Journal of Information Hiding and Multimedia Signal Processing ©2018 ISSN 2073-4212
Ubiquitous International Volume **9**, Number **2**, March **2018**

Multi-Step Segmentation Method Based on Adaptive Thresholds for Chinese Calligraphy Characters

Xiao-Dan Jiang

College of Electrical and Information Engineering
Quzhou University
Quzhou, 324000, P. R.China
16282409@qq.com

Zhe-Ming Lu*

School of Aeronautics and Astronautics
Zhejiang University
Hangzhou, 310027, P. R.China
*Corresponding author: zheminglu@zju.edu.cn

Hong Ye

College of Electrical and Information Engineering
Quzhou University
Quzhou, 324000, P. R.China

Jian-Wen Fang

College of Electrical and Information Engineering
Quzhou University
Quzhou, 324000, P. R.China

Received December, 2016; revised October, 2017

Abstract. *According to the writing characteristics of Chinese calligraphy, this paper proposes a multi-step segmentation method, which is the combination of column segmentation and character segmentation. The method of column segmentation based on adaptive gap thresholds gets each column on the Image. Character segmentation is divided into two steps. Firstly, we perform rough segmentation based on adaptive gap thresholds. Then we use the average height and average word spacing of each column to do the radical block merging based on adaptive threshold. Our method can extract the characters even though they have up-down structure. Experimental results show that the proposed method solves the problem of excessive segmentation and improves the accuracy of Chinese calligraphy character segmentation effectively.In addition, the robustness is strong.*

Keywords: Adaptive threshold, Column segmentation, Character segmentation, Radical merging.

1. **Introduction.** In the treasure of traditional Chinese culture, Chinese calligraphy, with its unique art form and rich connotation, occupies an important position. Many calligraphy works of the celebrities and masters become the classic examples for modern people who study history and learn calligraphy, such as Wang's 'Orchid Pavilion preface' [1],but as valuable calligraphy works, a large number of original calligraphy works are not accessible to many researchers. With the development of science and technology, utilizing

computer and related technology to convert the precious calligraphy to digital images, which not only preserves the historical treasures well, but also provides free access at any time and any where and the convenience of learning for the researchers and calligraphy lovers.

The researchers found that, regardless of studying the formation of different styles of calligraphy and identifying the authenticity of calligraphy works, the characters in a large number of calligraphy works are often needed to be searched, compared, counted and analyzed. It is very necessary to establish a sample library of single calligraphy. Therefore, single character segmentation, the record of related properties and the establishment of the sample library are powerful methods to study the calligraphy works.

At present, the segmentation methods for Chinese characters can be divided into three categories [2]: the methods based on the statistics [3, 4, 5], the methods based on the structures [6, 7, 8]and the methods based on the identification [9, 10, 11, 12]. The methods based on the statistics use the overall statistical distribution characteristics of the characters to determine the boundary between the characters [13], such as the most common histogram projection [14, 15]; The methods based on the structures analyze and search the segmentation rules by the space between characters and the structure of Chinese character itself [13], such as connected domain method [16]; The methods based on the identification identify a variety of possible segmentation results before the actual segmentation and select the final segmentation point through judging the identification results [13], such as the commonly used template matching segmentation algorithm [17, 18]. However, in view of the structure and characteristics of Chinese calligraphy works, most Chinese character segmentation algorithms do not apply to single calligraphy character image segmentation. Therefore, this paper proposes a simple and efficient adaptive threshold segmentation algorithm based on the Chinese character segmentation algorithm.

2. Adaptive Threshold Segmentation Algorithm.

The image of Chinese calligraphy is different from the modern Chinese character document. It is written in the direction from the right to the left, from the top to the bottom. It is written with the writing brush. The stroke is relatively thick and the gap is small. In this paper, we propose an adaptive threshold segmentation algorithm: first, the digital image of the calligraphy is processed by binarization to reduce the interference of background and noise. Then the image is split by column and each column can be obtained. At last, single calligraphy character is obtained by performing row splitting on each column. Column segmentation is based on the blanks in the calligraphy. Character segmentation is divided into two steps. Firstly, we use the blanks to perform rough segmentation. Then we use average height and average word space of each column to perform adaptive threshold radical block merging to reduce excessive segmentation. Experimental results show that the segmentation effect of the proposed algorithm in this paper is ideal. The algorithm has high accuracy and strong robustness.

2.1. Image Binarization.

Image binarization is a common method to study the gray level image using the gray level transformation, that is to set a threshold to divide the gray image into two parts: one part consisting of pixels greater than the threshold and the other part consisting of pixels less than the threshold. Assuming that the function of the input gray image is $\mathbf{f(x, y)}$, the function of the output binary image is $\mathbf{g(x, y)}$, then

$$g(x, y) = \begin{cases} 0 & f(x, y) < T \\ 255 & f(x, y) \geq T \end{cases} \tag{1}$$

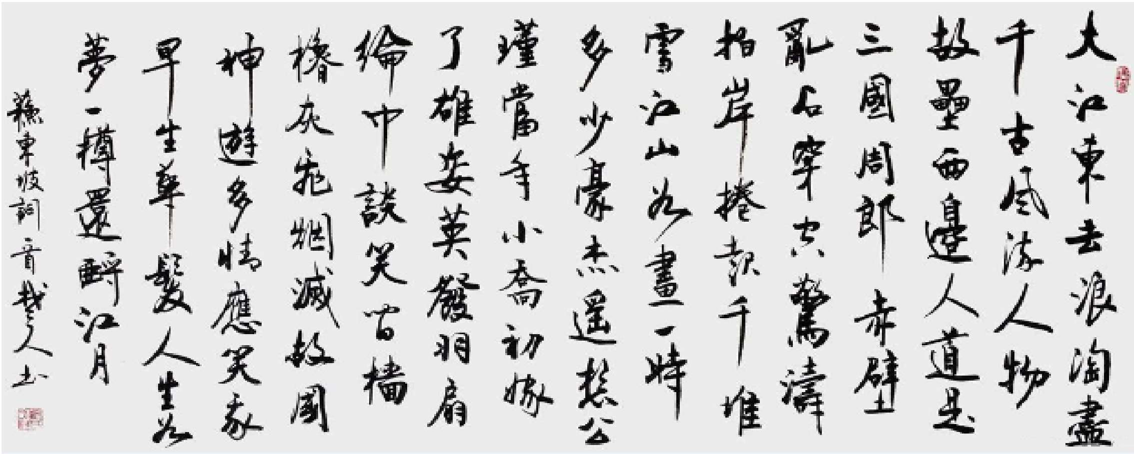


FIGURE 1. Original sample image of running hand

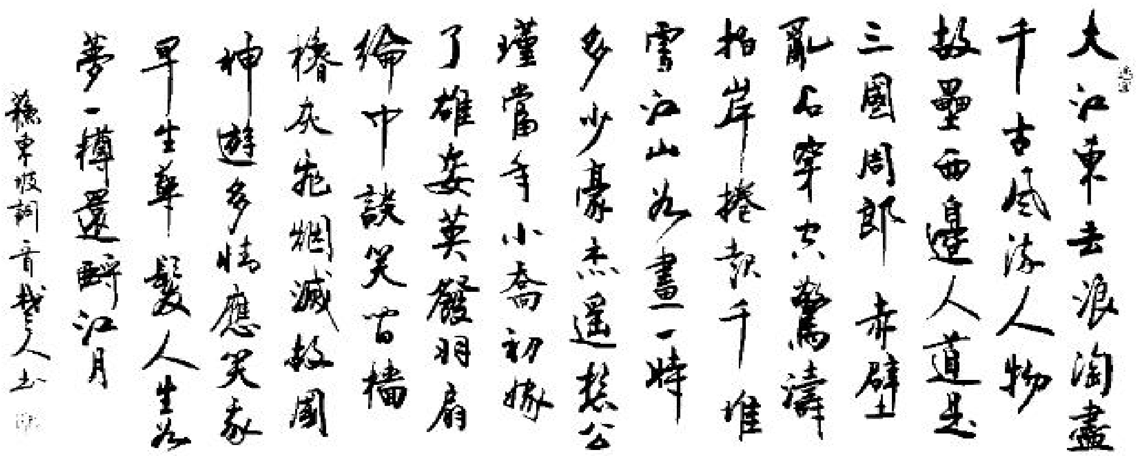


FIGURE 2. The rendering after binarization

In Eq.(1), T is the threshold. Its purpose is to separate the target from the background. So the threshold selection is very important. It not only saves the image information as much as possible, but also reduces the interference of the background and noise.

Through the binarization processing for the calligraphy image, the gray value of the pixel in the image is set to 0 or 255, which makes the whole calligraphy image appear obvious black and white effect. It is helpful for image segmentation. Because the gray level distribution of calligraphy characters and background noise is relatively balanced, this paper uses the simplest fixed threshold method. Set the threshold parameter T to 127, that is, scan the pixel values in the calligraphy image, the output pixel value of the pixel whose value is less than 127 is set to 0 (that is the target area), while the output pixel value is set to 255 (i.e., the background region). Experiments show that the target domain and the background domain of the binary image are obvious, and the basic information of the original calligraphy is retained. Fig.1 and Fig.2 are the images before and after the binarization respectively.

2.2. **Column Segmentation of Calligraphy Works.** After binarization, there is a space between two columns. It is the key to split by column. According to the existing white space in the image, record the start and end positions of each column. So the position of the calligraphy character column is determined. Because of much noise in calligraphy image, there may be some noise columns in the image after binarization. The adaptive threshold segmentation algorithm proposed in the paper can better eliminate noise column. Its main idea is: counting the black pixel values in each column after segmentation and calculating the percentage of the black pixels in this column. If the percentage is less than five percent, the column is treated as noise column; If the width of this column calligraphy character is less than half of average column width, the column is also treated as noise column and eliminated. The detailed steps of the column segmentation of the calligraphy image are as follows:

Step 1: Calculate the black pixel values P_y of each column in the image. Let $\boldsymbol{I}$ denote the calligraphy image, $\mathbf{I}(\mathbf{x}, \mathbf{y})$ denotes the pixel value of $\mathbf{x}$ row and $\mathbf{y}$ column in the image. The value of $\mathbf{I}(\mathbf{x}, \mathbf{y})$ is set to 1 when the pixel is black, and the value of $\mathbf{I}(\mathbf{x}, \mathbf{y})$ is set to 0 when the pixel is white. The statistics of the black pixel value P_y of each column is calculated according to Eq.(2), where $\boldsymbol{H}$ is the image height:

$$P_y = \sum_{x=1}^{H} I(x, y) \tag{2}$$

Step 2: Create an one-dimensional projection array $M[W]$ and calculate $M[W]$. It requires to meet the following conditions:

$$M[W] = \begin{cases} 0 & P_y \leq 8 \\ 1 & P_y > 8 \end{cases} \tag{3}$$

The array length W denotes the image width. If the calculated value of P_y in Step 1 is less than or equal to 8, the array will be treated as the white space column of the image. Put 0 into the corresponding location of the array M, otherwise put 1 into the position. After this operation, the operation on two-dimensional image is transformed into the operation on one-dimensional array M.

Step 3: Partition specific columns. Create two one-dimensional arrays **StartCol** and **EndCol** and store the start position and end position of each character column respectively. Put the positions which converse from 0 to 1 or from 1 to 0 in array M into **StartCol** and **EndCol** respectively. This can easily split each column out independently.

Step 4: Remove the noise column. The existing noise in the image or the writing habit of the calligraphy character will cause a part of useless noise columns. The column segmentation algorithm based on the adaptive threshold can eliminate the noise column well. According to the obtained column width in Step 3, calculate the percentage of black pixels in each column. According to the statistics of the sample image, if the percentage is less than 5 or the width of each column is less than half of the average width of calligraphy character, these columns are regarded as noise column and eliminated. Each retained column will be split by single character in next step.

2.3. **Calligraphy Word Segmentation.** The image will be performed with single character segmentation on the basis of column segmentation. Because many Chinese characters have the up-down structure, if fully applied the idea of column segmentation to the image, it will lead to the separation of radical blocks. So the character segmentation algorithm is more complex and more difficult. The proposed character segmentation in the paper can be divided into rough segmentation and radical block merging. Use the blank space to perform rough segmentation first. Then we use the average height and average character spacing of each column to perform radical block merging based on adaptive threshold to reduce excessive segmentation.

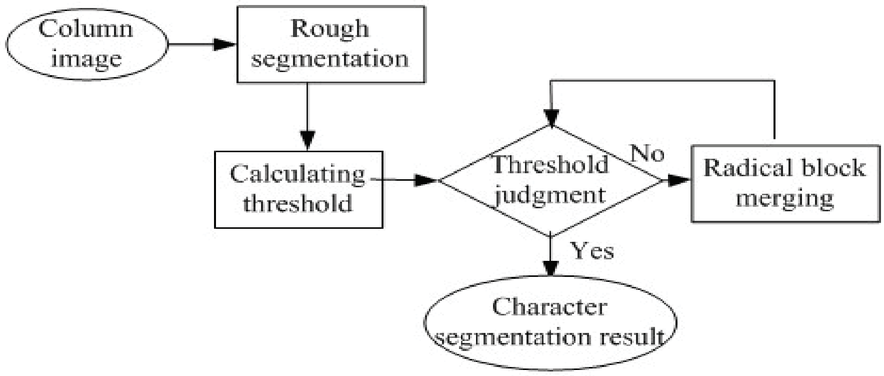


FIGURE 3. Character segmentation algorithm flow chart

2.3.1. *Segmentation threshold determination.* After obtaining the column images, we perform the rough segmentation on each column first. The rough segmentation algorithm is similar to the column segmentation algorithm. It mainly splits the single column horizontally to get the approximate position of a single character in the column. In order to avoid excessive segmentation caused by rough segmentation, we treat the Chinese character radical as an independent character. So it is critical for character segmentation to set proper threshold. According to the statistics of the sample image, the single character size and the spacing between characters are uniform basically. So this paper combines these information and uses character segmentation based on an adaptive threshold. Its main idea is stated as follows: rough segmentation is performed on the column image according to the blank space first. Then the average height of single character and average spacing between characters of each column are calculated. Then the segmentation experience is adopted to set a proper threshold. If the threshold condition can not be satisfied, radical block merging is performed. The detailed algorithm flow is as follows.

From Fig. 3, we can see that the inscribed part of each row are preserved in the array **StartRow** and array **EndRow** respectively through rough segmentation. Then, let the average line-height of single character of each column be **LineHight** and the average spacing between characters be **LineGrap**, we calculate the average height of single character based on Eq. (4) and average spacing in each column based on Eq. (5). Finally, we perform radical block merging by judging the threshold. It is the key of the algorithm. If it is mishandled, the segmentation effect can not achieve the desired result. The accuracy of single character segmentation is low.

$$LineHight = \frac{1}{n} * \sum_{i=0}^{n-1} \left(EndRow\,[i] - StartRow\,[i] \right) \tag{4}$$

$$LineGrap = \frac{1}{n-1} * \sum_{i=0}^{n-2} \left(StartRow\,[i+1] - EndRow\,[i] \right) \tag{5}$$

2.3.2. *The radical block merging of calligraphy character.* The coordinate values that are saved in the array **StartRow** and array **EndRow** store are for a single character or for its radical can be judged according to the preset threshold. Array **StartRow[i]** and array **EndRow[i]** keep the start position and end position of initial character in row **i**. According to Eq.(4) and the Eq.(5),if the adjacent two rows meet Eq.(6) and Eq.(7), we merge the adjacent rows:

$$StartRow\,[i+1] - EndRow\,[i]\,\epsilon\,[0, \alpha * LineGrap] \tag{6}$$

$$EndRow\,[i] - StartRow\,[i] < \beta * LineHight \tag{7}$$

According to the segmentation experience $\alpha = 0.67, \beta = 0.5$. Eq.(6) denotes the range of the distance between two adjacent rows is , Eq.(7) denotes the height of the current row is less than half of the average single character height. If the two equations are satisfied at the same time, the row is regarded as having up-down structure. Then we perform radical block merging.

2.4. **Noise block removal.** After calligraphy image binarization, if the noise in the image is much, some useless noise block will exist in the single character block after segmentation. Therefore, the paper has performed noise block removal in the preservation of single calligraphy character. The following two conditions are set for single calligraphy character after segmentation:

$$EndCol\,[i] - StartCol\,[i] \geq \delta \qquad i = 1, 2, 3, \cdots\cdots n \tag{8}$$

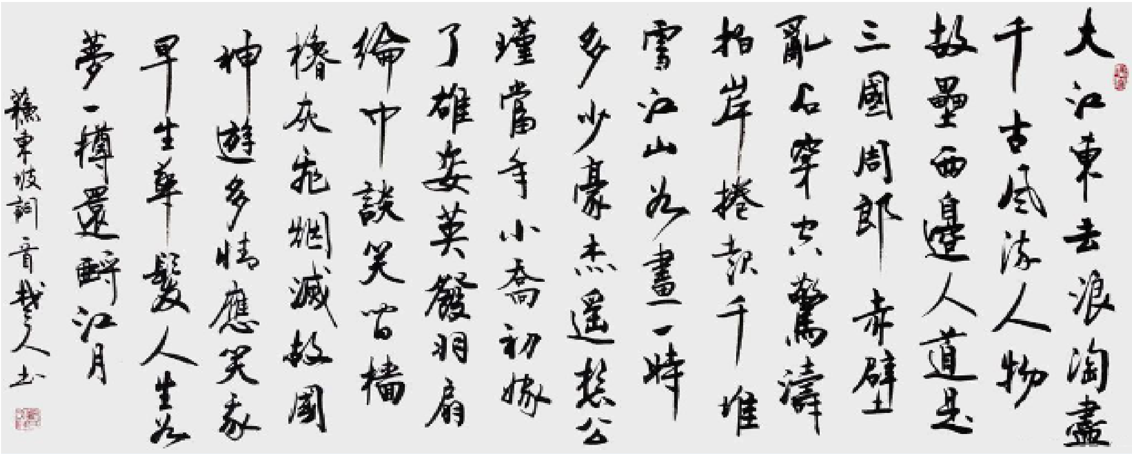
$$RowHight \geq \varepsilon * LineHight \tag{9}$$

In Eq.(8), $\delta = 10$. According to the experimental test sample statistics, the width of normal single calligraphy character is generally greater than 10 pixels. In Eq.(9), **RowHight** denotes the height of single calligraphy character block,$\varepsilon = 0.5$. If the single character block does not satisfy Eq.(8) or Eq.(9), it will be regarded as the noise block and removed. It is not the research object.

3. **Experimental results and analysis.** This paper selects five calligraphy fonts: regular script, official script, running script, cursive and seal. 30 test samples are used for the algorithm implementation under the environment of Visual Studio 2010. Fig.4(a) and Fig.5(a) are used as the sample images. We can see that the blank space of row and column and the font size are uniform. In the experiment, the fixed threshold segmentation algorithm is chosen as the comparison. The algorithm sets fixed segmentation threshold between characters in the process of character segmentation. The segmentation effect varies obviously when the algorithm is applied to those calligraphy images whose spacing differs a lot.

Fig.4(b) and Fig.5(b) are the segmentation results of the fixed threshold algorithm, the difference between the two results is huge. Fig.4(b) achieves the effect of character segmentation well, while Fig.5(b) hardly completes character segmentation. The reason is that the fixed threshold conditions above are set for Fig.4(b). If the experimental sample is changed, the segmentation effect of the fixed threshold algorithm may not be obvious. So the robustness of the fixed threshold segmentation algorithm is weak. Fig.4(c) and Fig.5(c) are the effect images of the adaptive threshold segmentation algorithm in the paper. They realize the character segmentation effect in the calligraphy works well. The experimental results show that the accuracy and robustness of the proposed algorithm is better.

The author does some statistical analysis of the experimental results of different calligraphy fonts. Table 1 records the experimental results of 30 different fonts sample images. From Table 1, we can see that the accuracy of fixed threshold segmentation method is unstable. The reason is mainly that the fixed threshold segmentation method relies on the

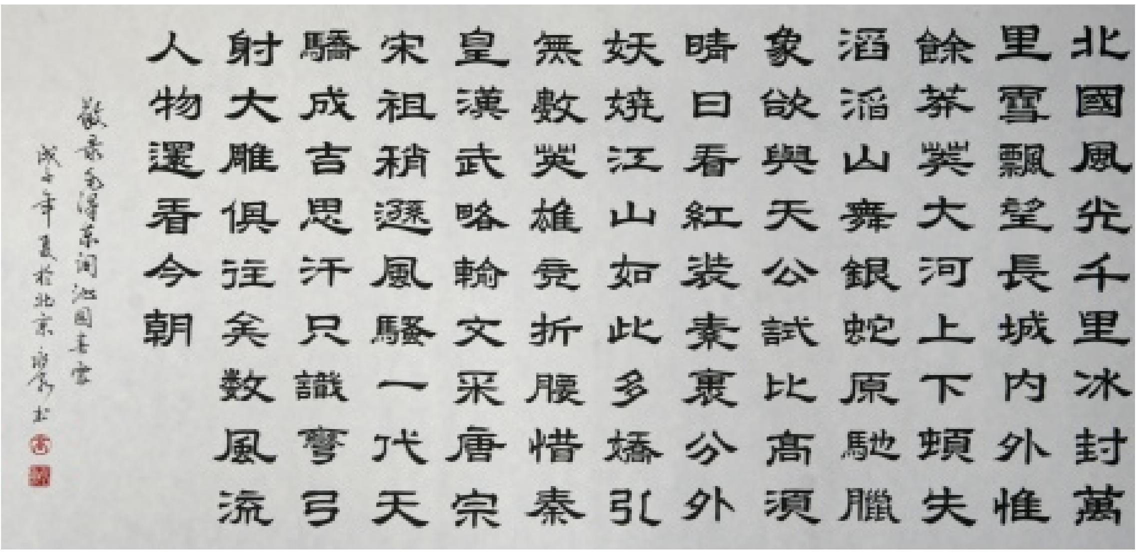


(a)

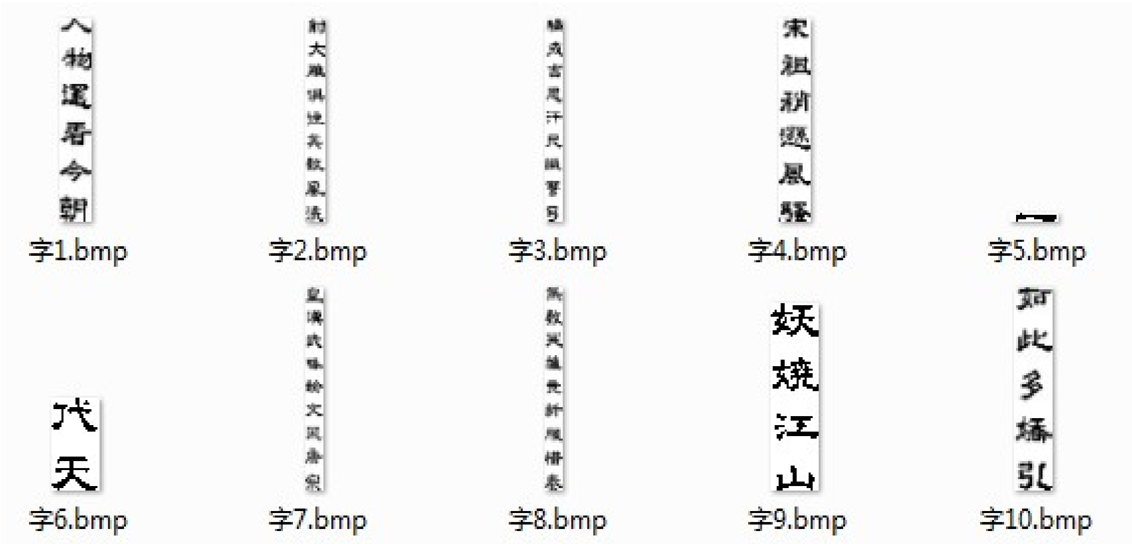
(b)

(c)

Figure 4. Comparison of segmentation with fixed threshold and adaptive threshold for a sample image of running script: (a)The original sample image of running script, (b)The effect image of fixed threshold segmentation (local), (c)The effect image of adaptive threshold segmentation (local)



(a)



(b)

(c)

FIGURE 5. Comparison of segmentation with fixed threshold and adaptive threshold for a sample image of official script: (a)The original sample image of official script, (b)The effect image of fixed threshold segmentation (local), (c)The effect image of adaptive threshold segmentation (local)

TABLE 1. The comparisons of the experimental results of 30 sample images.

Calligraphy class	Sample number	Total number of samples	Correct segmentation of sample		Correct rate	
			Fixed threshold algorithm	adaptive threshold algorithm	Fixed threshold algorithm	adaptive threshold algorithm
Regular script	6	430	415	424	96.5%	98.6%
Official script	6	322	188	316	58.4%	98.1%
Seal character	5	125	117	123	93.6%	98.4%
Running script	7	633	285	606	45.02%	95.8%
Cursive	6	331	227	293	68.6%	88.5%

sample image. If different sample images spacing differs a lot, the fixed threshold segmentation method is basically not suitable. However, the adaptive threshold segmentation method can maintain stable segmentation efficiency for different sample images.

Because the four fonts, i.e., regular script, official script, seal character and running script, are written regularly and the font size and spacing are uniform, the segmentation effect of using the proposed algorithm is ideal. However, for the cursive script, the writing is more casual. There are some conjoint strokes between characters. So the accuracy of the adaptive threshold segmentation algorithm declines, but on the whole the segmentation effect is ideal.

4. **Conclusion.** This paper presents a calligraphy character segmentation algorithm based on adaptive thresholds, which can segment the characters efficiently based on the characteristics that the calligraphy character size and the spacing between the characters in the calligraphy image are uniform.Experiments show that it can achieve good segmentation effect for a variety of calligraphy fonts. It is helpful to improve the automatic degree of feature extraction of calligraphy characters and the identification of calligraphy works. In the paper, for the noise such as seal and creases in the calligraphy image, the effect of noise block removal is not ideal. For the segmentation of conjoint characters, it also needs further research and discussion.

Acknowledgments. This work was partly supported by the Public Good Research Project of Science and Technology Program of Zhejiang Province, China. (NO.2016C31097 and NO.2015C33230), the Science and Technology Project of QuZhou, Zhejiang, China. (NO.2015Y005).

REFERENCES

[1] X. F. Zhang, Y. T. Zhuang, W. M. Lu , F. Wu, Chinese calligraphic character retrieval based on shape similarity, *Journal of Computer-aided Design & Computer Graphics*, vol. 17, no. 11, pp.2565–2569, 2005.

[2] E. Z. Ni, M. J. Jiang, C. L. Zhou, Research on segmentation of historical Chinese books, *Computer Engineering and Applications*, vol. 49, no.2, pp.29–33, 2013.

[3] C. Hong, G. Loudon, Y. Wu, and R. Zitserman, Segmentation and recognition of continuous handwriting Chinese text, *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 12, no.2, pp. 223–232, 1998.
[4] X. Zhang, and V. Sugumaran, Content based search engine for historical calligraphy images, *International Journal of Intelligent Information Technologies*, 2014, 10(3):1-18.
[5] Y. T. Ma, Z. S. Tao, J. H. Zhang, and X. W. Yang, Optimal model for handwritten Chinese character segmentation, *Computer Engineering and Applications*, vol. 44, no.2, pp.227–229, 2008.
[6] Y. Lu, P. F. Shi, and K. H. Zhang, Segmentation of free-format handwritten Chinese characters based on structure features of characters, *Acta Electronica Sinica*, vol. 28, no.5, pp.102–104, 2000.
[7] X. Y. Li, F. Yang, and W. B. Zhang, Segmentation of connected handwritten Chinese characters based on structure cluster and strokes analysis, *Computer Engineering and Applications*, vol.44, no.34, pp. 163–165, 2008.
[8] H. Huang, Z. K. Huang, M. Y. Lin, and L. Y. Hou, Character segmentation of Chinese rubbing images using total variation preprocess, *Applied Mechanics and Materials*, vol. 734, pp. 590–595, 2015.
[9] C. L. Liu, M. Koga, and H. Fujisawa, Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no.11, pp.1425–1437, 2002.
[10] Y. Jiang, X. Q. Ding, and Z. Ren, Handwritten Chinese address segmentation and recognition based on structural matching, *Journal of Tsinghua University (Science and Technology)* , vol. 46, no.7, pp.1235–1238, 2006.
[11] T. H. Su, T. W. Zhang, D. J. Guan, H. J. Huang, Off-line recognition of realistic Chinese handwriting using segmentation-free strategy, *Pattern Recognition*, 42, pp.167–182, 2009.
[12] Z. Cai, J. Wang, and Y. Sun, Segmentation of handwritten Chinese character strings based on improved Algorithm Liu, *Indonesian Journal of Electrical Engineering and Informatics*, vol. 2, no.4, pp. 161–169, 2014.
[13] Y. Gao,and Y. Yang, Survey of unconstrained handwritten Chinese character segmentation, *Computer Engineering*, vol. 30, no.5, pp.144–147, 2004.
[14] Y Lu, Machine printed character segmentationan overview, *Pattern Recognition*, vol. 28, no.1, pp. 67–80, 1995.
[15] W. Zheng, X. N. Huang, and Y. Y. Ma, Method for character segmentation of shopping receipts of POS machine, *Applied Mechanics and Materials*, Vols. 543-547, pp. 2274–2277, 2014.
[16] R. G. Casey, and E. Lecolinet, A survey of method and strategies in character segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no.7, pp.690–706, 1996.
[17] M. Yu, Y. Y. Zhang, and C. H. Xue, Image segmentation algorithm of single handwritten Chinese characters, *Computer Engineering and Applications*, vol. 46, no.9, pp. 180–182, 2010.
[18] Y. Zhang, Z. Q. Zha, and L. F. Bai, A license plate character segmentation method based on character contour and template matching, *Applied Mechanics and Materials*, vols. 333-335, pp. 974–979, 2013.