Extracting Captions from Videos Using Temporal Feature

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

ABSTRACT
Captions in videos provide much useful semantic information for indexing and retrieving video contents. In this paper, we present an effective approach to extracting captions from videos. Its novelty comes from exploiting the temporal information in both localization and segmentation of captions. Since some simple features such as edges, corners and color are utilized, our approach is efficient. It involves four steps. First, we exploit the distribution of corners to spatially detect and locate the caption in a frame. Then the temporal localization for different captions in a video is performed by identifying the change of stroke directions. After that, we segment the caption pixels in a clip with a same caption based on the consistency and dominant distribution of caption color. Finally, the segmentation results are further refined. The experimental results on two representative movies have preliminarily verified the validity of our approach.

Categories and Subject Descriptors
H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing – *Indexing methods*;
I.5.4 [**Pattern Recognition**]: Applications – *Text processing.*

General Terms
Algorithms, Performance, Design, Experimentation.

Keywords
Caption Localization, Caption Extraction.

1. INTRODUCTION
As the technology of digital multimedia develops rapidly, a large number of videos spring up in our daily lives. The superimposed captions in TV programs and movies on the Internet (e.g., videos with Real format) contain much useful semantic information for video content indexing and retrieval. Thus, caption extraction has become a hot topic recently. The researchers have summarized the related key techniques into three components: localization, segmentation and optical character recognition (OCR), as shown in Figure 1. Generally, text extraction involves detection, localization, and segmentation of captions. Detection and localization refer to judging the existence of captions and marking the corresponding regions; Segmentation aims to identify the pixels of captions from located caption regions to generate a binary image for optical character recognition (OCR) [1].

MM'10, October 25–29, 2010, Firenze, Italy.

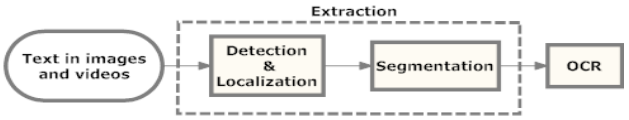

Figure 1. Illustration of text extraction.

For this topic, many methods have been proposed and they can be summarized into four categories according to the distinctive features utilized. Dense edges and corners are widely used by many researchers. For instance, Wong *et al.* [2] and Liu *et al.* [3] detect text regions by locating intensive edges; Hua *et al.* [4] utilize the SUSAN detector to find corners, and then identify the text regions based on the distribution of corners. Edge-based and corner-based methods are simple and easy to implement, but the high false alarms occur when complex backgrounds also contain many edges and corners. The color-based methods are also very popular. This kind of method is very effective for the videos in which the color of captions and background is much different, especially in the case of simple background. Hase *et al.* [5] gather the regions with similar color based on connected component analysis (CCA). Clustering techniques are widely used in the color-based method. Another category is the stroke-based method, in which the structure of stroke is utilized in text localization and extraction. In [6][7], the authors design a stroke filter to find the caption regions. Dinh *et al.* [8] localize the text areas based on the similar width of strokes. The text structure can also be treated as a kind of texture, and text regions are identified by a texture classifier. Jain *et al.* [9] use the Gabor filter to extract the texture feature. Many machine learning methods (neural networks [10], SVM [11]) have been applied in classifying text and non-text regions. The texture-based methods are relatively time-consuming and sometimes influenced by the fonts and styles of characters.

All the methods mentioned above are based on the spatial features. For captions in videos, besides the spatial features widely used for detecting text in static images, some researchers introduce the temporal features to detect and segment the captions in videos. Tang *et al.* [12] present a novel caption-transition detection scheme, which locates both spatial and temporal positions of video captions with high precision and efficiency. Tang *et al.* [13] compare the pixel values in the same location in consecutive frames, and those pixels with very small change are considered as caption pixels. The segmentation method in [13] cannot work well when the background is also unchanged in a clip containing the same caption. In this paper, we propose a new caption extraction method for videos, in which the temporal feature in both localization and segmentation is utilized, and our method also works well even when the background in the located caption region keeps unchanged. The whole approach only uses the simple spatial features such as corners, edges and color, which makes it very efficient and suitable for videos.

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.

2. OUR METHOD

Unlike texts in static images, a same caption in videos always last for a sequence of frames. We take advantage of this temporal redundancy in both localization and segmentation. Figure 2 shows the framework of our approach. To make our system more efficient, a video is uniformly sampled by 3 frames per second, and all the computations are performed on the sampled frames. The localization in our approach involves two aspects, spatial localization and temporal localization. First, our system detects and spatially locates the candidate caption regions for each sampled video frame by exploiting the distribution of corners. Then the temporal localization of captions is performed to decompose the video into a sequence of clips, and each clip contains a same caption. After that, a caption segmentation procedure based on the clips, not individual frames, and some post-processing computation, are performed to generate the suitable form, a binary image, for OCR modules. This clip-based segmentation makes the generated result more accurate and reliable by integrating the information from multiple frames.

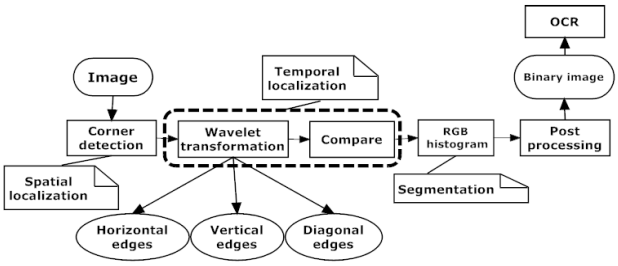

Figure 2. Framework of our approach.

2.1 Spatial Localization

Characters are made up of strokes, so some features associated with the structure of strokes, such as edges, corners and uniform color of stroke pixels, can be utilized to locate the candidate caption regions. As shown in Figure 3, the text region marked by rectangles generally contains many corners distributing densely labeled by the black spots in Figure 3(b).

Many methods can be chosen to detect the corners, such as the SUSAN detector [14] and the Harris detector. We adopt the method presented by [15]. Concretely, we consider a m *by* m neighborhood $S(p)$ of a pixel p, and calculate its difference correlation matrix D_p by

$$D_p = \begin{bmatrix} \sum_{S(p)}(\frac{dI}{dx})^2 & \sum_{S(p)}(\frac{dI}{dx}\cdot\frac{dI}{dy}) \\ \sum_{S(p)}(\frac{dI}{dx}\cdot\frac{dI}{dy}) & \sum_{S(p)}(\frac{dI}{dy})^2 \end{bmatrix}, \quad (1)$$

where I denotes a video frame. Let λ_{min}^p denotes the minimum eigenvalue of matrix D_p. The pixel p is declared as a corner if $\lambda_{min}^p > Q_c$, where $Q_c = 0.05 * \max_p \lambda_{min}^p$. We choose $m = 3$ in computing D_p.

To localize caption regions, we first divide a frame into *M*N* blocks uniformly as shown in Figure 4 (a). The value of *M* and *N* are related with the image size in videos, and a larger image size corresponds to larger *M* and *N*. For our experimental data, we choose *M=8, N=8,* so each frame is divided into 64 blocks. The number of corners in each block is denoted by $N_i, i = 1, \ldots, MN$).

Apparently, the text region contains quite a number of corners. As shown in Figure 4 (b), if quite a number of corners occur in the i^{th} block and its number exceeds a predefined threshold T_c, i.e., $N_i \geq T_c$, the i^{th} block is labeled as a text block. In our experiment, we choose $T_c = N_p * 0.003$, where N_p denotes the number of pixels in a block. Generally, captions in one frame are spatially continuous, so we discard the isolated text blocks and incorporate the remainders as the caption regions. The boundaries of the rectangle regions generated by incorporation are taken as the boundary of caption region.

Figure 3. (a) Original frame. (b) Corners detected for (a).

Figure 4. (a)The partition of a frame. (b)The distribution of corners.

2.2 Temporal Localization

In a video, two different captions appearing in order can be temporally separated by a clip composed of the frames with no caption, or they are continuous. The temporal localization aims to find the boundaries of consecutive different captions, so that a video can be decomposed into the clips, and each of them contains no frame, or a same caption. For the former case, using the techniques described in section 2.1, the clip with no caption can be identified. For the remaining part of a video, the temporal localization is completed by detecting whether caption change occurs between two sampled frames.

Each subtitle in videos always lasts for a few seconds, so the characters are unchanged within the caption region in the short time interval. The characters, especially the Chinese, are composed of strokes, so we can exploit the variation of strokes to judge whether the captions change or not. We assume that the characters primarily consist of four types of basic strokes, i.e., horizontal strokes, vertical strokes, up-right-slanting and up-left-slanting strokes. In these four directions, the stroke segments contain rich high frequency energy. We use a single level Haar wavelet transform W_H to decompose a caption region B_t into four components, an approximation component $A_t(u, v)$ to the original image and three detail components $H_t(u, v), V_t(u, v)$, $D_t(u, v)$ corresponding to horizontal, vertical and diagonal directions respectively. We count the pixels which have response values in three detail components $H_t(u, v), V_t(u, v), D_t(u, v)$ to form a 3-bin histogram $h = \{N_i | i = 1,2,3\}$, where N_i denotes the number of pixels in the i^{th} bin.

In Figure 5, the different color histograms correspond to two caption regions in consecutive frames respectively. Obviously, the histograms for a same caption are similar, and different captions result in very different histograms. To facilitate the comparison,

we compute the normalized histogram H by $H \stackrel{\text{def}}{=} \{H(i)|H(i) = N_i/N$, and $N = \sum_i N_i\}$. Then the similarity of two normalized histograms H_t and H_{t+1} is defined by the histogram intersection,

$$S(H_t, H_{t+1}) = \sum_i min\,(H_t(i), H_{t+1}(i)). \quad (2)$$

If the similarity of the histograms of two consecutive frames is lower than a predefined threshold T_h, i.e., $S(H_t, H_{t+1}) < T_h$, our system will consider a new caption appears at time *t+1*. Then frame *t* is the last frame for the current caption, and frame *t+1* is the start frame for a new caption. The results of temporal localization are utilized for the following character segmentation.

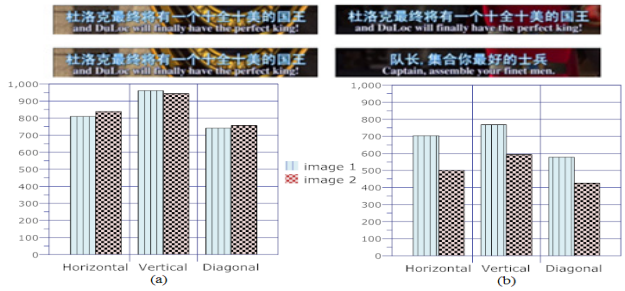

Figure 5. Compare two caption regions of consecutive frames. (a) In the same clip. (b) In different clips.

2.3 Caption Segmentation

Our system adopts a simple and efficient color-based method to segment text pixels. For each clip with caption generated by temporal localization, text pixels in caption regions always have homogeneous color both spatially and temporally, and their quantity should dominate the region relative to non-text pixels with different colors. In order to identify text colors more easily from the histogram of pixel colors in caption regions, more frame samples are desirable. Due to efficiency consideration, we only random sample another frame, besides the first frame and the last frame of a clip, to compute the color distribution of pixels in the caption region. In the RGB color space, we uniformly quantize each channel into eight bins to partition the whole color space into 512 cubes $C_k(k = 1,2,\dots,512)$. For each video clip generated by temporal localization, the 512-bin color histogram $\widetilde{H}(k)$ $(k = 1,2,\dots,512)$ of caption regions in the frames selected is calculated. The color cube C_i representing the color range of text pixels is identified by,

$$\widetilde{H}(i) \geq \forall \widetilde{H}(j) \;\; (j = 1,2,\dots,512). \quad (3)$$

So the pixels in the i^{th} cube are considered as candidate text pixels. In addition, our experiences show caption pixels generally have relatively high brightness so that they can easily stand out from background. Thus, we also take into account the pixel brightness in the segmentation of text pixels. Concretely, we first calculate the brightness L for each pixel in C_i by

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

Then, if the brightness of a pixel exceeds a predefined threshold T_L (in our system, $T_L = 160$), the pixel will finally be classified as text pixel. So a binary image (255 for text pixels and 0 for background pixels) for each clip is obtained by applying logical AND operation on the segmentation results of each frame selected.

2.4 Post-processing

No matter what kind of segmentation method is used, the results still contain some false stroke areas or text pixels, which will affect the recognition rate of OCR. A post-processing procedure is usually required to further refine the segmentation results.

The gaps always exist among different strokes of a same character, as well as between different characters, it is true for both Chinese characters and English characters. That means text pixels are always mixed with non-text pixels locally. So we set up a sliding window, 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 so that it can cover different strokes or different characters. 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. Since too many text pixels in sliding window means there is no gap between characters or strokes, all the pixels in the window are labeled as background and the sliding window moves forward 8 pixels. On the other hand, the sliding window will move forward only three pixels.

In the segmentation of text pixels described above, each pixel is classified independently, but the strokes of a character are always continuous. So the morphological dilation operation is performed to link broken strokes and fill missing pixels.

3. EXPERIMENTAL RESULTS

To evaluate the validity of our proposed approach, we select two representative movies as test dataset. Each video lasts from 30 to 40 minutes. One has complex background, and the other has simple background. Their resolutions are both 1024*576. The character set in each video involves Chinese, English, punctuation symbols, and digits. Some examples of experimental results are shown in Figure 6. The first and second rows show the original images, and the third row gives the corresponding segmentation results. Each column corresponds to a clip in the two videos respectively.

First, we evaluate the performance of temporal localization by two metrics, precision *P* and recall *R*, and they are defined as,

$$P = N_c/N_d \;\;,\;\; R = N_c/N_g \quad (5)$$

where N_c denotes the correct number of boundaries detected, N_d is the total number of boundaries detected by our system, and N_g denotes the number of boundaries labeled manually. In labeling, the boundaries involve three cases: (1) from a caption to a different caption; (2) from no caption to a caption; (3) from a caption to no caption. The two movies contain 589 and 539 boundaries respectively. The experimental results are summarized in Table 1. Obviously, the localization results for video with simple background are better than those for complex background video. Further analysis of experimental results shows that many edges associated with complex background are mistaken as strokes.

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

$$Q = \frac{The\ \ number\ of\ characters\ correctly\ recognized}{The\ total\ number\ of\ characters}. \quad (6)$$

In our experiments, we random sample the segmented captions and input them to the OCR module. The corresponding experimental results are summarized in Table 2. Obviously, the recognition rate for simple background is higher than that for complex background video, since better segmentation is expected to be obtained in simple background. In addition, the complex

background may contain the pixels with the similar color as captions, and it is very difficult for our method to identify them. But the acceptable results are generated for different categories of characters, Chinese, English, punctuation symbols, and digits. A little lower recognition rate for English letters results from high similarities between English letters, for example, 'v' and 'y', 'e' and 'c'. In addition, the size of English characters in our dataset is smaller than the size of Chinese characters. The low resolution makes precise segmentation more difficult. The imprecise segmentation is easy to make them indiscernible.

Figure 6. Some examples of captions extracted.

Table 1. The temporal localization results

	N_g	N_d	N_c	P	R
Complex background	589	603	547	0.907	0.929
Simple background	539	553	508	0.919	0.942

Table 2. The recognition results (N_t denotes the total number of the characters, N_c denotes the number of the characters correctly recognized)

		N_t	N_c	Q
Simple background	Chinese characters	377	377	1.000
	English Letter	1086	982	0.904
	Punc. Symb./Digits	163	160	0.982
	Sum	1626	1519	0.934
Complex background	Chinese characters	755	709	0.939
	English Letter	2442	2265	0.928
	Punc.Symb./Digits	250	230	0.920
	Sum	3447	3104	0.900
Sum(Two videos)		5073	4623	0.911

4. CONCLUSION

In this paper, we propose an efficient and effective caption extraction method for videos, in which we exploit the temporal information in caption localization and segmentation. Our approach can still work well when background in caption regions keep unchanged. The encouraging experimental results (91.1% averagely) on 5073 characters have preliminarily validated our approach.

5. ACKNOWLEDGMENTS

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.

6. REFERENCES

[1] K. Jung, K. I. Kim, and A. K. Jain, "Text information extraction in images and video: A survey", Pattern Recognit., vol. 37, no. 5, pp. 977-997, May 2004.

[2] E. K. Wong and M. Chen, "A new robust algorithm for video text extraction", Pattern Recognition 36, pp.1397-1406, 2003.

[3] C. Liu, C. Wang and R. Dai, "Text Detection in Images Based on Unsupervised Classification of Edge-based Features", IEEE ICDAR, pp. 610-614, 2005.

[4] 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.

[5] H. Hase, T. Shinokawa, M. Yoneda, and C. Y. Suen, "Character String Extraction from Color Documents", Pattern Recognition, 34 (7), pp.1349-1365, 2001.

[6] X. L. Li, W. Q. Wang, S. Q. Jiang, Q. M. Huang and W. Gao, "Fast and Effective Text Detection", IEEE International Conference on Image Processing, San Diego, California, U.S.A., pp.969-972, Oct. 2008.

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

[8] V. C. Dinh et al, "An Efficient Method for Text Detection in Video Based on Stroke Width Similarity", ACCV, Part I, LNCS 4843, pp. 200–209, 2007.

[9] A. K. Jain and S. Bhattacharjee, "Text Segmentation using Gabor Filters for Automatic Document Processing", Machine Vision and Applications, vol.5, pp.169-184, 1992.

[10] S. H. Park, K. I. Kim, K. Jung, and H. J. Kim, "Locating Car License Plates using Neural Networks", IEE Electronics Letters, 35 (17), pp.1475-1477, 1999.

[11] K. C. Jung, J. H. Han, K. I. Kim, and S. H. Park, "Support vector machines for text location in news video images", in Proc. IEEE Region 10 Conf. Syst. Technolog. Next Millennium, vol. 2, pp. 176–180, 2000.

[12] X. Tang, X. Gao, J. Liu, and H. J. Zhang, "A spatial temporal approach for video caption detection and recognition", IEEE Trans. on Neural Networks, special issue on intelligent multimedia processing, July, 2002.

[13] X. Tang, B. Luo, X. Gao, E. Pissaloux, and H. Zhang, "Video text extraction using temporal feature vectors", in Proc. of IEEE International Conference on Multimedia and Expo, Lausanne, Switzerland, Aug. 2002.

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

[15] J. Shi, C. Tomasi, "Good features to track", 9th IEEE Conference on Computer Vision and Pattern Recognition, June 1994.