Adaptive Graph Cut Based Binarization of Video Text Images

Cunzhao Shi, Baihua Xiao, Chunheng Wang, Yang Zhang
State Key Laboratory of Intelligent Control and Management of Complex Systems
Institute of Automation, Chinese Academy of Sciences
Beijing, China
{cunzhao.shi, baihua.xiao, chunheng.wang, yang.zhang}@ia.ac.cn

***Abstract*—Interactive image segmentation which needs the user to give certain hard constraints has shown promising performance for object segmentation. In this paper, we consider characters in text image as a special kind of object, and propose an adaptive graph cut based text binarization method to segment text from background. The main contributions of the paper lie in: 1) in order to make the binarization local adaptive with uneven background, the text region image is firstly roughly split into several sub-images on which graph cut is applied; and 2) considering the unique characteristics of the text, we propose to automatically classify some pixels as text or background with high confidence, severed as hard constraints seeds for graph cut to extract text from background by spreading the seeds into the whole sub-image. The experimental results show that our approach could get better performance in both character extraction accuracy and recognition accuracy.**

Keywords-text binarization; graph cut; hard constraints seeds; adaptive; split-merge; sub-image.

I. INTRODUCTION

As text in images and videos has different sizes, resolutions and complex backgrounds, applying traditional optical character recognition (OCR) technology directly on the text image often leads to poor recognition rate. Thus, an effective binarization method is necessary for good system performance. In this paper, we focus on binarization of text region images from videos with relatively complex background.

A large number of approaches have been proposed on text binarization during the past years. Most of them could be classified into two major categories: statistical threshold methods derived from traditional document binarization methods [4], [10], [11], [12], [13] and the machine learning methods such as color clustering [6], [14], statistical modeling [2], [3], [5], [7], [15] and so on. Global threshold method, such as the method proposed by Otsu [11], is effective to binarize document images or text images with high contrast and uniform background. Approaches proposed by Niblack [10] and Sato et al. [12] compute the local threshold, one based on local mean and variance and another using directional filters to compute the probability of each pixel being on a text stroke. However, threshold methods would fail when the text and background have similar textures or colors. Gllavata et al. [6] and Song et al. [14] use K-means color clustering to separate text from image. The performance of color clustering method is dependent on the color consistency and also sensitive to noise and text resolution. Chen et al. [2], [3], Gao et al. [5] and Ye et al. [15] adopt different approaches to build models (GMM or MRF) for pixels to extract text from images.

Threshold methods don't make full use of the structure of the text characters whereas the model trained on one dataset might not generalize well on other datasets. Actually, as text is a special kind of object, we could make use of the various object segmentation methods. Interactive image segmentation which needs the user to give certain hard constraints is becoming more and more popular, and graph cut has been widely used in this area [16], [19]. As text has some unique characteristics, we could automatically get some hard constraints for graph cut. Therefore, in this paper we propose an adaptive graph cut based text binarization method. In order to adaptively binarize text from complex uneven background, we firstly split the text region images into several sub-images on which graph cut is applied. Considering the unique characteristics of the text, a new method is proposed to automatically classify some pixels with high confidence, served as the hard constraints seeds for graph cut to spread the seeds into the whole sub-image so as to binarize (segment) the sub-image.

The rest of this paper is organized as follows. Section Ⅱ details the proposed method. Experiments are presented in Section Ⅲ and conclusions are drawn in Section Ⅳ.

II. THE PROPOSED METHOD

A. Method Overview

The proposed method consists of three stages: 1) adaptively splitting text region image into several sub-images, 2) determining the polarity of the text region and automatically finding some text and background pixels with high accuracy, based on which graph cut is used to segment the sub-images, and 3) merging the segmented sub-images. The flowchart of the proposed method is shown in Fig. 1 and the corresponding detailed description will be presented in the following sections.

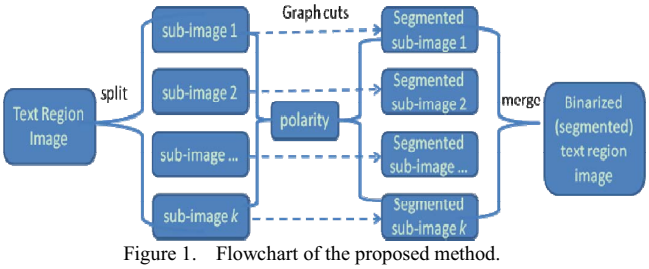

Figure 1. Flowchart of the proposed method.

B. Splitting

1) Finding the Seeds： In order to split the input text images into several sub-images, we need to find at least one

978-0-7695-4661-2/12 $26.00 © 2012 IEEE
DOI 10.1109/DAS.2012.15

seed image (character image), which is the basis for further splitting. Details of the process and the corresponding analysis to find the seeds are given below.

- Based on the fact that characters in images usually have distinct contrast in edges or colors to their background for reading, the edge map of the original image is firstly detected. In this paper the Canny edge [1] detector is used due to its continuity and integrity. Then connected components analysis (CCA) is operated on the edge image to get the connected components (CCs) as candidate seed areas.
- Although most plain background is suppressed in the edge image, there still exist some edges from background which sometimes link several character areas together. As characters have some similar features, we could further analyze the geometric characteristic of the CCs to filter out those disturbing CCs.
- Finally, we calculate the average width and height, according to which the final seed images are acquired.

2) Splitting Images By Seeds： Next, the text image will be split into sub-images according to the seed images. The splitting stage is the premise for later stages, which means if we lose text information at this stage, the final extracted text information will definitely be incomplete. Thus, we strictly obey the principle that all the non-seed regions would be split according to the average width and height. Edge density is then used to exclude those non-character areas which have few edges. Fig. 2 shows the overall splitting procedure.

Figure 2. Stages of splitting method.

In Fig. 2, all the characters in the left original image are considered as seeds and these seeds are the final split sub-images, whereas on the right side of (b) not all the characters are detected as seeds, so further segmentation is needed to get the final split sub-images in (c). Images in (d) are the Otsu [10] binarization results on the whole input images, and images in (e) are the merging results of each binarized sub-images using Otsu [10] without any de-noising. Comparing results in (d) and (e), we find that even without any de-noising on the sub-images, the merging result has less noise, suggesting that some noise caused by uneven background is removed during the splitting process.

Figure 3. Illustration of the polarity criterion.

3) Estimating the Polarity： As the following step needs the polarity of the image, we propose a new and effective method to decide the polarity of the image by fusing the polarities of sub-images. For each preliminary binarized sub-image, stroke widths of the image, the dilated image and the eroded image are calculated based on which the polarity is estimated. The same mask is used for the dilation and erosion operation. The judging criterion is detailed as:

$$Foreground = \begin{cases} 1(white) & if\,(Stroke\_width_{dilate} > Stroke\_width_{origin}\ \&\& \\ & Stroke\_width_{erode} < Stroke\_width_{origin}) \\ 0(black) & if\,(Stroke\_width_{dilate} < Stroke\_width_{origin}\ \&\& \\ & Stroke\_width_{erode} > Stroke\_width_{origin}) \end{cases} \quad (1)$$

where $Stroke\_width_{origin}$, $Stroke\_width_{dilate}$, $Stroke\_width_{erode}$ represent the stroke width of the original image, the dilated image and the eroded image respectively. This criterion means that if the stroke width gets thinner after erosion and wider after dilation, the foreground of the binarized image is 1 (white), and vice versa. Fig. 3 illustrates the process. We can see that as images in the first two columns have black foreground, their stroke width becomes wider after erosion and thinner after dilation accordingly, whereas in the last two columns stroke width becomes thinner after erosion and wider after dilation as a result of the white foreground. Edge images are used to calculate the stroke width due to its equal response to images with either white or black foreground. After the polarity of each sub-image is calculated, the final polarity is determined by the votes of sub-image polarities.

C. Segmenting Sub-images

Given a sub-image, the goal is to extract the foreground (character strokes) from the background. The interactive image segmentation technique, graph cut, is used. While interactive image segmentation needs the user to give hard constraints by indicating several pixels (seeds) definitely to be the part of the foreground or that of the background [16], we propose to automatically acquire these hard constraints based on the characteristics of the strokes. The details of the algorithm are given below.

1) Graph Cut： In this section we describe how to use graph cut in the context of our segmentation method. An undirected graph $G = \{V, E\}$ is composed of nodes (vertices V) and undirected edges (E) that connect these nodes. Each edge in the graph is assigned a nonnegative weight w_e as the cost. There are two terminals, the background and the foreground terminals. A cut is a subset combination of the edges so that each node is assigned to either of the terminals. The cost of a cut is usually defined as the sum of the costs of the edges [16]:

$$|C| = \sum_{e \in C} w_e. \quad (2)$$

The cost function that we use as the soft constraints for segmentation needs to include both region and boundary properties [16]. Let P be all the pixels in the image and N be a set of pairs $\{p,q\}$ of the neighboring pixels in P. $L=\{L_1,L_2...,L_p...\}$ is a binary vector whose components L_p specify the labels of pixel p in P. Each L_p could be either 1 (foreground) or 0 (background). The cost function $E(L)$ for each segmentation L is defined as [16]:

$$E(L)=\lambda R(L)+B(L) \tag{3}$$

where

$$R(L)=\sum_{p\in P}R_p(L_p) \tag{4}$$

$$B(L)=\sum_{\{p,q\}\in N}B_{\{p,q\}}*\delta(L_p,L_q) \tag{5}$$

and

$$\delta(L_p,L_q)=\begin{cases}1 & if\ L_p\neq L_q\\ 0 & otherwise.\end{cases} \tag{6}$$

The coefficient $\lambda>=0$ is a trade-off factor between the region cost $R(L)$ and the boundary cost $B(L)$. $R(L)$ measures the individual penalties for labeling the pixel p as foreground or background and each pixel has two cost weights $R_p(1)$ and $R_p(0)$, corresponding to linking cost to foreground and background respectively. $B(L)$ reflects penalties for discontinuity between neighboring pixels. $B_{\{p,q\}}$ should be large when p and q are similar and vice versa. Reference [16] proved that the minimum cut would give a segmentation minimizing (3) among all segmentations satisfying the given hard constraints. In this paper, the min-cut/max-flow algorithms in [17] is used. The algorithm for hard constraints, $R(L)$ and $B(L)$ would be given in the following sections.

2) Automatically Acquiring Hard Constraints： While interactive image segmentation needs the user to give hard constraints by indicating several pixels definitely to be the part of the foreground or that of the background [16], we propose to automatically acquire these hard constraints based on the characteristics of the strokes. As we can see in Fig. 4, character strokes have the following characteristics: 1) for the same character, the strokes have relatively similar width; 2) pixels inside a stroke have the same color or intensity; and 3) pixels near the strokes usually have different colors or intensities from that of the strokes for reading. Based on the above observations, the hard constraints seed pixels are acquired as follows:

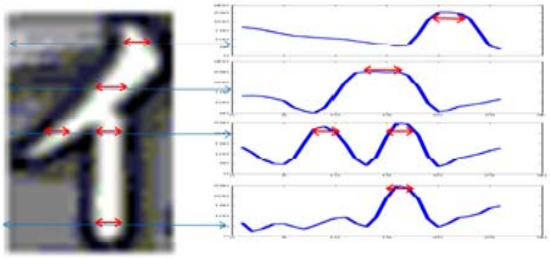

Figure 4. Intensity waves of character image.

(a) (b) (c) (d) (e)

Figure 5. Illustration of the seeds aquiring process: (a) Original sub-image. (b) Label image before seeds growing. (c) Label image after seeds growing. (d) Label image after mean shift clustering. (e) Segmentation result by graph cut.

- Scan the image horizontally and vertically for the intensity changes. In Fig. 4, intensity waves for four horizontal lines are drawn on the right, each corresponding to the horizontal intensity changes on the left image. According to the intensity waves, we detect the stroke-like crests and troughs as candidate strokes and background;
- If the polarity of the image is 1 (the foreground of the image is white), the crests with width bigger than 1 and smaller than half of the width of the sub-image, and with intensity above average intensities are further labeled as candidate strokes seeds whereas the troughs are labeled as candidate background seeds and vice versa. As shown in Fig. 4, since the polarity is 1, the crests labeled in red are candidate strokes seeds;
- Seeds growing are applied on the candidate pixels. For each pixel in the image, if there exists a seed pixel in its neighborhood and their intensity difference is within a certain value (set to 10), assign the pixel the same label with the seed pixel;
- To improve the reliability of the seed pixels, mean shift clustering [20] is used to cluster the candidate foreground and background pixels using color features, and only those pixels whose distance from the clustering center is within a predefined value (set to 25) are chosen as hard constraint seeds. Fig. 5 shows the hard constraints acquiring process where green pixels are for foreground pixels and red ones are for background pixels.

3) Region and Boundary Cost Algorithms： In this section we detailed the region and boundary cost algorithms. Each pixel of the image is a node in the graph and edges are composed of the standard 8 neighborhood system. As mentioned before, region cost $R(L)$ should reflect the individual penalties for assigning pixel p to foreground or background. To this end, we assume that if the color of the pixel is nearer to that of the foreground seed pixels, the cost $R_p(1)$ should be smaller and $R_p(0)$ should be larger and vice versa. The details are given below.

- Clustering the foreground and background hard constraint seeds by mean shift clustering method [20]. Assume we get n foreground and m background clustering centers $Center\{fore\}_n$ and $Center\{back\}_m$.

- For each pixel p, calculate the distance to each clustering center, represented as $Dist\{back\}_m^p$ and $Dist\{fore\}_n^p$ respectively.
- $R_p(1)$ and $R_p(0)$ are defined as follows:

$$R_p(1) = \min\{Dist\{fore\}_k^p\}, k = 1,2,...n \quad (7)$$

$$R_p(0) = \min\{Dist\{back\}_k^p\}, k = 1,2,...m. \quad (8)$$

Considering the local discontinuity penalty $B(L)$, it could decrease as a function of the distance between pixels p and q [16]. In this paper we use the function:

$$B_{\{p,q\}} = \exp(-\frac{(color_p - color_q)^2}{2\sigma^2}) \quad (9)$$

where $color_p$, $color_q$ represent the color features, and σ is a scale factor fixed to be 0.25 in the experiments.

Since we have defined the region cost and the boundary cost, we use min-cut/max-flow algorithms in [17] to segment the image. An example of the segmented sub-image is presented in Fig. 5.

D. Merging Segmented Sub-images

Given the segmented sub-images with white pixels for text and black pixels for background, all we need to do is linking all the segmented sub-images on the black background.

III. Experiments and Results

A. Data Set and Evaluation Methods

Figure 6. Some examples from the dataset.

As there are no suitable public data sets for text binarization from video text images, we collect 586 text region images from videos as our data set. The data set consist of images of different languages including Chinese, Japanese and numbers. These images vary a lot in character sizes, fonts, colors and background complexities. Fig. 6 shows some examples from the data set.

Two evaluation methods are used to verify the binarization performance. One is the character extraction rate (CER). The other is the character recognition rate (CRR). They are defined as:

$$CER = N_{segment} / N \qquad CRR = N_{recognize} / N \quad (10)$$

where $N_{segment}$ is the number of characters completely extracted without lost strokes or connecting to background, $N_{recognize}$ is the number of characters correctly recognized by an OCR engine, and N is the total number of the characters.

B. Performance of the Polarity Estimating Method

As acquiring the foreground and background seeds needs the polarity of the image, the proposed polarity estimating method is firstly evaluated. Among the 586 images being tested, only 4 images, most of which have hollow characters and complex uneven backgrounds, are classified incorrectly, reaching a classification rate of 99.32%.

C. Evaluation of Split-merge Technique

The split-merge technique plays an important role in the overall performance of the proposed method. As not only is it the basis for polarity estimation, but it also improves the effectiveness of the graph cut segmentation results. For comparison, we apply the proposed method on the whole image to get hard constraints seeds for graph cut. The CER and CRR are 88.44% and 73.86% respectively, compared to 95.03% and 78.34% of the proposed method.

D. Evaluation of the Validity of the Hard Constraints

As the hard constraints are essential for the following semi-supervised image segmentation, we need to testify the validity of the constraint seeds we got. The semi-supervised image segmentation methods in [18] and graph cut in this paper are applied with the same constraint seeds acquired by our approach. The CER and CRR results are shown in Fig. 7. From the results we can see that both semi-supervised classification methods achieve much better result compared to Otsu [11], verifying the effectiveness of the hard constraints acquired by our approach.

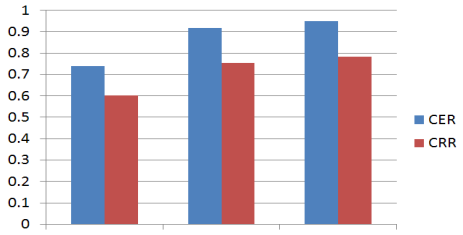

Figure 7. CER and CRR results.

E. Comparing Results Using Different Methods

The proposed binarization method is compared with four methods proposed by Otsu [11], Niblack [10], Chen et al. [4], and Lyu et al. [9] respectively. The reason to choose these methods is that Otsu is a simple but classical method widely used in many text segmentation applications; Niblack's method is proved to be one of the best local threshold methods; Chen's method is a variant of Niblack's method by selecting the window size adaptively; and Lyu's method shows good performance on their data set. In experiments, we ran Niblack's method with $k = -0.2$ and window size $r = 30$, while for Chen's method k is fixed to be 0.6 and T_σ is fixed to be 60. Commercial OCR software ABBYY FineReader 9.0 [21] is used for recognition. Table Ⅰ shows the CER and CRR results. Some binarzation results of the above five methods are displayed in Fig. 8. As we can see in Fig. 8, the unsatisfactory binarization results of the traditional threshold methods illustrate the complexity of the background.

Figure 8. Comparing results of the five methods.

The OCR engine performs badly when the binarization input has much noise as in the case of Otsu, Niblack and Chen's methods. The results demonstrate that the proposed method performs much better than the other four methods both in CER and CRR. The main advantages of our approach lie in: 1) local adaptive split-merge technique which not only excludes some noise caused by uneven background but also improves the performance of graph cut; 2) the validity and effectiveness of the automatically acquired hard constraints for graph cut; and 3) graph cut based semi-supervised classification method which could spread the seeds into the whole sub-image.

TABLE I. RESULTS OF DIFFERENT METHODS

Evaluation Methods	Text Binarization Methods				
	Lyu et al. [9]	*Otsu [11]*	*Niblack [11]*	*Chen et al. [4]*	*Our approach*
CER (%)	83.64	74.02	62.90	68.09	**95.03**
CRR (%)	65.24	60.38	55.43	58.32	**78.34**

IV. CONCLUSIONS

In this paper, an adaptive graph cut based text binarization method is proposed. Due to split-merge technique, the proposed approach is local adaptive, which could adaptively handle text regions with uneven background. Moreover, as character strokes have some unique characteristics, a new method is proposed to automatically give effective hard constraints for graph cut to spread these hard constraints into the whole sub-image. Experimental results demonstrate the validity of our method. In the future, more robust and effective hard constraints and more unique semi-supervised learning algorithm for text binarization could be further researched.

ACKNOWLEDGMENT

This work is supported in part by the National Natural Science Foundation of China under Grant No. 60802055, No. 60933010 and No. 60835001.

REFERENCES

[1] J. Canny, "A Computational Approach to Edge Detection," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp. 679-698, 1986.

[2] D. Chen, J.M. Odobez, and H. Bourlard, "Text segmentation and recognition in complex background based on markov random field," Pattern Recognition, 4:40227, 2002. ISSN 1051-4651.

[3] X. Chen , J. Yang , J. Zhang and A. Waibel, "Automatic detection and recognition of signs from natural scenes," IEEE Trans. Image Process., vol. 13, p.87-99 , 2004.

[4] X. Chen and A. Yuille, "Detecting and Reading Text in Natural Scenes," Proc. Int'l Conf. Computer Vision and Pattern Recognition, pp. II:366-373, 2004.

[5] Gao, J. and Yang, J., "An Adaptive Algorithm for Text Detection from Natural Scenes," Proc. Int'l Conf. Computer Vision and Pattern Recognition, vol. 2, pp.84 - 89 , 2001.

[6] J. Gllavata, R. Ewerth, T. Stefi, and B. Freisleben, "Unsupervised text segmentation using color and wavelet features," Image and Video Retrieval, pages 1967–1967, 2004.

[7] M. Li, M. Bai, C. Wang, B. Xiao, and Y. Lv, "Conditional random field for text segmentation from images with complex background," Pattern Recognition Letters, 2010. ISSN 0167-8655.

[8] R. Lienhart and A. Wernicke, "Localizing and segmenting text in images and videos," IEEE Trans. Circuits Syst. Video Technol., vol. 12, no. 4, pp.256 - 268 , 2002.

[9] M. R. Lyu , J. Song and M. Cai, "A comprehensive method for multilingual video text detection, localization, and extraction," IEEE Trans. Circuit and Systems for Video Technology, vol. 15, p.243 , 2005.

[10] W. Niblack. An Introduction to Digital Image Processing. Prentice-Hall, 1986.

[11] N. Otsu. A threshold selection method from gray-level histograms. Automatica, 11: 285–296, 1975.

[12] T. Sato, T. Kanade, E. K. Hughes, and M. A. Smith, "Video OCR for digital news archives," Proc. IEEE Int. Workshop on Content-Based Access of Image and Video Database (CAVID'98), pp.52 - 60 , 1998.

[13] J. Sauvola, T. Sepp\änen, S. Haapakoski, and M. Pietik\äinen, "Adaptive document binarization," Proc. Int. Conf. Document Analysis and Recognition, vol. 1, pp.147 - 152 , 1997.

[14] Y. Song, A. Liu, L. Pang, S. Lin, Y. Zhang, and S. Tang, "A Novel Image Text Extraction Method Based on K-means Clustering," Seventh IEEE/ACIS International Conference on Computer and Information Science, 14-16 May 2008, pp. 185-190.

[15] Q. Ye, W. Gao, and Q. Huang, "Automatic text segmentation from complex background," IEEE International Conference on Image Processing, Singapore, Vol.5, pp.2905-2908, Oct. 2004.

[16] Y.Y. Boykov and M.P. Jolly, "Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in n-d Images," Proc. IEEE Int'l Conf. Computer Vision, pp. 105-112, 2001.

[17] Y. Boykov and V. Kolmogorov, "An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 26, no. 9, pp. 1124-1137, Sept. 2004..

[18] Shiming Xiang, Feiping Nie, Changshui Zhang, "Semi-Supervised Classification via Local Spline Regression," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 32, no. 11, pp. 2039-2053, 2010.

[19] C. Rother, V. Kolmogorov, and A. Blake, "`GrabCut'—Interactive Foreground Extraction Using Iterated Graph Cuts," *ACM Trans. Graphics,* vol. 23, no. 3, pp. 309-314, 2004.

[20] K. Fukunaga and L.D. Hostetler,"The estimation of the gradient of a density function, with applications in pattern recognition," IEEE Trans. Information Theory, vol. 21, pp. 32-40, 1975.

[21] ABBYY Finereader 9.0. http://www.abbyy.com/.