Video OCR for Digital News Archives*

Toshio Sato Takeo Kanade Ellen K. Hughes Michael A. Smith
School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213
{tsato, tk, ln, msmith}@cs.cmu.edu

Abstract

Video OCR is a technique that can greatly help to locate topics of interest in a large digital news video archive via the automatic extraction and reading of captions and annotations. News captions generally provide vital search information about the video being presented – the names of people and places or descriptions of objects. In this paper, two difficult problems of character recognition for videos are addressed: low resolution characters and extremely complex backgrounds. We apply an interpolation filter, multi-frame integration and a combination of four filters to solve these problems. Segmenting characters is done by a recognition-based segmentation method and intermediate character recognition results are used to improve the segmentation. The overall recognition results are good enough for use in news indexing. Performing Video OCR on news video and combining its results with other video understanding techniques will improve the overall understanding of the news video content.

1 Introduction

Understanding the content of news videos requires the intelligent combination of many technologies: speech recognition, natural language processing, search strategies, image understanding, etc.

Extracting and reading news captions provides additional information for video understanding. For instance, superimposed captions in news videos annotate the names of people and places, or describe objects. Sometimes this information is not present in the audio or cannot be obtained through other video understanding methods. Performing Video OCR on news video and combining its results with other video understanding techniques will improve the overall understanding of the news video content.

Although there is a great need for integrated character recognition in text-based video libraries [1], we have seen few achievements. Automatic character segmentation was performed for titles and credits in motion picture videos in [2] and [3]; however, both papers have insufficient consideration of character recognition.

There are similar research fields which concern character recognition of videos [4] [5]. In [4], character extraction from the car license plate using video images is presented. In [5], characters in scene images are segmented and recognized based on adaptive thresholding. While these results are related, character recognition for the news video presents its own difficulties because of different conditions of the character size and complex backgrounds. In news video captions, resolution of character is lower; also, the background complexity is more severe than in other research.

An example of these issues is illustrated in Figure 1 which shows one frame from CNN Headline News. The video frame has a caption with low resolution characters with heights of ten pixels or less. The background in the example is complicated and in some areas has little difference in hue and brightness from that of the characters. The importance of the correct extraction and recognition of the caption can be understood through analysis of the audio and closed captions. The information contained in this caption, and many others like it, is not in the audio and can only be obtained through accurate Video OCR.

The first problem is low resolution of the characters. The size of an image is limited by the number of scan lines defined in the NTSC standard, and characters of the video caption are small to avoid occlusion of interesting objects such as people's faces. Therefore, the resolution of characters in the video caption is insufficient to implement stable and robust Video OCR systems. This problem can be even more serious if inadequate compression methods are employed. The problem of resolution is discussed in OCR of World Wide Web images [6]. They propose two recognition

*IEEE International Workshop on Content-Based Access of Image and Video Databases (CAIVD'98)

techniques for low resolution images. However, it is convenient if this problem is solved in preprocessing to use achievements of researches on recognition.

Another problem is the existence of complex backgrounds. Characters superimposed on news videos often have hue and brightness similar to the background, making extraction extremely difficult. For the cases of the license plate research [4], the scene image research [5] and the document image analysis [7], the difficulty of complex backgrounds is rarely a problem although examples with plain backgrounds are illustrated. In [3], there is an assumption that characters are drawn in high contrast against the background.

Object detection techniques such as matched spatial filtering are alternative approaches for extracting characters from the background. In [8], several template matching methods are compared to determine which is the best for detecting an eye in a face image. Since every character has a unique shape, simple template matching methods are not adequate for extracting whole character patterns. It is assumed that, for characters composed of line elements, a line detection method [9] is more applicable.

We also have a problem with segmenting characters in low quality images. Although there are many approaches for character segmentation [10], errors still remain because most methods analyze only a vertical projection profile. Character segmentation based on recognition results [11] offers the possibility to improve the accuracy. However, the huge computational cost to select proper segments from combinations of segment candidates is prohibitive.

In Section 2, we describe image enhancement methods which cope with both problems. In Section 3, character extraction methods for the problem of complex backgrounds are explained. In Section 4, character segmentation and recognition methods are presented. Section 5 contains results obtained by using Video OCR on seven news videos. The results of post-processing are described in Section 6.

2 Selection and Enhancement of Text Region

2.1 Detection

Since a video news program comprises huge numbers of frames, it is computationally prohibitive to detect each character in every frame. Therefore, we first roughly detect text regions in groups of frames to increase processing speed.

Some known constraints of text regions can reduce the processing costs. A typical text region can be characterized as a horizontal rectangular structure of

Figure 1: (a)Examples of caption frame. (b)Magnified image of character. (c)Binary image.

clustered sharp edges, because characters usually form regions of high contrast against the background.

Smith and Kanade describe text region detection using the properties in [2]. The speed is fast enough to process a 352×242 image in less than 0.8 seconds using a workstation (MIPS R4400 200MHz).

By utilizing this method, we select frames and extract regions that contain textual information from the selected frame. For extraction of vertical edge features, we apply a 3×3 horizontal differential filter to the entire image with appropriate binary thresholding. If a bounding region which is detected by the horizontal differential filtering technique satisfies size, fill factor and horizontal-vertical aspect ratio constraints, it is selected for recognition as a text region.

Detection results are selected by their location to extract specific captions which appear at lower positions in frames.

2.2 Improvement of Image Quality

In television news videos, the predominant difficulties in performing Video OCR on captions are due to low resolution characters and widely varying complex backgrounds. To address both of these problems, we have developed a technique which sequentially filters the caption during frames where it is present. This technique initially increases the resolution of each caption through a magnifying sub-pixel interpolation method. The second part of this technique reduces the variability in the background by minimizing (or maximizing) pixel values across all frames containing the caption. The resulting text areas have good resolution and greatly reduced background variability.

2.2.1 Sub-pixel Interpolation

The size of a character image in the video caption of CNN Headline News is less than 10×10 pixels. This resolution is insufficient for robust character recognition. To obtain higher resolution images, we expand the low resolution text regions by applying a sub-pixel interpolation technique. To magnify the text area in each frame by four times in both directions, each pixel of the original image $I(x,y)$ is placed at every fourth pixel in both x and y direction to obtain the four times resolution image $L(x,y) : L(4x,4y) = I(x,y)$. Other pixels are interpolated by a linear function using neighbor pixel values of the original image weighted by distances as follows:

$$L(x,y) = \frac{\sum_{(x_0,y_0)\in\mathcal{N}(x,y)} d(x-x_0, y-y_0)\cdot I(\frac{x_0}{4},\frac{y_0}{4})}{\sum_{(x_0,y_0)\in\mathcal{N}(x,y)} d(x-x_0,y-y_0)} \quad (1)$$

where $\mathcal{N}(x,y) = \{(x_0,y_0) \mid x_0 \in \{\lfloor\frac{x}{4}\rfloor\cdot 4, \lceil\frac{x}{4}\rceil\cdot 4\}, y_0 \in \{\lfloor\frac{y}{4}\rfloor\cdot 4, \lceil\frac{y}{4}\rceil\cdot 4\}\}$ and $d(x,y) = ||(x,y)||^{-1}$ (Also see Figure 2).

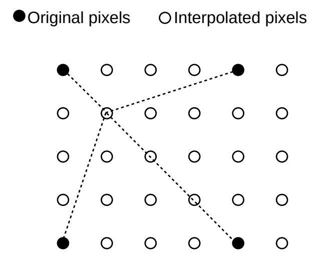

Figure 2: Sub-pixel interpolation (The case of interpolation factor 4 is illustrated).

2.2.2 Multi-frame Integration

For the problem of complex backgrounds, an image enhancement method by multi-frame integration is employed using the enhanced resolution interpolation frames. Although complex backgrounds usually have movement, the position of video captions is relatively stable across frames. Furthermore, we assume that captions have high intensity values such as white pixels. Therefore, we employ a technique to minimize the variation of the background by using a time-based minimum pixel value search. (For black characters, the same technique could be employed using a time-based maximum search.) With this technique, an enhanced image is made from the minimum pixel value in each location that occurs during the frames containing the caption. By taking advantage of the video motion of non-caption areas, this technique results in text areas with less complex backgrounds while maintaining the existing character resolution. (See Figure 3).

The sub-pixel interpolated frames, $L_i(x,y)$, $L_{i+1}(x,y), \ldots, L_{i+n}(x,y)$ are enhanced as $L_m(x,y)$ via

$$L_m(x,y) = min(L_i(x,y), L_{i+1}(x,y), \ldots, L_{i+n}(x,y)) \quad (2)$$

where (x,y) indicates the position of a pixel and i and $i+n$ are the beginning frame number and the end frame number, respectively. These frame numbers are determined by text region detection [2].

Figure 3: Improving image quality by multi-frame integration.

An example of effects of both the sub-pixel interpolation and the multi-frame integration for the original image in Figure 1 is shown in Figure 8.

3 Extraction of Characters

3.1 Character Extraction Filter

To further reduce the effect of complex backgrounds, a specialized filter based on correlation is used. We recognize that a character consists of four different directional line elements: vertical, horizontal, left diagonal, and right diagonal. We employ a filter which integrates the output of four filters corresponding to those line elements.

To make learning data for filters, we select pixels from actual television captions which correspond to vertical, horizontal, left diagonal and right diagonal line elements, shown as white pixels in Figure 4.

The size of the filter is defined to include only a line element of characters. In this paper, we consider 15×3, 3×7, 9×7 and 9×7 filters to detect vertical, horizontal, left diagonal and right diagonal line elements, respectively. Values of the filters are determined by averaging pixels for each position of the neighboring area according to the learning data. Each filter image is normalized to have an average value of zero. All filter values are multiplied by a common fixed coefficient to make the output pixel values remain in 8-bit data. Figure 5 shows data of the filter values.

Figure 4: Learning data of filters (white pixels). vertical / horizontal / left diagonal / right diagonal.

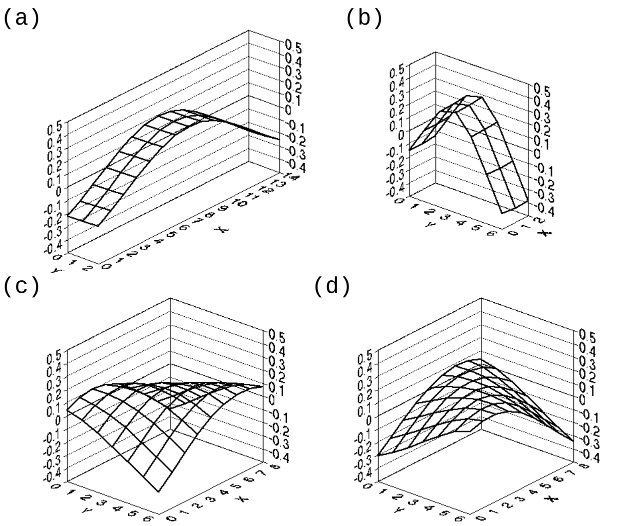

Figure 5: Character extraction filters $F_i(x, y)$. (a)Vertical. (b)Horizontal. (c)Left diagonal. (d)Right diagonal.

At the time of processing, a preprocessed image $L_m(x, y)$ is filtered by calculating correlation with the filters $F_i(x, y)$, where $i = \{0, 1, 2, 3\}$, to extract each line element.

$$L_{f,i}(x, y) = \sum_{y_0=-h_i}^{h_i} \sum_{x_0=-w_i}^{w_i} L_m(x + x_0, y + y_0) \cdot F_i(x_0 + w_i, y_0 + h_i) \tag{3}$$

where w_i and h_i represent the area of the filters.

Positive values at the same position among filtered images are added to integrate four directional line elements.

$$L_f(x, y) = \sum_{i=0}^{3} L'_{f,i}(x, y) \tag{4}$$

$$L'_{f,i} = \begin{cases} 0 & \text{if } L_{f,i}(x, y) \leq 0 \\ L_{f,i}(x, y) & \text{otherwise} \end{cases} \tag{5}$$

Pixels for characters have high values, so thresholding at a fixed value θ is applied to reduce noise for the final output of the character extraction filter L_{filter}.

$$L_{filter} = \begin{cases} 0 & \text{if } I(x, y) \leq \theta \\ L_f(x, y) & \text{otherwise} \end{cases} \tag{6}$$

Figure 9 shows an example of results for the character extraction filter.

3.2 Thresholding/Projection

Thresholding at a fixed value for the output of the character extraction filter L_{filter} produces a binary image which is used to determine positions of characters and recognize characters.

Horizontal and vertical projection profiles of the binary image are used to determine candidates for character segmentation.

3.3 Simple Segmentation

As we search for white characters, peaks in a vertical projection profile indicate boundaries between characters. As shown in Figure 6, a position at which the projection value changes from low to high compared with a fixed threshold value is detected as a left edge of the peak. In the same manner, a right edge of the peak is detected as high to low transition. The left and right edge positions of the peak correspond to left and right boundaries of a character segment candidate, respectively.

Spurious valleys in the projection profile sometimes cause a character to be segmented into two or more pieces as shown in Figure 10. Since the edges of the peaks include correct segment boundaries of the character, we still devise a method for selecting the best combination of edges of the peaks to find character segments. The method integrates the edge positions with the character recognition results, as explained in the next section.

Top and bottom boundaries of character segments are simply determined by detecting beginning and end positions in a horizontal projection profile.

4 Integration of Recognition and Segmentation

4.1 Character Recognition

We use a conventional pattern matching technique to recognize characters. An extracted character segment image is normalized in size and converted into a blurred gray scale image by counting the number of neighbor pixels. This preprocessing makes the recognition robust to changes in thickness and position. The normalized gray scale data $n(x,y)$ are matched with reference patterns $ref_c(x,y)$ based on a correlation metric. The matching metric m_c is described as follows:

$$m_c = \frac{\sum n(x,y) \cdot ref_c(x,y)}{\sqrt{\sum (n(x,y))^2}\sqrt{\sum (ref_c(x,y))^2}} \qquad (7)$$

A category c which has the largest m_c in reference patterns is selected as the first candidate. In the same manner, second and third candidates are selected. The reference patterns $ref_c(x,y)$ are built by averaging sample data.

4.2 Selecting Segmentation Result by Character Recognition

Segment candidates which are detected by the simple segmentation method may include over-segmented characters as explained in Section 3.3.

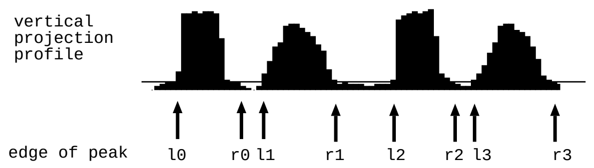

Figure 6: Edge of peak in vertical projection profile.

To select a correct pair of edges which represents a character segment, correspondences between character segments and their similarities m_c are evaluated.

The peaks of the vertical projection $\boldsymbol{P}$ consist of pairs of left edge l_i and right edge r_i.

$$\boldsymbol{P} = \{(l_i, r_i) \mid i = 0, 1, 2, \ldots, M\} \qquad (8)$$

where $M+1$ is the number of peaks.

A candidate of character segmentation $\boldsymbol{C_j}$ is defined as a series of segments (l_b, r_e), where l_b and r_e are left and right edges that appear in $\boldsymbol{P}$:

$$\boldsymbol{C_j} = \{(l_0, r_\alpha), (l_{\alpha+1}, r_\beta), \ldots, (l_{\gamma+1}, r_M)\} \qquad (9)$$

where $0 \le \alpha < \beta < \ldots < \gamma + 1 \le M$.

A segmentation is evaluated on how well it can result in a series of readable characters. That is, for a segmentation $\boldsymbol{C_j}$, an evaluation value $E_v(\boldsymbol{C_j})$ is defined as the sum of the maximum similarities for each segment normalized by the number of segments $N_{um}(\boldsymbol{C_j})$.

$$E_v(\boldsymbol{C_j}) = \sum_{(l_b, r_e) \in \boldsymbol{C_j}} m_m(l_b, r_e) / N_{um}(\boldsymbol{C_j}) \qquad (10)$$

where $m_m(l_b, r_e)$ is the maximum similarity of a segment between l_b and r_e among the reference patterns $ref_i(x,y)$ in Eq. (7).

The best character segmentation $\boldsymbol{C}$ is the one with the largest evaluation value.

$$\boldsymbol{C} = \arg\max_{\boldsymbol{C_j} \in \boldsymbol{P}} (E_v(\boldsymbol{C_j})) \qquad (11)$$

The search for the best segmentation is performed efficiently by a dynamic programming method with the constraint of the character width to reduce the calculation cost.

We now explain the detailed procedure of the search method for determining the best segmentation $\boldsymbol{C}$:

$$\boldsymbol{C} = \{(l_0, r_{s(1)}), (l_{s(1)+1}, r_{s(2)}), \ldots, (l_{s(n)+1}, r_M)\} \qquad (12)$$

where $n \le M$.

We process two consecutive segments at a time; we consider the first segment of the two to be correct and fixed if both the first and the second segments have high similarities. This process is repeated to fix all segments. As the first segment starts from the first left edge l_0, the end point of the first segment is determined as $r_{s(1)}$ via

$$s(1) = \arg\max_{u=\{0,\ldots,M-1\}} \{m_m(l_0, r_u) + m_m(l_{u+1}, r_v)\} \qquad (13)$$

where $r_u - l_0 < w_c, r_v - l_{u+1} < w_c$ and w_c is the maximum width of a character. Then, the start point of the next segment is l_{u+1}.

We continue to determine the end points of segments $r_{s(k+1)}$ step by step.

$$s(k+1) = \arg\max_{u=\{s(k)+1,\ldots,M-1\}} \{m_m(l_{s(k)+1}, r_u) + m_m(l_{u+1}, r_v)\} \qquad (14)$$

where $r_u - l_{s(k)+1} < w_c, r_v - r_{u+1} < w_c$.

If v reaches M, the process terminates and both segments are determined to be results. Further, if $(l_{u+1} - r_u)$ in Eq. (14) exceeds a fixed value, the r_u is considered to be the end of a word and the process starts from Eq. (13) for the next word.

This method enables us to determine the best segmentation $\boldsymbol{C}$ faster than by calculating the evaluation value for all combinations of segments in Eq. (11).

5 Evaluation with News Video

We evaluate our method using CNN Headline News broadcast programs, received and digitized. Video data is encoded by MPEG-1 format at 352×242 resolution. We use seven 30-minute programs; the total number of characters for the captions is 5,076, which includes 2,370 Roman font characters (395 words) and 2,706 Gothic font characters(419 words).

Our method includes the sub-pixel interpolation, the multi-frame integration, the character extraction filter and the segmentation results selection by character recognition. Using a workstation (MIPS R4400 200MHz), it takes about 120 seconds to process a caption frame block and about 2 hours to process a 30-minute CNN Headline News program.

Table 1 shows character recognition rates for Video OCR using seven 30-minute CNN Headline News videos. The percentage of correct Roman characters (76.2%) is lower than that of Gothic characters (89.8%) because Roman font characters have thin line elements and tend to become scratchy. The total recognition rate is 83.5%.

To compare our results with those from a conventional OCR approach, we implement a simple OCR program. It consists of binarization of an image by straightforward thresholding at a fixed value (no sub-pixel interpolation, no multi-frame integration or no character extraction filter), character extraction by horizontal and vertical projections of the binary image (no integrating with character recognition), and matching by correlation (the same algorithm as ours).

Table 2 shows character recognition results of the conventional OCR. The recognition rate is 46.5%, which is almost half of our results and less than half of an average commercial OCR rate for documents [12]. The recognition rate of Roman font characters is lower because of their thin lines which correspond to one or less pixel in the binary image.

Figure 7 illustrates why the conventional OCR method has difficulties with video data. Each character has poor resolution which is less than 10×10 pixel size. Both words in Figure 7 have low contrasts against the background so that extraction of characters does not correspond to proper character segments.

Let us see how our approach improves the results. An effect of the sub-pixel interpolation and the multi-frame integration is demonstrated in Figure 8. The original image in Figure 7, which has less than 10×10 pixel size for each character, is enhanced to have about 30×40 pixel size by this method. Characters have smooth edges in contrast with notches of the original image in Figure 7.

Results of the character extraction filter are shown in Figure 9. Each filter emphasizes pixels which correspond to a line direction. The integration of four directional filtered images shown in Figure 9(e) indicates that the filter extracts characters from complex backgrounds; the backgrounds are not eliminated by a simple thresholding method in Figure 7.

Thresholding at a fixed value for the integrated image shown in Figure 9(f) is applied to determine a bounding rectangle for a character by vertical and horizontal projection of extracted pixels. White lines in Figure 10 illustrate detected edges of peaks in a vertical projection profile. Eight characters out of 18 are over-segmented though the edges include proper character segments. The edges of the peaks are selected by character recognition to segment characters shown in Figure 11.

While we acquired training data by using Roman font characters to design character extraction filters, the filters also work well with Gothic font character images. Figure 12 shows results of character extraction for Gothic font. Although the filter output appears weaker at intersections of strokes, it does not present much of a problem in segmenting and recognizing characters.

Table 1: Character recognition of Video OCR

	Roman	Gothic	Roman+ Gothic
Correct characters	1807	2430	4237
Total characters	2370	2706	5076
Recognition rate	76.2%	89.8%	83.5%

Table 2: Character recognition of conventional OCR

	Roman	Gothic	Roman+ Gothic
Correct characters	921	1439	2360
Total characters	2370	2706	5076
Recognition rate	38.9%	53.2%	46.5%

Figure 7: (a) Original image (204×14 pixels). (b) Binary image. (c) Character extraction using a conventional technique.

Figure 8: Result of sub-pixel interpolation and multi-frame integration (813×56 pixels).

Figure 9: Result of character extraction filter. (a) Vertical. (b) Horizontal. (c) Left diagonal. (d) Right diagonal. (e) Integration of four filters (f)Binary image.

Figure 10: Edges of peak in vertical projection profile.

Figure 11: Result of segmentation.

Figure 12: Result of extraction for Gothic font characters. Sub-pixel interpolated and multi-frame integrated image (top). Character extraction filter (middle). Character segments in binary image (bottom).

6 Postprocessing for Word Recognition

To acquire text information for content-based access of video databases, high word recognition rates for Video OCR are required. The Video OCR recognizes only 48.3% (393 out of 814 words), even though the recognition rate of characters is 83.5%.

We apply postprocessing, which evaluates differences between recognition results with words in the dictionary, and select a word having the least differences. The differences are measured among three candidates of the recognition result weighted by the similarity m_c. The dictionary consists of two different sources: the *Oxford Advanced Learner's Dictionary* [13] for general words (69,517 words) and noun word collections which are compiled by analyzing closed caption information of videos to obtain people, organization or place names (3,035 words) [14].

As shown in Table 3, this postprocessing improves the word recognition rate to 65.2%. Although 34 words which are recognized correctly are not in our dictionary, 172 words are recovered through postprocessing.

Table 3: Word Recognition rate with postprocessing (total 814 words)

	Correct words	Rate	Corrected by post.	Missed by post.
w/o Post.	393	48.3%	—	—
with Post.	531	65.2%	172	34

7 Conclusions

Accurate Video OCR provides unique information about the content of video news data, but poses challenging technical problems. Low resolution data and complex backgrounds are two problems which contribute to news caption degradation. By combining sub-pixel interpolation on individual frames, multi-frame integration across time, character extraction filtering and recognition-based character segmentation, we obtained satisfactory Video OCR performance for video news indexing. On actual news data, we attained 83.5% correct character recognition and 65.2% correct word recognition rates – which is almost double that obtained using conventional OCR techniques on the same data. The sample data which we used contains one of the most difficult designs to recognize. With plain background patterns and thick characters, which are fairly common, the recognition rate will be improved.

Information gained through Video OCR is often unique and unobtainable from other video understanding techniques, making accurate Video OCR a vital technology for searching news video archives. Furthermore, the Video OCR results represent a more important possibility – combination of image data and closed captioning text information. For instance, OCR results can improve a system that associates names and faces in news videos [15] to enhance query functions.

Acknowledgments

This work has been partially supported by the National Science Foundation under grant No. IRI-9411299. The authors would like to thank Shin'ichi Satoh and Yuichi Nakamura for valuable discussions and for providing postprocessing data.

References

[1] H. D. Wactlar, T. Kanade, M. A. Smith, and S. M. Stevens, "Intelligent access to digital video: The Informedia project," *IEEE Computer*, vol. 29, no. 5, pp. 46-52, May 1996.

[2] M. A. Smith and T. Kanade, "Video skimming and characterization through the combination of image and language understanding technique," In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 775-781, 1997.

[3] R. Lienhart and F. Stuber, "Automatic text recognition in digital videos," In *Proceedings of SPIE Image and Video Processing IV*, vol. 2666, pp. 180-188, September 1996.

[4] Y. Cui and Q. Huang, "Character extraction of license plates from video," In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 502-507, 1997.

[5] J. Ohya, A. Shio, and S. Akamatsu, "Recognizing characters in scene images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 214-220, 1994.

[6] J. Zhou, D. Lopresti, and Z, Lei, "OCR for World Wide Web images," In *Proceedings of SPIE Document Recognition IV*, vol. 3027, pp. 58-66, 1997.

[7] V. Wu, R. Manmatha, and E. M. Riseman, "Finding text in images," In *20th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3-12, Philadelphia, 1997.

[8] R. Brunelli and T. Poggio, "Template matching: Matched spatial filters and beyond," *Pattern Recognition*, vol. 30, no. 5, pp. 751-768, 1997.

[9] H. A. Rowley and T. Kanade, "Reconstructing 3-D blood vessel shapes from multiple X-ray images," In *AAAI Workshop on Computer Vision for Medical Image Processing*, San Francisco, CA, March 1994.

[10] Y. Lu, "Machine printed character segmentation - an overview," *Pattern Recognition*, vol. 28, no. 1, pp. 67-80, 1995.

[11] S.-W. Lee, D.-J. Lee, and H.-S. Park, "A new methodology for gray-scale character segmentation and recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 1045-1050, October 1996.

[12] Information Science Research Institute, "1994 annual research report", http://www.isri.unlv.edu /info/publications/anreps.html.

[13] The Oxford text archive. http://ota.ox.ac.uk/.

[14] Y. Nakamura and T. Kanade, "Semantic analysis for video contents extraction - spotting by association in news video," In *ACM Multimedia 97*, Seattle, 1997.

[15] S. Satoh and T. Kanade, "NAME-IT: Association of face and name in video," In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 368-373, 1997.