Extracting Captions in Complex Background from Videos

Xiaoqian Liu[1]
Graduate University of Chinese Academy of Sciences[1]
Beijing, China
xqliu@jdl.ac.cn

Weiqiang Wang[1,2]
Key Lab of Intell. Info. Process.Inst. of Comput.Tech.,[2]
CAS, Beijing, China
wqwang@jdl.ac.cn

Tingshao Zhu[1]
Graduate University of Chinese Academy of Sciences[1]
Beijing, China
tszhu@gucas.ac.cn

***Abstract*—Captions in videos play a significant role for automatically understanding and indexing video content, since much semantic information is associated with them. This paper presents an effective approach to extracting captions from videos, in which multiple different categories of features (edge, color, stroke etc.) are utilized, and the spatio-temporal characteristics of captions are considered. First, our method exploits the distribution of gradient directions to decompose a video into a sequence of clips temporally, so that each clip contains a caption at most, which makes the successive extraction computation more efficient and accurate. For each clip, the edge and corner information are then utilized to locate text regions. Further, text pixels are extracted based on the assumption that text pixels in text regions always have homogeneous color, and their quantity dominates the region relative to non-text pixels with different colors. Finally, the segmentation results are further refined. The encouraging experimental results on 2565 characters have preliminarily validated our approach.**

Keywords- captions; extracting; temporal features;

I. INTRODUCTION

As the Internet develops rapidly, the digital media resources from text, images, to videos have become more and more abundant today. Captions in TV programs and movie subtitles contain a great deal of semantic information which is very useful in video content indexing and retrieval. Thus, many researchers have been investigating text localization, segmentation and optical character recognition (OCR), and try to extend the related techniques to more applications such as electronic eye, traffic monitoring. Text extraction in this paper refers to the process which extracts text pixels corresponding to text embedded in images and video to provide a valid input of binary image for the OCR module. Generally, the process consists of text localization, segmentation, as shown in Figure 1.

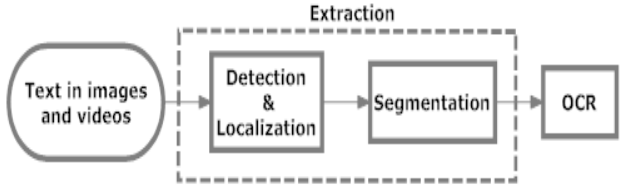


Figure 1. Illustration of text extraction.

So far, much progress has been made in text extraction. The main methods can be categorized into five categories according to different features associated with text utilized. Concretely, (1) Connected component analysis(CCA) methods. This category of methods [1][2] assume the pixels in text regions have homogeneous color, intensity, texture, so the clustering techniques are applied to identify connected components corresponding to character strokes. The color-based methods are simple, and very suitable in simple background. (2) Edge-based methods [3][4], which detect text regions by locating intensive edges. The edge-based methods are generally simple and effective, and most homogeneous regions are excluded. But the poor outcome possibly occurs if a large number of edges appear in background. Chen *et al.* [3] utilized the canny detector to find the edges. Liu *et al.* [4] introduced adaptive criterion to enhance the canny detector. (3) Corner-based methods [5][6]. For example, Zhong *et al.* [5] utilized the intensity information to extract corner, Hua *et al.* [6] utilized the SUSAN corner detector[7] to generate the corresponding corner reflection. The corner based methods are more effective, but detecting corners generally is a time-consuming task. (4) Texture-based methods [8][9][10]. This category of method assumes text regions have some kind of special textures. Many different techniques, such as Gabor filtering, FFT and wavelet transform, were investigated to extract the texture feature, and then the machine learning methods (neural networks, SVM) were applied to classify text and non-text regions. The texture-based methods are time-consuming and sometimes influenced by the fonts and styles of characters. (5) Stroke-based methods [11][12][13]. The stroke methods capture the intrinsic characteristics of text strokes, so the better detection results have been obtained even in complex background.

This paper presents a new caption extraction method in complex background for videos, in which the information of edges, color and strokes is integrated into the detection and segmentation process. Compared with the existing methods, our method truly utilizes the unique temporal feature of videos, and temporally segments a video into different clips. Our method aims to guarantee each clip contains a same caption, so that the efficiency and accuracy of text detection and segmentation can be greatly improved.

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.790

IEEE computer society

The rest of this paper is organized as follows. Section 2 presents our method. The experimental results are reported in Section 3. Section 4 concludes the paper.

II. OUR METHOD

The framework of our approach is shown in Figure 2. Different from images with embedded text, a video contains a great number of captions, so it is not an efficient and smart way to independently detect and segment captions for each video frame. In the framework of our approach, the temporal segmentation of captions is first performed to decompose a video into a sequence of clips, and we expect each clip has a same caption at most. The temporal segmentation brings two advantages. First, it makes the extraction of captions from video very efficient, due to no need to detect and segment captions for each frame. Second, since the information of multiple frames, instead of one frame, can be integrated to localize and segment a caption, more accurate results are expected. Section 2.1 details the temporal segmentation computation. Then, the localization and spatial segmentation of captions on each generated clip are performed, which are described in Section 2.2. Finally, the extraction results are further refined, and the related details are given in Section 2.3.

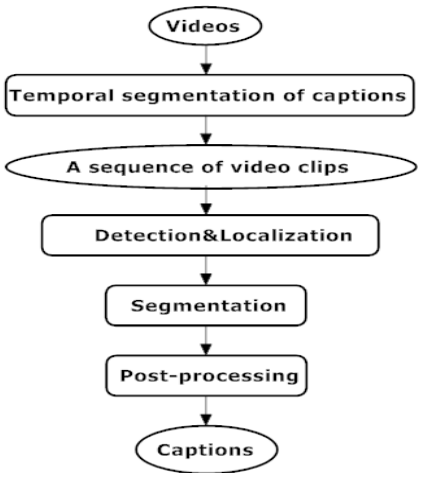


Figure 2. Framework of our approach.

A. Temporal Segmentation of Captions

A great number of captions exist in movie videos. Generally text regions are associated with intensive edges, and each subtitle in videos always lasts for a few seconds. Thus, our method identifies those regions with intensive edges, and locates the boundaries between different captions by verifying whether the spatial distribution of edge pixels changes greatly.

Our system samples three frames per second, and applies the canny detector to locate the edges in the video frames. For each edge pixel, the gradient direction is estimated based on Sobel operator (Figure 3). Let G_x and G_y denote the outputs of Sobel filters in the x-axis and the y-axis direction respectively, the direction of gradient is computed by

$$\theta = tan^{-1}(G_y/G_x). \quad (1)$$

If θ equals 0, it means a vertical edge between the left dark pixels and the right bright pixels. Figure 4 give an example to show the related results.

Figure 3. (a) The original image (b) Canny edge image (c) Edges generated by horizontal Sobel operator (d) Edges generated by vertical Sobel operator.

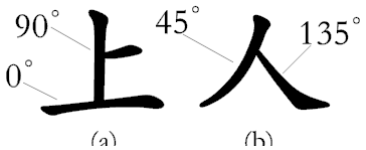


Figure 4. Gradient directions of strokes.

Further, the gradient directions are uniformly quantized into eight representative directions, i.e., 0°, 45°, 90°, 135°, 180°, -135°, -90°, -45°, which correspond to the following eight ranges [-22.5°,22.5°), [22.5°,67.5°), [67.5°, 112.5°), [112.5°, 157.5°), [157.5°,180°)∪[-180°,-157.5°), [-157.5°, -112.5°), [-112.5°, -67.5°), [-67.5°, -22.5°), as shown in Figure 4. Since text is composed of strokes, the change of subtitles correspondingly results in the change of the directions of strokes. So we use the histogram of the gradient directions of edge pixels to characterize the captions. Let h_t^i, $h_{t'}^i$ denote the number of the pixels in the i^{th} bin, $i = 1,2,\dots,8$ for two successive frames at time t and t' respectively, and we choose the Euclidean distance S to calculate the difference of the distribution of gradient directions between two frames, i.e.,

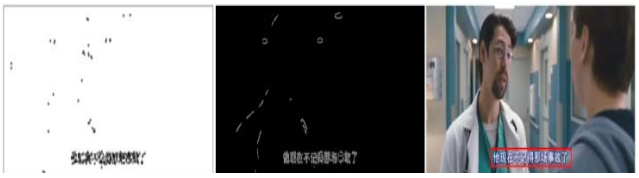
$$S = \sqrt{\sum_{i=1}^{8}(\overline{h_t^i} - \overline{h_{t'}^i})^2}\,, \quad (2)$$

where $\overline{h_t^i}$, $\overline{h_{t'}^i}$ are the i^{th} bin of the normalized versions $\overline{h}_t, \overline{h}_{t'}$ corresponding to histogram $h_t, h_{t'}$. If $S > T_s$, where T_s is a predefined threshold, the captions in the two frames are different. Otherwise, they share the same caption. We choose $T_s = 0.018$ in our experiment.

B. Localization and Spatial Segmentation

Besides edges, corners in text regions are also widespread, since the strokes with different directions often cross each other, which is particularly true for Chinese

characters. Figure 5 gives an example. So we can detect and localize text regions by analyzing the spatial distribution of edge pixels and corners.



(a) (b) (c)

Figure 5. (a) Detected corners marked by the black dots. (b) Edges extracted from the frame. (c) Text region marked by the red line.

Concretely, each video frame is partitioned into 32 by 32 pixels blocks, and then the number of corners and edge pixels within each block is counted. For a pixel block, if the number of corners N_c and the number of edge pixels N_e meet $N_c > T_c$, $N_e > T_e$, where T_c, T_e are two predefined thresholds respectively, the block is marked as a text block. Finally, all the blocks marked as text blocks are merged into a caption region. The caption region is often a rectangular area at the bottom of a video frame, and the following constraints or strategies are exploited to direct the merging procedure,

1) The width of a caption area must be greater than or equal to the height.
2) Start merging procedure from the marked text block at the middle and the lowest of a frame, and extend the mergence outwards step by step.
3) Two text blocks can be merged, only if they are adjacent to each other horizontally or vertically.

Once a caption region is localized, our system adopts a simple and efficient color-based method to spatially segment text pixels. We assume that text pixels in caption regions always have homogeneous color, and their quantity dominates the region relative to non-text pixels with different colors. First, the color histogram of the pixels in caption regions is computed in the RGB color space. Each channel is uniformly quantized into four bins, so the whole color space is partitioned into 64 cubes. For each video clip generated by the procedure in Section 2.1, the 64-bins color histogram $\{H_i | \, i = 1,2,\dots,64\}$ is computed, and the color cube k with the largest number of pixels are identified , i.e.,

$$H_k \geq \forall H_j (j = 1,2,\dots,64). \quad (3)$$

Then H_k is considered as a reasonable estimation of text color range. So the pixels in the k^{th} cube are considered as candidate text pixels.

Our experiences show caption pixels generally have high brightness so that they can easily stand out from background. Thus, we also take into account the brightness factor and the pixel brightness values are calculated by

$$L = 0.299 * R + 0.587 * G + 0.114 * B. \quad (4)$$

For each candidate text pixel p, if its brightness $L(p)$ exceeds a predefined threshold T_L, the pixel p will be labeled as text pixels. As a result, a binary image is obtained in which the pixel value 255 represents text pixels and 0 represents non-text pixels.

C. Post-processing

In the output binary image, there still possibly exist some false stroke areas or text pixels. A post-processing procedure is designed to further refine the segmentation results. The refinement process follows the following observations:

1) There should be some constraints for the width of strokes. Some gaps must exist between different characters and the strokes of a character.
2) A certain number of edge pixels and corners distribute a little evenly.
3) A stroke has continuity.

Concretely, a sliding window is set up, and the size of the sliding window depends on the size of characters. At least, the length of its side should be larger than the width of strokes. In our implementation, an 8 by 8 window is chosen. At each sliding position in caption regions, the number of text pixels in the window is counted. According to the first observation, too many text pixels means it corresponds to a potential background, and the pixels in the window are labeled as background. As described in the second observation, if it contains fewer corners or edge pixels, the area is also marked as background. Finally, the morphological dilation operation is performed to link broken strokes and fill missing pixels, so that strokes in the segmentation results look smooth and continuous.

III. Experimental Results

To evaluate the validity of our proposed approach, we select representative 22 videos with complex background as the experimental data set. The videos in our data set cover multiple different resolutions, such as 640*272, 576*324,…, 880*492,1024*576, different sizes of captions, as well as different overlap styles. The character set in the videos involves Chinese, English, punctuation symbols, and digits. Each video lasts from 20~35 seconds. Some experimental results are shown in Figure 6. The first row shows the original images, and the second row gives the corresponding segmentation results.

We directly input our segmentation results into the Hanwang OCR software, and evaluate the performance of our approach by the recognition rate of our segmentation results. The character recognition rate Q is defined as

$$Q = \frac{The\ \ number\ of\ characters\ correctly\ recognized}{The\ total\ number\ of\ characters}. \quad (5)$$

82 frames are randomly sampled from 22 videos as the input of OCR software. The final experimental results are summarized in Table 1. The results show that the recognition rate of Chinese characters is higher than English letters and digits. A possible explanation is that Chinese characters have more complicated structure, so more

recognition features can be exploited. So, slight segmentation error cannot result in a great decrease of recognition rate. But English letters have comparatively fewer strokes, and high similarity between the letters, such as ‘e’ and ‘c’, ‘h’ and ‘n’, etc, also makes them more sensitive to segmentation errors. Punctuation symbols have very small size, and simple structure, which makes them more difficult to be recognized under the same error. Another potential factor is that English letters have smaller font size (i.e., lower resolution) than Chinese characters in our dataset.

TABLE I. THE RECOGNITION RESULTS (N DENOTES THE TOTAL NUMBER OF THE CHARACTERS, R DENOTES THE NUMBER OF THE CHARACTERS CORRECTLY RECOGNIZED)

	Chinese characters	**English Letter/Digits**	**punctuation symbol**	**sum**
N	930	1542	93	2565
R	919	1435	79	2493
Q	98.82%	93.06%	84.95%	94.85%

IV. CONCLUSION

This paper presents a new caption extraction method in complex background for videos, in which the information of edges, color and strokes is integrated into the detection and segmentation process. Compared with the existing methods, our method truly utilizes the unique temporal feature of videos, and temporally segments a video into different clips. The encouraging experimental results (94.85% averagely) on 2565 characters have preliminarily validated our approach.

ACKNOWLEDGMENT

This work was supported in part by National Natural Science Foundation of China under Grant 60873087 and by National Key Technologies R&D Program under Grant 2006BAH02A24-2.

REFERENCES

[1] J. Yi , Y. X. Peng and J. G. Xiao, “Color-based clustering for text detection and extraction in image”, Proceedings of the 15th international conference on Multimedia, pp.25-29, September 2007.

[2] S. C. Pei and Y. T. Chuang, “Automatic Text Detection using Multi-layer Color Quantization in Complex Color Images”, In Proc. of 2004 IEEE International Conference on Multimedia and Expo. Taipei, Taiwan, vol.1, pp.619-622, 2004.

[3] D. Chen, K. Shearer and H. Bourlard, “Text Enhancement with Asymmetric Filter for Video OCR”, In Proc. of International Conference on Image Analysis and Processing, pp. 192-197, 2001.

[4] Y. Liu, H. Lu, X. Y. Xue and Y. P. Tan, "Effective video text detection using line features", In Proc. 8th International Conference on Control, Automation, Robotics and Vision, Vol. 1, pp. 1528-1532, 2004.

[5] Y. Zhong, H. Zhang and A.K. Jain, “Automatic Caption Localization in Compressed Video”, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 22, No. 4, pp. 385-392, 2000.

[6] X.S. Hua, X.R. Chen, W.Y. Liu and H.J. Zhong, “Automatic location of text in video frames”, In Proc. of the 3rd Intl Workshop on Multimedia Information Retrieval, Ottawa, Canada, October, 2001.

[7] S. M. Smith and J. M. Brady, “SUSAN- A New Approach to Low Level Image Processing”, Int. Jour. of Computer Vision. 23(1), pp. 45-78, May 1997.

[8] B. Sin, S. Kim, and B. Cho, “Locating Characters in Scene Images using Frequency Features”, In Proc. of International Conference on Pattern Recognition, Vol. 3, pp. 489-492, 2002.

[9] I. Ar, and M. E. Karsligli, "Text Area Detection in Digital Documents Images Using Textural Features", CAIP, LNCS 4673, Springer-Verlag, 555-562, 2007.

[10] W. Mao, F. Chung, K. K. M. Lam and W. Siu, “Hybrid Chinese/English Text Detection in Images and Video Frames”, ICPR, Volume 3, pp. 1015- 1018, 2002.

[11] Q. Liu, C. Jung and Y. Moon, “Text segmentation based on stroke filter”, In Proc. of the 14th Annual ACM international Conference on Multimedia (Santa Barbara, CA, USA, October), NY, pp. 129-132, 2006.

[12] Q. Liu, C. Jung, S. Kim, Y. Moon and J. Kim, “Stroke filter for text localization in video images”, IEEE Conf. Image Processing, pp. 1473–1476, 2006.

[13] V.C. Dinh, S. S. Chun, S. cha, H. Ryu, and S. Sull, “An Efficient Method for Text Detection in Video Based on Stroke Width Similarity”, ACCV, Part I, LNCS 4843, pp. 200–209, 2007.

Figure 6. Some examples of captions extracted.