Text Segmentation based on Stroke Filter

Qifeng Liu
Computing Lab,
Samsung Advanced Institute of Technology
Zhongguan. No.9, Beijing, China
qifeng.liu@samsung.com

Cheolkon Jung
Computing Lab,
Samsung Advanced Institute of Technology
P.O.Box 111, Suwon 449-712, Korea
cheolkon.jung@samsung.com

Youngsu Moon
Computing Lab,
Samsung Advanced Institute of Technology
P.O.Box 111, Suwon 449-712, Korea
mys66@samsung.com

ABSTRACT

Most existing methods of text segmentation in video images are not robust because they do not consider the intrinsic characteristics of text. In this paper, we propose a novel method of text segmentation based on stroke filter (SF). First, we give the definition of text, which is realized in the form of stroke filter based on local region analysis. Based on stroke filter response, text polarity determination and local region growing modules are performed successively. The effectiveness of our method is validated by experiments on a challenging database.

Categories and Subject Descriptors

H.3.1 [**Content Analysis and Indexing**]:-*Indexing methods*; I.5.4 [**Applications**]:– *Text processing*

General Terms

Algorithms, Performance, Design, Experimentation

Keywords

Text segmentation, image processing, text polarity determination

1. INTRODUCTION

It is well known that text extraction, including text detection, localization, segmentation and recognition is very important for video auto-understanding. In this paper, we only discuss text segmentation, which is to separate text pixels from complex background in the sub-images from videos.

Text segmentation in video images is much more difficult than that in scanning images. Scanning images generally has clean and white background, while video images often have very complex background without prior knowledge about the text color.

Although there have been several successful systems of video text extraction [1], few researchers specially study text segmentation in video images deeply. The used strategies could be classified into two main categories: (1) difference (or top-down) and (2) similarity based (or bottom-up) methods. The first methods are based on the foreground-background contrast. For example, fixed threshold value method [2], Otsu's adaptive thresholding method [3], global & local thresholding method [4], Niblack's method [5] and improved Niblack method [6], et al. In general, they are simple and fast, but fail when foreground and background are similar.

Alternatively, the similarity based methods cluster pixels with similar intensities together. For example, Lienhart uses the split & merge algorithm [7], Wang et al. combine edge detection, watershed transform and clustering [8]. However, these methods are unstable, since they exploit many intuitive rules about text shape. As an alternative, Chen et al. convert text pixel clustering to labeling problem using Gibbsian EM algorithm [9]. This method is effective but too time consuming.

Only a few papers address text polarity determination. Instead, most existing methods assume that the text color is always lighter or darker than background. Obviously, this assumption limits the practical application of these methods. Chen et al. determine text polarity by multiple hypotheses testing using OCR. It is effective, but too time consuming [9]. Antani et al. analyze connected component with some prior knowledge [10]. This method is not robust since text components are often connected to background. An effective method is proposed by Song et al., which is based on the intrinsic relationship between text and background edges [11]. However, this method fails when text localization is not accurate and edge clutter in background is included.

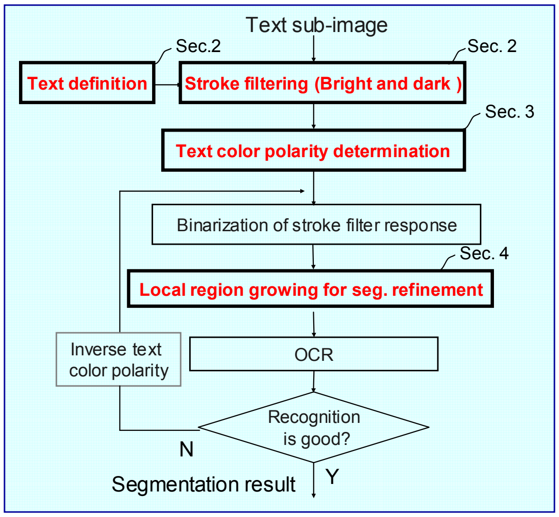

Text sub-image
Sec.2
Sec. 2
Text definition
Stroke filtering (Bright and dark)
Sec. 3
Text color polarity determination
Binarization of stroke filter response
Sec. 4
Local region growing for seg. refinement
Inverse text color polarity
OCR
Recognition is good?
N
Y
Segmentation result

Figure 1. Flowchart of our method. The bold and red parts are key points of this paper.

MM'06, October 23-27, 2006, Santa Barbara, California, USA.
ACM 1-59593-447-2/06/0010.

The main problem of most existing methods is that they are sensitive to text color, size, font and background clutter, since they simply exploit either general segmentation method or some prior knowledge. In this paper, we try to discover the intrinsic characteristics of text and design a robust algorithm specially for text segmentation. Fig. 1 shows the flowchart of our method, of which the novelty could be summarized as that we propose:

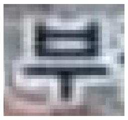
- stroke filter (SF), which can describe the intrinsic characteristics of text in terms of scale, orientation and response,
- stroke feature based text polarity determination method,
- local region growing method for segmentation refinement based on stroke features and global & local statistic similarities.

The remained parts of this paper are organized as follows. In Sec. 2, we address stroke filter. Text polarity determination and local region growing are described in Sec. 3 and Sec. 4, respectively. We give experiments in Sec. 5 and conclude this paper in Sec. 6.

2. STROKE FILTER

2.1 Text Definition

Based on the observation of text, we give the definitions of text and stroke-like structure as follows:



Definition 1. A sub-image is text, if and only if: (1) local constraint - there are many stroke-like structures in the sub-image, and (2) global constraint - the stroke-like structures have specific spatial distribution, where a stroke is defined as a straight line or arc used as a segment of a character.

Definition 2. A local rectangular image region is stroke-like structure, if and only if it is (in term of intensity): (1) different from its lateral regions, (2) with similar lateral regions, and (3) nearly homogenous.

2.2 Stroke Filter

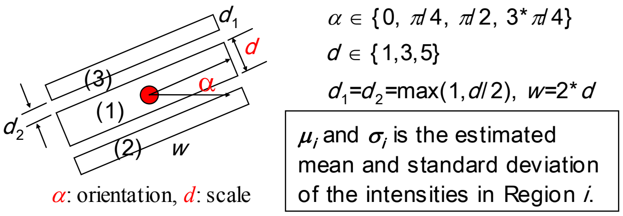

d_1
d
(3)
α
(1)
d_2
(2)
w
α: orientation, d: scale
$\alpha \in \{0, \pi/4, \pi/2, 3*\pi/4\}$
$d \in \{1,3,5\}$
$d_1=d_2=\max(1,d/2)$, $w=2*d$
μ_i and σ_i is the estimated mean and standard deviation of the intensities in Region i.

Figure 2. Local regions for stroke filter.

For each pixel in source image, we analyze local region difference. As shown in Fig. 2, the central point denotes an image pixel (x, y), around which there are three rectangular regions. The orientation and scale of these local regions are determined by α and d. According to the definition of stroke-like structure, we define *bright and dark* stroke filter responses of the pixel (x, y) as:

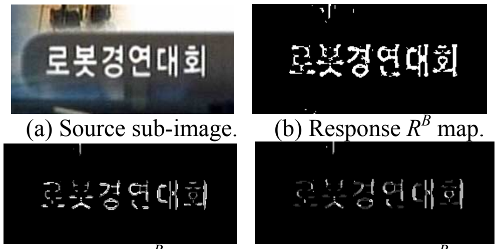
$$\begin{cases} \textit{Bright res.}: R^{(B)}_{\alpha,d}(x,y) = \mu_1 - \mu_2 + \mu_1 - \mu_3 - |\mu_2 - \mu_3| \\ \textit{Dark res.}: \quad R^{(D)}_{\alpha,d}(x,y) = \mu_2 - \mu_1 + \mu_3 - \mu_1 - |\mu_2 - \mu_3| \end{cases}, \qquad (1)$$

where μ_i is the average value of the pixel intensities in Region (i) as shown in Fig. 2. Eq. (1) has clear physical meanings related to the three conditions in Definition 2. The closer the pixel (x, y) is to the central of bright (dark) stroke-like structure, the larger its bright (dark) stroke filter response is.

By stroke filtering, we extract stoke feature (R^B, O^B, S^B, R^D, O^D, S^D) of any pixel (x, y) as follows (Note that (R^D, O^D, S^D) have similar expressions):

$$\begin{cases} R^{(B)}(x,y) = \max\limits_{(\alpha,d)} R^{(B)}_{\alpha,d}(x,y) & (\mathit{Response}) \\ O^{(B)}(x,y) = \arg\max\limits_{(\alpha)} R^{(B)}_{\alpha,d}(x,y) & (\mathit{Orientation}) \\ S^{(B)}(x,y) = \arg\max\limits_{(d)} R^{(B)}_{\alpha,d}(x,y) & (\mathit{Scale}) \end{cases}. \qquad (2)$$

An example of bright stroke filtering results is shown in Fig. 3. We find that stroke filter can filter out most step-like edges, and at the same time, the text parts are enhanced well.



(a) Source sub-image. (b) Response R^B map.

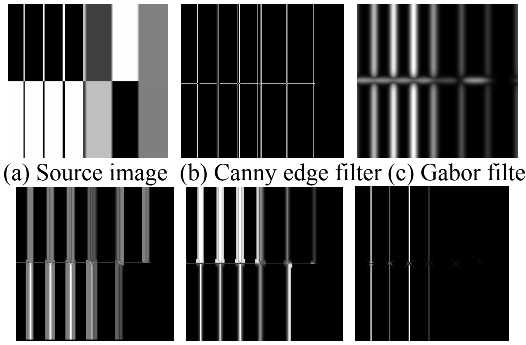
(c) Scale S^B map. (d) Orientation O^B map.

Figure 3. Bright stroke filtering results. The intensities indicate the stroke feature values.

2.3 Comparison with Related Filters

There are some related filters, which are based on local image difference between central and lateral regions, such as Canny edge filter [12], Gabor filter [13], Haar-like filter [14] and ratio line filter [15]. The difference lies in that stroke filter considers some intrinsic characteristics of text.

We compare the five filters on a typical image containing some kinds of edge, such as step edge, stroke edge (i.e., the left four vertical lines), bright edge and dark edge, as shown in Fig. 4. We find that compared with the other filters, stroke filter has the best performance to enhance stroke-like structures.



(a) Source image (b) Canny edge filter (c) Gabor filter

(d) Haar-like filter (e) Ratio edge filter (f) Stroke filter

Figure 4. Comparison of responses of some related filters.

In addition, we use integral image [14] to avoid convolution operation and thus speed up stroke filtering significantly. The computational cost is similar to that of Haar-like filter [14].

3. TEXT POLARITY DETERMINATION

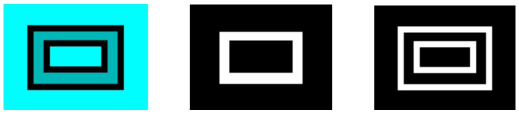
We first perform bright and dark stroke filtering to obtain R^B and R^D, respectively. Then we can obtain two features for text polarity determination. The first feature F_R is the ratio of the sums of bright and dark stroke filter response magnitude values:

$$F_R = \sum_{(x,y)} R^{(B)}(x,y) \Big/ \sum_{(x,y)} R^{(D)}(x,y) \cdot \qquad (3)$$

We think that bright (dark) text should have stronger response of bright (dark) stroke filter than that of dark (bright) stroke filter.

The second feature F_E, inspired by [11], is the ratio of the numbers of edge points in binarized bright and dark stroke filter response maps:

$$F_E = N^{(B)} \big/ N^{(D)} \,. \qquad (4)$$



(a) Source image (b) Bright SF resp. map (c) Dark SF resp. map

Figure 5. Bright and dark stroke filter (SF) response maps.

As shown in Fig. 5, $N^{(B)}$ is the number of edge points in (b) and $N^{(D)}$ in (c). F_E is useful for the case of bright text on bright background (*Bon*B) or dark text on dark background (*DonD*). By experiments, we find bright (dark) text has less edge points in the binarized bright (dark) stroke filter response map, i.e., $N^{(B)}<N^{(D)}$ ($N^{(B)}>N^{(D)}$).

We manually select about 100 samples (50 for training and 50 for testing) from about 800 video images, which are of complex conditions such as *BonB* or *DonD*. Some representative samples are plotted in the feature space spanned by (F_R, F_E), as shown in Fig. 6. Here we use general SVM classifier with RBF kernel. The accuracy is about 96%. This demonstrates the feature (F_R, F_E) is stable and distinctive.

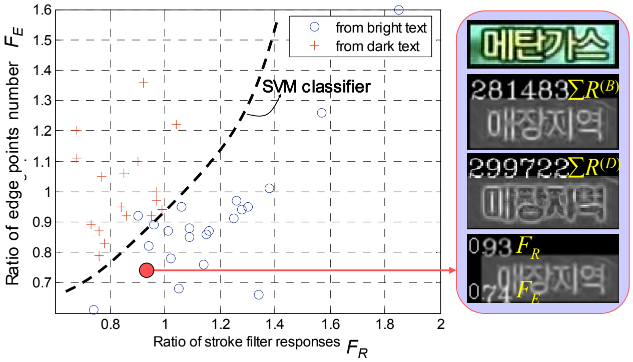


Figure 6. Feature space and SVM classifier.

4. LOCAL REGION GROWING

The binarized stroke filter response map can only be regarded as initial segmentation result, since many text pixels are missed. So in this section, we address how to recall the missed text pixels. The algorithm, named as local region growing, is described as Table 1, and Fig. 7 is a good reference for understanding.

Table 1. Algorithm of local region growing.

Input: I – initial segmentation result (binarized stroke filter response map obtained in Sec. 3); S – source sub-image.
Step 1: Combining I and S, we estimate the PDF (Probability Density Function) of text color.
Step 2: For each white pixel in I, if the number of white pixels in its 3*3 neighbors is within [3, 9], then go to Step 3, else Step 2.
Step 3: For each black pixel in the 3*3 regions, if it is (1) similar to its text neighbors, and (2) of high prob. according to PDF, it is labeled as text. Repeat Step 3 & 2, until no new text pixel is found.
Output: Refined segmentation result.

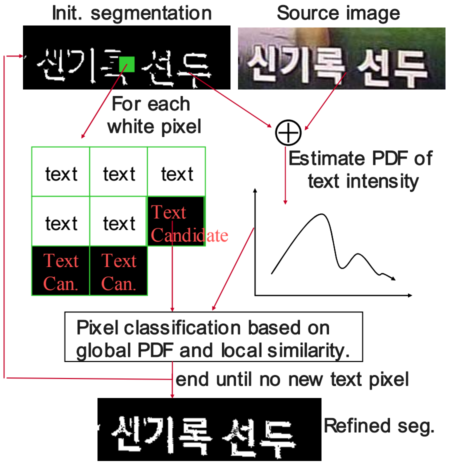


Figure 7. Flowchart of local region growing module.

The novelty of this method is that it is based on stroke filter response and combines global (PDF) and local similarities together for reliable region growing. Some results are shown in Fig. 8.

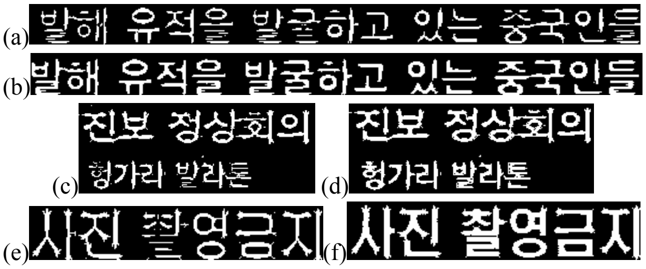

(a)
발해 유적을 발굴하고 있는 중국인들
(b)
발해 유적을 발굴하고 있는 중국인들
(c)
진보 정상회의 헝가리 발라톤
(d)
진보 정상회의 헝가리 발라톤
(e)
사진 촬영금지
(f)
사진 촬영금지

Figure 8. Some results of local region growing module. {(a), (c), (e)} and {(b), (d), (f)} indicate results before and after local region growing, respectively.

5. EXPERIMENTS

We do experiments with one CPU (1.4GHz) on Win2000 and VC++6.0. In our database, there are news images from about 4-hour videos of South Korea TV channels, whose size are 720*480 (text height varies from 10 to 250 pixels). In these images, 435 images are for testing and 357 for training. The time of processing a sub-image (140*40) is about 15 ms.

Some representative results are shown in Fig. 9. It is demonstrated that our method is robust against the challenging cases of *BonB* (Fig.9(b)(c)(d)(g)(k)) or *DonD*, complex background with various edge and color clutter (Fig.9(d)(g)(h)), color variation in one character (Fig.9(i)), et al. As we know, few methods can simultaneously handle so many different complex cases in video image, although someone may perform well in some specific conditions. For example, Song's text polarity classification method [11] may fail when complex background (Fig.9(d)(h)), and most existing segmentation methods may fail in the cases of *BonB* (Fig.9(b)(c)(d)(g)). These cases often occur in general videos.

The drawback of our method is that it is unstable when (1) the height of stroke-crowded text is less than 10 pixels (Fig. 9(k)), and (2) one text sub-image contains two different polarities (Fig. 9(j)). Future work will be done to solve these problems.

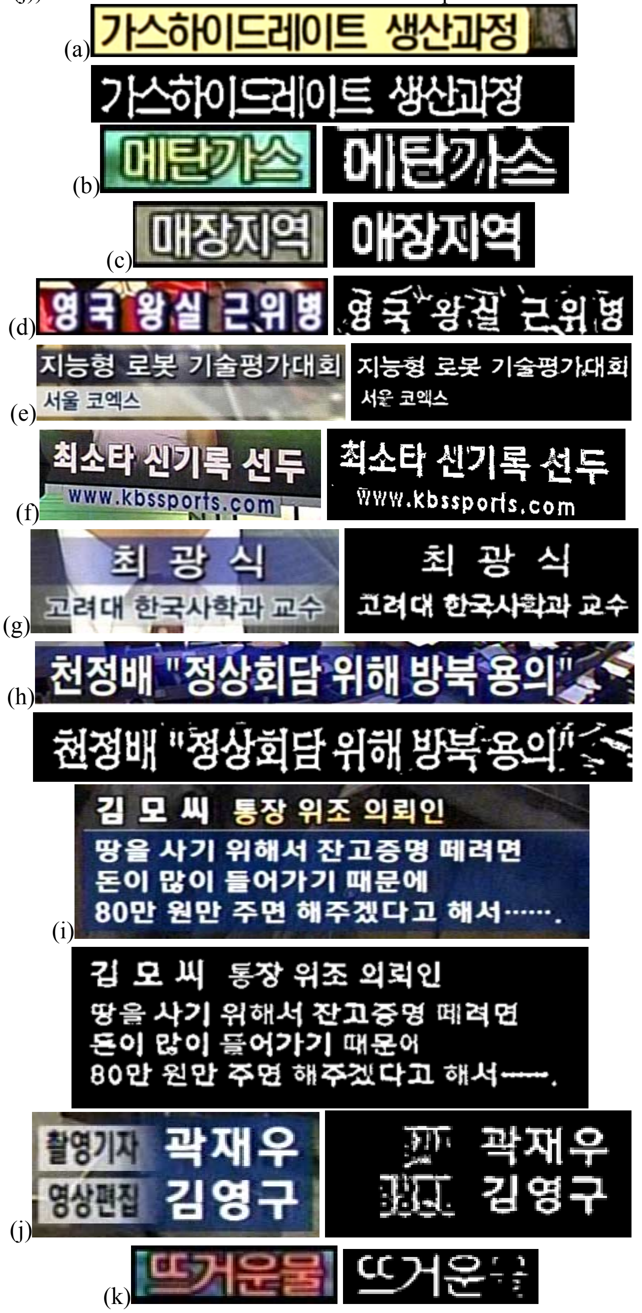


Figure 9. Representative results of our method.

6. CONCLUSION

In this paper, we propose stroke filter and demonstrate that stroke filter can play an important role in text segmentation in video images. Our method consists of three parts: stroke filtering, text polarity determination and local region growing. Experiments show that our method is robust against many challenging difficulties in video images.

7. REFERENCES

[1] K. Jung, et al, "Text Information Extraction in Images and Video: A Survey," *Pattern Recognition,* vol. 37, pp. 977-997, 2004.

[2] T. Sato, et al., "Video OCR: Indexing Digital News Libraries by Recognition of Superimposed Caption," *ACM Multimedia Systems Special Issue on Video Libraries*, Feb., 1998.

[3] N. Otsu, "A Threshold Selection Method from Gray-Scale Histogram," *IEEE Trans. On System Man Cybern*, vol. 1, pp. 62-66, 1979.

[4] F. Chang, et al., "Caption Analysis and Recognition for Building Video Indexing System," *Multimedia Systems*, vol. 10, pp. 344-355, 2005.

[5] C. Wolf, et al., "Extraction and Recognition of Artificial Text in Multimedia Documents," *Pattern Analysis and Applications*, vol. 6, pp. 309-326, 2003.

[6] K. Zhu, et al., "Using Adaboost to Detect and Segment Characters from Natural Scenes," *intl. Workshop on Camera-based Document Analysis and Recognition*, Aug. 2005

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

[8] K. Wang, et al., "Character Segmentation of Color Images from Digital Camera," *Intl. Conf. on Document Analysis and Recognition*, pp.210-214, 2001.

[9] D. Chen, et al., "Text Detection and Recognition in Images and Video Frames," *Pattern Recognition*, vol. 37, pp. 595-608, 2004.

[10] S. Antani, et al., "Video OCR for Digital News Archive," *Intl. Conf. on Pattern Recognition*, vol. 1, pp. 831-834, 2000.

[11] J. Song, et al., "A Robust Statistic Method for Classifying Color Polarity of Video Text," *IEEE intl. conf. on Acoustics, Speech, Signal Processing*, vol. 3, pp. 581-584, 2003.

[12] J. F. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. on PAMI*, pp. 679-698, 1986.

[13] D. Chen, et al., "Text Enhancement with Asymmetric Filter for Video OCR", *Intl Con. Image Analysis & Processing*, 2001.

[14] R.Lienhart and J. Maydt. "An Extended Set of Haar-like Features for Rapid Object Detection," *IEEE Conf. Image Processing*, vol. 1, pp. 900-903, Sep. 2002.

[15] F. Tupin, et al., "Detection of Linear Features in SAR Images: Application to Road Network Extraction," *IEEE Trans. on GeoScience and Remote Sensing*, vol. 36, pp. 434-453, 1998.