h(d)$ is the frequency of distance d. The *SW* is defined as the distance with the highest value of $h(d)$. This technique accurately measures the *SW* in highly degraded images. Fig. 1 shows the $h(d)$ and measured *SW* for a typical document image.

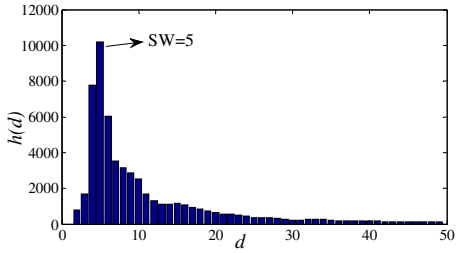


Fig. 1 The histogram of the distances between two successive edge pixel and measured SW

2) Setting the rate of rainfall process

The rate of rainfall process determines how much the height of local minimum is raised when a drop of water fills it. We use the SW and double edge features of text and set the rate of rainfall process so that the average water amount of each pound increases by one in each iteration of rainfall process. As shown in Fig. 2 in horizontally scanning of a character, for each double edge pixels, there are approximately SW text pixels. We use this feature of text to set the rate of rainfall process.

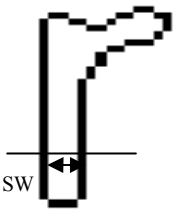


Fig. 2 number of edge pixels of a character in horizontal scan line

During the rainfall process, the drops of water fall down onto the edge pixels and fill the text pounds. Suppose a character has N edge pixels. Therefore, in each iteration of rainfall process, N drops of water fall onto its edges and flow into the region related to that character. The area of this region approximately is N×SW/2. Therefore, if we set the rate of rainfall process into SW/2, the average water amount of that region, pound, increases by one. Whereas the average water amount of pounds that do not follow the SW and double edge features increase less than one. This knowledge helps us to separate text from background. We define the average water amount, AWA, as follow:

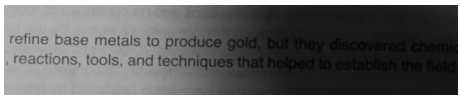
$$AWA = \frac{\text{the amount of water fills the pound}}{\text{the area of related pound}} \quad (3)$$

3) Our modified water flow model

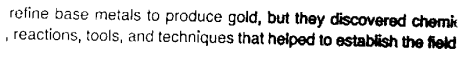
In our algorithm, the drops of water are poured onto the edge pixels and the rainfall process iteratively continues *w* times then it stops. At equilibrium, the difference between water filled and original terrain is defined as filled water. We extract the water-filled regions and apply a labeling process on them to extract pounds. The AWA is computed for each pound and is utilized for pound classification.

Classification of pounds by using AWA is straightforward. Since we know that the AWA of a text pound approximately increases by one in each iteration, we expect that the AWA of a text pound approximately become *w* after *w* iteration. Therefore, we use the following rule for pound classification.

Where the *AWA (k)* represents the average water amount of pound *k* and *T* is a constant parameter that is experimentally set into 0.7. In our work, *w* is set into 10. Fig. 3 (b) shows the result of our contrast independent binarization algorithm applied on Fig. 3(a).



(a)



(b)

Fig. 3 (a) A typical badly illuminated document image (b) the result of our contrast independent binarization algorithm

Extensive experiments show that although our contrast independent binarization algorithm efficiently extracts the text and eliminates the background, it has the following drawbacks.

1- It produces broken characters

2- It produce a binary image in which the stroke width is variable

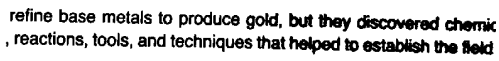
The drawbacks of our binarization algorithm decrease the recognition rate when we apply the OCR software on the binary image. Therefore, we combine our binarization algorithm with the Niblack's method to get a reliable binarization algorithm.

B. Combination of two binarization algorithm

To combine two binarization algorithms, we concentrate on their advantages and disadvantages. We know that Niblack's method distinguish text from background in the area close to the text and our binarization algorithm effectively eliminate background and extract text. Therefore, we apply a morphological dilation operator on the binary image produced by using our contrast independent binarization algorithm to extract the area close to the text. After finding the regions close to the text, we use the Niblack's method to binarize them and label the regions far from the text as background.

In our work, a *2SW×2SW* rectangular structuring element is utilized for dilation. The result of finding the regions close to the text in Fig. 3(a) is shown in Fig. 4 (a) and Fig. 4 (b) shows the result of applying Niblack's method on the regions close to the text and final binary image.

(a)



(b)

Fig. 4 (a) text region extraction by applying dilation on binary image produced by our binarization method (b) result of applying Niblack's method on regions close to the text

IV. EXPERIMENTAL RESULT

The proposed binarization algorithm was tested on a set of degraded document images and its performance was compared with four well-known document image binarization algorithms. Based on visual criteria, proposed algorithm outperforms all algorithms that are tested. Example results are shown in Fig. 5. We briefly explain the reasons that other binarization algorithm fail to binarize badly illuminated document. Niblack's method [9] introduces some parts of the regions inside the sliding window as text, so in the regions far from text, some part of background are labeled as text. Sauvola's method [7] is very sensitive to the predetermined parameters and in document with variable foreground and background we cannot set its

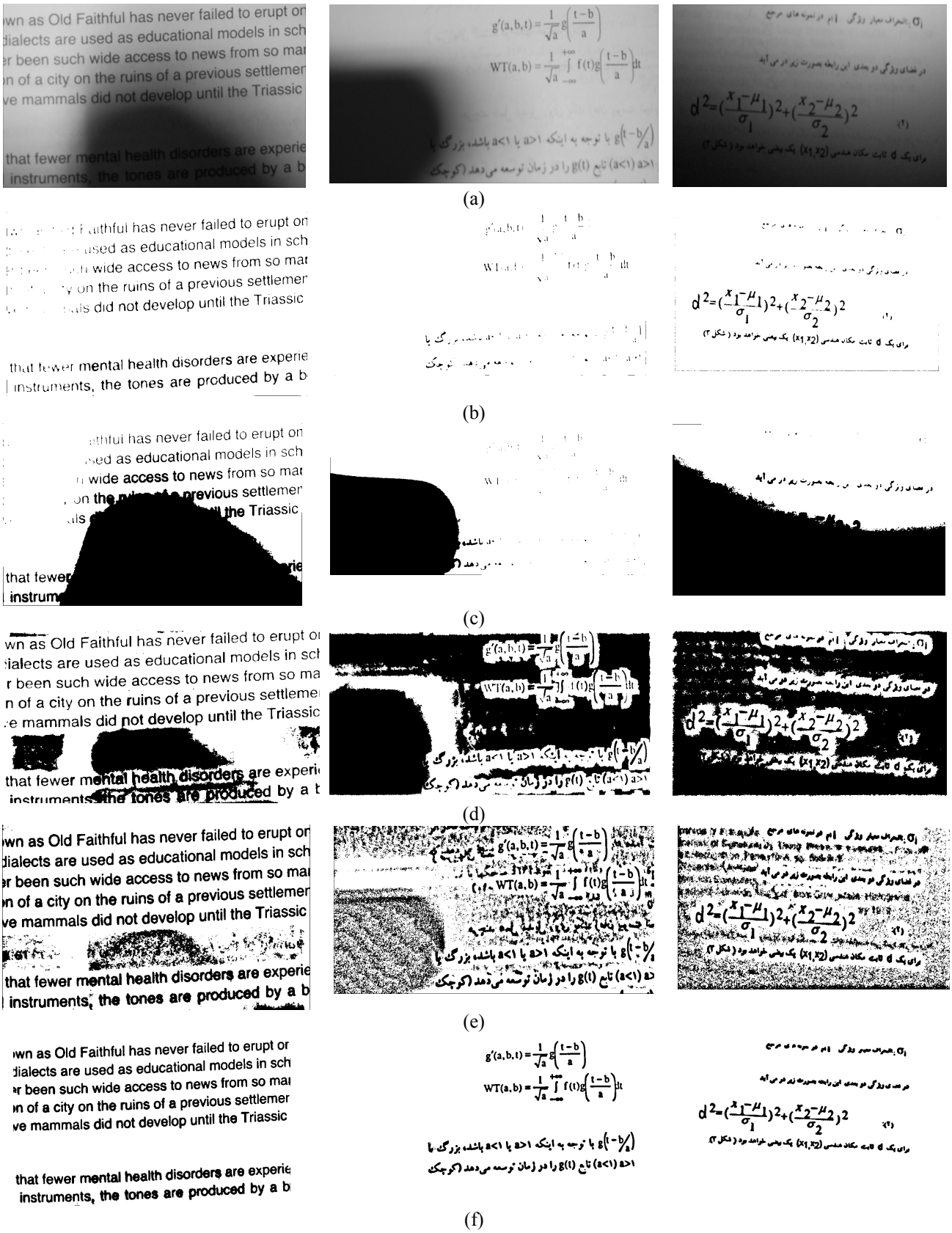


Fig. 5 (a) Badly illuminated document images. (b) results of Sauvola's method. (c) results of Otsu's method. (d) results of Kim's method. (e) results of Niblack's method. (f) results of our proposed method.

parameter's value properly. Otsu's algorithm [1] fails to binarize these images properly because the gray level of foreground and background is not separable. Kim's method [14] uses a global thresholding to binarize filled water, and it cannot separate text from the low gray level background in badly illuminated document images.

V. CONCLUSION

In this paper, we propose a contrast independent binarization algorithm that effectively eliminates background and reliably extracts some parts of each character. Beside all advantages, this method produces broken character that reduces the efficiency of character recognition algorithms. Dealing with this problem, we combine our binarization algorithm with the Niblack's method which complements our algorithm. The main idea is to select the better algorithm in each part of document image. Since Niblack's method effectively distinguishes the text from the background in the areas closed to the text, we use our algorithm to find these areas and binarize them by Niblack's method. The regions far from the text are labeled as background by our algorithm. After extensive experiment, the proposed binarization algorithm demonstrate superior performance against four well-know binarization algorithms on a set of degraded document images captured with camera.

REFERENCES

[1] N. Otsu, "A threshold selection method from grey level histogram," *IEEE Trans. Syst. Man Cybern,* vol. 9, pp. 62–66, 1979.

[2] J.N. Kapur, P.K. Sahoo, and A.K.C. Wong, "A new method for graylevel picture thresholding using the entropy of the histogram," *Computer Vision, Graphics and Image Processing,* vol. 29, pp. 273–285, 1985.

[3] J. SWhite and A. Rosenfield, "Histogram Modification for Threshold Selection," *IEEE Transactions on Systems, Man, Cybernetics,* vol. 9, pp. 38-52, 1979.

[4] Y. Liu and S.N. Srihari, "Document image binarization based on texture features," *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 19, pp. 540-544, 1997.

[5] A. Dawoud and M.S. Kamel, "Iterative Multimodel Subimage Binarization for Handwritten Character Segmentation," *IEEE TRANSACTIONS ON IMAGE PROCESSING,* vol. 13, pp. 1223-1230, 2004.

[6] B. Gatos, I. Pratikakis, and S.J. Perantonis, "Adaptive degraded document image binarization," *Pattern Recognition,* vol. 39, pp. 317 – 327, 2006.

[7] J. Sauvola and M. Pietikainen, "Adaptive document image binarization," *Pattern Recognition,* vol. 33, pp. 225–236, 2000.

[8] J. Bernsen, "Dynamic thresholding of grey-level images," in *Proceedings of the Eighth International Conference on Pattern Recognition*, Paris, 1986.

[9] W. Niblack, *An introduction to digital image processing*. NJ, USA: Prentice Hall, Englewood Cliffs, 1986.

[10] Y. Yang and H. Yan, "An adaptive logical method for binarization of degraded document images," *Pattern Recognition,* vol. 33, pp. 787-807, 2000.

[11] M. Kamel and A. Zhao, "Extraction of binary character/graphics images from grayscale document images," *Graphical Models Image Processing,* vol. 55, pp. 203-217, 1993.

[12] Q. Chena, Q. Suna, P.A. Heng, and D. Xia, "Adouble-threshold image binarization method based on edge detector," *Pattern Recognition,* vol. 41, pp. 1254 – 1267, 2008.

[13] X. Ye, M. Cheriet, and C.Y. Suen, "Stroke-Model-Based Character Extraction from Gray-Level Document Images," *IEEE TRANSACTIONS ON IMAGE PROCESSING,* vol. 10, pp. 1152-1161, 2001.

[14] I.K. Kim, D.W. Jung, and R.H. Park, "Document Image Binarization Based on Topographic Analysis Using a Water Flow Model," *Pattern Recognition,* vol. 35, pp. 265-277, 2002.