AUTOMATIC TEXT SEGMENTATION FROM COMPLEX BACKGROUND

Qixiang Ye[1,2], *Wen Gao*[1, 2] *Qingming Huang*[1]

[1]Graduate School of Chinese Academy of Sciences, China
[2]Institute of Computing Technology, Chinese Academy of Sciences, China

ABSTRACT

In this paper, we proposed an automatic method to segment text from complex background for recognition task. First, a rule-based sampling method is proposed to get portion of the text pixels. Then, the sampled pixels are used for training Gaussian Mixture Models of intensity and hue components in HSI color space. Finally, the trained GMMs together with the spatial connectivity information are used for segment all of text pixels form their background. We used the word recognition rate to evaluate the segmentation result. Experiments results show that the proposed algorithm can work fully automatically and performs much better than the traditional methods.

1. INTRODUCTION

Text is a kind of important object in images and videos. It plays an important role in image understanding and content-based video retrieval tasks. This inspires the long researches on text recognition in images and videos [1]. However, located text lines in image or videos cannot put directly into OCR for recognition for the reason that these texts are of low resolution, of any color (not always white or black) and maybe embedded in complex background. It is necessary to develop techniques to segment text clearly from its background.

Researchers have done lots of works on text segmentation in the past years. Threshold methods have been developed to segment characters in document images with relatively simple background. In this kind of method, Niblack's local threshold method is proved to be the best one [2]. In [3], Wu uses a local threshold method to segment text from image blocks containing text. The local threshold is picked at the first "valley" on the smoothed intensity histogram. By considering that text in image and videos are always colorful, Tsai [4] developed a threshold method using intensity and saturation features to segment text in color document images. Threshold method cannot separate text clearly with its background when there are pixels in the background whose color is similar with the foreground pixels. Lienhart [5] and Bunke [6] use color clustering algorithm for text segmentation. The performance of this method is good when the color of the foreground pixel is uniform, but will drop a lot when the color of foreground pixels is not uniform which is induced by the limitation of clustering algorithm. Furthermore, the method is sensitive to noise and text resolution. Some of the existing works also consider the spatial information of the characters. In [7], Tang uses the morphological "open" operation to break the bridges across the character and the background. In [8], Chen first learns the GMMs of text pixels, and then uses Markov Random Field (MRF) to segment text from its background. For that text color differs in different images, users are requested to supply different training samples, which will bring burdens to the users. Multi-frame integration is often used for text segmentation in videos [9][10], which can enhance the foreground but will fail for moving text.

In this paper, by regarding that a text pixel always lies between an "edge couple", we proposed a rule-based sampling method which enables the truly unsupervised segmentation algorithm despite that a supervised learning method is employed in the algorithm. After building the color model for text pixels, spatial connectivity information of the character strokes is integrated to build the joint probability of each pixel to be text pixel. A threshold is selected to separate text with background pixels. After the final connected component analysis, binarized text is gotten.

The rest of the paper is as follows: We present the segmentation algorithm in section 2 and the experimental result in section 3. Conclusion is given in section 4.

2. UNSUPERVISED TEXT SEGMETNAION

Text lines are first located in images or video frames using the proposed text detection algorithm [11]. Figure 1 gives some examples of detected text lines.

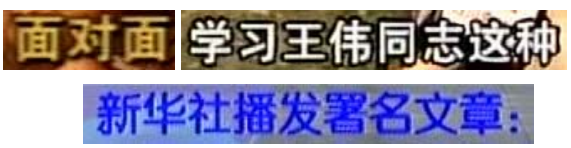


Figure 1. Examples of several detected text lines.

It can be seen in the above images that text background is complex. Simple threshold method will

fail to segment these texts from their background. At the same time, noise, compression, decompression and change of resolution can make the text pixel change in color. They maybe change both in intensity and hue. The pixels on the character's boundaries maybe blurred by the background and then have a large discrimination with the original printed character. That is the reason that we employ GMMs (Gaussian mixture models) to build the color models of text pixels. There are also pixels in the background whose color is quite similar with the foreground pixels. And there are noise and breaks on the character strokes. Spatial connectivity of character components should be integrated with color information in the segmentation algorithm.

2.1 Rule-based sampling

In this section, we will introduce the rule-based sampling which can supply training samples for text color models.

Before the sampling operations, a text image will be expanded by a sub-pixel interpolation technique [12], which can improve the resolution of the characters in the text line and make the amount of sampled pixels large enough to train the GMMs. A text line image has a height of at least 50 pixels after the interpolation in this paper.

We take the text image in figure 2 as an example to state the sampling procedure. Vertical, horizontal and diagonal edges are first computed by a Canny edge detector [14]. Horizontal (vertical, diagonal) edges include positive edges h^+ (v^+, d^+) and negative edges h^- (v^-, d^-) which is gotten on the positive and negative gradient and negative respectively. If we can find the corresponding pixel h^- (v^-, d^-) for a h^+ (v^+, d^+) pixel within the set distance w, we say that the two pixels form an edge couple $h^\pm$ ($v^\pm$, $d^\pm$). The value of w should be the width of character strokes. Before we can get the accurate width of character component, the mean value of distances between all of the edge couples used as the value of w whose largest value are limited in terms of the height of the present text image. Text pixels are supposed to lie between these edge couples. Some of these pixels will be sampled to build the color models of all of the text pixels.

To improve the reliability, only when there are more than 3 connected $h^\pm$ on the same horizontal row, that is, they construct a parallel edge lines, the pixels between them may be sampled as training samples. Figure 2 c) shows all of horizontal and vertical parallel edge lines in an image. All of the pixels between the parallel edge lines are extracted to be training samples (as shown in figure 2 d)). In the sampling procedure, only horizontal and vertical edge couples are used while diagonal edge couples are discarded because that it is easier to detect parallel lines in horizontal and vertical directions than in diagonal directions.

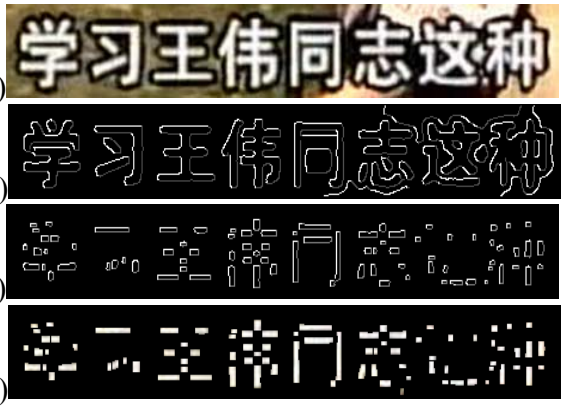


Figure 2. a) Original image. b) Canny edge map where white pixels are positive edges and gray pixels are negative edges, c) Parallel edge lines, d) Sampled pixels.

2.2 Building color models for text pixels

Hue and intensity components in HSI color space [14] are selected as color features. The intensity can discriminate most of the text with its background and hue component is useful in some special issues when the text's intensity is similar with its background (such as red text in green background). The Saturation component is discarded. This will reduce the dimension from 3 to 2 and then avoid short of training samples.

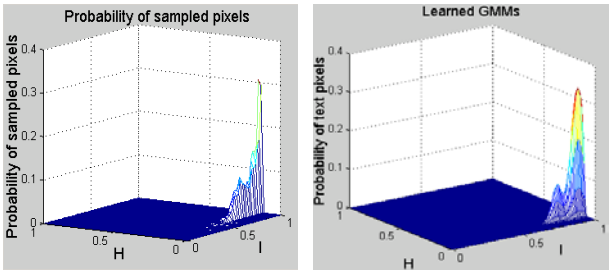
Under the finite mixture models to be fitted in this paper, each sampled pixels can be regarded as arising from the all text pixels G which is a finite number, g, of text pixels $G_1,\ldots,G_g$ in some proportions $\pi_1,\ldots,\pi_g$, where $\sum_{i=1}^{g}\pi_i = 1$ and $\pi_i \geq 0$. The probability density function of an observation x (of d dimensionality) in the finite mixture form of

$$\begin{aligned} p(x;\phi) &= \sum_{i=1}^{g}\pi_i \cdot p_i(x;\theta) \\ &= \sum_{i=1}^{g}\pi_i \cdot \frac{1}{(2\pi)^{d/2}\left|\Sigma_i\right|^{1/2}}\exp\left(-\frac{1}{2}(x-\mu_i)^T(\Sigma_i)^{-1}(x-\mu_i)\right) \end{aligned} \quad (1)$$

where θ represents unknown parameters including mean vector μ_i and the elements of covariance matrices Σ_i for $i = 1,\ldots,g$ [15]. An EM algorithm can be use to estimate all of the parameters given enough training samples.

A crucial problem of this method is how to adaptively determine the number of Gaussian mixtures *K*, since color composition differs in different locations. In this paper, the initial number of mixture component *K* is set as 1. We

increase K by one each time and test the number of recognition characters. The final value selected for K should ensure that the most characters in the text line are correctly recognized. In the following experiments, we found that 2 and 3 are the mostly used values for K. Figure 3 gives the built GMM with two mixture components by sampled text pixels.



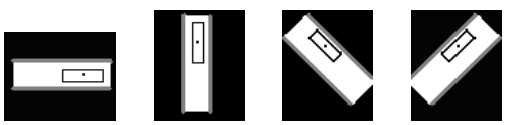
a) b)

Figure 3. Leaning models. a) Probability of sampled pixels, b) Learned GMMs.

2.3 Text segmentation using color and spatial connectivity information

After getting the color models of the foreground pixels, the probability of each pixel in the image to be text pixel can be calculated. According to the probability, most of the foreground pixels can be segmented from the background. But for characters of noise and breaks, spatial connectivity should be integrated to improve the segmentation performance.

The spatial connectivity information used in this paper is simple but effective. For each pixel, we find the nearest edge couples around it. If the pixel lies between a horizontal (vertical, diagonal) edge couple, we calculate the joint probability of the pixels by the neighborhood in horizontal (vertical, diagonal) direction. The neighborhood is a rectangle whose width is 1/3 of the width of character stroke w and length is same as w. For the pixels that do not lies between any parallel edge couple lines, the neighborhood is set as one pixel on itself. Figure 4 give examples of pixel's rectangle neighborhood.



a) b) c) d)

Figure 4. Pixel's neighborhood on a) horizontal, b) vertical, and c),d) diagonal strokes respectively. The gray lines are edge couples, the black rectangles around the pixel are the neighborhoods of the pixel.

The final probability of a pixel $x_{0,0}$ to be foreground pixel is calculated by a Gaussian function as

$$p(x_{0,0}) = \sum_{m=0}^{M}\sum_{n=0}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\alpha \cdot m^2 + n^2}{2\sigma^2}\right) \cdot p(x_{m,n}) \quad (2)$$

where, M, N are the sizes of neighborhood rectangle, $\alpha = N/M$, $p(x_{m,n})$ is probability of the pixel whose distance is m and n in the neighborhood. This joint probability can ensure that noise pixels and break pixels on the character strokes be correctly segmented as foreground pixels.

After getting the probability map, a threshold T is used for distinguish foreground and background pixels. Pixels whose probability is larger than T is segmented as foreground and others background. T is gotten by statistic method which shows that 0.06 performs best for all kinds of texts in the training set.

2.4 Connected components analysis

For the detected foreground pixels, connected component analysis based on the geometry properties is employed to eliminate non-text regions. To break the bridges across the character and the background, morphological "open" is first used. After the open operation, connected components whose sizes are smaller than 20 pixels (the value can be adjust according to image size) are eliminated. The region whose width/height ratio is larger than 10.0 or smaller than 0.1 will be eliminated. If there are too many pixels in a region lying on the boundaries of the image, the region will also be eliminated. Figure 5 is an example of segmented text.

a)
b)

Figure 5. a) Detected foreground pixels using GMMs with spatial connectivity information, b) Final segmentation result after connected component analysis.

3. EXPERIMENTAL RESULT

In our work, 537 text lines detected in 400 color images (300 video frames and 100 images from WWW) are selected for experiments. There are 4861 Chinese words in these text lines. The quality of these texts is decreased by in the MPEG/JPEG decoding procedure. Word recognition rate (WRR) by OCR is used to evaluate the the segmentation algorithm. The WRR is defined as

$$WRR = \frac{N_r}{N} \quad (3)$$

where N_r is number of correctly recognized words and N is the number of all words in the test set.

The experimental results are listed in table 1. We also select three methods for performance comparison, in which Local threshold and K-mean method are two kinds of mostly used methods and Chen's MRF-based method [8] is reported to be performance-best method. The WRR of our algorithm is 85.5%, which is higher that of Local threshold method and K-mean clustering method. The WRR of our algorithm is a little lower that Chen's method but our algorithm is fully automatic while Chen's method need to label training samples by hand.

The speed of the segmentation algorithm is 252 words per second (do not include the OCR procedure). For the reason that there are only several text lines in an image or video frames, the speed can feed the real-time demand in images and videos. The speed is much higher than other three kinds of methods.

Table 1. Performance comparison of three algorithms

Algorithms	WRR	Speed (words/s)
Our algorithm	85.5%	252
Local threshold	79.3%	266
K-Mean	81.6%	117
MRF method	86.7%	93

To evaluate the robustness of the proposed algorithm, we do experiment on texts with very low quality which are gotten by adding Gaussian noise to the images in the original test set and scale them down to low resolution. The decreases of WRR of the three algorithms are listed in table 2. The performance decreases of our method on noise text and low resolution text are 1.3% and 1.7% respectively which are the lowest in all of the compared methods. This shows the robustness of our method.

Table 2. WRR decrease on low quality texts.

Algorithms	WRR decrease on text with noise	WRR decrease on low resolution text
Our algorithm	1.3%	1.7%
Local threshold	4.5%	3.3%
K-Mean	3.0%	3.8%
MRF method	2.1%	2.9%

4. CONCLUSION AND FUTURE WORKS

In this paper, an unsupervised method is proposed to segment text in complex background for recognition task. The method can be used to extract text embed in videos frame, images or natural scenes The method is fully automatic and has a good performance proved by OCR recognition result. Speed of the proposed method can feeds real-time demanding even for text segmentation in videos. In the future work, more sophisticated parameter determination method should be developed to improve the robustness of the proposed method.

5. ACKNOWLEDGMENT

This work is supported in part by Bairen project of Chinese acadamy of science and JDL-NEC sports video analysis and retrieval project No. 0P2004001.

6. REFERENCES

[1] K. Jain and B. Yu, "Automatic Text location in Images and Video Frames," *Pattern Recognition*. Vol. 31, No. 12, pp. 2055-2076, 1998.

[2] Trier, A.K Jaind, "Goal-Directed Evaluation of Binarization Methods," I*EEE Trans. on PAMI*, Vol. 17, No. 12, pp.1191-1202, Dec, 1995.

[3] V. Wu, R. Manmatha, and E. M. Riseman, "Textfinder: An Automatic System to Detect and Recognize Text in Images," *IEEE Trans. on PAMI*, Vol. 20, pp. 1224-1229, 1999.

[4] C.M Tsai and H.J Lee, "Binarization of Color Document Images via Luminance and Saturation Color Features," *IEEE Trans. on Image Processing*, Vol.11, No.4, 2002.

[5] R. Lienhart, "Automatic Text Recognition in Digital Videos," *in Proc. SPIE, Image and Video Processing IV*, , pp. 2666-2675, Jan. 1996.

[6] K. Sobottka, H. Bunke, and H. Kronenberg, "Identification of Text on Colored Book and Journal Covers," *in Int. Conf. on Document Analysis and Recognition*, pp.57-63, 1999.

[7] X. Tang, X. B. Gao, J. Liu and H. Zhang, "Spatial-Temporal Approach for Video Caption Detection and Recognition," *IEEE Trans.on Neural Networks*, Vol. 13, No. 4, July, 2002.

[8] D. Chen, J-M. Odobez, and H. Bourlard, "Text Segmentation and Recognition in Complex Background Based on Markov Random Field," *in Proc. of the Int. Conf. on Pattern Recognition*, 2002.

[9] M. Cai, J. Song and M.R. Lyu, "A New Approach for Video Text Detection," *in Int. Conf. On Image Processing*, Rochester, New York, USA, September 22-25,2002.

[10] H.Li and D. Doermann, "Text Enhancement in Digital Video Using Multiple Frame Integration," *in ACM Multimedia*, pp.385-395, 1999.

[11] Q. Ye, W. Gao, W. Wang, W. Zeng, "A Robust Text Detection Algorithm in Images and Video Frames," *in Fourth IEEE Pacific-Rim Conference On Multimedia, Singapore*, 2003.

[12] A. R. Smith, "Color Gamut Transform Pairs," *Computer Graphics*, Vol.12 No.3, pp.12-19, Aug. 1978.

[13] T.Sato, T.Kanade, E.K.Jughes, M.A. Smith and S.Satoh, "Video OCR: Indexing Digital News Libraries by Recognition of Superimposed Captions," *ACM Multimedia Systems: Special Issue on Video Libraries*, Vol.7 No. 5, pp. 385-395, 1999. 7(5).

[14] J. Canny, "Computational Approach to Edge Detection," *IEEE Trans. On Pattern Analysis and Machine Intelligence*, Vol. 8, No. 6, pp. 679-698, 1986,

[15] B.S. Everitt and D. J. Hand, Finite Mixture Distributions, *Chapman and Hall*, New York, 1981.