2013 IEEE International Conference on Systems, Man, and Cybernetics

A Novel Approach for Binarization of Overlay Text

Zhike Zhang Weiqiang Wang
University of Chinese Academy of Sciences, Chinese Academy of Sciences, Beijing, China
zhangzhike11@mails.ucas.ac.cn wqwang@ict.ac.cn

***Abstract*—In this paper, we presents a new binarization approach to extract text pixels from complex background in video frames. The binarization computation is a crucial step for video text recognition, which can greatly increase the recognition accuracy of an OCR software. The proposed approach consists of four phases. First, the text polarity is determined, i.e. light text with dark background or dark text with light background. Then the pixels in the given image are clustered into K clusters using the K-means algorithm in the RGB color space and the text cluster is selected based on the text polarity. Further, the MRF Model is exploited to get the binarization result. Finally, the result is further refined by the Log-Gabor filter. The Experimental results on a large dataset show that the significant gains have been obtained according to the segmentation performance on the pixel level as well as the OCR accuracy.**

***Index Terms*—video text, binarization, K-means, MRF**

I. Introduction

With the development of multimedia technology, the amount of video data available on the WWW is constantly growing. There is an increasing demand for the research of video indexing and summarization. Since text in videos contains rich semantic information, accurate recognition of it can significantly contribute to video indexing and retrieval. Text in videos is generally classified into two types: the overlay text which is added artificially during the editing process and the scene text existing in the real-world scenes. This paper focuses on the former type. The overlay text is usually closely related to the video content, so it can be reliably used for video indexing and retrieval. However, due to the diverse difficulties, *e.g.*, complex background, low resolution, unknown text color, the extraction of overlay text in videos is not as easy as the text extraction from scanned documents which has been well solved. The whole process of video text recognition generally contains four steps: detection, localization, extraction, and recognition [1]. The detection step aims to roughly identify text regions and non-text regions. Then the localization step determines the accurate positions of text regions. Further, the text extraction step binarizes pixels in the text regions into text pixels and background pixels. Finally, the recognition step is used to convert the binarized pixel text into the encoded text (unicode text), which is the work done by the OCR software. This paper mainly discusses the text extraction (text binarazition).

We classified the text binarization methods into four classes: the first is the threshold-based method where the local threshold and the global threshold can be used; the second uses diverse clustering techniques (K-means, mean-shift, gaussian mixture modeling) to segment text pixels; the third extracts text based on some stroke-based filters; the last is based on a special MRF model which can be solved by min-cut/max-flow algorithms [24], and here we call it the graph-cut method. The features for text binarization mainly include intensity, color and stroke. The threshold-based methods based on intensity are usually simple and efficient, but they often produce poor result due to complex background. The color-based clustering methods are more effective but still introduce noise if the background has similar color with text. The stroke-like filters are often used to enhance text pixels and to suppress background pixels.

In this paper, we propose a new approach in the graph-cut framework for overlay text binarization, which can integrate more features to obtain better result. First, we determine the text polarity, *i.e.*, judge that the text pixels are light or dark. Subsequently, we use K-means clustering and choose the text cluster with the help of Otsu's global threshold based on the fact that it can extract the majority of text pixels even with background noises. Further, we use the graph-cut method to get the binarization result and apply stroke-like (Log-Gabor) filters to enhance the result.

The rest of the paper is organized as follows. Section 2 presents the related work. In Section 3, the proposed approach is described in details. The experimental results are reported in Section 4. Section 5 concludes the paper.

II. Related Works

Among four kinds of methods mentioned previously, the traditional threshold-based binarization can be further categorized into two categories: the global-threshold method [2][3] and the local-threshold method [4]. Besides, multi-threshold and multi-stage threshold [5] are common techniques to improve the performance. Recently, Zhou *et al* [6] presented to select the local thresholds based on edge information. Bolan *et al* [8] proposed a thresholding method using the contrast defined by the local intensity maximum and minimum. Both of them obtained superior performance for some applications. The threshold-based methods mainly process the gray-level image but the color information is also a crucial clue for binarization. Although these methods perform well in some cases, they usually fail when applied in complex background. An review of the threshold-based binarization can be seen in [9].

The clustering is another technique for binarization. The overlay text often has an uniform color. It is appropriate to

978-1-4799-0652-9/13 $31.00 © 2013 IEEE
DOI 10.1109/SMC.2013.726

cluster all pixels into several color classes and choose one of them as text pixels. Leydier *et al*. [10] used a serialization of the K-means algorithm for ancient documents with heavy defects. Two color spaces RGB and HSL (Hue, Saturation, Luminosity) were chosen to handle different degradations. However, this method needs a user interface to define the number of logical classes and select the multiple color samples as original centers of the logical classes. Fu *et al*. [11] used the K-means clustering algorithm in YCbCr color space to generate several layers, and heuristic rules were employed to select text layer. Huang *et al*. [12] segmented a text line into single characters and then performed K-means clustering with the combination of Euclidean distance and the Cosine similarity in RGB space in single character. They defined a method based on some values called cen-deviation and cen-mean to select text cluster. Kita *et al*. [13] also used K-means clustering for binarization. However, to deal with multicolored characters they tested $2^k - 2$ combinations of k clusters to select the optimal combination as the binarization result. The selection is based on the degree of "character-likeness" calculated by a well trained SVM.

The third class uses the stroke-based filters to retrieve text regions. Chen *et al*. [14] proposed two groups of asymmetric gabor filters which can efficiently extract the orientation and scale of the stripes in a video image. These features were used to enhance contrast only at those edges most likely to represent text. However, this method is sensitive to text size. Li *et al*. [15] proposed two-threshold method using stroke edge filter, which can effectively identify stroke edges in subjective evaluation. First a new stroke edge filter is applied to obtain stroke edge map. Then a two-threshold method based on the improved Niblack thresholding technique is utilized to identify stroke edges. Those pixels between the edge pairs above the high threshold are collected to estimate the representative of stroke color, so that stroke pixels are further extracted by computing the color similarity. Finally some heuristic rules are devised to integrate stroke edge and stroke region information to obtain better segmentation results. Some researchers [16][17] applied the Log-Gabor filters for text extraction. The Log-Gabor filters can catch the spatial information as well as the frequency information in a certain direction and circumvent some limitation that the Gabor filters suffer from. Both of them used only two directions for the filter, *i.e.*, the horizontal and the vertical one. However, Mancas-Thillou *et al*. [16]. estimated the important parameters of frequency and bandwidth for filters by a simple calculation of stroke thickness based on skeleton and an OCR engine's recognition feedback, while Huang *et al*. [17] just picked parameters experimentally.

Recently, the last class, the Markov Random Field (M-RF) based binarization has been extensively applied. Wolf *et al*. [18] proposed binarization in an energy minimization framework and applied a less powerful and computationally expensive simulated annealing(SA) for energy minimization. The MRF based binarization was proposed and applied for the hand-device captured document images [19], where the authors first used the thresholding based technique to produce a binary image and then applied the graph-cut algorithm to remove noises and smooth binarization output. Anand Mishra *et al*. [20] used the MRF model for binarization of natural scene text. The method is robust to the variations of foreground and background colors, since it exploits the GMM for modeling color distributions, and the result is refined by using an iterative graph-cut scheme.

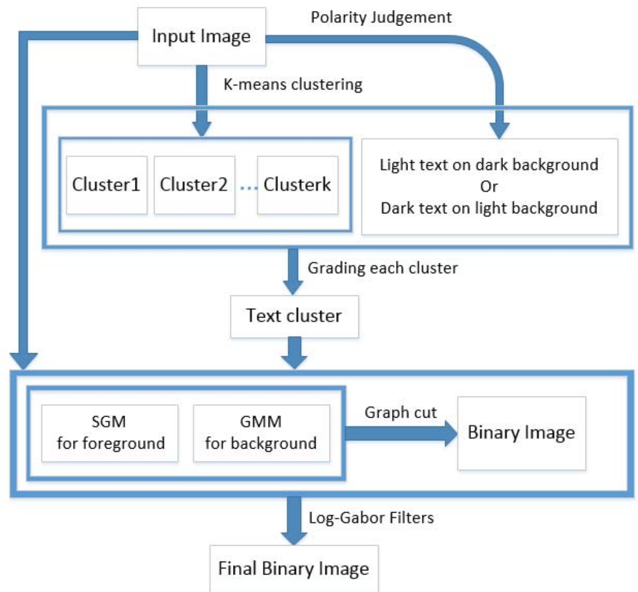
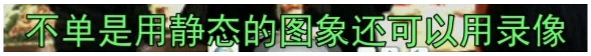


Fig. 1. The overview of our approach

Besides the methods mentioned above, the method based on the Convolutional Neural Network (CNN) [21][22] is worthy of our attention, due to its suitability and effectiveness for many problems. There are also some techniques which can improve recognition accuracy, *e.g.*, multi-frame integration [17], segmenting a text line into characters before binarization [12][22][23].

III. The Proposed Approach

The framework of the proposed approach is shown in Figure 1. Given a text line localized in videos with complex background, the output of our system is a binary image that corresponds to overlay text and background. It consists of four main phases. First, the polarity of the overlay text is determined, *i.e.*, judging whether it is light text on dark background or vice-versa. This phase can be skipped if we know the polarity based on prior knowledge. Then the pixels in the given text line region are clustered into k classes using the K-means algorithm in the RGB color space, and one of them is selected as text cluster based on the text polarity. Further, the MRF Model is exploited to get the binarization result. Finally, we utilze the Log-Gabor filters to refine the binary image obtained. The technical details of the proposed approach are described in the following subsections.



(a) Original image

(b) Binary image: light text on dark background

(c) Binary image: dark text on light background

(d) After removing boundary connected component for (b)

(e) After removing boundary connected component for (c)

Fig. 2. An example for polarity judgement: the scores of (d) and (e) by OCR engine's recognition feedback are 0.55 and 0.21 respectively, so the text polarity is light text on dark background

A. Polarity Judgement

The researchers have proposed some methods for polarity judgement. Zhou *et al.* [6] used the grayscale histogram to achieve this goal. Their system processes the histogram with a Gaussian filter recursively until only two peaks are obtained. They assumed that the background always corresponds to the higher peak in the output histogram. However, this is not always true in practice. Instead, Yang *et al*. [7] proposed a skeleton based method. They first exploited the Otsu's method to obtain two candidate binary images, and the one is considered as the final binary image with correct text polarity, if its skeleton has fewer pixels on the boundaries.

For a given text line color image, the proposed method first converts it into an intensity image. Similar to Yang's method [7], we also apply Otsu's method [2] to create two candidate binary images. However, we choose the one which has a higher confidence score feedback in the recognitin computation of an OCR engine. As shown in Figure 2, for each candiate binary image, all connected components consisting of white pixels are removed as noises if they are connected to the boundary of text line region, since we assume the white pixels stand for text pixels in binary image. Then, the processed binary images are fed to the OCR engine, and the one with the higher confidence score is selected as the real binary image with correct text polarity, denoted by I. It is exploited to help select the text cluster in the subsequent K-means clustering.

B. K-means Clustering

In this phase, we use the K-means clustering in the RGB space to cluster the pixels of the image into k classes, and thus gets k binary images if we treat each class as foreground and the others as background. For each class, we use the following formula,

$$Score(i) = \alpha\frac{|I_f \cap C_i|}{|C_i|} + \beta(1 - N(v_i)) + \gamma(1 - N(b_i)), \tag{1}$$

Fig. 3. The score of each binary image corresponding to its cluster. From top to bottom: original image, cluster1: 0.9403, cluster2: 0.1495, cluster3: 0.7786, cluster4: 0.4908. So we take the first as text class.

to compute the score for it, and the one with the highest score is selected as the text class. Here i is the index of classes. I_f denotes the set of foreground pixels in binary image I obtained in the polarity judgement. C_i denotes the set of pixels in cluster i. $|\cdot|$ stands for the size of the set. $N(\cdot)$ denotes the normalization computation by dividing the maximum value of the corresponding variable. v_i is the variance of the number of pixels of connected component in class i, i.e.,

$$v_i = \frac{1}{N_i}\sum_j [area(l_{ij}) - \frac{1}{N_i}(\sum_j area(l_{ij}))]^2, \tag{2}$$

where N_i is the number of connected components in cluster i and $l_{ij} \subset C_i$ denotes connected component. b_i is the number of boundary pixels in class i:

$$b_i = \sum_{p \in C_i} BoundaryTest(p) \tag{3}$$

$$BoundaryTest(p) = \begin{cases} 1, & p \ is \ a \ boundary \ pixel, \\ 0, & otherwise. \end{cases} \tag{4}$$

According to the formula 1), we expect the chosen text class has the characteristics in three aspects. First, it should have more consistent pixels with the set of foreground pixels I_f in binary image I obtained in the polarity judgement. It should be noted that the Otsu's thresholding algorithm rarely misses text pixels, although it is possible that some background noises are introduced due to some background pixels with similar intensity to text pixels. Second, the variance of the number of the pixels corresponding to connected components in the text class should be small. Third, the text class should have few pixels on the boundary of text regions. Parameter α, β, γ are used to reflect the weights of three factors. An example in this phase is shown in Figure 3.

C. Graph-Cut Based Binarization

Through the above computation, a binary image $\hat{I}$ corresponding to text class is obtained. Then the MRF model is applied to remove some noises and smooth this result. We treat each pixel of image $\hat{I}$ as a node in a Markov Random

Field and formulate the problem of binarization in the energy minimization framework. The energy function is defined as

$$E(\mathbf{x}, \boldsymbol{\theta}, \mathbf{c}) = E_i(\mathbf{x}, \boldsymbol{\theta}, \mathbf{c}) + E_{ij}(\mathbf{x}, \mathbf{c}), \tag{5}$$

where $\mathbf{x} = \{x_1, x_2, ..., x_n\}$ is a set of labels (1 for foreground, 0 for background) for all pixels, $\boldsymbol{\theta}$ denotes the set of model parameters which is learnt from the foreground/background color distributions and $\mathbf{c} = \{c_1, c_2, ..., c_n\}$ denotes the RGB value of pixels. The unary term can be expressed as

$$E_i(\mathbf{x}, \boldsymbol{\theta}, \mathbf{c}) = -\sum_i \log p(x_i|c_i), \tag{6}$$

and the pairwise term is defined as

$$E_{ij}(\mathbf{x}, \mathbf{c}) = \lambda \sum_{(i,j)\in \mathbf{N}} [x_i \neq x_j] exp(-\eta ||c_i - c_j||^2). \tag{7}$$

Here $p(x_i|c_i) = p(x_i, c_i)/p(c_i) = p(c_i|x_i)p(x_i)/p(c_i)$. For a given c_i, we can ignore $p(c_i)$ and we assume that the foreground and background share the same prior probability, *i.e.*, $p(x_i = 1) = p(x_i = 0)$. Under these constraints, we do not distinguish $p(x_i|c_i)$ and $p(c_i|x_i)$. $\mathbf{N}$ denotes the four neighborhood system in the image. The pairwise term or the smoothness term only exists when pairwise pixels in the neighborhood system share different labels. The more similar the colors of pairwise pixels are, the more cost it takes to mark them as different labels.

In order to learn the probability $p(c_i|x_i)$, we use a Single Gaussian Model (SGM) to characterize the color distribution of foreground pixels, since text pixels usually have the uniform color, and use the Gaussian Mixture Model (GMM) to characterize the color distribution of background pixels due to the complexity of background, *i.e.*,

$$p(c_i|x_i = 1) = p^f(c_i) = \mathcal{N}(c_i|\mu^f, \Sigma^f), \tag{8}$$

$$p(c_i|x_i = 0) = p^b(c_i) = \sum_{k=1}^{K} \pi_k^b \mathcal{N}(c_i|\mu_k^b, \Sigma_k^b). \tag{9}$$

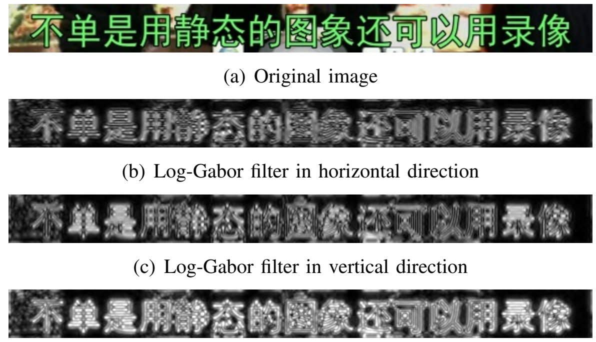
For simplicity and efficiency, we use a GMM-like approach instead in practice:

$$p(c_i|x_i = 0) = p^b(c_i) = \max_{k=1}^{K} \mathcal{N}(c_i|\mu_k^b, \Sigma_k^b). \tag{10}$$

Here $\mathcal{N}(\cdot)$ denotes the Gaussian distribution,*i.e.*,

$$\mathcal{N}(c|\mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} (det(\Sigma))^{\frac{1}{2}}} \exp\{-\frac{1}{2}(c-\mu)^T \Sigma^{-1} (c-\mu)\}. \tag{11}$$

The parameters of foreground and background distributions are estimated from the corresponding pixels of binary image $\hat{I}$. We use the graph cut [24] algorithm to solve this energy minimization problem.



(a) Original image

(b) Log-Gabor filter in horizontal direction

(c) Log-Gabor filter in vertical direction

(d) The fusion of two directions

Fig. 4. One example of Log-Gabor filters.

D. Log-Gabor Filters

Although the graph-cut computation based on color information can remove some noises, it cannot remove noisy background pixels that have similar color with text yet. The Log-Gabor filter can catch spatial information as well as frequency information in a certain direction and circumvent some limitation that the Gabor filter suffers from. Thus, we use the Log-Gabor filters which can utilize stroke information to further refine the output. In the polar coordinates, the Log-Gabor filter in frequency domain can be defined as $H(f, \theta) = H_f \times H_\theta$, and the radial component $H_f = exp\{\frac{-[\ln(f/f_0)]^2}{2[\ln(\sigma_f)]^2}\}$, the angular component $H_\theta = exp\{\frac{-(\theta-\theta_0)^2}{2\sigma_\theta^2}\}$. Here f_0 denotes the central frequency, θ_0 denotes the filter direction, σ_f defines the radial bandwidth B in octaves with $B = 2\sqrt{2/\ln 2} * |ln(\sigma_f)|$ and σ_θ defines the angular bandwidth $\Delta\Omega$ with $\Delta\Omega = 2\sigma_\theta\sqrt{2\ln 2}$.

In practice, we use two directions for the filter, *i.e.*, $\theta_0 \in \{0, \pi/2\}$. The filter outputs are denoted by R_h, R_v which correspond to the horizontal and the vertical directions. Then they are merged based on $R = \sqrt{R_h^2 + R_v^2}$ to generate the final result R. An example is shown in Figure 4. For the two filters, we use a fixed angular bandwidth $\Delta\Omega = \pi/2$, f_0 is estimated by the method in [16], and σ_f is set to 0.65. The Log-Gabor filters are applied to remove those connected components whose pixels have small average response values. Before applying the Log-Gabor filters, we first remove some boundary noises by dam point labeling and inward filling [1].

IV. Experimental Evaluation

Since there is no standard dataset for overlay text in videos, we have designed a powerful tool, and it can automatically extract a large number of text lines from videos and generate the binary groundtruth images on pixel level simultaneously. To be more persuasive, we have set up an evaluation dataset consisting of 9549 images including 87282 characters, which is collected from a variety of sources including television series, movies, cartoon films and lectures, and the overlay text in them covers different color, font and size. We focus on Chinese character recognition for it is harder than Latin alphabet. Since the extraction computation have nothing to do with languages, and the recognition performance more depends

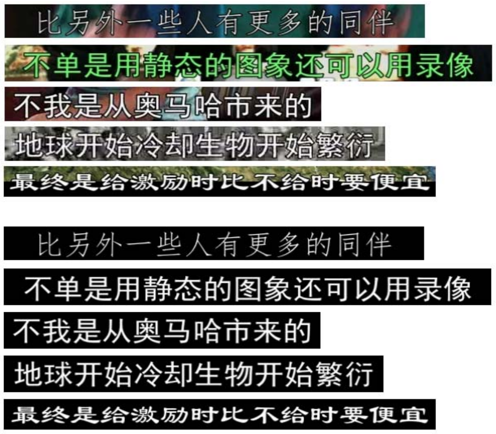


Fig. 5. Some sample images and their groundtruth images for experiment in this work

on the OCR engine, the proposed approach should be suitable for other languages, if an universal OCR engine is used, such as Tesseract which can recognize tens of languages. Some sample images we collect for experiments are shown in Figure 5.

To evaluate the performance of the proposed binarization algorithm, we compare it with the well-known threshold-based binarization techniques like Otsu [2], Kittler [3], Niblack [4]. For fair comparison we always make these methods binarize images using correct polarity. During the experiments, K in K-means clustering is set 3 or 4, thus 2 or 3 for GMM-like in Equation 10. We set $\alpha = 0.5, \beta = 0.3, \gamma = 0.2$ in Equation 1, λ and η in Equation 7 are set experimentally. We demonstrate the performance of our approach from both qualitative analysis and quantitative evaluation, and the latter includes results based on the OCR accuracy as well as pixel-level accuracy.

A. *Qualitative Analysis*

In order to intuitively understand the influence of each phase in our approach. An example is shown in Figure 6. We can see that the 3rd phase and the 4th phase do remove some noises, since the graph-cut computation can utilize the spatial neighborhood information apart from the color information and the Log-Gabor filter can exploit the stroke information to remove the background noises which have similar color with text pixel. The comparison between the threshold-based algorithms and the proposed method on some sample images are also shown in Figure 7. Obviously, our method is more effective than the threshold-based algorithms and it can produce cleaner binary image. However, the proposed method sometimes removes some text pixels due to their low Log-Gabor responses, *e.g.*, the first character in the last image in Figure 7.

B. *Quantitative Evaluation*

1) OCR accuracy: We evaluate the performance of the proposed method and the other methods according to the OCR accuracy. The binarization results of all methods are fed into the Google's OCR engine Tesseract [25], then the character recognition rate (CRR) and the image recognition rate (IRR) are computed. The CRR is defined that the proportion of the characters correctly recognized to total groundtruth characters and the IRR is defined that the proportion of text line images correctly recognized to total groundtruth images. The experimental results are summarized in Table I, and they show that the proposed approach has significant improvement on the OCR accuracy. It also shows that the graph-cut computation and the Log-Gabor filters can contribute to better result, especially the Log-Gabor filter. Addtionally, The parameter K in K-means clustering is also very important. As parameter

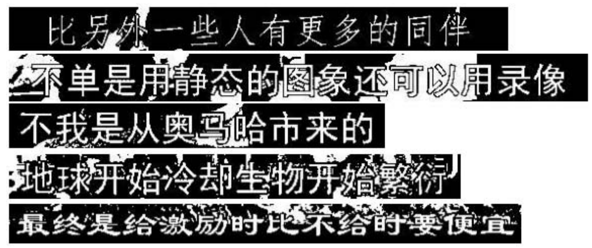
(a) Original image

(b) Final binary image

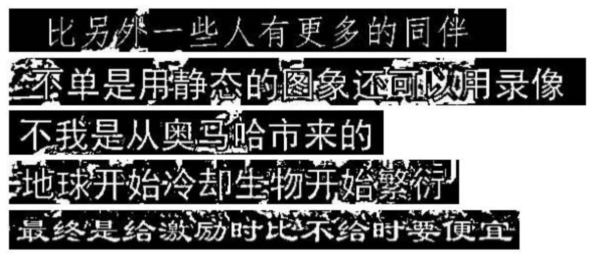
(c) Binary image without graph-cut

(d) Binary image without log-gabor

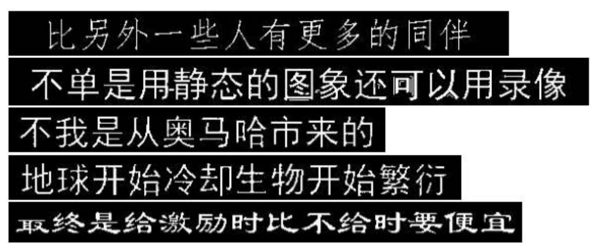
Fig. 6. The effect of each phase in our approach



(a) Results of Otsu's method



(b) Results of Niblack's method



(c) Results of proposed method

Fig. 7. Comparison of threshold-based methods and ours

TABLE I
OCR ACCURACY(IN %)

Method	CRR	IRR
Otsu	75.90	29.85
Local Ostu	75.33	32.64
Niblack	74.90	28.10
Kittler	55.93	16.99
proposed without graph cut (K=4)	87.98	44.04
proposed without log-gabor (K=4)	85.34	39.35
proposed (K=4)	88.94	44.05
Proposed (K=3)	90.73	50.68

TABLE II
PIXEL LEVEL ACCURACY WITH RESPECT TO WELL-KNOWN EVALUATION MEASURES

Method	Precision	Recall	F-score
Otsu	0.73	0.91	0.81
Local Otsu	0.82	0.92	0.87
Niblack	0.82	0.83	0.83
Kittler	0.53	0.87	0.66
proposed without graph cut (K=4)	0.97	0.73	0.84
proposed without log-gabor (K=4)	0.95	0.73	0.83
Proposed (K=4)	0.97	0.73	0.84
Proposed (K=3)	0.96	0.81	0.88

K increases, it becomes more difficult to select text cluster correctly.

2) Pixel-level accuracy: We also compare the performance of these algorithms based on pixel-level accuracy according to the well-known metrics like precision, recall, F-score. The experimental results are summarized in table II. As we have mentioned previously, the threshold-based methods can extract the majority of text pixels, so they often lead to a high recall, e.g., the recall for Otsu's method exceeds 90%. However, the precision of our method is far higher than those of the threshold-based methods. The high OCR accuracy of the proposed method also demonstrates that despite missing some foreground pixels, the characters in binary images of our method are still recognizable, since the characters are just thinner than those of groundtruth. We also observe that a larger K results in the low recall. As parameter K increases, text cluster generally gets less pixels.

V. CONCLUSION

We present a novel binarization approach for overlay text. The proposed method can effectively integrate multiple features including intensity, color, stroke to achieve better performance. The experimental results on a large evaluation dataset show that the binarization results generated by our method can well retain the most informative text pixels and they can result in much higher OCR accuracy even in the case of lower pixel-level recall.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61271434, No. 61232013, No. 61175115.

REFERENCES

[1] Michael R. Lyu, Jiqiang Song, Min Cai, *"A Comprehensive Method for Multilingual Video Text Detection, Localization, and Extraction,"* IEEE Trans. on Circuits and Systems for Video Technology. 15(2):243-255, 2005.

[2] N. Otsu, *"A threshold selection method from gray level histogram,"* IEEE Transactions on System, Man, Cybernetics, vol. 19, no. 1, pp. 62-66, 1978.

[3] J. Kittler, J. Illingworth, and J. Föglein, *"Threshold selection based on a simple image statistic,"* Computer Vision, Graphics, and Image Processing, vol. 30, no. 2, pp. 225-236, 2000.

[4] W. Niblack, *An Introduction to Digital Image Processing,* Englewood Cliffs, New Jersey: Prentice-Hall, 1986.

[5] S. Wu and A. Kangus, *"Automatic thresholding of gray-level using multi-stage approach,"* in Proc. ICDAR 2003, pp. 493-497

[6] Z. Zhou, L. Li, and C. L. Tan, *"Edge based binarization for video text images,"* in Proc. of 20th Int. Conf. on Pattern Recognition, Singapore, 2010, pp. 133-136.

[7] Haojin Yang, Quehl B., Sack H.,*"A skeleton based binarization approach for video text recognition,"* Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), 2012, pp. 1-4.

[8] S. Bolan L. Shijian and Chew Lim Tan, *"Binarization of Historical Document Images Using the local Maximum and Minimum,"* In Proc. DAS 2010, pp. 159-165.

[9] P. Stathis, E. Kavallieratou, and N. Papamarkos, *"An evaluation technique for binarization algorithms,"* J. UCS, vol. 14, no. 18, pp. 3011-3030,2008.

[10] Y. Leydier, F. Le Bourgeois, and H. Emptoz, *"Serialized Unsupervised Classifier for Adaptive Color Image Segmentation: Application to Digitized Ancient Manuscripts,"* in Proc. ICPR, 2004, pp. 494-497

[11] Libo Fu, Weiqiang Wang, and Yaowen Zhan, *"A Robust Text Segmentation Approach in Complex Background Based on Multiple constraints,"* Lecture Notes in Computer Science, Springer-Verlag, Berlin Heidelberg, Vol. 3767. pp.594-605, 2005.

[12] X. Huang, H. Ma, and H. Zhang, *"A New Video Text Extraction Approach,"* in Proc. ICME, 2009, pp. 650-653.

[13] K. Kita and T. Wakahara, *"Binarization of color characters in scene images using k-means clustering and support vector machines,"* in Proc. ICPR, 2010, pp. 3183-3186.

[14] D. Chen, K. Shearer and H. Bourlard,*"Text Enhancement with Asymmetric Filter for Video OCR,"* ICIAP, 2001, pp. 192-197.

[15] Xiaojun Li, Weiqiang Wang, Qingming Huang, Wen Gao, Laiyun Qing, *"A Hybrid Text Segmentation Approach,"* Proceeding of IEEE International Conference on Multimedia and Expo, 2009, pp.510-513.

[16] Céline Mancas-Thillou and Bernard Gosselin,*"Character Segmentation-by-Recognition Using Log-Gabor Filters,"* in Proc. ICPR, 2006, pp. 901-904.

[17] Xiaodong Huang,*"A novel video text extraction approach based on Log-Gabor filters,"* International Congress on Image and Signal Processing, 2011, pp. 474-478.

[18] C. Wolf and D. S. Doermann, *"Binarization of low quality text using a markov random field model ,"* In Proc. ICPR 2002, pp. 160-163.

[19] X. Peng, S. Setlur, V. Govindaraju and R. Sitaram, *"Markov random field based binarization for hand-held devices captured document images ,"* in ICVGIP, 2010.

[20] A. Mishra, K. Alahari and C. V. Jawahar, *"An MRF Model for Binarization of Natural Scene Text ,"* In Proc. ICDAR 2011, pp. 11-16.

[21] Z. Saidane and C. Garcia, *"Robust Binarization for Video Text Recognition,"* In Proc. ICDAR 2007, pp. 874-879.

[22] Z. Saidane and C. Garcia, *"An automatic method for video character segmentation,"* In Proc. International Conference on Image Analysis and Recognition, 2008, pp. 557-566.

[23] Trung Quy Phan, Shivakumara P., Bolan Su, Tan C.L., *"A Gradient Vector Flow-Based Method for Video Character Segmentation,"* In Proc. ICDAR 2011, pp. 1024-1028.

[24] Y. Boykov and V. Kolmogorov, *"An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision,"* IEEE Trans. Pattern Anal. Mach. Intell., vol. 26,no. 9, pp. 1124-1137, 2004.

[25] OCR Engine used:http://code.google.com/p/tesseract-ocr/.