Accurate Overlay Text Extraction for Digital Video Analysis

Dongqing Zhang, and Shih-Fu Chang

Electrical Engineering Department,
Columbia University, New York, NY 10027.
(Email: dqzhang, sfchang@ee.columbia.edu)

Abstract

This report describes a system to detect and extract the overlay texts in digital video. Different from the previous approaches, the system used a multiple hypothesis testing approach: The region-of-interests (ROI) probably containing the overlay texts are decomposed into several hypothetical binary images using color space partitioning; A grouping algorithm then is conducted to group the identified character blocks into text lines in each binary image; If the layout of the grouped text lines conforms to the verification rules, the bounding boxes of these grouped blocks are output as the detected text regions. Finally, motion verification is used to reduce false alarms. In order to achieve real time speed, ROI localization is realized using compressed domain features including DCT coefficients and motion vectors in MPEG videos. The proposed method showed impressive results with average recall 96.9% and precision 71.6% in testing on digital News videos.

1. Introduction

Videotext detection and recognition has been identified as one of the key components for the video retrieval and analysis system. Videotext detection and recognition can be used in many applications, such as semantic video indexing, summarization, video surveillance and security, multilingual video information access, etc. Videotext can be classified into two broad categories: Graphic text and scene text. Graphic text or text overlay is the videotext added mechanically by video editors, examples include the news/sports video caption, movie credits etc. Scene texts, are the videotexts embedded in the real-world objects or scenes, examples include street name, car license number, the number/name on the back of a soccer player. This report is to address the problem of accurately detecting and extracting the graph videotexts for videotext recognition.

Although the overlay text is manually added into the video, the experiments showed they are even as hard to extract as many video objects, such as face, people etc. This is due to the following reasons: 1. Many overlay texts present in the cluttered scene background; 2. There is no consistent color distribution for texts in different videos. Consequently, the color-tone based approach widely used in face or people detection application actually cannot be applied in text detection.; 3. The size of the text regions may be very small such that when the color segmentation based approach is applied, the small text region may merge into the large non-text regions in its vicinity.

There has been much prior work for videotext detection and extraction. T.Sato et al. [1] investigated superimposed caption recognition in News video. They use spatial differential filter to localize the text region, and size/position constraints to refine the detected area. Their algorithm is used in the specific domain, such as CNN news. R.Lienhart et al. [2] gave an approach using texture, color segmentation, contrast segmentation, and motion analysis. In their system, color segmentation by region merging is done in global video frame without a localization process. Thus the accuracy of the extraction may heavily rely on the segmentation accuracy. Li et al. [3] uses Haar wavelet to decompose the video frame into subband images. Neural network is used to classify the image blocks into text or non-text based on the subband images. The Haar wavelet filtering is essentially a process of texture energy extraction. Y. Zhong [4] et al, employs DCT coefficients to localize text regions in MPEG video I-frame. They did not use color information and layout verification, therefore false alarms are inevitable in those texture-like objects, such as building, crowd. M. Bertini [5] presented a text location method using salient corner detection. Salient corner detection may produce false positives in the cluttered background without good verification process. J. Shim and C. Dorai et al. [6] uses a region-based approach for text detection. They use gray level images generated from the color video frames. And texture features are not used in detection. L. Agnihotri and N. Dimitrova.[7] uses a texture-based approach to locate the text region, without using the color segmentation or decomposition. A. Jain [8] presented a method for text location in image and video frames. They use color space decomposition and integration, texture and motion model is not used for detection. And a simple layout verification method is used based on vertical projection profile.

Some of these work did not explicitly address the problem of accurate text boundary extraction. But accurate boundary extraction of videotexts is important for recognition, because the recognition error due to ill extraction is often unrecoverable using enhancement or language/knowledge model.

Our method attempts to address the limitations of the previous systems by avoiding background disturbance and reducing the false alarms. We use a multiple hypothesis testing approaches: The region-of-interests (ROI) probably containing the overlay texts are decomposed into several hypothetical binary images using color space partitioning; A grouping algorithm then is conducted to cluster the identified character blocks into text lines in each binary image; If the layout of the grouped text lines conforms to the verification rules, the bounding boxes of these grouped blocks are output as the detected text regions. Motion verification is also used to reduce false alarms. In order to achieve real time speed, ROI localization is realized using compressed domain features including DCT coefficients and motion vectors in MPEG videos.

The overall system can be illustrated in the following diagram:

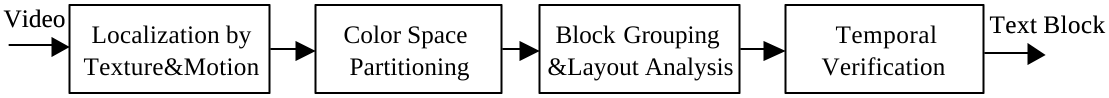

Video
Localization by Texture&Motion
Color Space Partitioning
Block Grouping &Layout Analysis
Temporal Verification
Text Block

Figure 1. System Flowchart

In the diagram, texture and motion analysis is used to localize the region-of-interests (ROIs) of the videotexts by extracting texture and motion energy from compressed domain features. The color space partitioning is to divide the HSV color space into a few of partitions, and the hypothetical binary images are generated for each partition.

Character block grouping and layout analysis cluster the character-like blocks into text regions and layout analysis is performed to verify if the text regions are able to form text lines. The temporal consistency analysis then is conducted to eliminate the false alarms staying in the video frames with too short duration.

The report is organized as follows: section 2 describes the localization algorithm to detect the Region of Interests. Section 3 describes the hypothetical binary image generation using color space partitioning. Section 4 describes a rule-based approach for character block grouping and using layout analysis to do verification. Section 5 presents the temporal verification procedure to eliminate false alarms.

2. Localization using Compressed Domain Features

A typical size of the video frame is 320x240. Without the localization of the interest region, the detection program is difficult to realize realtime speed. Furthermore, the localization using texture or motion features can filter out irrelevant regions that would result in false alarms. However, capability of the motion texture based approach to extract the accurate boundary of the text region is usually poor, therefore an algorithm relying alone on these features may not be suited for accurate text detection. The problem becomes more severe if the background is cluttered.

2.1 Texture Energy and Motion Energy

Texture features may be the most widely used features for videotext detection. The intuition behind texture based approach is videotexts are often with high contrast against their background and with sharp stroke edges. These features make the text line hold high energy in the high frequency band of the Fourier spectrum. For many videos, such as news and sports video, using texture feature alone is able to detect most of the text region from the video frames, since texts in these videos are deliberately rendered with high contrast for the audience.

The method used here is similar to that used by Y. Zhong and Jains [4]. The texture energy in the horizontal and vertical direction is extracted from the 8x8 DCT coefficient block in a MPEG-1 video:

$$\begin{aligned} E_h(x,y) &= \sum_{1\le k\le 6} |C_{0k}(x,y)| \\ E_v(x,y) &= \sum_{1\le k\le 6} |C_{k0}(x,y)| \end{aligned} \qquad (1)$$

Where *i,j* is the coordinates of a DCT block. E_h is called the horizontal Texture Energy Map and E_v is called the vertical Texture Energy Map.

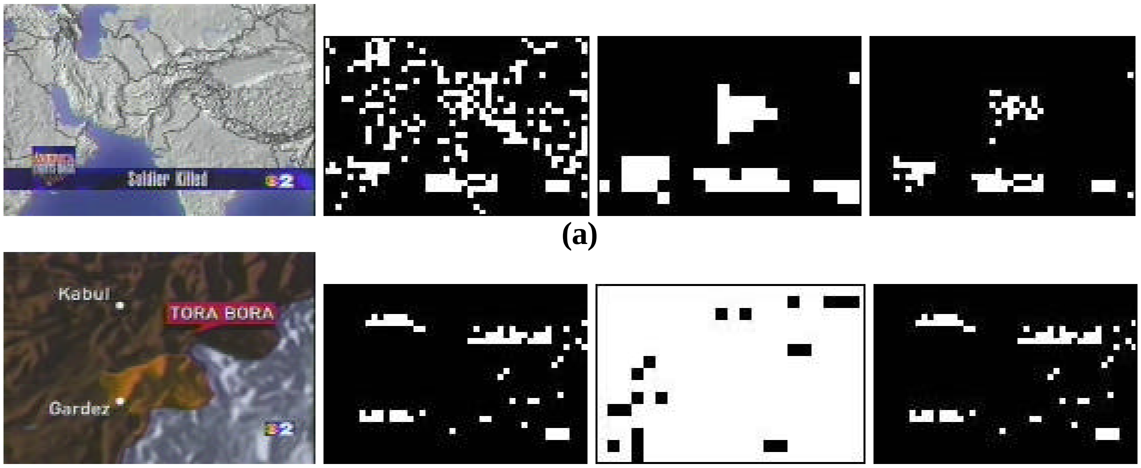
For some video genres, like news and sports, most of the videotexts are static. Thus the motion features can also be used in text extraction for those static text blocks, the method has been successfully used in [9] for extraction of sports video score box. Here a measurement called Motion Energy (ME) is used to characterize the motion intensity of the regions. ME is the length of the motion vector of each macro block in the B or P frame. All Motion Energies of macro blocks in B or P frame form Motion Energy Map (MEM), which exhibits the motion intensity at different locations in a video frame.

2.2. Combining Texture and Motion Energy

Motion energy map is often unstable due to the inaccuracy of the MPEG motion estimation algorithm. Therefore one cannot rely alone on the features extracted from the MPEG motion vector. We use a joint measurement combining both texture and motion energy. The combination occur in each I frame, where the DCT features come from the current I-frame, and the motion features come from the latest B or P frame (temporal morphological filter can be used to stabilize the MEM using multiple B and P frames). Since the motion energy map is extracted from the macro block. Their resolution is half of the texture map. In order to be combined with texture energy map, they are upscaled to the identical size with the texture map. The combination makes the high joint measurement corresponding to high texture energy and low motion energy, thus we have the following equation:

$$TM = (E_h > l_1) \wedge (E_v > l_2) \wedge (U(E_m, 2) < l_3) \qquad (2)$$

Where E_m the motion energy map, $U(E_m, 2)$ is the operator of upscaling by 2, which is actually upsampling followed by an interpolation operator. The constants l_1, l_2, l_3 are the thresholds for the horizontal, vertical texture energy and motion energy. The binarized map may be with irregular boundary, but they can be further de-noised using morphological filters. Figure 2 illustrate the texture energy map, motion energy map and combined measurements.



(a)

(b)

Fig 2. Localization by Texture-Motion Analysis. (a) Frame with Intensive motion (zoom in) (b) Frame without any motion. From left to right: original image; texture energy map (after tresholding); motion energy map (after negative, upscaling, thresholding); combined measurement (before morphological filter).

3. Color Space Partitioning for Hypothetical Binary Image Generation

The idea of color space partitioning is based on the fact that most of the text overlays are of uniform color or near uniform color. This also means the color distribution of a videotext line is very localized in the color space. On the hand, the background is often rendered with high color contrast to the text overlay. Thus partitioning the color space into a few of sub regions can separate the color layer of the text overlay from its background.

The color space based approach has been used by previous algorithms, for example A.K.Jain [8] uses color space decomposition to separate the layer of the texts in images. Also some approaches use color segmentation method in color space or gray space, such as [2][6]. A problem of color segmentation is that the number of the color clusters is hard to determine a prior. Another problem is the texts with too small sizes may be merged to the background object if the number of the color cluster is not selected properly.

Here we use a straightforward approach: we first convert the RGB color space into HSV color space. Afterwards we divide the whole HSV space into $n \times m \times l$ cubes. Which means the H direction is divided into n segments, S direction m segments, V direction l *segments* (segments can be slightly overlapped). Suppose a color vector in color space is denoted as $C = (H_C, S_C, V_C)$. Then the partition i can be represented as color range from C_i to C_i'. A binary image is generated for each color space partition as:

$$B_i = (I \geq C_i) \wedge (I < C_i') \qquad (3)$$

Where I is the given color image. These binary images will be verified using the character block grouping and layout analysis.

4. Character block grouping and Layout Analysis

The idea of using layout analysis is to verify the binary text regions in each generated binary image, and a text region is identified if one of these binary images contains texts. The layout analysis procedure includes three phases: connected component analysis, grouping of the character blocks, layout verification.

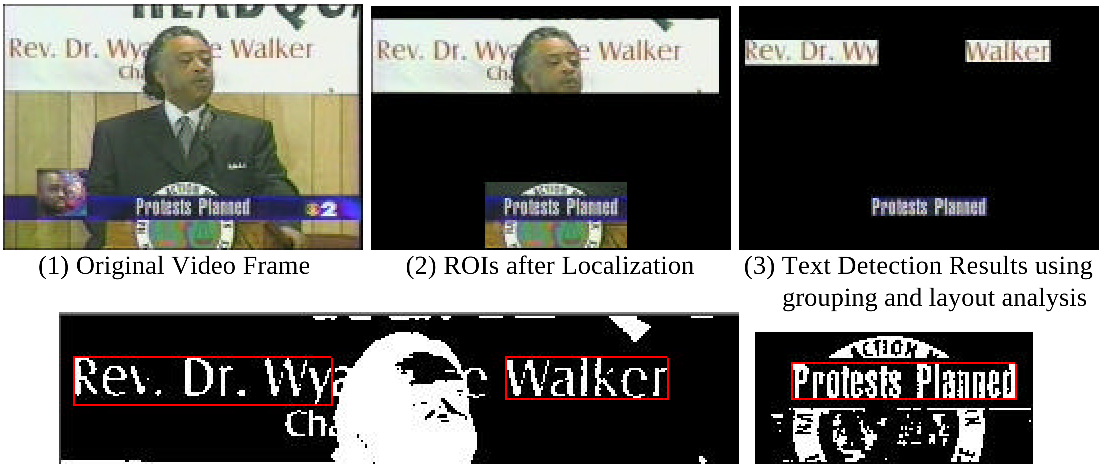
In each binary image, the connected component analysis (CCA) algorithm will be first performed to generate the character blocks. The character block is the outmost bounding box of a given binary component after using CCA. Size and shape filters will be performed to eliminate the blocks with too small or too large sizes, and regions with abnormal aspect ratio.

The grouping algorithm uses the following rules to cluster two blocks:

1. The blocks should be aligned such that they are with the same bottom line.
2. The heights of the blocks should be close to each other.
3. The blocks should be “close enough”, which means the nearest distance of two blocks should not exceed certain ratio of the average height.

The grouping procedure would group the individual blocks into hypothetical text lines. Then the verification procedure is performed in these text lines. We use a simple verification process: The character number in a text line should exceed certain constant.

The typical value of the constant is 3, because the lengths of most words in English are larger than 3. One example of grouping and layout analysis is shown in Figure 3.



(1) Original Video Frame

(2) ROIs after Localization

(3) Text Detection Results using grouping and layout analysis

(4) Character Grouping and layout analysis in ROIs

Figure 3. Character Block Grouping and Layout Analysis.

5. Temporal Consistency Verification

Multiple frame verification is to filter out the detection false alarms using temporal consistency verification of the bounding boxes corresponding to the text line regions. A text line on video usually stays on the video with a significantly long time. But the false alarm regions are usually transient in one or two I-frames.

Temporal consistency verification is performed for the detected bounding boxes. It calculates the overlap ratio of each bounding box in the current I-frame with all bounding boxes in the previous I-frame. And take the bounding box with the largest overlap ratio as the most likely match (MLM). The overlap ratio is calculated as following:

$$O(r_1, r_2) = \frac{2a(r_1 \cap r_2)}{a(r_1) + a(r_2)} \quad (4)$$

Where $a(r)$ is the area of region *r*. If the overlap ratio with the MLM is significant (larger than certain constant), then the two bounding boxes are regarded as temporarily consistent. N Bounding boxes are said to be n-consistent if each of them are temporarily consistent with the bounding box in the nearest I-frames in *n* consecutive I-frames. A videotext line is detected if the text line is n-consistent, where n is larger than a constant, otherwise it is classified as a false alarm.

6. Experiments

The algorithm is tested using NIST TREC-2002 benchmark, and caption detection in News video.

TREC Video Track is an open metric-based evaluation system aiming at promoting communication and progress in digital video retrieval field. The task of TREC-2002 is to index and retrieve TREC video based on concept detection (including face, indoor, outdoor, text overlay etc.) and query-retrieval framework.

The benchmark used 23.26 hours videos for the development of concept detectors and 5.02 hours videos as the Feature Test Set (FTS) (i.e. for feature extraction evaluation).

The testing results using our developed method on 5 hours Feature Validate (FV) set in TREC-2002 video data showed 0.4164 average precision, which compares with the average precision 0.3271 of Dorai, Shim and Bolle's system (DSB system) [6][11], and average precision 0.5018 of B.Tseng, C.Lin and J.Smith's fusion system results [10], which combines the described system and the DSB system. The average precision of our system on the TREC Test Set is 0.2941, while the DSB system achieves precision of 0.3324 and the fused system [10] achieves 0.4181 precision. [1]

Figure 5 shows some text detection results on TREC-2002 video data.

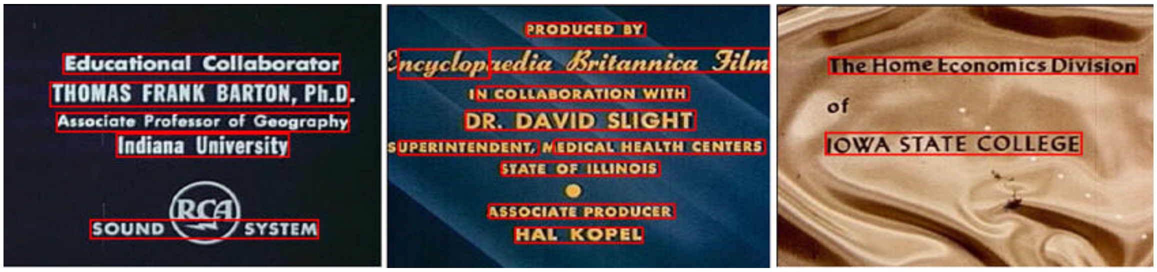


Figure 5. Overlay Text Detection Example of TREC 2002 Videos

Yet another testing excrement is conducted on News videos, News videos include a large amount of overlay texts. The detection and recognition of the text overlay is very useful for News video retrieval. The text overlay in News videos may present in various positions, thus the fixed position assumption used in our sports video application [] does not work very well for general overlay text detection. The experimental data include four News videos, three of which are US news videos from three different channels, one of which is Taiwan News video. The overall length of the videos is 2.13 hours. The caption styles on these videos are different from each other as well as fonts. These videos are of MPEG-1 format with 320x240 resolutions. The overall length of video is about 2 hours. The following table shows the detection results. Some of the detection results are shown in Figure 6.

Table 1. Text Overlay Detection in News Video

	US 1 (Ch 7)	US 2 (Ch 11)	US 3 (Ch 2)	TW 1 (TTV)	Average
Recall	97.2% (105/108)	95.7% (90/94)	95.7%(90/94)	98.1%(154/157)	96.9%
Precision	77.2% (105/136)	58.4% (90/154)	62.5%(90/144)	86.3%(154/179)	71.6%

Legends: Ch: Channel, US: United States, TW: Taiwan

The experiments showed that the performance on News video is better than the performance in TREC video. This is because most of overlay texts in News videos are

[1] Acknowledgement to Dr. Belle T. Tseng and Dr. Ching-Yung Lin to provide TREC-2002 text overlay testing results

with clean background and high contrast. Figure 6.shows some examples of text detection results in News video.

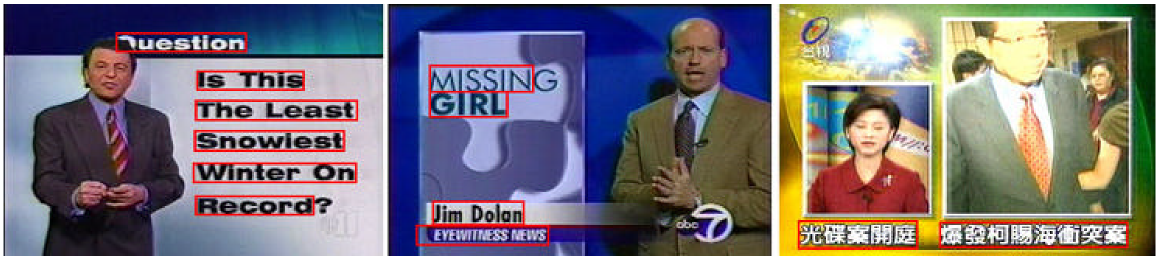


Figure 6. Examples of Overlay Text Detection Results on Digital News Video

7. Conclusions and Future Work

The report gives a rule-based approach to detect and extract the text lines on the video frames. The system includes the compressed domain feature processing to extract Region of Interests, color space partitioning, region grouping and layout analysis, At last temporal verification is used to enhance the stability and reduce the false alarms. Future work is to extend the rule-based approach to the probabilistic framework to general the accuracy of the whole system.

8. Acknowledgement

Acknowledgement to Dr. Belle L. Tseng and Dr. Ching-Yung Lin, IBM Watson Research Center for helpful discussion and testing results on TREC-2002 video data.

9. Reference

[1] T. Sato, T. Kanade, E. Hughes, and M. Smith, "Video OCR: Indexing Digital News Libraries by Recognition of Superimposed Captions", Multimedia Systems, 7:385-394, 1999.

[2] R.Lienhart, W. Effelsberg, "Automatic text segmentation and text recognition for video indexing", Multimedia System, 2000.

[3] H. Li, D. Doermann, O. Kia, “Automatic text detection and tracking in digital video”, IEEE Transaction. on Image processing, Vol 9, No. 1, January 2000.

[4] Y. Zhong, H, Zhang, and A. K.Jain, "Automatic Caption Localization in Compressed Video", IEEE Trans on PAMI, Vol 22. No.4 April 2000.

[5] M. Bertini, C. Colombo and A. D. Bimbo, "Automatic caption localization in videos using salient points". In Proceedings IEEE International Conference on Multimedia and Expo ICME 2001, Tokio, Japan, August 2001.

[6] J.C. Shim, C. Dorai and R. Bolle, “Automatic Text Extraction from Video for Content-Based Annotation and Retrieval," in Proc. 14th International Conference on Pattern Recognition, vol 1, pp. 618-620, Brisbane, Australia, August 1998.

[7] L. Agnihotri and N. Dimitrova. Text Detection in Video Segments. Proc. Of workshop on Content Based access to Image and Video Libraries, pp 109-113, June 1999.

[8] A.K. Jain and B.Yu, “automatic Text Location in Images and Video Frames”, Pattern Recognition, vol.31, no.12, pp. 2055-2076, 1998.

[9] D. Zhang, and S.F. Chang, “General and Domain-specific Techniques for Detecting and Recognizing Superimposed Text in Video”, Proceeding of International Conference on Image Processing, Rochester, New York, USA.

[10] B.L.Tseng, C.Y. Lin, D. Zhang and J.R. Smith, “Improved Text Overlay Detection in Videos Using a Fusion-Based Classifier”, IBM T.J. Watson Research Center, Yorktown Heights, New York, 2002.

[11] C. Dorai, “Enhancements to Videotext Detection Algorithms, Confidence Measures, and TREC 2002 performance”, IBM T.J. Watson Research Center, Yorktown Heights, New York, September 2002.