Text Localization, Enhancement and Binarization in Multimedia Documents

Christian Wolf[1] Jean-Michel Jolion[1] Françoise Chassaing[2]

[1] Lab. Reconnaissance de Formes et Vision, INSA de Lyon
Bât Jules Verne 20, Avenue Albert Einstein
Villeurbanne, 69621 cedex, France
{wolf,jolion}@rfv.insa-lyon.fr

[2] France Télécom R&D
4, rue du Clos Courtel - BP 59
35512 Cesson-Sévigné cedex, France
francoise.chassaing@francetelecom.fr

Abstract

The systems currently available for content based image and video retrieval work without semantic knowledge, i.e. they use image processing methods to extract low level features of the data. The similarity obtained by these approaches does not always correspond to the similarity a human user would expect. A way to include more semantic knowledge into the indexing process is to use the text included in the images and video sequences. It is rich in information but easy to use, e.g. by key word based queries. In this paper we present an algorithm to localize artificial text in images and videos using a measure of accumulated gradients and morphological post processing to detect the text. The quality of the localized text is improved by robust multiple frame integration. A new technique for the binarization of the text boxes is proposed. Finally, detection and OCR results for a commercial OCR are presented.

1. Introduction

Extraction of text from images and videos is a very young research area, which nevertheless attracts a large number of researchers. The first algorithms, introduced by the document processing community for the extraction of text from colored journal images, segmented characters before grouping them to words and lines [1]. These methods worked fine for high resolution images as journals but failed in the case of low resolution video, where characters are touching and the font size is very small. New methods based on edge detection or texture analysis soon followed [2, 3, 7, 9] as well as algorithms working in the compressed domain [11].

Text appears in videos in a wide range of writings, fonts, styles, colors, sizes, orientations etc., which makes an exact modeling almost impossible. Since the motivation behind our method is semantic indexing, we concentrated on horizontally aligned, artificial text, as e.g. news captions.

The paper is outlined as follows: In section 2 we describe our system[1] and the intermediate steps detection, tracking, image enhancement and binarization. Section 3 describes the experiments we performed to evaluate the system and compares the results with other methods. Section 4 gives a conclusion and an outlook on our future research.

2. The proposed system

A scheme of our proposed system is presented in figure 1. The detection algorithm is applied to each frame of the sequence. The detected text rectangles are passed to a tracking step, which finds corresponding rectangles of the same text appearance. From the frames of an appearance a single enhanced image is generated and binarized before passing it to a standard commercial OCR software. The respective steps are given in the next sub sections.

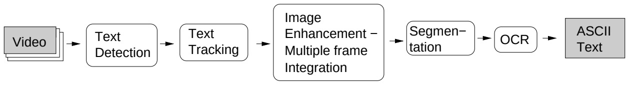


Figure 1. The scheme of our system.

2.1. Detection

Our detection method is based on the fact that text characters form a regular texture containing vertical strokes which are aligned horizontally. We slightly modified the algorithm of LeBourgeois [2], which detects the text with a measure of accumulated gradients:

$$A(x,y) = \sum_{i=-t}^{t} \frac{\partial I}{\partial x}(x+t,y)$$

[1] This work is supported by France Télécom R&D in the context of project ECAV. The first part resulted in patent ref. # FR 01 06776, submitted in May 25th 2000 by France Télécom.

The parameters of this filter are the implementation of the partial derivative (We chose the horizontal version of the Sobel operator, which obtained the best results in our experiments) and the size of the accumulation window, which depends on the size of the characters and the minimum length of words to detect. Since the results or not very sensitive to this characters we fixed it to $2t + 1 = 13$ pixels. The filter response is an image containing for each pixel a measure of the probability to be part of text.

Binarization of the accumulated gradients is done with a two-threshold version of Otsu's global thresholding algorithm [5]. We changed the binarization decision for each pixel as follows:

$$\begin{array}{lll} I_{x,y} < k_l & \Rightarrow & B_{x,y} = 0 \\ I_{x,y} > k_h & \Rightarrow & B_{x,y} = 255 \\ k_l > I_{x,y} > k_h & \Rightarrow & B_{x,y} = \begin{cases} 255 & \text{if there is a path to} \\ & \text{a pixel } I_{u,v} > k_h \\ 0 & \text{else} \end{cases} \end{array}$$

where k_h is the optimal threshold calculated by Otsu's method and k_l is calculated from the k_h and the first mode m_0 of the histogram: $k_l = m_0 + \alpha(k_h - m_0)$. The parameter α was fixed in experiments to 0.87.

A phase of mathematical morphology follows the binarization step for several reasons:

- To reduce the noise.
- To correct classification errors using information from the neighborhood of each pixel.
- To connect characters in order to form complete words.

The morphological operations consist of the following steps:

- Close (1 iteration).
- Suppression of unwanted bridges between components (Suppression of all pixels which are part of a column of height 2 pixels or less).
- Conditional dilation (16 iterations) followed by a conditional erosion (16 iterations).
- Horizontal opening (6 iterations).

The conditional dilation and erosion algorithms have been developed in order to connect characters to words. This is necessary to make the detection results less sensitive to the size of the accumulation window and in cases where the distances between characters are large. Dilation is based on a standard dilation operation with the structural element $B_H = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$ and the following conditions, which are verified before each pixel P is dilated:

- The relative difference of the heights of the connected component including pixel P and the neighboring component to the right does not exceed a threshold t_1.
- The relative difference of the positions of these two connected components does not exceed a threshold t_2.
- The relative difference of the heights of the connected component including pixel P and the height of the bounding box including both components does not exceed a threshold t_3.

The dilated pixels are marked with a special label different from the labels "text" and "non-text". The conditional erosion step uses the same structuring element B_H, but erodes only pixels marked with the special label. Finally, all pixels marked with the special label are marked as text. The effect of this morphological step is the connection of the all connected components which are horizontally aligned and whose heights are similar.

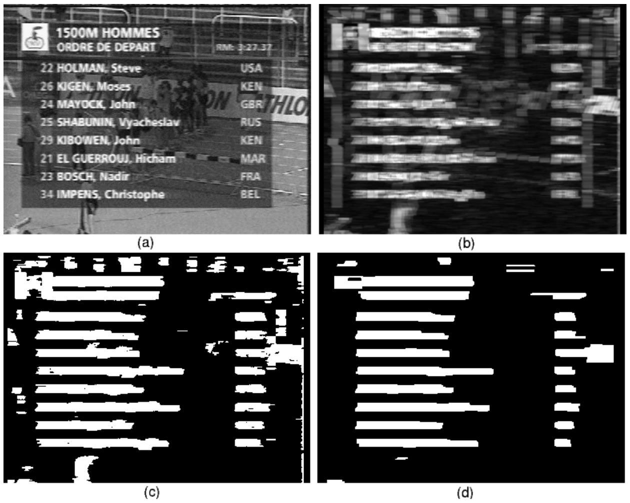


Figure 2. The intermediate results during the detection process: The input image (a), the gradient (b), the binarized image (c), The image after morphological post processing (d).

Figure 2 shows the intermediate results during the detection phase. Figures 2a and 2b display the original image and the accumulated gradients. The text areas are visible as emphasized white rectangles. Figure 2c shows the binarized image which still contains noise due to the background texture in the original image. Almost all of this noise is removed after post processing, shown in figure 2d.

After the morphological post processing, geometrical constraints are imposed on the rectangles (width/height, number of pixels/area of the bounding box etc.) in order to further decrease the number of false alarms. Finally some special cases are considered to increase the size of the bounding boxes of the rectangles to include the heads and tails of the characters.

2.2. Tracking

The goal of the tracking step is the association of detected text rectangles in successive frames to create appearances of text. To achieve this, we use the overlap information between the list of rectangles detected in the current frame and the list of currently active rectangles (i.e. the text detected in the previous frames). We use the length of the appearance as a measure of stability, which allows us to further reduce the number of false alarms.

2.3. Multiple frame integration

Before passing the images to the OCR software we enhance their contents. We also increase their resolution, which does not add any additional information, but is necessary because the commercial OCR programs have been developed for scanned document pages and are tuned to high resolutions. Both steps, interpolation and multiple frame enhancement, are integrated into a single, robust algorithm.

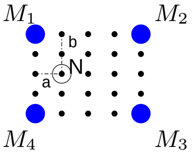


Figure 3. Bi-linear interpolation.

The area of the enhanced image $\hat{F}$ consists of the bounding box of all text images F_i taken from the frames i of the sequence. For each pixel p the temporal mean $M(p)$ and the standard deviation $S(p)$ are calculated. Then, the resolution of each frame image F_i is increased by bi-linear interpolation[2] (see figure 3). In this algorithm, the gray value of each pixel is a linear combination of the gray values of its neighbors, where the weight is calculated from the distances between the pixel and the respective neighbor M_k as follows: $w_k = (1-a)\cdot(1-b)$. The interpolation is applied to all frame images F_i, the final enhanced image $\hat{F}$ being the mean of the interpolated images.

To increase the robustness of the integration process, we added an additional weight g_k^i to the interpolation scheme which decreases the weights of outlier pixels:

$$g_k^i = \left(1 + \frac{|F_i(M_k) - M(M_k)|}{1 + S(M_k)}\right)^{-1}$$

The final weight for neighbor M_k of Frame i is therefore $w_k^i = (1-a)\cdot(1-b)\cdot g_k^i$.

2.4. Binarization

As a final step, the enhanced image needs to be binarized. Several binarization algorithms have been developed by the computer vision and the document analysis community. In our experiments we found out, that for our purposes, the simpler algorithms are more robust to the noise present in video images. For our system we chose Niblack's method [4], which also performed one of the best in a well known survey [6]. Niblack's algorithm calculates a threshold surface by gliding a rectangular window across the image. The threshold T for the center pixel of the window is computed using the mean m and the variance s of the gray values in the window: $T = m + k \cdot s$, where k is a constant set to -0.2. The results are not very sensitive to the window size as long as the window covers at least 1-2 characters. However, the drawback is noise created in areas which do not contain any text (due to the fact that a threshold is created in these cases as well). Sauvola et al. [8] solve this problem by adding a hypothesis on the gray values of text and background pixels, which results in the following formula for the threshold:

$$T = m \cdot (1 - k \cdot (1 - \frac{s}{R}))$$

where R is the dynamics of the standard deviation fixed to 128. This method gives better results for document images, but creates additional problems for video frames whose contents do not always correspond to the hypothesis. To overcome this, we changed the formula in order to normalize the contrast and the mean gray level of the image:

$$T = m - k\,\alpha\,(m - M) \quad , \quad \alpha = 1 - \frac{s}{R} \quad , \quad R = max(s)$$

where M is the minimum graylevel of the image and R is set to the maximum of the standard deviations of all windows. More details on the derivation of this method can be found in [10].

3. Experimental results

To estimate the performance of our system we carried out exhaustive evaluations using a video database containing 60.000 frames of different MPEG videos (384×288 pixels). The videos contain 371 appearances of artificial text with 3519 characters (See figure 4). Recognition was done by the commercial product Abbyy Finereader 5.0.

Figure 5a shows results of the different binarization techniques described in the previous section. Niblack's method segments the characters very well, but suffers from noise in the zones without text. Sauvola's algorithm overcomes this problem with the drawback that the additional hypothesis cause thinner characters and holes. Our solution solves this problem keeping the good results concerning the zones without text.

Figure 5b shows the detection and OCR results. We achieve a detection rate of 93.5% of the text appearances and 85.4% of the characters have been recognized correctly

[2] bi-cubic interpolation results in images of higher quality for a human viewer but does not improve the OCR performance.

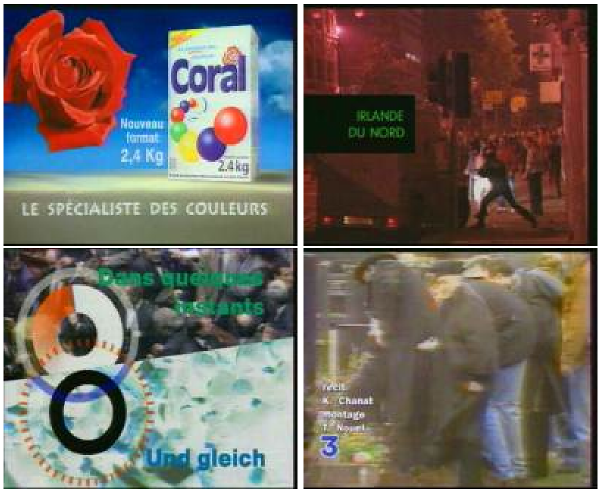


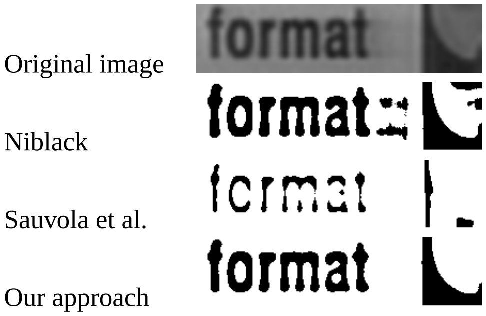
Figure 4. Examples of the video database used in our experiments.

by the OCR. The OCR also confirms the betters results of our binarization technique.

The comparison with other established methods is very difficult, if not impossible, due to the lack of a common video test database. We therefore only show the figures given in the articles, but all direct comparison has to be taken “with a grain of salt”. Doermann et al. [3] achieve a recognition rate of 87%-89% for movie credits. Lienhart [9] uses high resolution video (up to 1920×1280 pixels) and segments 80% of the characters. Of these, 90% are recognized correctly. Kanade et al.[7] detect 76% of the words correctly and recognize 70% of the characters.

4. Conclusion and Discussion

In this paper we presented an algorithm to detect and process artificial text in images and videos. A very high detection rate is obtained with a simple algorithm where only 7.5% of the text boxes in the test data are missed. Our method contains an integral approach beginning with the localization of the text to the multiple frame enhancement and the binarization of the text boxes before they are passed to a commercial OCR. The main contributions lie in the robustness of the algorithm and the exploitation of the additional constraints of artificial text used for indexing. One problem we are still working on is the high number of false alarms produced by the system. Our future research will be concentrated on text with fewer constraints (scene text, general orientations, moving text) and on image enhancement.



(a)

Detection	OCR		
	Method	Recall	Precision
Recall 93.5%	Otsu	47.3%	90.5%
	Niblack	80.5%	80.4%
	Sauvola	72.4%	81.2%
	Our method	**85.4**%	**90.7%**

(b)

Figure 5. Results of the binarization methods (a) Detection and OCR results (b).

References

[1] A. Jain and B. Yu. Automatic Text Location in Images and Video Frames. Technical Report MSU-CPS-97-33, PRIP Lab., Department of Computer Science, 1997.

[2] F. LeBourgeois. Robust Multifont OCR System from Gray Level Images. In *Proceedings of the 4th Int. Conference on Document Analysis and Recognition*, pages 1–5, Aug. 1997.

[3] H. Li and D. Doerman. A Video Text Detection System based on Automated Training. In IEEE Computer Society, editor, *Proceedings of the ICPR 2000*, pages 223–226, 3 Sept. 2000.

[4] W. Niblack. *An Introduction to Digital Image Processing*, pages 115–116. Englewood Cliffs, N.J.: Prentice Hall, 1986.

[5] N.Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

[6] O.D.Trier and A. Jain. Goal-Directed Evaluation of Binarization Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(12):1191–1201, Dec. 1995.

[7] T. Sato, T. Kanade, E. Hughes, M. Smith, and S. Satoh. Video OCR: Indexing digtal news libraries by recognition of superimposed captions. *ACM Multimedia Systems: Special Issue on Video Libraries*, 7(5):385–395, 1999.

[8] J. Sauvola, T. Seppänen, S. Haapakoski, and M. Pietikäinen. Adaptive Document Binarization. In *International Conference on Document Analysis and Recognition*, volume 1, pages 147–152, 1997.

[9] A. Wernike and R. Lienhart. On the segmentation of text in videos. In *Proc. of the IEEE Int. Conference on Multimedia and Expo (ICME) 2000*, pages 1511–1514, July 2000.

[10] C. Wolf and J. Jolion. Extraction de texte dans des vidéos: le cas de la binarisation. In *13ème congrès francophone de reconnaissance des formes et intelligence artificielle*, volume 1, pages 145–152, Jan. 2002.

[11] Y. Zhong, H. Zhang, and A. Jain. Automatic Caption Localization in Compressed Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(4):385–392, Apr. 2000.