# Automatic Text Regions Location in Video Frames ◊

Bassem Bouaziz [1]
Walid Mahdi [1]
Abdelmajid BEN Hamadou [1]

*[1] LARIM, Institut Supérieur d'Informatique et du Multimédia de Sfax, Tunisie*
*bassem.bouaziz@fsegs.rnu.tn; {walid.mahdi, benhamadou.abdelmajid} @isims.rnu.tn*

## Abstract

*Content-based information retrieval from digital video databases and media archives is a challenging problem and is rapidly gaining widespread research and commercial interest. For a reliable retrieval and intelligent access to video programs, indexing should provide semantic descriptors. One way to include more semantic knowledge into the indexing process is to use the text embedded within images and video sequences programs such as credit titles, ellipse, etc. Text in video is rich in information and easy to use, e.g. by key word based queries. In this paper we propose an automatic text regions location technique in digital video frames. The detected text boxes can then be passed to standard commercial OCR software to obtain the full texts used in the video indexing purpose. Our method makes use of four main techniques in image processing, that is an adaptive binarization, multi-resolution, histogram segmentation and morphologic operations to locate text regions. A new technique for histogram segmentation based on Optimum thresholding is then proposed. The quality of localized text is improved by experimental results that we have driven on a large sample of video frames selected from various kinds of video programs (commercials, TV news, full-length films, etc.). Finally, the results of text regions localization are presented.*

*◊ This work is subscribed among the CMCU Project undertaken in LIRIS Laboratory, CNRS, Ecole Centrale de Lyon, France, and in collaboration with Pr. Liming CHEN director of the laboratory and Mr. Mohsen ARDABILIAN FARD assistant professor in Ecole Centrale de Lyon, France .*

## 1. Introduction

Efficient indexing and retrieval of digital video is an important function of video database. Content-based video indexing aims at providing an environment both convenient and efficient for video storing and retrieving [1], especially for content-based searching as those which exists in traditional text-based database systems. Most of research works have been focused on the extraction of the structure information or simple spatial-temporal based features within a video, such as the basic shot boundaries detection [2], color histogram [3][4], texture or movement, for indexing purpose. However, besides these basic features which are important to characterize the video content, one has also to segment or provide underlying semantic descriptors in order to fully enable intelligent video content access.

In our last work, we have made use of the video syntax (special effect, rhythm, specific syntax for news) to drive some semantic hierarchical video structure embedded within a video [5][6][7]. Unfortunately, such a semantic hierarchical video structure, described as shots, scenes and sequences does not provide the semantic description of each segmented units. One automatic way to reach such a semantic description is to segment texts which often occur in video images, especially for TV news. For instance, most movies begin with credit titles, and when a jump occurs in the film narration, it is often indicated by textual terms, called ellipses, such as *"5 months before"* or *"3 years latter"*. TV news make an extensive use of textual information to indicate the subject of each topics, the name of interviewee, the place of report, etc. Thus textual information embedded within a video are very high level clues for the content-based video indexing.

In this paper, we propose an efficient automatic text regions locating technique for video images. These text regions resulted from our technique can then be submitted to OCR in order to obtain the full texts. The experiences that we have driven on images of various movies show the efficiency of our approach.

The rest of the paper is organized as follows. In section 2, we briefly review related works in the literature, emphasizing strengths and drawbacks of each technique. In section 3, first we define text features in video programs, and we then present the general principle of our text region localization approach. In section 4 we present a new approach for introduce the multi-resolution approach for text line detection in video images including an adaptive binarization technique. Section 5 presents our approach for selecting effective text regions from all probable text regions detected in video images. In section 6, we propose a merge process which aims to improve text regions localization. The experimental results are presented in section 7. Our experimentation has investigated different kinds of movie : dramatic film, commercials and newscasts. Concluding remark and further work direction are depicted in the last section.

## 2. Related Works

Text detection in real-life videos and images is still an open problem. The first approach proposes a manual annotation [8]. Clearly manual-based solution is not scalable as compared to the amount of video programs to be indexed. Current automatic text detection methods can be classified into three categories. The first category is connected components-based methods [9][10], which can locate text quickly but have difficulties when text embedded in complex background or touches other graphical objects. For example, Zhong et al [9] extracted text as those connected components of monotonous color that follow certain size constraint and horizontal alignment constraint. The second category is texture based [11][12][13]. This category is hard finding accurate boundaries of text areas and usually yields many false alarms in "text like" background texture areas. Indeed, texture analysis-based methods could be further divided into top-down approaches and bottom-up approaches. Classic top-down approach is based on splitting images regions alternately in horizontal and vertical direction based on texture, color or edge. On the contrary, the bottom-up approach intends to find homogenous regions from some seed regions. The region growing technique is applied to merge pixels belonging to the same cluster [14][15]. Some existing methods do solve the problem to a certain extent, but, not perfectly. The difficulty comes from the variation of font-size, font-color, spacing, contrast and text language, and mostly the background complexity. The third category is edge-based methods [16][17][18]. Generally, text regions can be decomposed by analyzing the projection profiles of edge intensity maps. However, this kind of approaches can hardly handle large-size text. As conclusion, the three categories methods presented above are limited to many special characters embedded in text of video frames, such as text size and the contrast between text and background in video images. To detect the text efficiently, all these methods usually define a lot of rules that largely dependent of the content of video.

Unlike other approaches, our approach has nor restriction on the type of character neither the place where text regions should appear within an image. Our text region detection technique is based on the basic features of texts appearing in digital videos. It is oriented to texts generated by video title machines rather than on scene text.

## 3. Text region characteristics

Textual information in audiovisual program can be classified into two kinds: natural text which appears as a part of the scene (e.g. street names or shop names in the scene), and artificial text which is produced separately from the video shooting and inserted into the scene during the post-processing step by video title machines. Both of them, when they occur within a video program, are of capital importance and they are good clues for content-based indexing and retrieval. However, by the opposition to the natural text which accidentally appear in the scene, the inclusion of artificial text is carefully selected, and thus is subject to many constraints so that the video program is easily read by viewers. Below we summarize the main characteristics of these constraints by the following features:

- Text characters are in the foreground.
- Text characters contrast with background since artificial text is designed to be read easily.
- Text characters are monochrome.
- Text characters are generally upright.
- Text character has size restrictions.
- Character size should not be smaller than a certain number of pixels otherwise they are illegible to viewers.

Our method makes use of these basic features to localize text regions in video shots. It also takes into account the characteristics of video programs, such as the low resolution, presence of noises, the absence of control parameters in the cameras. In our approach we do not use a filtering method to reduce image noises because it may cause problems for the detection of the characters having a very small size. Actually, the majority of text region detection methods in the literature support the hallucination of some text regions which is more acceptable than the false detection. Making use of these basic features, the experimentation that we have conducted on various types of video programs shows that our technique is robust for contrast, font-size, font-color, language, and background complexity. Besides we also note in our experimentation that our method decreases the number of false detection (false alarm).

## 4. Probable text regions localization

The localization of probable text regions within an image generally is a part of pre-processing which is very important for text detection purpose. There are some methods from the pattern recognition field which are based on the thresholding, regrouping or edges detection making use of statistics methods, fuzzy logic or neural networks. There is no method which can be considered as adequate with all applications. In the following, we propose an image contrast correction method then a multi-resolution approach in order to locate probable text regions. We then apply a post-processing to the image resulting from the multi-resolution to locate the boundary of each region.

### 4.1. Contrast correction:

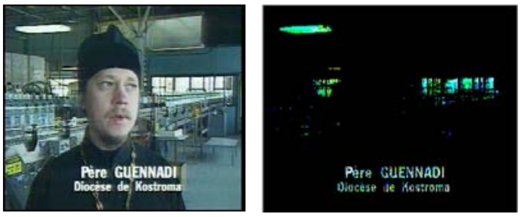
Contrast correction is used to enhance the visual appearance of an image. Contrast modification of an image is defined as follow:

$$S_i = \varepsilon * (g_i - \bar{i}) + \bar{i}$$

where $S_i$ is the ith gray level value of the contrast enhanced image, $\varepsilon$ is a specified contrast parameter, $g_i$ is the ith graylevel value of the original image, and $\bar{i}$ is the mean value of the original image given by:

$$\bar{i} = \left[1/\left(N_x N_y\right)\right] * \sum\sum g\left(X,Y\right)$$

where $N_x$ and $N_y$ are the dimension of the image in the x and y directions respectively, and $g(X,Y)$ is the graylevel value of the pixel at the $X,Y$ coordinate.



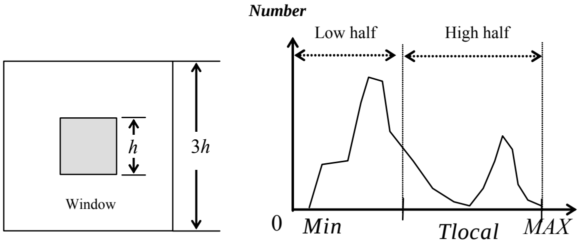
***(a)*** ***(b)***

***a)*** *the original image* ***b)*** *the contrast corrected image with* $\varepsilon$ *=2.0.*

**Figure 1.** Contrast Correction

### 4.2. Adaptive binarization

The image obtained after contrast correction is scanned by a window of size $H \times W$ (H and W are proportional to image height and width respectively). The window scans the entire image step by step first in horizontal direction then in vertical direction. In each step, the origin of window only moves W/2 in horizontal direction (or H/2 in vertical direction) so that the inaccuracy caused by splitting a character with window border can be compensated. The value of points covered by the window is the local area to be analyzed.



***(a)*** ***(b)***

***(a)*** *Window Extract.* ***(b)*** *Local threshold.*

**Figure 2.** Adaptive binarization technique

The threshold is found from the local histogram of this area. Let *MAX* and *MIN* be the highest peak and the lowest peak respectively. We find the low peak at the low half of [*MIN*, *MAX*] and the high peak at the high half of that, and then determine the new threshold (*T*local) ( Figure 2.b), as the lowest position between the low peak and the high peak. The points in this area whose luminance are lower than *T*local are marked with a flag. After the entire image has been scanned, all points with the flag are removed. Applying selective local threshold, we obtain a simplified image.

### 4.3. Multi-resolution approach

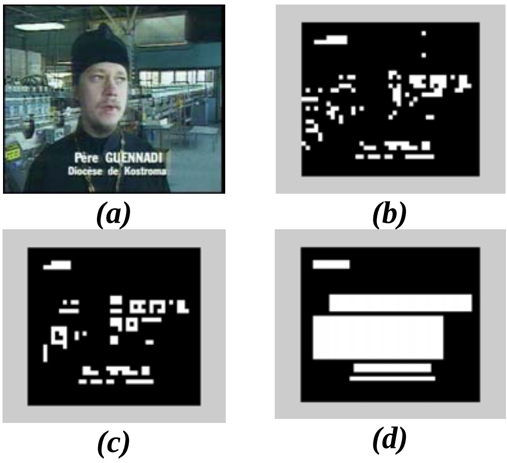
The use of the multi-resolution method for text lines localization is based on the basic feature that a text line generally appears in the shape of a full line in an image with low resolution. Thus we first convert a gray levels image source into a binary image by thresholding. The output binary image *BW* has value of 0 (black) for all pixels in the input image with luminance less than the adaptive threshold and 1 (white) for all other pixels. The multi-resolution process, when applied to an image *BW* returns an image *newBW* that is *M* times the size of *BW*. If *M* is between 0 and 1.0, *newBW* is smaller than *BW*. If *M* is greater than 1.0, *newBW* is larger than *BW*. This passage from an initial resolution of the image *BW* to a new resolution *newBW* is ensured by an interpolation method. Interpolation aims to finds values of a two-dimensional function *f(x,y)* underlying the data at intermediate points. In our experimentation, we have fixed $M$ = 0.125 and we have used the nearest neighbour interpolation method. The multi-resolution method makes possible to filter the input image and to keep only connected components with homogenous color corresponding to a meaningful area.

## 4.4. Image post-processing

This phase follows immediately the multi-resolution and binarization and consist on applying some specific morphological operation to the binary image *BW* for eliminating negative form in the BW image, correcting classification errors using information from the neighbourhood of each pixels and also connecting characters in order to capture complete words. there morphological operations consist of the following steps:

- Negative elimination form:
  - Elimination of all pixels connected to the four edges of the image.
  - Elimination of all pixels connected to the horizontal lines and larger than a predefined threshold *lt*. (In our experimentation $lt \leq 75\%$ of the binary image *BW* width. .
- Removal of isolated pixels (individual white pixels that are surrounded by black pixels).
- Bridges previously unconnected pixels. Diagonal fill to eliminate 8-connectivity of background.

This set of conditional morphologic operations is repeated until the image no longer changes. This is necessary to make the detection results less sensitive to the size of the image and in cases where the distances between characters are large and characters styles not uniform. Figure 3.d illustrates the final state of image shown in figure 3.a applying the two steps multi-resolution and image post-processing. We show then all probable text regions. For capturing text regions on the source image( Figure 3.a), it is sufficient to locate the mapping between coordinates of text regions detected in the binary image (Figure 3.d) and coordinates in the original image.



*(a)* *(b)* *(c)* *(d)*

***a)*** *An input video shot.* ***b)*** *An output image obtained by applying a multi-resolution technique.* ***c)*** *The output image "b" after negative form elimination.* ***d)*** *Image "c" after post-processing.*

**Figure 3.** Probable text region localization process.

However, we can notice that figure 3.d contains five areas candidates of text regions whereas on the source image there are only two effective text regions. The next section tackles this problem by a checking process for each candidate text regions.

## 5. Text regions checking

The false detection of text regions shown in figure 3.d can be explained by the fact that the multi-resolution method based on Bi-level (binary image) thresholding is a method rather efficient for textual document applications for which a pixel belongs to the background or to the certain image object. In the case of video documents, an image generally is made of several objects with different colors. Consequently it is possible to have some false detection of text regions when applying multi-resolution method based on Bi-level thresholding to such kind of images.

In our approach the results obtained from the previous step constitute a first localization of candidate regions susceptible to contain a text. We actually perform a further step which aims to checking each candidate region already localized as a probably text region and decide whether it is an effective text region or not. For this purpose we apply at first an intensity level slicing method to each candidate text region based on intensity optimum thresholding method to separate an object's pixels from the background pixels, then we verify if the contrast within each region is more significant to make us sure that the concerned region contain effectively a texts objects.

## 5.1. Optimum thresholding

This technique is useful when different features of a candidate text region are contained in different gray level. Indeed, if the background is simple, a text string, even of low contrast, can be easily detected by a low threshold, whereas a text string embedded in a complex background needs a higher threshold to further simplify the background. Therefore, it is necessary to determinate a proper threshold for each candidate text region according to its background complexity. For this, we firstly assume that each candidate text region contain effectively a text object. Then, it is possible to assume that histogram of each candidate text region contains two predominant peaks, one due to the background and one due to an object e.g. text object. The goal of this operation is to find for each candidate text region the valley $\mu$ between the two peaks and threshold each region at this gray level value. This means that each gray level $\mu \in [0, L]$ will be mapped into a gray level $v \in [\mu, L]$ according to a transformation depicted by equation 1 :

$$v = f(\mu) \qquad (1)$$

which can be simply defined by Equation 2 :

$$v = \begin{cases} \mu, & a \leq \mu \\ L, & otherwise \end{cases} \qquad (2)$$

To determine the optimum graylevel value dynamically we proceed as follow. First we start by computing the histogram of the candidate text region. Then we smoothe the histogram curve using 41x1 mean filter to reduce error in finding the optimum threshold value. Finally, ignoring the two end points of the histogram curve, we locate the valley $\mu$ between the two peaks and uses this graylevel value as the optimum threshold value.

## 5.2. Text region validation

When each candidate text region is processed in the way described by *optimum thresholding* method described above, a simple analysis of the color spatial variation of all transformed candidate text regions allows us to identify effective text regions. This analysis is based on a basic text characteristics which says that text characters generally contrast with background since artificial text is designed to be read easily. All we need to do is to quantify and to locate each thresholded text region, pixels belonging to the background and pixels belonging eventually to the text object. A simple method to do such operation, is to locate on the histogram curve of each thresholded text region the two more peaks respectively within the interval $[0, \mu]$ and $[\mu+1, 255]$ then we determine their positions $P_1$ and $P_2$. A spatial variation of each candidate text region is then characterized by equation 3 and aims at quantifying the contrast variation.

$$D(P_1, P_2) = abs\ (P_1 - P_2) \qquad (3)$$

If the distance $D(P_1, P_2)$ is greater than a predefined threshold σ, the candidate text region is classified as an effective text region, otherwise it will be ignored. In our experimentation σ = 110. As we can see in figure 3, it is clear that regions 1, 2 and 3 have a weak spatial variation. Consequently these regions are ignored.

## 6. Improving text region localization

This stage aims at capturing complete text area by recursively applying the intensity level slicing process to each effective text region already transformed. Indeed, the text regions obtained by all the treatments presented previously may lead to a non complete text area which is possibly caused by the interpolation method used in the multi-resolution process.

## 6.1. Horizontal delimitation of text region boundaries

We first select a representative horizontal line $Rh_{lg}(i)$ among all lines of an identified text region. For the choice of $Rh_{lg}(i)$ we propose to select the one which is formed by the maximum of pixels horizontally aligned and belonging to characters. Generally the selected line $Rh_{lg}(i)$ will be the one formed by the maximum pixel number having a value equal to L since after transformation process characters are assumed to be monochrome and contrasting with their background. Next, we compare Rhlg(i) with adjacent line $Rh_{lg}(i-1)$ that immediately precedes it ( respectively follows it $Rh_{lg}(i+1)$) . We use the spatial color value distribution and connected monochrome pixels principle as merging criterion. Let $Pos_{Rhlg(i)}$ and $Pos_{Rhlg(i-1)}$ (respectively $Pos_{Rhlg(i+1)}$) to be two sets that describe pixel positions in line $Rh_{lg}(i)$ and $Rh_{lg}(i-1)$ (respectively $Pos_{Rhlg(i+1)}$) , having color value equal to L.

$$Pos_{Rhlg(i)} \cap Pos_{Rhlg(i+1)} \neq \varnothing \qquad (4)$$

If the equation 4 is verified, *Rhlg(i)* will be replaced by $Rh_{lg}(i-1)$ (respectively $Pos_{Rhlg(i+1)}$), and the process is recursively applied until the complete definition of the low and high horizontal boundaries of the text region.

## 6.2. Vertical delimitation of text region boundaries

For the vertical delimitation of text area boundaries, we propose to add to the representative line $Rh_{lg}(i)$ all pixels which satisfy the following conditions :

- Only pixels which are on the left or on the right of pixels forming the representative line Rhlg(i) are treated.
- Only pixels having the same color value than Rhlg(i) pixels, are added to Rhlg(i).
- Pixels addition to the Rhlg(i) line must respect the negative form elimination principle presented in section 4.4.

## 7. Evaluation and experimental results

The experiments are performed following the algorithms presented in this paper. The experimental data are from various videos of some movies and various genres: commercials, newscasts, TV new, and feature. The total length of these videos is more than 100 minutes. All experimental data have been chosen for the variety of text style and image complexity.

For the statistical experimentation, we have carried out exhaustive evaluations and comparison with other interesting methods presented respectively by Wenyin Liu et al [18] and wolf et al [19] using a video database containing 205 video frames (*Total _frames*) containing 964 text regions (Total_text_Regions). However, we note that the comparison with these two methods is very difficult due to the lack of a common video test database. We note then that direct comparison has to be taken, with a “grain of salt“. The statistical experimental results are listed respectively in Table 1.

**Table 1.** Statistical Detection Results for our method.

| *Method* | W.L | W | Our |
|---|---|---|---|
| *Total_Missed_Text_Regions* | 17 | 59 | 28 |
| *Total_False_Alarms* | 97 | 63 | 31 |
| *Correct_Localization_Rate* | 88.17 % | 87.34 % | 94,07 % |
| *False_Alarm_Rate* | 10,06 % | 6.54 % | 3.21 % |

$$\text{False\_Alarm\_Rate} = \frac{Total\_False\_Alarms}{Total\_text\_Regions}$$

$$\text{Correct\_Localization\_}Rate = \frac{Total\_Detected\_Text\_Region}{Total\_text\_Regions}$$

Total_Detected_Text_Region = (Total_ text_Regions - Total_Missed_Text_Regions - Total_False_Alarms);

The experimental results presented by column 4 show that the text localization rate of our proposed method are 94%. Our technique is able to detect text regions with different character size, different style, even in the case of complex image background. Thanks to the selection process of effective text regions our method decreases significantly the false detection rate. The false alarm rate is 3%. Note that we means by false alarm the hallucination of text region.

## 8. Summary and future work

For automatic video indexing purpose we have introduced a new text regions locating technique in digital video images using the basic features of text appearance. The experimentation that we have driven on a large video images selected from various kinds of movies shows that our technique is very efficient, capable to locate text regions with different character sizes and different styles, even in case of texts occuring within complex image background. The future work will be:

- Apply some supervised classification methods in text region localization to reduce false alarm.
- Integrate OCR module and character segmentation to make correct intra-word segmentation.

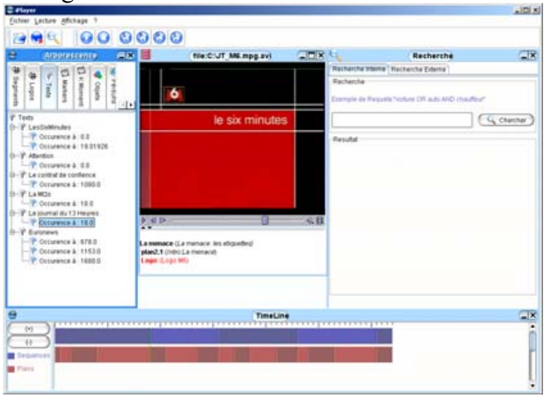


**Figure 4.** Main interface of I-Player

In the other hand, many applications can be derived from this automatic text locating technique. For instance, the automatic extraction of credit titles, an automatic generation of content table video sequence and exploration of other spatial clues, to define other techniques enabling further video browsing and retrieval. In this context we work on complete tools as a new prototype product for video structuring and indexing. This tools is named “I-player“ and it will be industrialized in few years. Figure 4 show the main window of the “I-player “.I-player is composed by four principals components. The first component named “Tree view“ provide an hierarchical view of video sequence as a table of content [5][6][7]. The “Tree view“of the Figure 4 show the table of content generated from the video according Text index. In our tool we provide seven tabbedpage representing “tree view” index (Texts, Logos, key moments, persons, logos, markers and objects). The second one named “Visulazor“is a screen used to visualize scenes within all localized index. The third component is a keyword based search engine. The last component presents the “timeline”.

## 6. References

[1] Petajan, E. D., Bischoff, B., Bodoff, D., and Brooke, N. M., “*An improved automatic lipreading system to enhance speech recognition*,” CHI 88, pp. 19-25, 1988.

[1] Ph. Aigrain, Ph. Joly et V. Longueville, *“Representation-based user interfaces for the Audiovisual Library of Year 2000”*, Proceedings of IS&T/SPIE Conference on Multimedia Computing and Networking, pages 35-45, 1995.

[2] M. Ardebilian, X. W. Tu, L. Chen, *“Improvement of Shot Detection Methods Based-on Dynamic Threshold Selection”,* Proc. SPIE : Multimedia Strage and Archiving Systems II, Dallas, USA, 1997.

[3] G. Pass, R. Zabih, J. Miller, *“Comparing images using color coherence vectors“*, ACM Multimedia proceeding 1996.

[4] M. Swain and D. Ballard, *“Color Indexing“*, International Journal of Computer Vision, Vol. 7 (N°. 1, 1991), pages 11-32.

[5] W. Mahdi, M. Ardebilian, L. Chen, *“Automatic Video Content Parsing Based on Exterior and Interior Shots Classification “*. In the sventh Iternational conference on ADVANCED COMPUTER SYSTEMS, ACS'2000 October 23-25, 2000, ISBN 83-87352-24-7, Szczecin, POLAND, pp. 571-578.

[6] W. Mahdi, M. Ardebilian, L. Chen, *“Automatic Video Scene Segmentation based on Spatial-temporal Clues and Rhythm “*, in International Journal of Networking and Information Sytsems . EDITION HERMES, Vol N°3, n° 1/2000, pp 27-51, ISSN 1290-2926, ISBN 2-7462-0288-3, http://www.hermes-journals.com.

[7] W. Mahdi, L. Chen, Y. Chahir, D. Tsishkou, Y. Liu , *“Segmentation en Sujets dans les Journaux Télévisés”*, dans la conférence Internationale TAIMA :Traitement et Analyse d’Images : Méthodes et Applications , Hammamet (Tunisie) du 8 au 12 octobre 2001, pp 259-264

[8] D. Marc, *“ Media stream : Representing Video for Retrieval and Repurposing“*, ACM Multimedia proceeding, page 478-479, sans Fransisco, CA, ,USA, octobre 15-20, 1994.

[9] A.K. Jain and B. Yu, *“Automatic Text Location In Images and Videos Frames“*, Pattern Recognation, Vol. 31, N° 12, pp. 2055-2076, 1998.
[10] Y. Zhong, K. Karu and A.K. Jain, “ Locating text in complex color images“, Pattern Recognation, 28(10):1523-1535, 1995.

[11] H. Li, D. Doermann, and O. Kia, *“ Automatic Text Detection and Tracking in Digital Video“*, IEEE Trans, Image Processing, Vol. 9, N°1, Jan 2000.

[12] Y. Zhong, H. Zhang, and A. K. Jain, *“ Automatic Caption Extraction of Digital Video“*, Proc ICIP’99, Kobe, 1999.

[13] T. Sato and T. Kanade, *“ Video OCR: Indexing digital news livrairies by recognation of superimposed caption“*, ICCV Workshop on Image and Video retrieval, 1998.

[14] V. Wu, R. Manmatha, and E.M. Riseman*,“ Finding text in images“*, in Proc of the 2nd Intl., Conf on Digital Librairies. Philadalphia, PA, pp 1-10, Juuly 1997.

[15] K. Sobottka, H. Bunke, and H. Kronenberg, *“Identification of Text on Colored Book and Journal Covers“*, in Proc of the 5th Intl., Conf on Document Analysis and Recognation, pp. 57-62, 1999.

[16] T. Sato, T. Kanade, E. Hughes, and M. Smith, *“Video OCR for digital News Archives“*, IEEE International Workshop on Content-Based Access of Images and Video Databases, pp. 52-60, January, 1998.

[17] W. Qi, et *al*., *“Integrating Visual, Audio and Text Analysis for news Video“*, 7th IEEE International Conference on Image Processing (ICIP2000), Vancouver, British Columbia, Canada, 10-13 September 2000.

[18] Y. Hao, Y. Zhang, H. Zeng-guang and T. min, *“Automatic Text Detection In Video Frames Based on Bootstrap Artificial Neural Network And CED“*, in Journal of WSCG, Vol.11, N°1., ISSN 1214-6972 WSG‘2003, February 3-7, 2003, Plzen, Czech Republic. Copyright UNION Agency – Science Press.

[19] C. Wolf , J.M Jolion and F. Chassaing. *Text Localization, Enhancement and Binarization in Multimedia Documents*. In Proceedings of the International Conference on Pattern Recognition (ICPR) 2002, volume 4, pages 1037-1040, IEEE Computer Society. August 11th-15th, 2002, Quebec City, Canada