MVA2000 IAPR Workshop on Machine Vision Applications, Nov. 28–30, 2000, The University of Tokyo, Japan

3—33

A Character Pattern Extraction Method Applicable to Various Types of Images

Hideaki Goto *
Education Center for Information Processing,
Tohoku University

Hirotomo Aso †
Graduate School of Engineering,
Tohoku University

Abstract

As the applications of the OCR are widely expanding recently, computers have to extract character patterns from various types of images such as scanned document images, scene images and video frames. Many methods for character pattern extraction have been proposed so far. They were designed and tuned for a special type of image and for a certain input device. However, we have many images which cannot be processed by the conventional framework. It is necessary for us to develop an all-purpose, unified processing which can extract character patterns from various types of images using various input devices.

To realize the unified processing, we previously developed a character pattern extraction method based on the local multilevel processing and the region growing. The performance of the method was evaluated using scanned document images, however, no detailed evaluation has been performed for scene images. By the new experiments using scene images, we have verified that the method works well even for scene images. The rate of text line extraction is 75%, which is almost same as that of a conventional method. Although the rate is not very high, the remarkable property of our method is that it is applicable to various types of images.

1 Introduction

With the recent expanding application of the OCR, computers have to extract character patterns from various types of images including scene images and video frames[1]. Scene images often suffer from greater shading, include much noise, and have more complex backgrounds than scanned document images. Many methods have been studied and proposed for scene images[2, 3]. However, they do not work well for the main text regions in document images, because they were designed to pick up rather large characters only, and because they can handle only the images in which text strings appear sparsely.

Today we have a large number of documents in which text characters are printed on colored backgrounds, colorful textures, or complex backgrounds. To extract text information from such documents, we need a process which can separate character patterns and backgrounds. For separation of character patterns and complex backgrounds, many methods have been proposed [3, 4, 5]. However, they can handle only clean, plain document images which have almost no shadings. They often fail to extract character patterns from picture regions in document images.

In recent years, the techniques for document image capturing by digital still cameras or video cameras have been studied actively. The document images captured by a camera have similar properties as scene images. Even the document processing systems used in offices require techniques developed for scene image analyses.

Putting the problems mentioned above together, it is necessary for us to develop an all-purpose, unified processing which can extract character patterns from various types of images (Fig.1). Such a unified processing must have all the following properties.

1. It can extract from very small characters with thin strokes to large characters with thick strokes from images with complex backgrounds.
2. It is tolerant of the shadings of image.
3. It can handle images with very complex layout.

To realize the unified processing, we developed a method for character pattern extraction[6]. The method is based on the local multilevel processing and the region growing. Experimental results for scanned document images give satisfactory rates of text line extraction: 81.7% for colorful magazine

* Address: Kawauchi, Aoba-ku, Sendai-shi 980–8576, Japan. E-mail: hgot@ecip.tohoku.ac.jp , WWW: http://www.ecip.tohoku.ac.jp/~hgot/

† Address: Aramaki, Aoba-ku, Sendai-shi 980–8578, Japan. E-mail: aso@ecei.tohoku.ac.jp

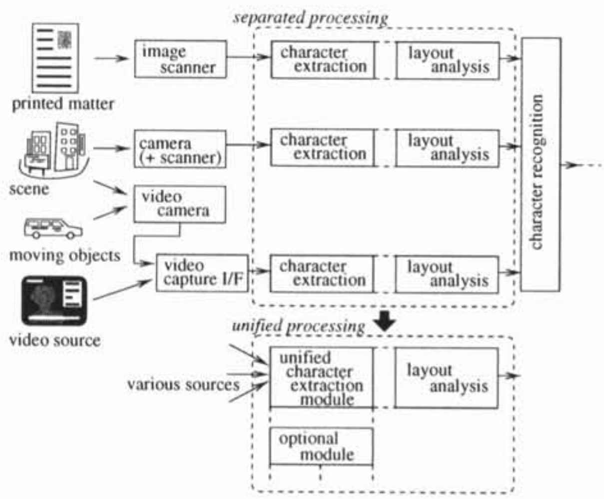

Figure 1: Unification of the character pattern extraction processes

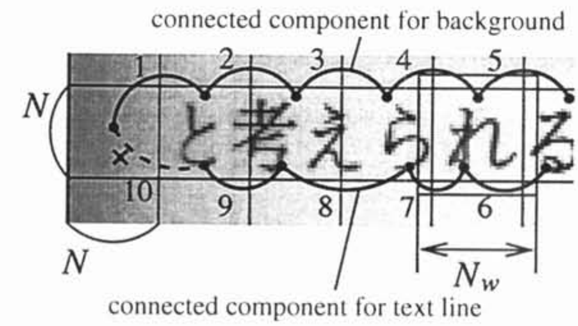

Figure 2: Region growing

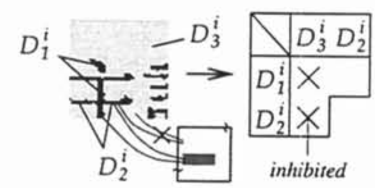

Figure 3: Creation of merging inhibition table

covers, 98.5% for magazine pages with colorful backgrounds, and 100% for technical articles with shadings. We also applied the method to some scene images, however, no detailed evaluation has been performed. In this paper, we describe the detailed experimental results for scene images, comparing the performance of the method with a binarization-based method[2]. The computation cost of our method is also discussed.

2 Character pattern extraction based on local multilevel thresholding and region growing

The overview of the method is as the following. First, the input image is partitioned into subimages of size $N \times N$ by the square grid (Fig.2). A co-centering window of size $N_w \times N_w$ is assigned to each subimage. For each subimage, the representative gray levels are found by detecting the peaks in the intensity histogram using the pixel values within the corresponding window. The local image is segmented by the multilevel thresholding. We used $N = 50$ and $N_w = 60$ for scanned document images.

For each subimage, the merging inhibition table is created. Figure 3 shows that there are three representative gray levels, D_1^i, D_2^i and D_3^i, in the i-th subimage. If D_1^i or D_2^i is merged with D_3^i, it causes the fusion of the character patterns and the backgrounds. Such merging must be inhibited.

Then, the connected components of the representative gray levels are found over the image, comparing the gray level difference between 8-connected subimages with an incremental threshold. This procedure corresponds to the region growing. For each possible connection between the gray levels, the merging inhibition tables are referred over the pair of the connected components. If an inhibited merging is detected, the connection is canceled. For example, the connection from 1 through 5 and from 6 through 9 in Fig.2 are allowed, while the connection 10 is inhibited. This procedure ensures that the fusion of character strokes and backgrounds will never occur over the image. Each connected component of the representative gray levels corresponds to the set of black pixels in a decomposed image. The input image is decomposed into several binary images by the method. A text line printed by the same gray is expected to appear in a single decomposed image.

We have found some character patterns are split into several fragments if the method is applied to low resolution scene images. Since the subimage is too large compared to the local structure of the images, some stroke fragments are connected to the surrounding backgrounds before they are connected to adjacent stokes. An example is shown in the left column of Fig.4. Aiming for reducing such undesirable split of strokes, we tried to reduce N to 20 without changing the algorithm. We did not change the value of N_w because the precision of the multilevel thresholding gets worse as N_w is reduced.

3 Experimental results

The performance of the proposed method has been evaluated using 33 scene images taken by a digital camera with 850k pixel CCD, both indoors and outdoors. 11 images out of them have charac-

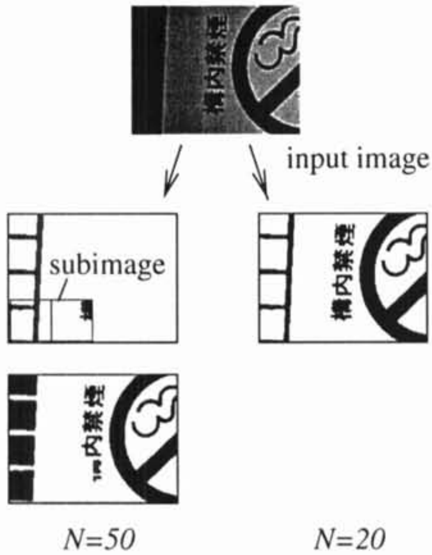

Figure 4: Undesirable split of strokes for large N

Table 1: The rates of text line extraction

attr.	#lines	$N = 50$	$N = 20$	binarization
simple	156	86%	81%	96%
complex	101	63%	67%	46%
Total	257	75%	75%	76%

ter patterns which appear on complex backgrounds. The size of each image is 1024×768 pixels. We used the local adaptive binarization for comparison[2].

3.1 Rates of text line extraction

Table 1 shows the rates of text line extraction. For images with simple backgrounds, the binarization works better than our method. Most of the failures in our method were due to the low contrast of the image. Some improvements are required in the local multilevel thresholding. We can see some improvement of the rates for images with complex backgrounds at $N = 20$, however, no significant difference can be found in the total. The processing time of the region growing at $N = 20$ is about 2.3 times of that at $N = 50$. We need to search another way of suppressing the split of strokes.

Figure 5 shows an example of the character pattern extraction from a journal cover image. Both the proposed method and the adaptive binarization are tolerant to the smooth shading in the image. Though we cannot see the difference in Figure 5, the adaptive binarization produces clearer character patterns than the proposed method. In binarization-based methods, it is assumed implicitly

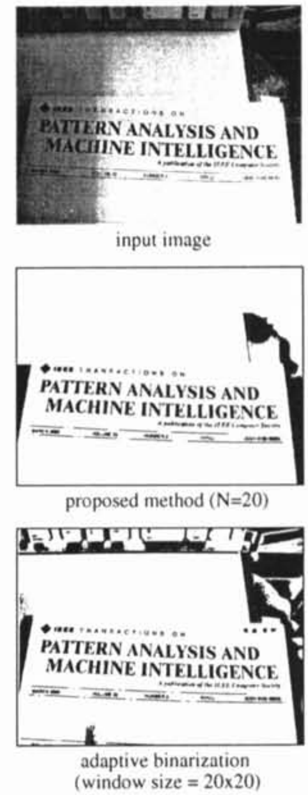

Figure 5: Character pattern extraction from the image with simple background (IEEE Trans. PAMI, Vol.22, No.3, 2000.)

Figure 6: Character pattern extraction from the image with complex backgrounds (UNIX MAGAZINE, June, 2000, ASCII CORP.)

Table 2: Processing time

N (# of subregions, ratio)	Thresholding	Region growing	Total (ratio)
N=50 (336 , 1.00)	0.692	0.554	1.246 (1.00)
N=40 (520 , 1.55)	0.834	0.623	1.457 (1.17)
N=30 (910 , 2.71)	1.147	0.819	1.966 (1.58)
N=20 (2028, 6.04)	1.686	1.188	2.874 (2.31)

User times (sec.) on Sun Ultra30 (CPU: UltraSPARC-II, 300MHz)

that "the image is of bilevel." This assumption is very strong and advantageous for extracting character patterns from images with smooth backgrounds. On the other hand, the appropriate number of gray levels must be found automatically in the multilevel thresholding, and therefore, the multilevel thresholding is slightly disadvantageous for such images.

Figure 6 shows an example of the character pattern extraction from a book cover image with complex background. The image have 16 text lines. All of them are extracted correctly by our method, while the binarization method can extract only 6 text lines. Although we have no significant difference of the text extraction rates statistically, note that our method works far better for scanned document images with complex backgrounds than binarization-based methods, without a modification of the algorithm and parameters[6].

3.2 Processing time

Processing times of our method were measured by a Workstation, Sun Ultra30. The program was written in C++ language. Table 2 shows the processing times in average for various values of N. It is difficult to estimate the total computation costs theoretically because they depend strongly on the contents of input images. Fortunately, in our implementation, we have no tremendous slowing down as N decreases. According to the values in the table, the dependency of the processing time with respect to the number of subregions K is nearly $O(\sqrt{K})$.

4 Conclusion

In this paper we have introduced the framework of the unified processing of character pattern extraction. In order to expand the application of OCR, such a framework is necessary. To realize the unified processing, we previously developed a character pattern extraction method based on the local multilevel thresholding and the region growing. New experiments have been performed and the results have shown that our method works well even for scene images. The rate of text line extraction from scene images is 75% in average, which is almost same as that of a conventional binarization-based method. The remarkable advantages of the method are that it works far better for scanned document images with complex backgrounds than conventional methods, and that it can process various types of images. The proposed method is expected to contribute to the development of universal document analysis systems, office robots, information capturing cameras, and so on.

We need some improvements of the performance of the method, especially the performance for the images with plain backgrounds. It is one of our future work to improve the performance.

References

[1] H. Fujisawa, H. Sako, Y. Okada, and S.W. Lee, "Information capturing camera and developmental issues," Proc. 5th Int. Conf. on Document Analysis and Recognition (ICDAR'99), pp. 205–208, 1999.

[2] J. Ohya, A. Shio, and S. Akamatsu, "Recognizing characters in scene images," IEEE Trans. on PAMI, Vol.16, No.2, pp.214–220, 1994.

[3] Y. Zhong, K. Karu, and A.K. Jain, "Locating text in complex color images," Pattern Recognition, Vol.28, No.10, pp.1523–1535, 1995.

[4] A.K. Jain and B. Yu, "Automatic text location in images and video frames," Pattern Recognition, Vol.31, No.12, pp.2055–2076, 1998.

[5] K. Sobottka, H. Kronenbern, T. Perroud, and H. Bunke, "Text Extraction from Colored Book and Journal Covers," International Journal on Document Analysis and Recognition (IJDAR), vol.2, no.4, pp.163–176, 2000.

[6] H. Goto and H. Aso, "Character pattern extraction based on local multilevel thresholding and region growing," Proc. 15th Int. Conf. Patt. Recogn. (ICPR2000), Vol. 4, pp.430–433, 2000.