Journal of Computational Information Systems 7: 6 (2011) 1886-1892
Available at http://www.Jofcis.com

Fast Document Image Binarization Based on an Improved Adaptive Otsu's Method and Destination Word Accumulation

Yudong ZHANG , Lenan WU†

School of Information Science and Engineering, Southeast University, Nanjing China

Abstract

This paper presents an improved adaptive method for fast document image binarization. The proposed method first uses wiener filter to reduce noises; second uses an improved adaptive Otsu's method for binarization; and third uses dilation and erosion operators to preserve stroke connectivity and fill possible breaks, gaps, and holes. Besides, packed binary format and destination word accumulation are used to hasten the dilation and erosion processing. The experiments demonstrate that our method is effective and outperforms global Otsu's method and adaptive Otsu's method, and the computation time of morphological operators is faster than traditional morphological operators.

Keywords: Document Image Binarization; Mathematical Morphology; Destination Word Accumulation; Wiener Filter; Otsu's Method

1. Introduction

In automatic document processing, text binarization is critical, since it allows the documents to be recognized, stored, and retrieved more efficiently [1]. The first attempts towards binarization utilized a statistically defined global threshold [2]. These methods, though simple, exhibit poor results when they deal with degraded documents such as shadows, non-uniform illumination, and ink seeping [3]. Afterwards, scholars prefer to employ adaptive local binarization methods which ignore the edge property and lead to erroneous results due to the creation of fake shadows [4].

However, traditional adaptive method can not distinguish blocks as containing text or not, so it will still output binarization result for background blocks [5]. Therefore, we proposed a novel methodology which used the maximum interclass variance to classify blocks as containing text or not. Besides, we chose the wiener filter [6] as the preprocessing and the mathematical morphological as the postprocessing. Moreover, a packed binary format was utilized to hasten the morphological operators.

The structure of this paper was organized as follows: Next section 2 gave the detailed methodology of our proposed approach, including the wiener filter as preprocessing, the improved adaptive Otsu's method as binarization, the mathematical morphology as the postprocessing, and the packed binary format as acceleration strategy. Experiments in Section 3 involved the binarization result of our proposed approach, and the time compare to validate the effectiveness of the proposed acceleration strategy. Final section 4 was

† Corresponding author.
Email addresses: zhangyudongnuaa@gmail.com (Yudong ZHANG), wuln@seu.edu.cn(Lenan WU)

June, 2011

devoted to the conclusion and discussion.

2. Method

2.1. Preprocessing

A preprocessing stage of the grey scale source image is essential for the elimination of noisy areas, smoothing of background texture as well as contrast enhancement between background and text areas [7]. There are a lot of filters can accomplish aforementioned goals. In this paper, we use the wiener filter because it is an adaptive filter and it tailors itself to be the "best possible filter" for a given signal [8]. Consider an image *X* is corrupted by noises. *Y*=*I*+*N*. The winner filter can be depicted as

$$H(f) = \frac{|I(f)|^2}{|I(f)|^2 + |N(f)|^2} \tag{1}$$

where *I*(*f*) denotes the power of uncorrupted image, and *N*(*f*) denotes the power of the noise. The filtered image is denotes as

$$I_F = I * h \tag{2}$$

where *h* is the inverse DFT of *I*(*f*), * denotes the convolution operator, and I_F denotes the filtered image.

2.2. Binarization

We use the adaptive Otsu's method by dividing the image into small 32*32 blocks, which overlay the image without overlap. If the blocks can not fit exactly over the image, we use partial blocks to pad the right or bottom of the image. Fig. 1 shows a 8-by-8 image divided into 9 3-by-3 blocks.

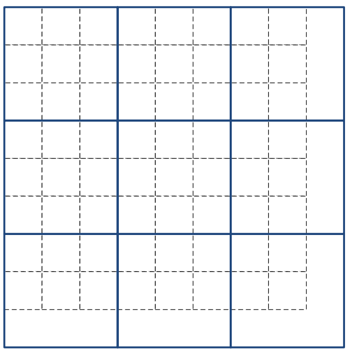

Fig. 1 8-by-8 Image Divided into 3-by-3 Blocks

The Otsu's method is used to binary each block. Otsu's method searches for the threshold that minimizes the intra-class variance which is defined as the weighted sum of variances of the two classes [9].

$$\sigma_\omega^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t) \tag{3}$$

where ω_i denotes the probabilities of the two classes separated by a threshold *t*, and σ_i^2 denotes the variances of these classes. Otsu has proven that minimizing the intraclass variance is the same as maximizing interclass variance [10]

$$\sigma_b^2(t) = \sigma^2 - \sigma_\omega^2(t) = \omega_1(t)\omega_2(t)\left[\mu_1(t) - \mu_2(t)\right]^2 \tag{4}$$

which is expressed in terms of class probabilities ω_i and class means μ_i, which in turn can be updated iteratively. Formula (4) is simpler than formula (3), therefore, we maximize formula (4) to get the Otsu's threshold. The procedures of Otsu's method can be depicted as follows:

Step 1. Compute histogram and probabilities of each intensity value;

Step 2. Set the initial value of $\omega_i(0)$ and $\mu_i(0)$;

Step 3. Loop for all possible thresholds t

Step 3.1. Update $\omega_i(t)$ and $\mu_i(t)$;

Step 3.2. Compute $\sigma_b^2(t)$;

Step 4. Choose the threshold t^* corresponding to the maximum of $\sigma_b^2(t)$.

Step 5. The binarization image I_B is expressed as

$$I_B(x,y) = \begin{cases} 0, I(x,y) < t^* \\ 1, I(x,y) \geq t^* \end{cases} \tag{5}$$

Here 0 denotes black representing the text, and 1 denotes white representing the background.

Those 32*32 blocks containing texts are called 'normal blocks'. On the counter part, the blocks containing degraded background instead of texts are called 'background blocks'. Otsu's method is effective for normal blocks but will introduce in artificial texts for background blocks [11]. Therefore, we propose a classification method: If σ_b^2 is higher than given threshold, the block is a normal one, otherwise is a background block. After hundreds of experiment, the threshold is determined as 0.74. Fig. 2 depicts the flowchart of our proposed improved adaptive Otsu's method.

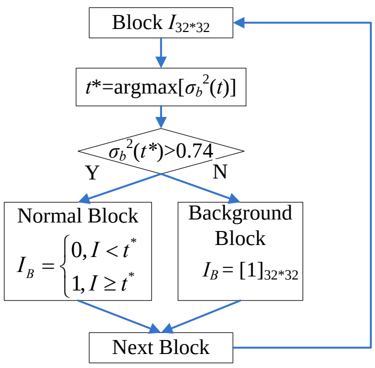

Fig. 2 Flow of Improved Adaptive Otsu's Method

2.3. Postprocessing

In this stage, the mathematical morphology is used to enhance the quality of the binarization image I_B which results in the previous step. We first proceed to a dilation operator, which ensures the character thickness remains the same while several gaps in the character body will diminish. Second we proceed to an erosion operator in order to further improve the quality of text regions and to preserve stroke connectivity by isolated pixel removal and filling of possible breaks, gaps, and holes.

Let I represent a binary image and S represent a structural element (SE). Both I and S have an origin; the origin of S is referred to as its center by convention. Then the dilation $\oplus$ and erosion $\ominus$ can be defined as follows [12]

$$I \oplus S = \bigcup_{b \in S} I_b = \bigcup_{\omega \in I} S_\omega \tag{6}$$

$$I \ominus S = \bigcap_{b \in \overline{S}} I_b = \bigcap_{\omega \in I} \overline{S}_\omega \tag{7}$$

where I_b denotes the translation of I along the pixel vector b, S_ω denotes the translation of S along the pixel vector ω. The union $\bigcup$ and intersection $\bigcap$ operators represent bitwise OR and AND, respectively. The set $\overline{S}$ denotes the bitwise NOT, viz.,

$$\overline{S} = \{-b \mid b \in S\} \tag{8}$$

2.4. Acceleration

Many available image processing programs treat binary images as a matrix of original size with uint8 class (8bpp), and uint32 class (32bpp) [13]. Those are inefficient in space storage and computation complexity. Each pixel in a binary image has only two possible values, either 0 or 1, therefore, we can map a group of 32 pixels into one uint32 word [14]. Suppose the size of original image is $M \times N$, the transformed image is $M \times (N/32)$. In order for 32-bit shift operations to move naturally across 32-bit boundaries, the packed data must have the most significant byte at the left. The byte order from left to right in a word is then 3-2-1-0 on little-endian machines, conversely, the byte order from left to right is 0-1-2-3 on big-endian machines. Fig. 3 gave a simple illustration of packing an 8×32 binary image into an 8×1 uint32 packed binary format.

	1	2			31	32
1	0	0	..	..	1	1
2	1	0	..	..	1	0
3	0	0	..	..	1	1
4	0	0	..	..	1	0
5	1	0	..	..	1	1
6	0	0	..	..	0	0
7	1	1	..	..	0	1
8	0	0	..	..	1	0

Packed Binary Format
$1*2^{31}+1*2^{30}+\ldots+0*2^1+0*2^0$
$0*2^{31}+1*2^{30}+\ldots+0*2^1+1*2^0$
$1*2^{31}+1*2^{30}+\ldots+0*2^1+0*2^0$
$0*2^{31}+1*2^{30}+\ldots+0*2^1+0*2^0$
$1*2^{31}+1*2^{30}+\ldots+0*2^1+1*2^0$
$0*2^{31}+0*2^{30}+\ldots+0*2^1+0*2^0$
$1*2^{31}+0*2^{30}+\ldots+1*2^1+1*2^0$
$0*2^{31}+1*2^{30}+\ldots+0*2^1+0*2^0$

Fig. 3 Illustration of Packed Binary Format

For the packed binary image, we can use word-sized partitioning approaches such as source word accumulation (SWA) or destination word accumulation (DWA) method. Both SWA and DWA methods are amenable to efficient implementation, but DWA is more easy to realize [15]. Suppose $I_\beta^N(\alpha)$ denotes the N pixels starting at location α and translated by β (α and β denote vectors on the two dimensional image), then the dilation and erosion of DWA can be expressed in the form

$$I \oplus S = \bigcup_{\omega \in I} \bigcup_{b \in S} I_b^{32}(\omega - b) \tag{9}$$

$$I \ominus S = \bigcap_{\omega \in I} \bigcap_{b \in \overline{S}} I_b^{32}(\omega - b) \tag{10}$$

where ω covers all indices (i, j) where j mod 32 = 0. The $I_b^{32}(\omega\text{-}b)$ takes 32 bits starting at ω-b and shifts them by b, therefore, write them into the 32-bit aligned word $I(\omega)$. Because words are computed independently, a union (intersection) over words should be used formally in (10) if the destination is initialized to OFF(ON) pixels. All source code is available at http://www.leptonica.com.

3. Experiments

The experiments were carried out on the platform of Windows XP on desktop PC with Intel Pentium4, 3GHz processor and 2GB memory. The algorithm was developed via Matlab 2010b.

3.1. Binarization Result

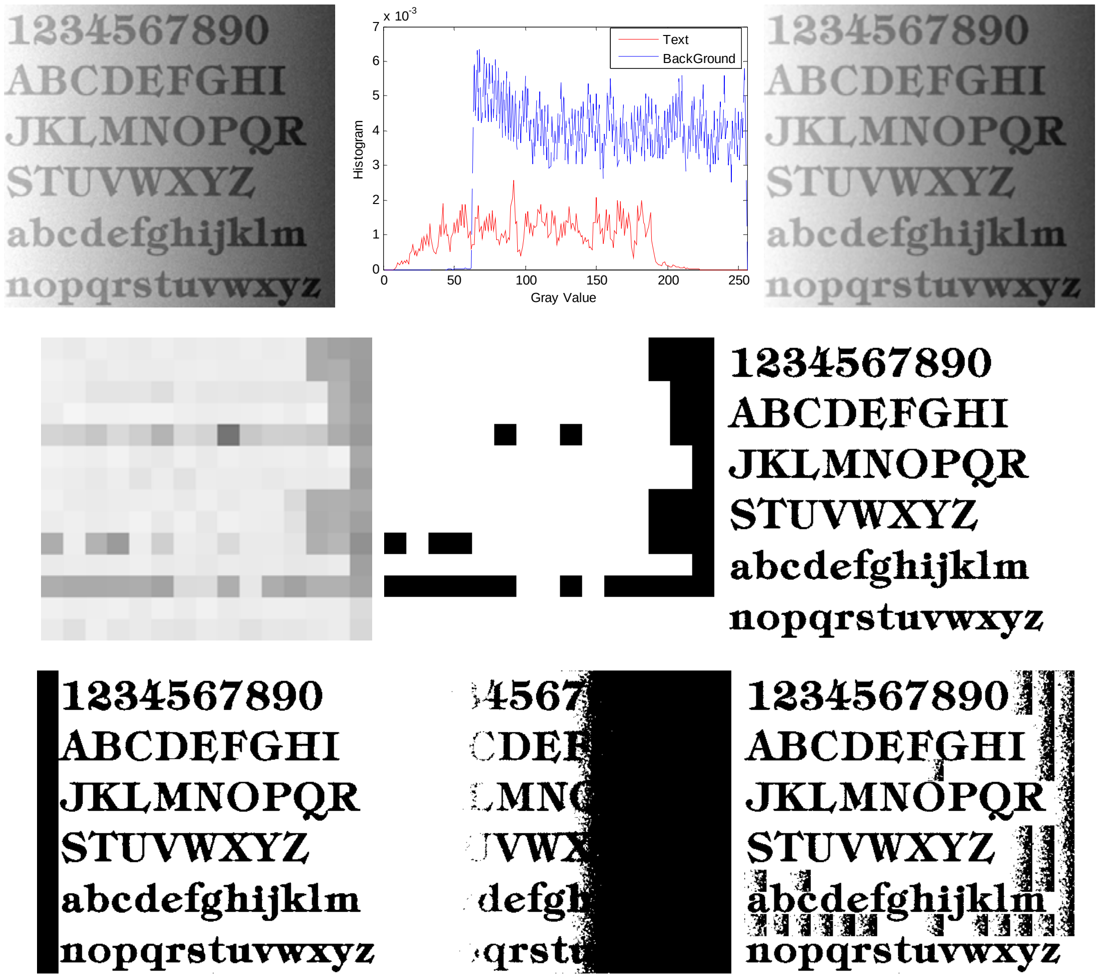

Fig. 4 A Typical Binarization. (a) Degradation Image; (b) Histogram of Text and Background; (c) Wiener Filtered Image; (d) Block-wise Maximum Interclass Variance; (e) Detected Background Blocks Shown in Black Squares; (f) Binarization Result; (g) Morphological Postprocess; (h) Global Otsu's Method; (i) Adaptive Otsu's Method.

Fig. 4(a) shows a typical degradation image with non-uniform illumination and Gaussian noise (variances=0.001). The text area and background area selected manually and their histograms are shown in Fig. 4(b), indicating that the histograms overlapped each other so global method could not binarize the degradation image efficiently. Fig. 4(c) shows the winner filtered result, of which the noises are obviously reduced. Fig. 4(d-e) shows the block-wise maximum interclass variance and the detected background blocks. Fig. 4(f) shows the binarization result using our method. Fig. 4(g) shows the final morphological postprocessing.

Besides, Fig. 4(h-i) shows the binarization results by global Otsu's method and adaptive Otsu's method, respectively. Based on visual criteria, we can state that the proposed algorithm outperforms the other two algorithms with respect to image quality and preservation of meaningful textual information.

3.2. Computation Time

Here we choose 10 different size document images, and perform morphological postprocessing with our proposed acceleration compared to no acceleration. The results are shown in Table 1. The time ratio is defined as the ratio of computation time with acceleration to computation time without acceleration, and is shown in Fig. 5. It indicates that the time ratio decrease with increase in the pixel number. For small document images (pixel number $< 3\times10^5$), the time ratio is about 55%. For median document images ($3\times10^5 <$ pixel number $< 1\times10^6$), the time ratio is about 34%. For large document images (pixel number $> 1\times10^6$), the time ratio is about 30%.

Table 1 Comparison of Morphological Computation Time

Image Size	Pixel Number	Time		Ratio
		No Acceleration	Acceleration	
250*200	50,000	0.001962	0.0011214	57.1551%
400*300	120,000	0.002771	0.0012996	46.8999%
450*300	135,000	0.0024249	0.0013242	54.6083%
450*350	157,500	0.0033532	0.0020391	60.8098%
800*600	480,000	0.008167	0.0028981	35.4861%
1024*768	786,432	0.012645	0.0040318	31.8848%
1080*810	874,800	0.013566	0.0046707	34.4289%
1467*1026	1,505,142	0.022427	0.0068408	30.502%
1600*1200	1,920,000	0.029545	0.0088503	29.9552%
2400*1556	3,734,400	0.053068	0.015794	29.7629%

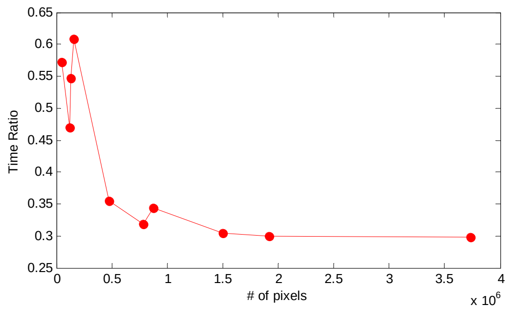

Fig. 5 Time Ratio

4. Conclusions

In this paper, we proposed a novel document image binarization method, which use the wiener filter as preprocessing, an improved adaptive Otsu's method as the binarization, and a mathematical morphology with acceleration as the postprocessing. The experiments demonstrate the effectiveness of the proposed approach and the rapidness of the acceleration strategy. The future work will mainly focus on generalizing the proposed method to other fields such as optical character recognition [16], protein folding model [17], and short-term load forecasting [18].

References

[1] R. Farrahi Moghaddam and M. Cheriet, "A multi-scale framework for adaptive binarization of degraded document images," *Pattern Recognition,* vol. 43, pp. 2186-2198, 2010.

[2] Y. Zhang and L. Wu, "Segment-based Coding of Color Images," *Science in China F edition: Information Science,* vol. 52, pp. 914-925, 2009.

[3] B. Gatos*, et al.*, "Improved document image binarization by using a combination of multiple binarization techniques and adapted edge information," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, 2008, pp. 1-4.

[4] N. Stamatopoulos*, et al.*, "A method for combining complementary techniques for document image segmentation," *Pattern Recognition,* vol. 42, pp. 3158-3168, 2009.

[5] B. Naegel and L. Wendling, "A document binarization method based on connected operators," *Pattern Recognition Letters,* vol. 31, pp. 1251-1259, 2010.

[6] S. Nakamori, "Square-root algorithms of RLS Wiener filter and fixed-point smoother in linear discrete stochastic systems," *Applied Mathematics and Computation,* vol. 203, pp. 186-193, 2008.

[7] Y. Zhang*, et al.*, "A hybrid recognition method for document images," in *12th IASTED International Conference on Intelligent Systems and Control, ISC 2009, November 2, 2009 - November 4, 2009*, Cambridge, MA, United states, 2009, pp. 62-66.

[8] J. Benesty*, et al.*, "On widely linear Wiener and tradeoff filters for noise reduction," *Speech Communication,* vol. 52, pp. 427-439, 2010.

[9] D. Deguchi*, et al.*, "New calculation method of image similarity for endoscope tracking based on image registration in endoscope navigation," *International Congress Series,* vol. 1256, pp. 460-466, 2003.

[10] Y. Ogasawara*, et al.*, "Cloning, Sequencing, and Functional Analysis of the Biosynthetic Gene Cluster of Macrolactam Antibiotic Vicenistatin in Streptomyces halstedii," *Chemistry & Biology,* vol. 11, pp. 79-86, 2004.

[11] Y. Zhang*, et al.*, "Remote-sensing Image Classification Based on an Improved Probabilistic Neural Network," *Sensors,* vol. 9, pp. 7516-7539, 2009.

[12] I. Bloch, "Duality vs. adjunction for fuzzy mathematical morphology and general form of fuzzy erosions and dilations," *Fuzzy Sets and Systems,* vol. 160, pp. 1858-1867, 2009.

[13] Y. Zhang*, et al.*, "Face orientation estimation by particle swarm optimization," in *2009 2nd International Symposium on Information Science and Engineering, ISISE 2009, December 26, 2009 - December 28, 2009*, Shanghai, China, 2010, pp. 388-391.

[14] A. Bourquard*, et al.*, "Optical imaging using binary sensors," *Opt Express,* vol. 18, pp. 4876-88, Mar 1 2010.

[15] D. S. Bloomberg, "Implementation Efficiency of Binary Morpholgoy," in *International Symposium for Mathematical Morphology VI*, Sydney, Australia, 2002.

[16] Y. Zhang*, et al.*, "Color Image Enhancement based on HVS and PC Neural Network," *Science in China E edition: Information Science,* vol. 53, pp. 1963-1976, 2010.

[17] Y. Zhang*, et al.*, "Polymorphic BCO for Protein Folding Model," *Journal of Computational Information Systems,* vol. 6, pp. 1787-1794, 2010.

[18] Y. Zhang*, et al.*, "Bacterial foraging optimization based neural network for short-term load forecasting," *Journal of Computational Information Systems,* vol. 6, pp. 2099-2105, 2010.