Chapter 2
Region-Based Caption Text Extraction

Miriam Leon, Veronica Vilaplana, Antoni Gasull and Ferran Marques

Abstract This chapter presents a method for caption text detection. The proposed method will be included in a generic indexing system dealing with other semantic concepts which are to be automatically detected as well. To have a coherent detection system, the various object detection algorithms use a common image description, a hierarchical region-based image model. The proposed method takes advantage of texture and geometric features to detect the caption text. Texture features are estimated using wavelet analysis and mainly applied for *text candidate spotting*. In turn, *text characteristics verification* relies on geometric features, which are estimated exploiting the region-based image model. Analysis of the region hierarchy provides the final caption text objects. The final step of *consistency analysis for output* is performed by a binarization algorithm that robustly estimates the thresholds on the caption text area of support.

Keywords Text detection and localization · Binary Partition Tree

M. Leon (✉) · V. Vilaplana · A. Gasull · F. Marques
Technical University of Catalonia,
Barcelona, Spain
e-mail: mleon@tsc.upc.edu

V. Vilaplana
e-mail: veronica.vilaplana@upc.edu

A. Gasull
e-mail: antoni.gasull@upc.edu

F. Marques
e-mail: ferran.marques@upc.edu

N. Adami et al. (eds.), *Analysis, Retrieval and Delivery of Multimedia Content*,
Lecture Notes in Electrical Engineering 158, DOI: 10.1007/978-1-4614-3831-1_2,

2.1 Introduction

Semantic image indexing relies on the annotation of the presence in the scene of some a priori defined semantic concepts. At a first level of abstraction, semantic concepts are commonly associated with objects. However, object detection is a non-solved problem in a general framework and the extraction of the text in the scene can provide additional relevant information for semantic scene analysis [1]. This is specially true for caption text which is usually synchronized with the scene content. Caption text is artificially superimposed on the video at the time of editing and it usually underscores or summarizes the video content. This makes caption text particularly useful for building keyword indexes [2]. For example, when recognizing a given location (for example, a street), in addition to the information obtained by recognizing the buildings in the image, a caption text associating the scene with a given city may help to confirm the location.

As proposed in [3], text detection algorithms can be classified in two categories: those working on the compressed domain and those working on the spatial domain. Independently of their domain, algorithms can be divided into three phases: (i) *text candidate spotting*, where an attempt to separate text from background is done; (ii) *text characteristics verification*, where text candidate regions are grouped to discard those regions wrongly selected; and (iii) *consistency analysis for output*, where regions representing text are modified to obtain a more useful character representation as input for an OCR. In this chapter, we develop the three phases of the algorithm within the context of caption text.

The caption text detector presented in this work will be included in a more generic indexing system. Actually, the global application is that of off-line enrichment of the current annotation of very large video databases (for instance, the whole repository of TV broadcasters) as well as of creation and instantiation of new descriptors for future annotation of new semantic concepts (for example, searching in the database for a person who previously did not require being explicitly annotated).

Two of the requirements imposed by this application are (i) analysis of the video at the temporal resolution provided by the key frames that are currently stored and (ii) use of an image representation and description which compacts all the scene information in a small number of elements and, at the same time, is as generic as possible, so that the representation can be reused in different contexts (for example, to detect other objects) [4].

Given the first constraint, we concentrate on the problem of caption text extraction in still images. Caption text presents some features that are typically used by text extraction algorithms. The horizontal intensity variations produced by the text are exploited in techniques that analyze the image in the transform domain, either using the DCT [5] or the wavelet transform [6]. Also spatial domain techniques take advantage of this feature by proposing edge detectors to spot the areas with high probability of containing text [7]. Next, spatial cohesion features, such as size, fill factor, aspect ratio or horizontal alignment, are applied to check if text candidate regions are consistent with its neighborhood and to discard false positives [8].

Note that all these techniques are specific for text detection and commonly independent of the approaches dealing with the detection of other semantic concepts. In the case of detecting text in a global indexing system, it is interesting to have a common image representation and a common set of descriptors.

Regarding the image representation, region-based image representations provide a simplification of the image in terms of a reduced number of representative elements, which are the regions. In a region-based image representation, objects in the scene are obtained by the union of regions in an initial partition. To reduce the number of possible region unions, it is useful to create a hierarchy of regions representing the image at different resolution levels. The idea is to have not only a single partition but a universe of partitions representing the image at various resolutions. In this context, object detection algorithms (and specifically text detection algorithms) only need to analyze the image at those positions and scales that are proposed by the regions in the hierarchy [4].

In a previous work, the tree of maxima (and minima) [9] was proposed as hierarchical region-based image model for text detection [10]. Nevertheless, in order to reuse the representation to detect other objects, the Binary Partition Tree (BPT) [11] is used in this work since its suitability for generic object detection was illustrated in [4] and, posteriorly, demonstrated in [12] for the case of various semantic objects of different nature such as human faces, sky regions, traffic signals and car plates.

Given these requirements, we proposed in [13] a method for caption text extraction in still images using a hierarchical region-based image representation. Here, improvements for the first two phases (*text candidate spotting* and *text characteristics verification*) and a solution for the third phase (*consistency analysis for output*) are proposed.

The presentation of these concepts is structured as follows. Section 2.2 summarizes the main ideas behind the image model [11] and its use for object detection and, specifically, text detection [4]. In Sect. 2.3, the region-based caption text detection approach is detailed. This section is structured in three sections in which every phase of the text detector is described. Section 2.3.1 discusses the use of wavelet information to spot the text candidates in the image [6]. The use of the Haar transform in the color domain is proposed to extract text candidates with low contrast in the luminance component. In Sect. 2.3.2, geometrical descriptors are used to confirm the spotted candidates and discard false positives [8]. In that case, we take advantage of the region-based representation to estimate the geometrical descriptors [13] and of the hierarchical image description to obtain the best set of text caption representatives. In turn, Sect. 2.3.3 describes the proposal for the final *consistency analysis for output* step. It is performed by an adaptive binarization algorithm that robustly estimates the thresholds on the area of support of the caption text candidate and provides the final input to the OCR. Section 2.4 discusses the results obtained by this technique. Finally, conclusions are drawn in Sect. 2.5.

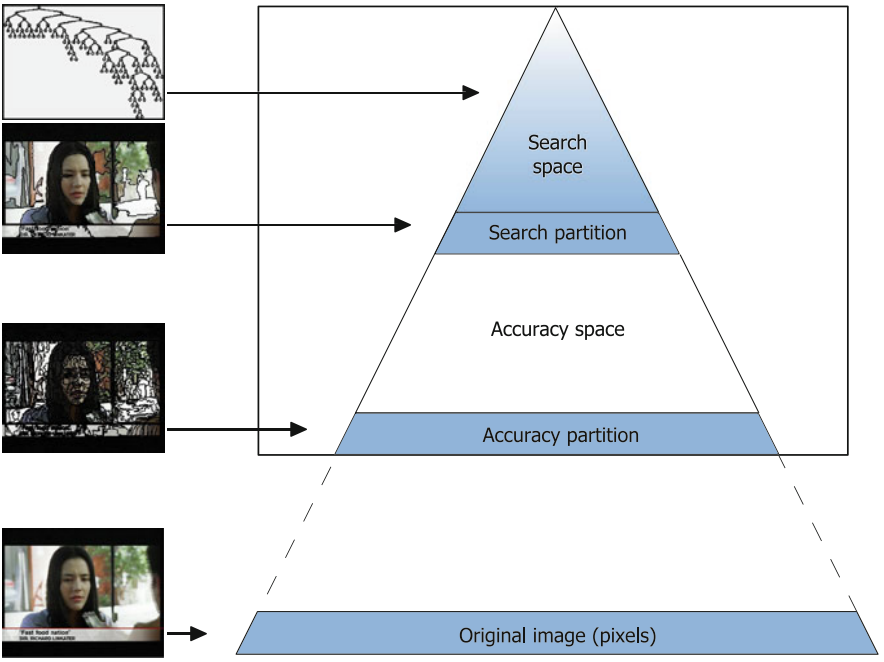


Fig. 2.1 Region-based hierarchical representation

2.2 Hierarchical Region-Based Image Model

The Binary Partition Tree (BPT) [11] reflects the similarity between neighboring regions. It proposes a hierarchy of regions created by a merging algorithm that can make use of any similarity measure. Starting from a given partition, the region merging algorithm proceeds iteratively by (1) computing a similarity measure for all pair of neighbor regions, (2) selecting the most similar pair of regions and merging them into a new region and (3) updating the neighborhood and the similarity measures. The algorithm iterates steps (2) and (3) until all regions are merged into a single region. The BPT stores the whole merging sequence from an initial partition to the one-single region representation. The leaves in the tree are the regions in the initial partition. A merging is represented by creating a parent node (the new region resulting from the merging) and linking it to its two children nodes (the pair of regions that are merged).

The BPT represents a set of regions at different scales of resolution and its nodes provide good estimates of the objects in the scene. Using the BPT representation in object detection, the image has to be analyzed only at the positions and scales that are proposed by the BPT nodes. Therefore, the BPT can be considered as a means of reducing the search space in object detection tasks.

The initial partition can be made of individual pixels or flat zones, which produce a very large BPT. In object detection applications, the use as initial partition of a very accurate partition with a fairly high number of regions is more appropriate [4]. Since this partition is used to ensure an accurate object representation, it is called the *accuracy partition* (see Fig. 2.1). Moreover, in the context of object detection, it is useless to analyze very small regions because they cannot represent meaningful objects. As a result, two zones are differentiated in the BPT: the accuracy space providing preciseness to the description (lower scales) and the search space for the object detection task (higher scales). A way to define these two zones is to specify a point of the merging sequence starting from which the regions that are created are considered as belonging to the search space. The partition that is obtained at this point of the merging process is called the *search partition* (see Fig. 2.1).

In the case of caption text detection, text bars are assumed to be the objects to be detected, and they are extracted by the analysis of the search space. In turn, in the case of scene text detection, characters are not always found in the search partition. This detection requires a more accurate image representation and it will be performed analyzing nodes in the accuracy space. Scene text detection will not be detailed since it is not under the scope of the work presented in this chapter.

2.3 Caption Text Detection Approach

Caption text can be described as text added inside a rectangular bar, horizontally aligned, which contrasts strongly with the bar background and has textured aspect. These features are commonly translated into two types of descriptors: texture and geometric descriptors which are typically used for *text candidate spotting* and *text characteristic verification*, respectively.

Textured areas can be detected using wavelet analysis. However, this approach produces many false positives (that have to be filtered out using geometric descriptors) and some misses in low contrast areas. On the other hand, given the generic framework of our application, the BPT has been created combining color homogeneity and contour complexity criteria [4]. Due to their homogeneous background and regular shape, caption text objects are likely to appear as single nodes in the BPT. Hence, we propose to combine the two approaches.

In a first stage, texture is estimated over the whole image by means of a multiresolution analysis using a Haar wavelet decomposition. Texture information is used to select highly textured regions (candidate regions) in the BPT. Candidate regions are anchor points for caption text detection. In order to correctly estimate useful descriptors to evaluate candidate regions, their area of support is conformed to the object shape characteristics. Region evaluation is carried out combining region-based texture information and geometric features. Among those nodes in the BPT that pass this text characteristic verification stage, final caption text nodes are selected analyzing the BPT structure.

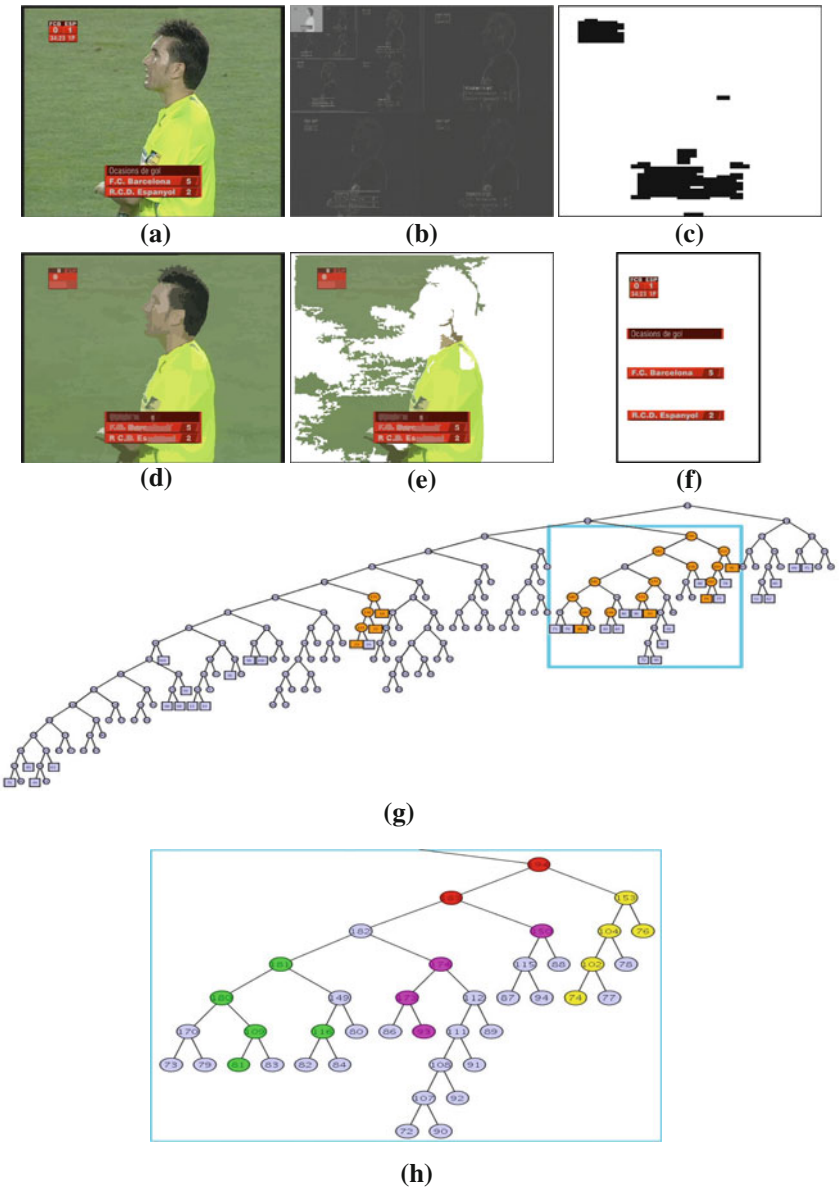


Fig. 2.2 Example of caption text detection. **a** Original image, **b** wavelet transform, **c** text candidate pixels, **d** search partition, **e** text candidate regions, **f** set of final selected regions, **g** BPT showing the selected leaves (*squared nodes*) and the candidate nodes (*orange nodes*), and **h** Detail of the BPT (*rectangle* in **g**) showing the final selected nodes for each text bar (*green*, *lilac and yellow nodes*) and the discarded nodes (*red nodes*)

2.3.1 Text Candidate Spotting

As proposed in [6], texture descriptors such as DWT coefficients give enough information to determine where textured areas can be found in an image. In [13] we proposed to use the power of the LH and HL subbands in a Haar transform (Fig. 2.2b) analyzed over a sliding window of fixed size H × W (W > H to consider horizontal text alignment):

$$P^l_{LH}(m,n) = \frac{1}{WH}\sum_{i=1}^{W}\sum_{j=1}^{H} LH^l(m+i,n+j)^2, \tag{2.1}$$

where l denotes the decomposition level and an analogous expression is used for P^l_{HL}. The window is moved over subbands of the transformed image with an overlapping of half the window size in both directions. Both subbands are analyzed because DWT power in windows containing text should present high values ($>T_1$) in at least one of the two subbands and relevant enough values ($>T_2$) in the other subband. This way, all pixels in a window are classified as text candidates if the power in the window satisfies the following condition:

$$\Big(\big(P^l_{LH} > T_1\big) \wedge \big(P^l_{HL} > T_2\big)\Big) \vee \Big(\big(P^l_{LH} > T_2\big) \wedge \big(P^l_{HL} > T_1\big)\Big), \tag{2.2}$$

where T_1 and T_2 are two thresholds, T_1 being more restrictive than T_2 ($T_1 > T_2$).

This wavelet analysis may produce misses in low contrasted areas. In the case of caption text, such misses are commonly related to text over a background with a similar luminance value but whose chrominance values are different enough to be distinguished by the human visual system. Taking into account this observation, the previous technique has been separately applied to the three YCbCr image components.

The final mask marking all the text candidates is obtained by performing the union of the (upsampled) masks at each decomposition level (Fig. 2.2c) and at each image component. For the results presented in Sect. 2.4, the size of the sliding window is 6×18, $l = 2$, $T_{1Y} = 1200$ and $T_{2Y} = 400$ for luminance, and $T_{1CbCr} = 18$ and $T_{2CbCr} = 10$ for chrominance.

Finally, regions in the search partition (Fig. 2.2d) are selected if they contain any text candidate pixel. Moreover, texture-based selection is propagated through the BPT so that all ancestors of the candidate regions are selected as well (Fig. 2.2g). This is a very conservative policy but, at this stage, it is important not to miss any possible region containing text (Fig. 2.2e).

2.3.2 Text Characteristic Verification

For every selected node, descriptors are estimated to verify if the region represents a caption text object. Initially, a region-based texture descriptor is computed as in Eq. (2.1) but now the sum is performed over interior pixels to avoid the influence of wavelet coefficients due to the gradient in the region boundary. This descriptor is mainly used to filter out regions that have been selected due to the presence in the mask (see Fig. 2.2c) of a few wrong candidate pixels in the surroundings of textured areas.

To complete the verification process, geometric descriptors are calculated for the remaining candidate nodes. Before computing these descriptors, the area of support of candidate nodes is modified by a hole filling process and an opening with a small structuring element (typically, 9×9).

This stage is needed to eliminate small leaks that the segmentation process may introduce due to the interlacing or to color degradation between regions. Such leaks result in very noisy contours that bias the geometric descriptor estimation. Finally, since the opening may split the region into several components, the largest connected component is selected as the area of support for computing geometric descriptors. Descriptors and the thresholds that nodes should accomplish (following a restrictive policy) are listed in the sequel. Values in brackets indicate the thresholds used for the experiments presented in Sect. 2.4 for standard PAL format 720×576 images.

- *Rectangularity* (R): R must be in the range [0, 1]. The calculation of the rectangularity is done using the Discrepancy method [14]. A rectangle is fitted to the region, and the discrepancies between the rectangle and region are measured. R must be greater than T_R; the nearer to 1, the more similar to a rectangle ($T_R = 0.85$).
- *Aspect ratio*: ($AR = Width_{BB}/Height_{BB}$) must be in the range $[T_{AR_1}, T_{AR_2}]$, the upper limit is not strictly necessary but is useful to discard line-like nodes ($T_{AR_1} = 1.33$, $T_{AR_2} = 20$). Given the regular shape (close to rectangular) of caption text objects, the AR is calculated with the bounding box (BB) of the node area of support.
- *Height*: must be in the range $[T_{H_1}, T_{H_2}]$ ($T_{H1} = 13$ pixels for character visibility and $T_{H_2} = 144$, a quarter of PAL format height).
- *Area*: must be in the range $[T_{A_1}, T_{A_2}]$ ($T_{A_1} = 225$, the area of a node with minimum height and minimum aspect ratio, and $T_{A_2} = 138.240$, a third of the PAL format image area).
- *Compactness*: ($CC = Perimeter^2/Area$) must be smaller than T_{CC}, to avoid nodes with long, thin elongations commonly due to interlacing ($T_{CC} = 800$).

The result of applying these descriptors and thresholds to the image shown in Fig. 2.2a, is presented in Fig. 2.2g, where the verified nodes are marked in orange.

At this stage, verified nodes may present two problems. First, as shown in Fig. 2.2h, several verified nodes may be in the same subtree; that is, several (complete or partial) instances of the same caption text object may be represented in a subtree. Second, if the image contains a collection of caption text bars laying close enough, they may

be merged into a single node; that is, a single subtree may represent several caption text objects that, due to their proximity, can be understood as a single one.

The first problem leads to the presence of unnecessary verified nodes, actually representing the same caption text object, that are to be processed in the *consistency analysis for output* step. In that case, the best node in the subtree has to be selected. The straightforward solution of selecting the highest node in the subtree may lead to non-optimal solutions, as discussed in [13]. In that work, a confidence value was estimated for each node, and nodes in the subtree with the highest confidence value were finally preserved.

Nevertheless, a second problem has been detected due to the presence of several caption text objects in the image merged as a single node in the tree, that pass the verification stage. Such configurations are very common, for instance, in sport events, where the data of several participants are jointly presented. In that case, the problem can be more severe due to possible differences in the colors of fonts and backgrounds used in the neighbor caption text bars. If all the bars are selected as a single object, these differences result in a decrease in the performance of the subsequent *consistency analysis for output* step. This step relies on a binarization of the validated caption text bar area of support; if the two classes (character and background) are not homogeneous in color, the binarization may fail.

Having in mind these two problems, we propose here a new strategy to jointly handle both situations in a more robust manner. Subtrees are traversed in postorder. For each subtree, a list of possible caption text objects is created. Verified nodes in the subtree are compared with the previous caption text objects already stored in the list. If the geometrical features of the verified node under analysis allow us to assume that this node belongs to a caption text object already in the list, the verified node under analysis is assigned to this caption text object and the description of this caption text object is updated. If the verified node under analysis cannot be assigned to any already existing caption text object, it is added to the list as a new caption text object.

All these comparisons are performed using only simple geometrical descriptors previously extracted from the tree nodes. In particular, the features that are compared between a verified node under analysis and an already existing caption text object are the coordinates of its center of mass as well as the height and the width of the modified node bounding box. Combining these three elements, the following situations can be detected:

1. The node completes an already existing caption text object: This is the case of a caption text object that is mostly represented by a single node in the BPT but some parts of it (for instance, its interior) are missing. In that case, neither the y-coordinate of the center of mass nor the height or the width of the BB present a substantial change. The node is assigned to this caption text object and the object description is updated.
2. The node horizontally extends an already existing caption text object: This is the case of a caption text object that has been split in the BPT into two horizontal-neighbor regions. The y-coordinate of the center of mass and the height of

the bounding box do not present a substantial change, whereas the width of the bonding box increases. The node is assigned to this caption text object and the object description is updated.

For other situations, the overlap between the node under analysis and the extension of the area of support of the caption text object is analyzed. If they overlap, the node is assumed to be part of the caption text object and its description is updated. If they do not overlap, a new caption text object is defined.

In the example of Fig. 2.2a the largest text box is detected as three separated text bar objects. Figure 2.2h shows a subtree whose root node represents this text box. The search algorithm detects the nodes with the same color as nodes which are part of the same text bar object, obtaining each text bar independently (see Fig. 2.2f).

2.3.3 Consistency Analysis for Output

For every caption text object, a binarization step is carried out. Given the specific characteristics of caption text bars, the binarization is performed by analyzing a few lines in the image. N (typically $N = 3$) equidistant horizontal line segments are selected within the area of support of the caption text object. The mean and the variance of the pixels in each line segment are computed. Line segments with high variance are assumed to be formed by text and background and are used to characterize the probability density function of the text, which is assumed to be Gaussian.

In turn, low variance line segments are supposed to represent the background and can be used to characterize its probability density function that is assumed to be Gaussian as well. Then, binarization is performed by a Maximum Likelihood approach. An example illustrating this process is presented in Fig. 2.3. As it can be seen (and it will be further discussed in next section), this approach leads to good results. Other binarization approaches have been also tested leading to lower performance.

The output of the binarization method is directly used as input for the OCR system. In this work, we have used the opensource **tesseract-ocr** system[1] which can be trained for a specific language and vocabulary.

As previously commented, the binarization approach assumes that background and text can be statistically modeled by Gaussian probability density functions. This assumption does not stand when, for instance, the various words in a given caption text bar are not homogeneous in color. This situation typically leads to a wrong binarization of some of the words. In order to solve this problem, words within the same caption text bar are segmented and a word-by-word binarization is implemented. The segmentation is carried out by applying first an edge detector (in our case, the Canny edge detector [15] but any other similar system could be used) to the caption text bar

[1] http://www.code.google.com/p/tesseract-ocr/

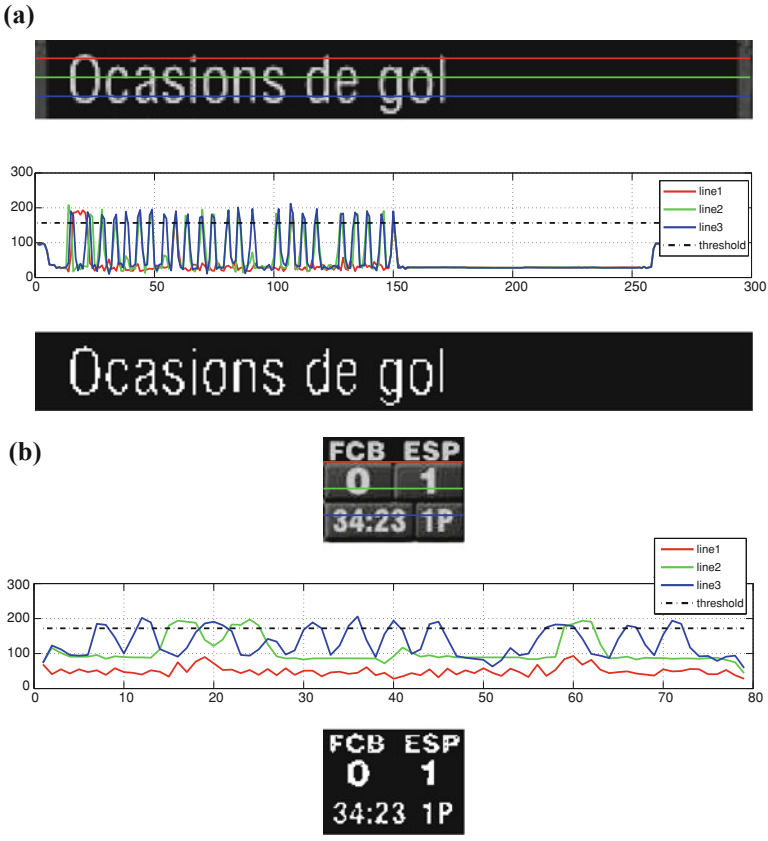


Fig. 2.3 Illustration of the caption textbinarization process for N = 3

area and second by performing a dilation of the detected contours using a rectangular structuring element. Every connected component is assumed to be a separate word and the previous binarization approach is applied.

An example of the usefulness of this word-by-word binarization process is illustrated in Fig. 2.4. The global binarization process fails due to the differences between the text representation in the first and third elements with respect to the second one. The result of the global binarization is illustrated in the second row of Fig. 2.4, where the second text element has been included in the background. The third row of Fig. 2.4 shows the correct result obtained when a word-by-word binarization is used. This example is further illustrated in Fig. 2.6b.

Fig. 2.4 Text bar binarization versus word-by-word binarization

Table 2.1 Detection results related to the number of objects in the first database

	Detected objects	% over 249 objects
Correctly detected	215	86.35 %
Partially detected	22	8.83 %
False negative	12	4.82 %

Table 2.2 Detection results related to the number of objects in the second database

	Detected objects	% over 2063 objects
Correctly detected	1758	85.21 %
Partially detected	40	1.93 %
False negative	265	12.84 %

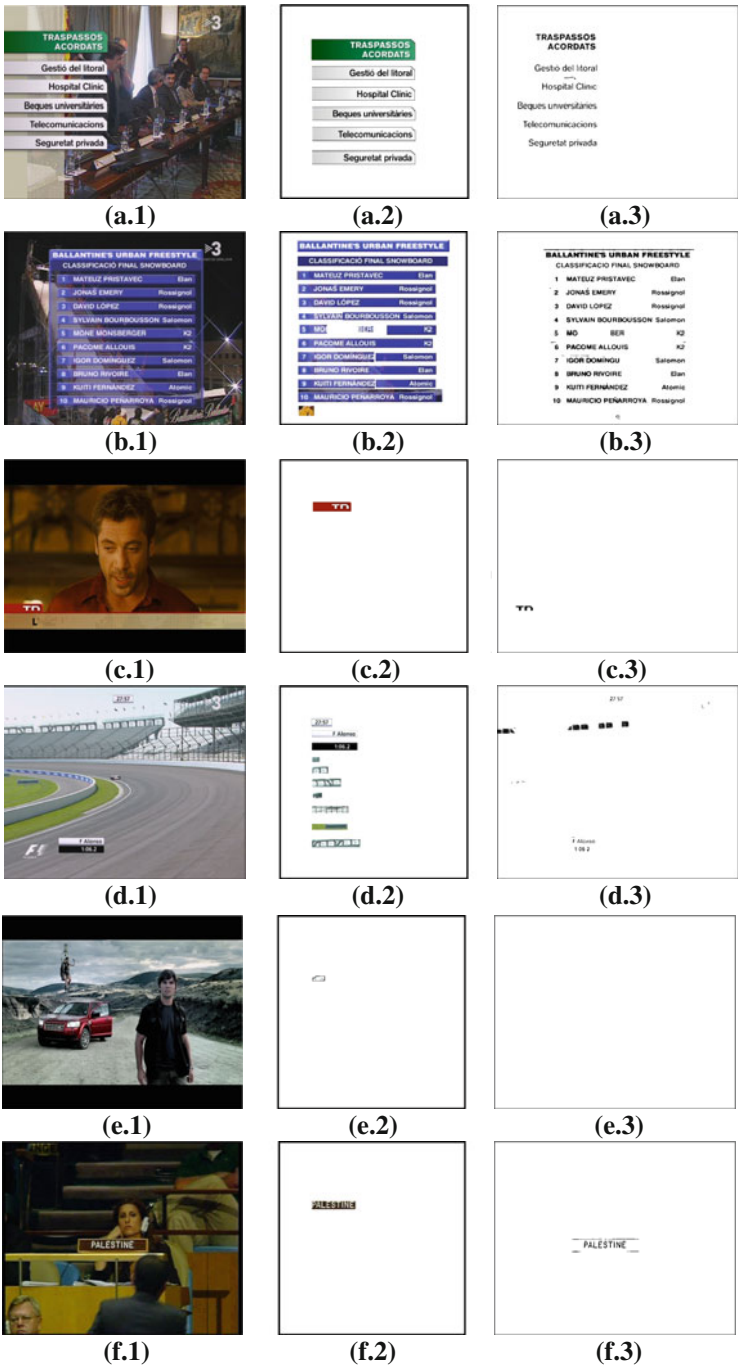
2.4 Results

The technique has been tested in two corpus, one formed by news and sport event videos, and the other one by sport event videos.[2] In the first corpus, there is a total of 249 caption text objects extracted from a set of 150 challenging images with text of different size and color, and complex background textures. Results, classified as correctly detected, partially detected, false positives and false negatives, are summarized in Tables 2.1 and 2.3, and illustrated in Figs. 2.5 and 2.6.

If these values are expressed in terms of recall and precision, partially detected objects (PDO) can be considered as false negative or as detected objects since they represent good anchor points for the following step (see Table 2.3). The number of false positives is 24. Results do not differ significantly from [13] but text bars are detected separately instead of together in a single text box.

In the second corpus there are 2063 caption text objects extracted from 649 key frames. The most remarkable result is that the number of false positives is very high due to the presence in the images of advertising panels and spectators, whereas the

[2] All images used in this chapter belong to TVC, Televisió de Catalunya, and are copyright protected. These key-frames have been provided by TVC with the only goal of research under the framework of the i3media project.



(a.1) (a.2) (a.3)
(b.1) (b.2) (b.3)
(c.1) (c.2) (c.3)
(d.1) (d.2) (d.3)
(e.1) (e.2) (e.3)
(f.1) (f.2) (f.3)

Fig. 2.5 Illustration of the caption text detection process. *First column* Original image; *Second column* List of the final selected regions; *Third column* Binarization result

Table 2.3 Detection results presented as precision and recall for the first and second databases, respectively

PDO	As outlier	As correct	As outlier	As correct
Recall	0.863	0.950	0.8521	0.871
Precision	0.885	0.894	0.692	0.697

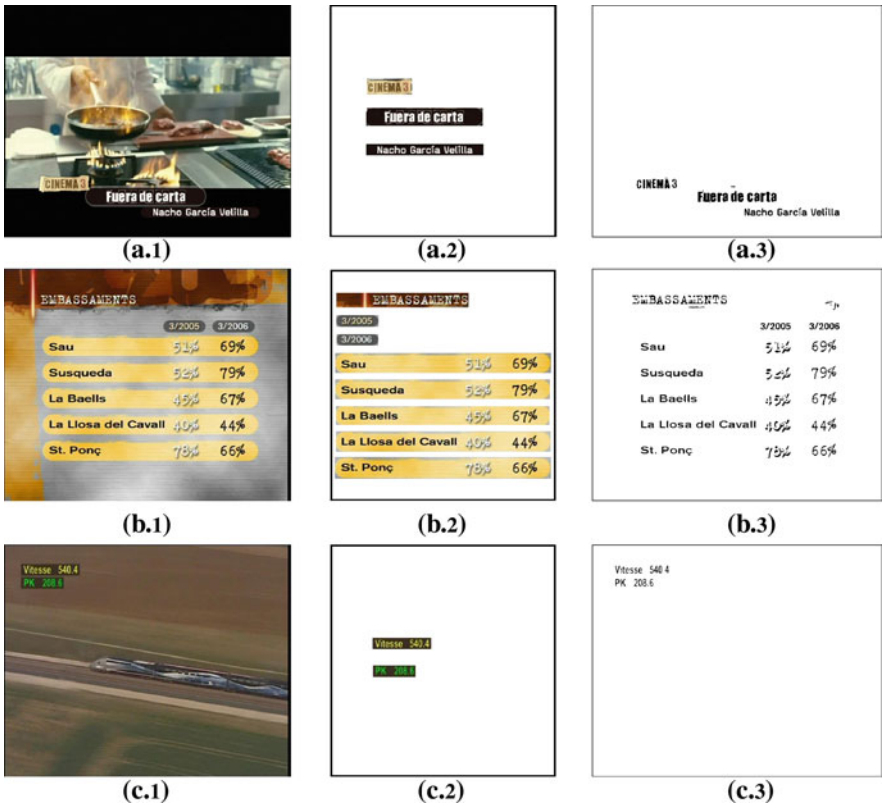


Fig. 2.6 Illustration of the caption text detection process. *First column* Original image; *Second column* List of the final selected regions; *Third column* Binarization result

number of detected text bar is satisfying (see Tables 2.2 and 2.3). Nevertheless, some of these elements are discarded in the third step (see Fig. 2.5d, e).

Figures 2.5 and 2.6 illustrate these results with some images that exemplify the performance and limitations of the algorithm. For every example, we present the different caption text bars that have been detected. To allow analyzing which bars have been detected isolately and which ones have been detected gathered in a single component, when necessary the detected text bars are separately presented regardless of their original position in the image.

Figure 2.5a presents an example of non-perfectly rectangular caption text objects. It is common that caption text objects present some modifications to make information more attractive for the viewer. As it can be observed, the assumed variability on the shape model allows the correct detection of such caption text objects.

Figure 2.5b is an example that illustrates false negatives and partial detections. The similarity between caption text background and objects around mislead the segmentation process and, in some cases, the caption text object is not correctly represented in the BPT. Moreover, we can illustrate as well an example of partial detection: caption text object marked with a "7" has been reported as partial detection since it has not been fully extracted as a single node.

Figure 2.5c is another example of false negative. In this case, although the bar is opaque and rectangular, the amount of text present in the lower caption bar (only a letter "L") does not produce an enough textured region to be detected in the *text candidate spotting* phase (See Sect. 2.3.1). Actually, this false negative is mostly due to a wrong selection of the key frame and in subsequent key frames the whole text in the caption bar appears and is completely detected.

Figure 2.5d, e shows representative examples of typical outliers. These structures correspond to highly textured, rectangular nodes in the BPT which are mistaken by caption text blocks. Nevertheless, they are removed in the following phase of *consistency analysis for output*. Figure 2.5d shows outliers, which are commonly present in sports sequences due to the presence in the image of stands or spectators.

Figure 2.5f shows an example of the behaviour of the algorithm in the presence of scene text. This type of text may present similar characteristics to the caption text (it may be placed in a close-to-rectangular bar and be highly contrasted to its background) and therefore it can be detected as such. Note that in the precision results provided in Table 2.3, 25 % of the detections classified as False Positive are related to scene text.

Figure 2.6a illustrates the robustness of the proposed algorithm to variations in the font type. Note that the algorithm exploits the texture appearance of the text (which is mostly common to all types of fonts) and, therefore, it presents similar performance independently of the font.

Figure 2.6b presents an example of the usefulness of the division of a caption text object into words and their subsequent separated binarization. In this example, the various text elements within each caption object do not share the same color features and, therefore, the global binarization, that assumes a Gaussian probability density function for all the text in the caption object, is wrong. The word-by-word binarization process allows us to correctly binaryze the various text elements.

Finally, Fig. 2.6c shows an image where the use of color information (see Sect. 2.3.1) provides good results. Letters in fluorescent green would be discarded in the *text candidate spotting* phase due to low contrast if only luminance information had been used.

2.5 Conclusions and Further Work

We have presented a new technique for caption text detection. This technique will be included in a global indexing system and, therefore, works on a common hierarchical region-based image representation. The technique combines texture information (through Haar wavelet decomposition) and geometric information (through the analysis of the regions proposed by the hierarchical image model) to robustly extract caption text objects in the scene.

Future work will focus on the creation of new text descriptors and on the analysis of the temporal redundancy of text. The former aims to improve the detection of text in textured areas. The latter, aims to take advantage of the fact that text has to appear at least 2 seconds on the screen so that the viewer can understand the information.

Acknowledgments This work was partially founded by the Catalan Broadcasting Corporation (CCMA) and Mediapro through the Spanish project CENIT-2007-1012 i3media and TEC2007-66858/TCM PROVEC of the Spanish Government.

References

1. Assfalg J, Bertini M, Colombo C, Del Bimbo C (2001) Extracting semantic information from news and sport video. In: Proceedings of the 2nd ISPA, pp 4–11
2. Crandall D, Antani S, Kasturi R (2002) Extraction of special effects caption text events from digital video. Int J Doc Anal Recog 2:138–157
3. Jung K, Kim K, Jain AK (2004) Text information extraction in images and video:a survey. Pattern Recog 37:977–997
4. Vilaplana V, Marqués F, Salembier P (2008) Binary partition trees for object detection. IEEE Trans Image Process 17(11):2201–2216
5. Zhong Y, Zhang H, Jain AK (2000) Automatic caption localization in compressed video. IEEE Trans PAMI 22(4):385–393
6. Li H, Doermann D, Kia O (2000) Automatic text detection and tracking in digital video. IEEE Trans Image Process 9(1):147–155
7. Tekinalp S, Alatan AA (2003) Utilization of texture, contrast and color homogeneity for detecting and recognizing text from video frames. In: IEEE ICIP 2003, Barcelona, Spain
8. Retornaz T, Marcotegui B (2007) Scene text localization based on the ultimate opening. Proc ISMM 1:177–188
9. Salembier P, Oliveras A, Garrido L (1998) Anti-extensive connected operators for image and sequence processing. IEEE Trans Image Process 7(4):555–570
10. Leon M, Mallo S, Gasull A (2005) A tree structured-based caption text detection approach. In: Proceedings of 5th IASTED VIIP, pp 220–225
11. Salembier P, Garrido L (2000) Binary partition tree as an efficient representation for image processing, segmentation and information retrieval. IEEE Trans Image Process 9(4):561–576
12. Vilaplana V, Marques F, Leon M, Gasull A (2010) Object detection and segmentation on a hierarchical region-based image representation. In: Proceedings of the ICIP-10, IEEE international conference on image processing, pp 3393–3396, Hong Kong, China
13. Leon M, Vilaplana V, Gasull A, Marques F (2009) Caption text extraction for indexing purposes using a hierarchical region-based image model. In: IEEE ICIP 2009, El Cairo, Egypt
14. Rosin PL (1999) Measuring rectangularity. Mach. Vis. Appl. 11(4):191–196
15. Canny J (1986) A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 8(6):679–698