Efficient Morphological Algorithms for Video Structuring and Indexing

Claire-Hélène Demarty and Serge Beucher

Centre de Morphologie Mathématique - Ecole des Mines de Paris
35, rue Saint-Honoré, 77305 Fontainebleau cedex, FRANCE
tel: 33 1 64 69 48 04 / 33 1 64 69 47 97
fax: 33 1 64 69 47 07
demarty@cmm.ensmp.fr / beucher@cmm.ensmp.fr

Abstract. In this paper[1] we propose automatic tools useful for the segmentation and indexing of video documents. These tools provide a first structure of a video document, which is a sound starting point allowing further sophisticated and dedicated algorithms to be run on its parts. Apart from being automatic, these tools are very simple, fast and efficient, due to the use of morphological filters. Two different detection algorithms of geometric (cuts, wipes) or chromatic (fades, dissolves) transitions are described, followed by a tool to select representative keyframes in a shot. Two other tools for inner shot change detection and detection of related shots are also presented, the latter being used in an application of newscaster detection.
Keywords: video indexing, content-based indexing, mathematical morphology

1 Introduction

In the context of multimedia indexing, the need for powerful tools is increasing. To reach a high level of abstraction and semantic information, it is commonly accepted that the whole indexing process cannot be totally automated. However, when faced with the large number of documents produced daily, parts of this process need to be automatic, if only for saving time for more sophisticated semi-automatic or manual parts. This is typically the case for transition extraction from video documents, for which several different techniques have already been proposed [1].

This paper therefore presents several automatic indexing tools which produce a first structure of a video document when combined. This structure can subsequently be used as a sound basis to which all semantic or syntactic additional information concerning the document can be related. The structuring and indexing tools proposed hereafter share the same essential qualities of being fast, very simple and reaching a high result level. Two transition detection algorithms for geometric (sec. 2.1) and chromatic transitions (sec. 2.2), are described first. The algorithm efficiency comes from the use of morphological filtering. The input sequences are color, non-encoded sequences, as not all video documents are encoded (old films, video documents directly at the source of applications such as teleconference, etc.). Added to the ability of applying these algorithms to all kinds of video documents, their simplicity allows one to reach at least the same efficiency (and even better) as the running time of MPEG based algorithms.

This is followed by the presentation of a tool for extracting representative key images from shots and for reducing this first selection to the most relevant images (sec. 3). We then describe a tool for detecting related shots in the document and present an application of newscaster detection. This application proves that these automatic and very simple tools listed above already allow us to construct an initial and nevertheless efficient syntactic structure of a video document, thus enabling one to answer questions of a rather high semantic level.

2 Temporal Splitting of the Document into Shots

As already mentioned in the introduction, two different algorithms are described according to whether the transition is *geometric* or *chromatic*. These two algorithms use morphological tools to extract particular shapes of 1D signals.

[1] This study is supported by the CNET-CCETT (France Télécom). The original images are copyright of the CCETT and of the French Television channels TF1, A2, FR3 and M6.

2.1 Extraction of Geometrical Transitions

A cut occurs in a document when two shots are simply concatenated. In this case, the whole frame is instantaneously affected by the transition. Geometric transitions act locally as a cut: for each pixel of the image, a local cut occurs, but not at the same time for each of them. The transformation proceeds according to a certain geometric model. A simple example is the basic horizontal wipe. The spatial model and the duration of the transformation are two main characteristics of the transition. The proposed algorithm is illustrated in fig. 1.

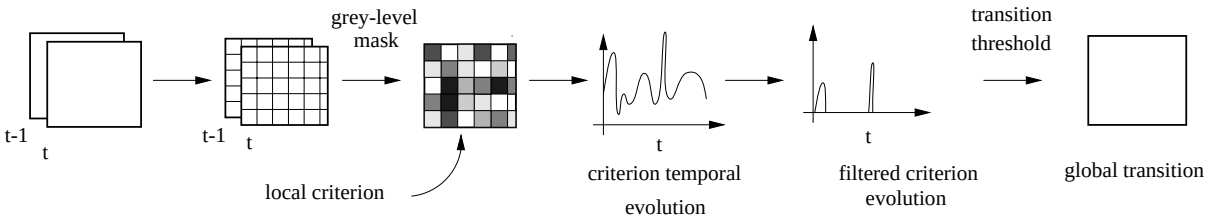

Fig. 1. Description of the local algorithm.

A Local Criterion Many transition detection algorithms compute a similarity criterion between two successive frames globally on the whole image. Depending on the choice of the criterion (point to point color difference, histogram difference or χ^2 computation), these algorithms are more or less sensitive to noise and to object or camera motions. In all cases, this results in the loss of spatial information, *i.e.* the geometric model of the transition. As in [8], we choose the alternative, which consists in splitting the image into small blocks (typically 20×20 pixels for CIF images) and then computing the chosen criterion locally on each pair of corresponding blocks in two successive images. Different block sizes were tested; a 20×20-pixel size appears to be the best compromise between computing time, detection results and keeping the spatial information, i.e. the transition geometry, as precise as possible (see section 2.1).

As for the choice of criterion, a mean distance in the RGB space for all pixels in two blocks is computed (complexity of $O(n)$). Contrary to what is done in [8], color histogram differences and χ^2 were set aside due to their computing cost and the loss of spatial information, although they usually lead to sharper criterion curves. This choice is balanced by a further filtering step (see next section).

After the local computation of the criterion, a grey level mask of the transition is obtained, each grey level corresponding to a criterion value on its corresponding block. We define our global criterion as the volume of this new image. As a global value is available for each pair of two successive images it is possible to compute the criterion temporal evolution curve for a given sequence (see left of fig. 2). Each peak corresponds to a cut, but at this stage the rough curve contains too much noise or strong variations, which prevents one from making a choice of a threshold for extracting the peaks. For this reason, an additional filtering step is applied to the curve before thresholding.

Morphological Filtering Mathematical morphology [9, 10] offers numerous powerful tools for filtering curves or images. This image processing technique, based on set theory, has already proved its efficiency in numerous image processing problems. In particular, the *top-hat* transform, TH, is especially well-designed for extracting small white details from images or peaks from one-dimensional curves. It consists of a comparison between a curve (or an image) and a structuring element, *i.e.* an object whose shape and size are chosen by the user. The top-hat of an object X is then the subset of X, obtained by keeping only the part of X in which the structuring element cannot be included (see fig. 3, part b). An alternative to the classical top-hat is the *inf top-hat*: instead of keeping the peaks in which the structuring element cannot be included exactly as on the original curve, we keep them with a height corresponding to the difference between the maximal and the minimal values under the peak (see fig. 3, part c). This leads to enhanced peak heights. In the case of the evolution curve, the structuring element is a small line of three pixels. The result of the inf top-hat transform as applied to the example in fig. 2 can be seen on the right part of the figure: only emphasized peaks remain (see, especially for peak #3, the usefulness of the inf top-hat).

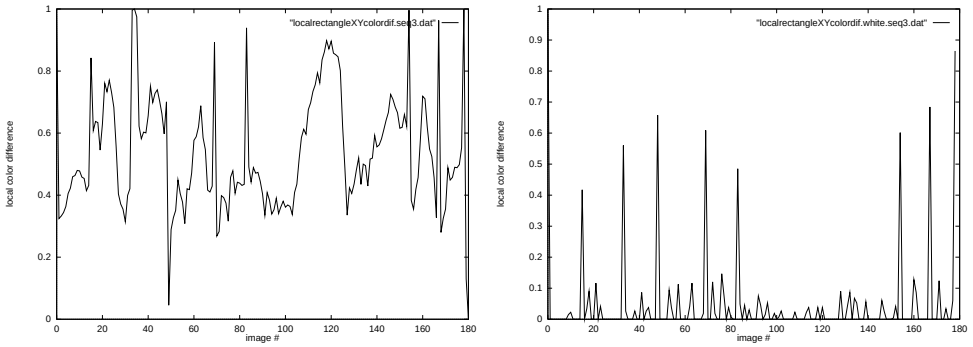

Fig. 2. Examples of evolution curves of the global criterion before any filtering (left), and filtered by a top-hat transform (right). Sequence *kart race*, 30 s, CIF format, frequency of 5 Hz, poor acquisition quality, objects moving fast and near the camera.

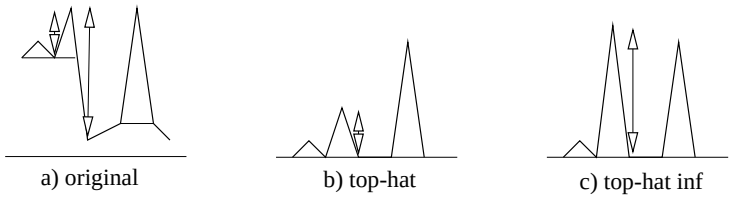

Fig. 3. Top-hat and inf top-hat of an object X by a line of length 3.

A *transition* threshold, set at 0.2 for all the test sequences, is then applied directly to this filtered curve. Other threshold values were also tested, but this one happened to be the best choice, leading to the conclusion that the parameter is not dependent of the input sequence: there is no real need for an automatic threshold adjustment. Tested on 22 video documents (with durations ranging from a few seconds to 15 minutes and 274 cuts), the algorithm works in 0.6 times real time and attains a mean detection rate of 98.3% for cuts, with a small rate of false detections at 1.5%.

The set of test sequences consists in a sample of a large variety of situations: objects moving fast and near to the camera, person crossing the stage, flashes, sequences of poor quality, transitions with duration of one frame only, indoor/outdoor scenes, sport, interviews, crowds, etc. All these different cases are well handled. The only difficulty remains in frames with spatial luminance changes of high frequency, combined with a small motion, for which numerous false alarms appear, as no motion compensation is done. Even in this particular case, the further relation establishment (section 4.1) helps to reduce afterwards the number of false alarms.

Geometric Model The use of a local criterion allows one to keep track of the transition geometry, which is not the case when using histogram or χ^2 comparisons. In the proposed algorithm, the spatial feature of each transition is directly accessible in what is called the *transition mask*. These masks can be grey level (each grey level corresponding to a certain criterion value) or binary ones (when thresholded).

The study of the temporal evolution of binary masks gives access to the geometry of the transition. An example of such an evolution is illustrated in fig. 4, part (a), in the case of a wipe (a wipe consists of a line passing through the image, with the current shot on one side and the next shot on the other side of the line). Mask geometry characterizes this image transformation, which is even more easily identified when morphological filtering (by opening and closing, its dual operator [9]) is applied to the mask. This suppresses erratic white rectangles and fills undesired holes, allowing a better recognition of the transition. This could also be improved by studying the union of the binary masks instead of the masks themselves. Figure 4, part (b), presents the resulting mask unions, once filtered by morphological operators. Once the masks are obtained, simple measures such as surface and intercepts in all directions are computed and their temporal evolution curves are compared with the precomputed curves of ideal transition models. The selected transition corresponds to the best correlated model. This technique should be further evaluated but preliminary tests currently give satisfactory results.

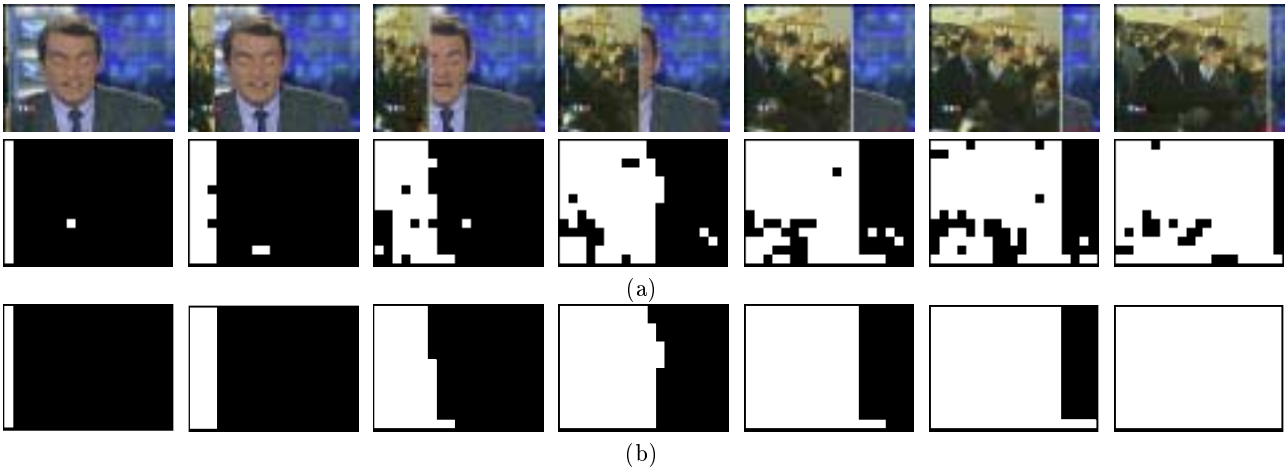

Fig. 4. Temporal mask evolution in the case of a wipe: Without (a) and with (b) morphological filtering.

Besides the recognition of the occuring transition model, the transition masks allow the definition and extraction of regions of interest as being local changes. Once these regions are extracted, they constitute simplified data, which can in their turn be processed by more sophisticated indexing tools, as text recognition or motion segmentation. In fig. 5, two such examples of transition masks are proposed.

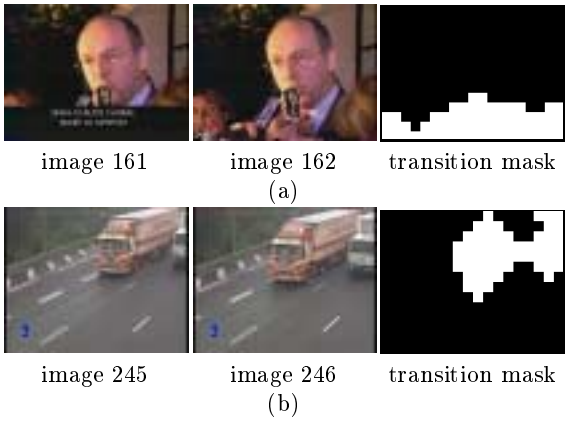

Fig. 5. Examples of transition masks, (a) in the case of anchored text and (b) for a moving object.

2.2 Hierarchical Algorithm for Chromatic Transition Extraction

In chromatic transitions the pixel grey levels are modified, not only because of the change from one shot to another as in a geometric transition, but also because of the transition itself. For example, during a dissolve the image grey levels are affected by some linear transformations from the grey levels of the preceding and following shots.

Another Similarity Criterion As in sec. 2.1 this second algorithm also produces a temporal evolution curve; the number of pixels with a non zero color difference between two successive frames is chosen as the similarity criterion. Contrary to the cut detection, no splitting into blocks is achieved, neither is the criterion computed globally on the image. The color difference is a point-to-point distance and for each pixel we decide to keep it or not. This local decision is derived from the theoretical observation that the distance between successive images is constant and non-zero during a dissolve from one non-moving

shot to another. In practice, provided that the image contains rather large and homogeneous regions, this assertion is also true inside the regions, and false on the contours. Therefore the distance has to be computed as locally as possible to decrease the contour influence.

For this criterion, dissolves appear as small hills whose width corresponds to the transition duration (see fig. 6, part (a)). Extracting these hills will give the exact location and duration of the dissolve. This extraction is also achieved by means of morphological filtering, but applied hierarchically to the curve, as explained in the following section.

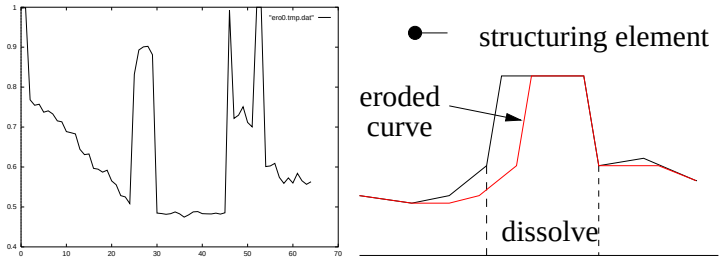

Fig. 6. (a) Example of criterion evolution curve. (b) Erosion of a 1D signal by a line of length 2.

Hierarchical Morphological Filtering Contrary to peak extraction, the top hat transform is not of any use for the detection of hills. We therefore first have to reduce the hill to a single peak. This is achieved by successive erosions [9] on the curve. This other morphological operator results in suppressing a small band (same width as for the structuring element) on one side of the hill, for a non-centered structuring element, as illustrated in fig. 6, part (b)). By repeating successive erosions of size 1, the hill width decreases until only a peak remains. A top hat applied after each erosion does not give any result until this last step. At this point, the number of erosions which are necessary to obtain this result corresponds exactly to the duration δ of the dissolve. Thus we also get the starting point of the transition which occurs δ seconds before the peak position. For this second algorithm the mean detection rate reaches 78.2% (on 24 chromatic transitions). The hierarchical part of the filtering does not really affect the whole algorithm speed, as it works in 0.8 times real time, on a pentium II, 400MHz. Moreover one should keep in mind that any code optimization could easily improve these speeds.

3 Extraction and Selection of Key Frames

3.1 Second Hierarchy of Peaks

On the filtered curve of fig. 2, a second hierarchy of smaller peaks appears, revealing minor changes in the sequence, such as disappearing/appearing anchored text/images, flashes, etc. For a given shot, we automatically select the corresponding images as representative frames. These keyframes, added to the first and the last frames, give a satisfactory summary for each shot: only 10.2% of the images are kept and new semantic information is retrieved as it can be seen in fig. 7. Furthermore no additional computation is made as key frames are extracted from the same evolution curve as for the cut detection. Some of these keyframes are, nonetheless, redundant due to the unconditional addition of the first and last frames, although this addition does not necessarily provide new information; in addition, some peaks of the second hierarchy do not lead to significant visual changes, because of noise or of the absence of motion compensation. For indexing purposes, the redundant keyframes need to be removed to reduce the stored information as well as to avoid redundant indexing of these images. This removal is achieved thanks to an inner change detection which we propose in the next section.

3.2 Detection of Inner Changes

In addition to the deletion of redundant frames, comparing the selected keyframes of each shot to one another also allows one to detect whether there were any changes in a particular shot. This comparison is

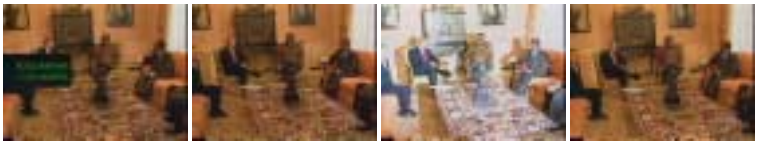

Fig. 7. (a) Example of selected key frames.

made by using the same local similarity criterion as in sec. 2.1 between two keyframes of a shot. According to another threshold, automatically set at one and a half times the value of the transition threshold, one decides whether a change occurs or not. By this change detection, the number of keyframes is reduced significantly (24.5% of the key frames are suppressed). In particular, most of the redundant keyframes extracted from the second peak hierarchy (see sec. 3.1) are removed. Coupling this detection with the study of the mask also gives more information on the exact location of the change. This indicator is interesting in order to run specific algorithms on the changing region afterwards (like text extraction, for example). As a side effect, this change detection allows confirmation of the transition detection step. When no change is detected, there is hardly any chance that a non-detected transition occurred in the current shot. However, a change detection could lead to a more precise study of this part of the document, for example by motion estimation. Once the keyframes are selected, the following indexing step consists of a segmentation of these images. To do this, a color segmentation algorithm dividing the image into a few large and homogeneous regions was proposed in [3].

4 Structure of the video document

4.1 Construction of Related Shots

Shot detection is the first step in the elaboration of a structure of the video document. Besides the shots, which are the first lower level of a temporal hierarchy, several other syntactic objects, such as scenes and sequences, are used to construct this hierarchy. Shots (frames recorded contiguously, same unity of place, action and time) can be grouped into scenes (continuity of actions sharing the same unity of place and time [2]) which can in turn also be grouped into sequences (same unity of subject) [7]. In this respect, establishing relations between shots is useful to extract different levels of the hierarchy [6].

At a low level of syntactic information, this relation detecting tool aims at answering the question: are these two shots similar? Here again, keyframes, but this time from different shots, are compared, still by the use of the same dissimilarity criterion as for the cut detection; when the criterion value is lower than a threshold (same value as for the *inner change threshold* and therefore automatically set, by the algorithm itself, as function of the *transition threshold*) the keyframes are related. One pair of related key frames between two different shots is sufficient to build a relation between these two shots.

Apart from the construction of higher levels of the hierarchy, we now have syntactic clues as to the organization of the document: a) when two groups of related shots are interlaced (as illustrated on fig. 8), this is a strong indicator in favour of an interview sequence; b) when a shot contains only one frame and is surrounded by two related shots it is worth determining if there was luminance flickering on this particular shot (a flash is a rather common event when dealing with TV newscasts, see fig: 7); c) when two successive shots are found to be related, it could mean that a false detection occurred.

4.2 Application to Newscaster Detection

This last section proposes a final indexing tool, which is a direct application of all the tools presented above. We emphasize that there has been no use of high semantic information. When considering the particular document class of TV newscasts, newscaster shots play a special role in the hierarchical organization of the document: they impose a structure on the newscast and separate the various news topics. To extract them, four criteria were elaborated and merged. Each of them leads to a probability of each shot being a newscaster shot. All of them are based on inherent properties of this kind of shots, whatever channel, land or time period they belong to: a) There is at least one person in front of the camera of a certain size and at a certain location in the frame. b) The shots reappear regularly throughout the whole newscast. c) They are often related to a shot at the beginning and at the end of the newscast. d) The background tends to be motionless.

Fig. 8. Groups of related shots for the sequence *Interview*.

The first criterion therefore measures the probability of having at least one connected component with the specific skin color [5], of a certain size and more or less in the middle of the frame. The size and the notion of "middle of the frame" are chosen experimentally on newscaster images, and are tested according to fuzzy sets. The second criterion consists in giving a probability to each group of related shots, directly proportional to the number of shots it contains. The regularity of the occurrences of such shots during the newscast is not taken into account at this point. For the third criterion the highest probability is given to the group containing shots at the beginning or at the end of the document. And finally, for the fourth criterion, we assume that a good estimation of the background motion is accessible outside a frame situated in the middle of the image and determined experimentally. This last probability is still computed by using the similarity criterion of sec. 2.1. All these criteria are then merged, using a simple weighted mean, in order to get a unique probability for each group of related shots, the maximal probability corresponding to the newscaster group. Figure 9 shows the selected newscaster group (with a probabilty of 87.7%) among 14 relation groups for a sequence, *jtv1*, simulating a TV newscast of 6 minutes. Due to the way the algorithm is implemented, any extra criterion can easily be added and its probability merged with the result.

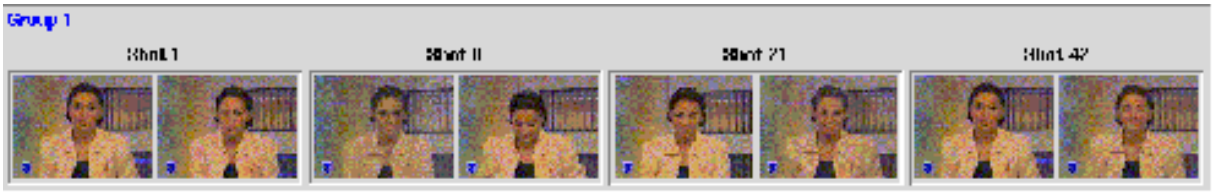

Fig. 9. Newscaster group for the sequence *Jtv1*, with a maximal probability of 87.7%.

5 Conclusion

Several automatic indexing tools have been presented. All of them provide good results, as proved by the tests, whilst remaining both simple and fast, and depending on the same parameters: the rectangle size and the *transition threshold*, which has a fixed value whatever the sequence is (another parameter should be added when dealing with binary transition masks). They allow one to start building a hierarchical representation of each video document and already give access to quite high level information when combined. The obtained hierarchical structure can further be considered as a starting point from which to proceed with manual and more sophisticated indexing of the structured documents.

References

[1] R. Brunelli, O. Mich, and C. M. Modena. A survey on the automatic indexing of video data. *Journal of Visual Communication and Image Representation*, 10(2):78–112, june 1999.

[2] G. Davenport, T. G. Aguierre Smith, and N. Pincevert. Cinematic primitives for multimedia. *IEEE Computer Graphics and Applications*, pages 67–74, july 1991.

[3] C. H. Demarty and S. Beucher. Color image segmentation using an hls transformation. In *International Symposium on Mathematical Morphology (ISMM'98)*, Amsterdam, The Netherlands, june 1998. Kluwer Academic Publishers.

[4] C.H. Demarty and S. Beucher. Morphological tools for video indexing. In IEEE Computer Society, editor, *International Conference on Multimedia Computing and Systems, ICMCS'99*, volume 2, pages 991–992, Florence, Italy, june 1999.

[5] David A. Forsyth and Margaret M. Fleck. Identifying nude pictures. In *3rd IEEE Workshop on applications of computer vision*, Sarasota, Florida, USA, december, 2-4 1996.

[6] Riad Hammoud, Liming Chen, and Dominique Fontaine. An extensible spatial-temporal model for semantic video segmentation. In *1st International Forum on Multimedia and Image Processing (IFMIP'98)*, Anchorage, Alaska, may, 10-14 1998.

[7] Rune Hjelsvold. Video information contents and architecture. In *Proceedings of the 4th International Conference on Extending Database Technology*, Cambridge, UK, March 28-31 1994.

[8] Akio Nagasaka and Yuzuru Tanaka. Automatic video indexing and full-video search for object appearances. In *2nd Working Conference on Visual Database Systems, IFIP WG 2.6.*, pages 119–133, Budapest, Hungary, october 1991.

[9] M. Schmitt and J. Mattioli. *Morphologie Mathmatique*. Masson, Paris, 1994.

[10] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, London, 1982.