EURASIP Journal on Applied Signal Processing 2002:4, 1–11

Overview of the MPEG-7 Standard and of Future Challenges for Visual Information Analysis

Philippe Salembier
Universitat Politècnica de Catalunya, Campus Nord, Modulo D5, Jordi Girona, 1-3, 08034 Barcelona, Spain
Email: philippe@gps.tsc.upc.es

This paper presents an overview of the MPEG-7 standard: the Multimedia Content Description Interface. It focuses on visual information description including low-level visual Descriptors and Segment Description Schemes. The paper also discusses some challenges in visual information analysis that will have to be faced in the future to allow efficient MPEG-7-based applications.

1. INTRODUCTION

The goal of the MPEG-7 standard is to allow interoperable searching, indexing, filtering, and access of audio-visual (AV) content by enabling interoperability among devices and applications that deal with AV content description. MPEG-7 specifies the description of features related to the AV content as well as information related to the management of AV content. As illustrated in Figure 1, the scope of the standard is to define the representation of the description, that is, the syntax and the semantics of the structures used to create MPEG-7 descriptions. For most description tools, the standard does not provide normative tools for the generation nor for the consumption of the description. This is not necessary to guarantee interoperability and, moreover, this allows future improvements to be included in MPEG-7 compliant applications. However, as will be discussed in this paper, in order to guarantee interoperability for some low-level features, MPEG-7 also specifies part of the extraction process.

MPEG-7 descriptions take two possible forms: (1) a textual XML form suitable for editing, searching, filtering, and browsing and (2) a binary form suitable for storage, transmission, and streaming. Overall, the standard specifies four types of normative elements illustrated in Figure 2: Descriptors, Description Schemes (DSs), a Description Definition Language (DDL), and coding schemes.

In order to describe AV content, a set of Descriptors has to be used. In MPEG-7, a *Descriptor* defines the syntax and the semantics of an elementary feature. A Descriptor can deal with low-level features, which represent the signal characteristics, such as color, texture, shape, motion, audio energy or audio spectrum as well as high-level features such as the title or the author. The main constraint on a descriptor is that it should describe an elementary feature. In MPEG-7, the syntax of Descriptors is defined by the *Description Definition Language (DDL)* which is an extension of the XML Schema language [1]. The DDL is used not only to define the syntax of MPEG-7 Descriptors but also to allow developers to declare the syntax of new Descriptors that are related to specific needs of their application.

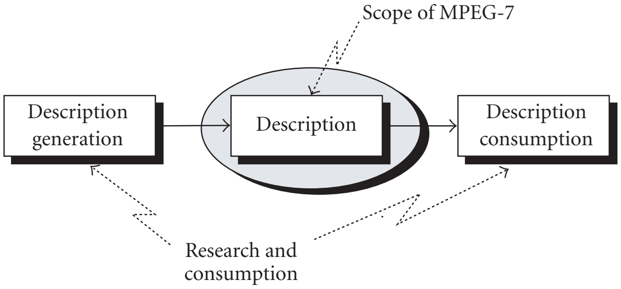


Figure 1: Scope of the MPEG-7 standard.

In general, the description of AV content involves a large number of Descriptors. The Descriptors are structured and related within a common framework based on *Description Schemes (DSs)*. As shown in Figure 2, the DSs define a model of the description using as building blocks the Descriptors. The syntax of DSs is also defined with the DDL and, for specific applications, new DSs can also be created.

When the set of DSs and Descriptors is instantiated to describe a piece of AV content, the resulting description takes the form on an XML document [2]. This is the first normative format in MPEG-7. This format is very efficient for editing, searching, filtering, and processing. Moreover, a very large number of XML-aware tools are available. However, XML documents are verbose, difficult to stream and not

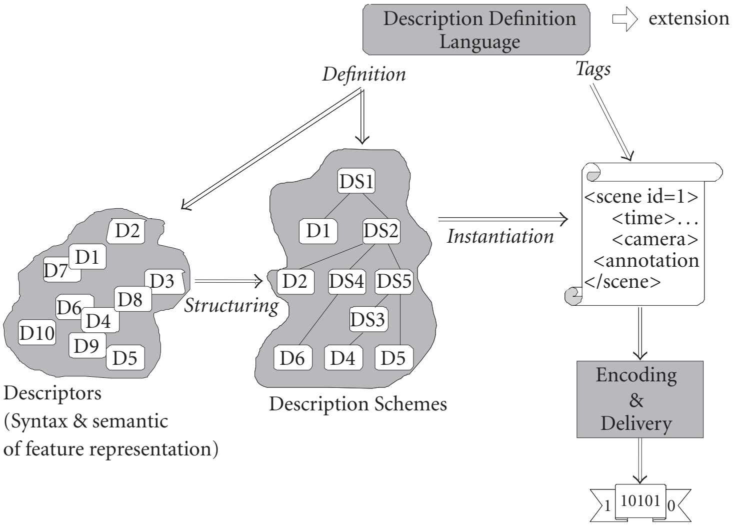


Figure 2: Main components of the MPEG-7 standard.

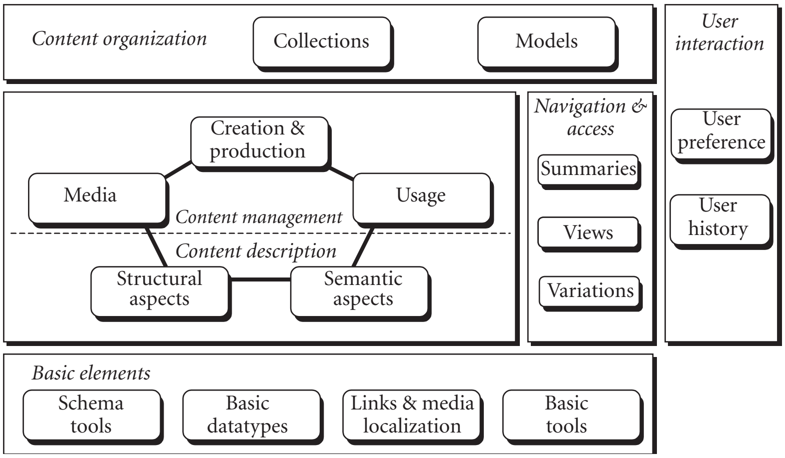


Figure 3: Overview of MPEG-7 multimedia DSs.

resilient with respect to transmission errors. To solve these problems, MPEG-7 defines a binary format *(BiM: Binary format for Mpeg-7)* and the corresponding encoding and decoding tools. This second format is particularly efficient in terms of compression and streaming functionality. Note that XML and BiM representations are equivalent and can be encoded and decoded losslessly.

The objective of this paper is to provide an overview of the MPEG-7 DSs and Descriptors focusing on the visual aspects (Section 2) and then, to discuss a set of visual information analysis challenges that could be studied to lead to very efficient MPEG-7-based applications (Section 3).

2. OVERVIEW OF MPEG-7

2.1. Multimedia Description Schemes

Figure 3 provides an overview of the organization of the multimedia DSs into different functional areas: basic elements, content management, content description, navigation and access, content organization, and user interaction. The MPEG-7 DSs can be considered as a library of description tools and, in practice, an application should select an appropriate subset of relevant DSs. This section discusses each of the different functional areas of multimedia DSs.

2.1.1 Basic elements

The first set of DSs can be seen on the lower part of Figure 3. They are called *Basic elements* because they provide elementary description functions and are intended to be used as building blocks for descriptions or DSs.

MPEG-7 provides a number of schema tools that assist in the formation, packaging, and annotation of MPEG-7 descriptions. An MPEG-7 description begins with a root element that signifies whether the description is complete or partial. A complete description provides a complete, standalone description of AV content for an application. On the other hand, a description unit carries only partial or incremental information that possibly adds to an existing description. In the case of a complete description, an MPEG-7 top-level element follows the root element. The top-level element orients the description around a specific description task, such as the description of a particular type of AV content (for instance an image, video, audio, or multimedia), or a particular function related to content management, (such as creation, usage, summarization, and so forth). The top-level elements collect together the appropriate tools for carrying out the specific description task.

In the case of description units, the root element can be followed by an instance of an arbitrary MPEG-7 DS or descriptor. Unlike a complete description which usually contains a "semantically-complete" MPEG-7 description, a description unit can be used to send a partial description as required by an application such as a description of a place, a shape, and texture descriptor and so on. It is also used to define elementary piece of information to be transported or streamed in case the complete description is too large.

Beside the schema tools, a number of basic elements are used as fundamental constructs in defining the MPEG-7 DSs. The basic data types provide a set of extended data types and mathematical structures such as vectors and matrices, which are needed by the DSs for describing AV content. The basic elements include also constructs for linking media files, localizing pieces of content, and describing time, places, persons, individuals, groups, organizations, textual annotation (including free text, structured annotation or annotation with syntactic dependency, etc.), classification schemes and controlled terms.

2.1.2 Content management

MPEG-7 provides DSs for AV content management. They describe information that generally cannot be perceived in the content itself but that is of vital importance for many applications. These tools describe the following information: (1) creation and production, (2) media coding, storage and file formats, and (3) content usage.

The creation information provides a title (which may itself be textual or another piece of AV content), and information such as creators, creation locations, and dates. It also includes classification information describing how the AV material may be categorized into genre, subject, purpose, language, and so forth. Finally, review and guidance information such as age classification, parental guidance, and subjective review are also given.

The media information describes the storage media including the format, the compression, and the coding of the AV content. The media information identifies the master media, which is the original source from which different instances of the AV content are produced. The instances of the AV content are referred to as media profiles, which are versions of the master obtained by using different encodings, or storage and delivery formats. Each media profile is described individually in terms of the encoding parameters, storage media information and location.

The usage information describes usage rights, usage record, and financial information. The rights information is not explicitly included in the MPEG-7 description, instead, links are provided to the rights holders and to other information related to rights management and protection.

2.1.3 Content description: structural aspects

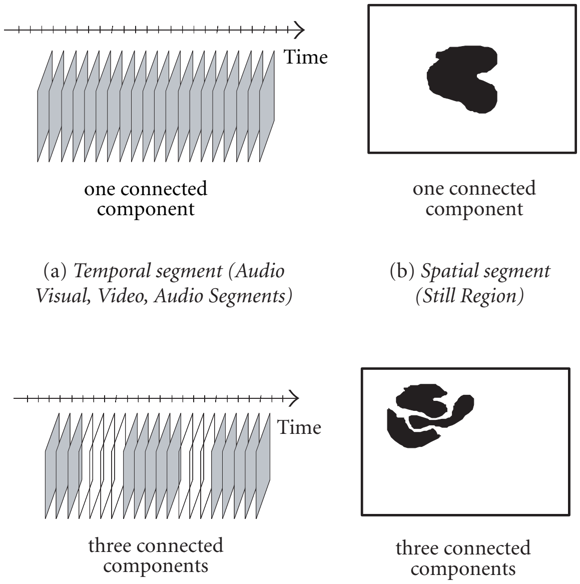
In MPEG-7, content description refers to information that can be perceived in the content. Two different view points are provided: the first one emphasizes the structural aspect of the signal whereas the second one focuses on the conceptual aspects of the content. This section presents the structural aspects with some details. Conceptual aspects will be briefly discussed in Section 2.1.4.

The description of the structure of the AV content relies on the notion of segments. The segment DS describes the result of a spatial, temporal, or spatio-temporal partitioning of the AV content. It can describe a hierarchical decomposition resulting in a segment tree. Moreover, the segment relation DS describes additional relationships among segments and allows the creation of graphs.

The segment DS forms the base type of the different specialized segment types such as audio segments, video segments, audio-visual segments, moving regions, and still regions. As a result, a segment may have spatial and/or temporal properties. For example, the audio segment DS describes a temporal interval of an audio sequence. The video segment DS describes a set of video frames. The audio visual Segment DS describes a combination of audio and visual information such as a video with synchronized audio. The still region DS describes a region of an image or a frame in a video. Finally, the moving region DS describes a moving region of a video sequence.

There exists also a set of specialized segments for specific type of AV content. For example, the Mosaic DS is a specialized type of StillRegion. It describes a mosaic or panoramic view of a video segment [3]. The VideoText is a subclass of the MovingRegion DS and describes a region of video content corresponding to text or captions. This includes superimposed text as well as text appearing in scene. Another example of specialized DS is the InkSegment DS which describes a segment of an electronic ink document created by a pen-based system or an electronic white-board.

The Segment DS contains elements and attributes that are common to the different segment types. Among the common properties of segments is information related to creation, usage, media location, and text annotation. The Segment DS can be used to describe segments that are not necessarily connected, but composed of several non-connected components. Connectivity refers here to both spatial and temporal domains. A temporal segment (VideoSegment, AudioSegment, and AudioVisualSegment) is said to be temporally connected, if it is a sequence of continuous video frames or audio samples. A spatial segment (StillRegion) is said spatially connected if it is a group of connected pixels. A spatio-temporal segment (MovingRegion) is said spatially and temporally connected if the temporal segment where it is instantiated is temporally connected and if each one of its temporal



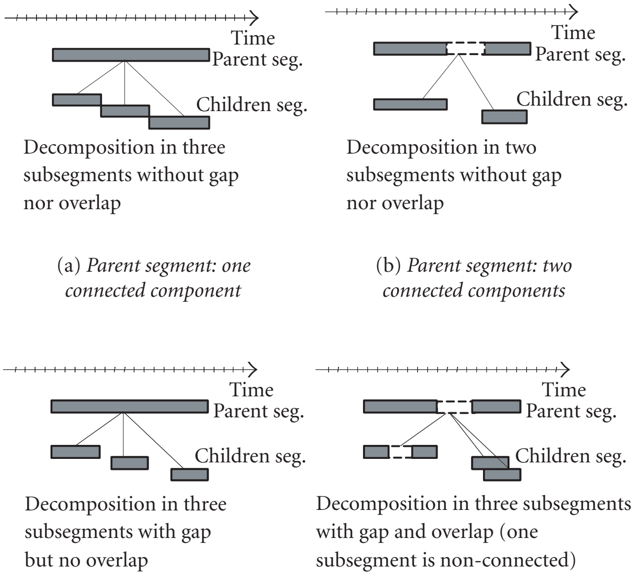
(a) *Temporal segment (Audio Visual, Video, Audio Segments)*

(b) *Spatial segment (Still Region)*

(c) *Temporal segment (Audio Visual, Video, Audio Segments)*

(d) *Spatial segment (Still Region)*

FIGURE 4: Examples of segments: (a) and (b) segments composed of one single connected component; (c) and (d) segments composed of three connected components.


Time
Parent seg.
Children seg.
Decomposition in three subsegments without gap nor overlap
Time
Parent seg.
Children seg.
Decomposition in two subsegments without gap nor overlap
Time
Parent seg.
Children seg.
Decomposition in three subsegments with gap but no overlap
Time
Parent seg.
Children seg.
Decomposition in three subsegments with gap and overlap (one subsegment is non-connected)

(a) *Parent segment: one connected component*

(b) *Parent segment: two connected components*

(c) *Parent segment: one connected component*

(d) *Parent segment: two connected components*

FIGURE 5: Examples of segment decomposition: (a) and (b) segment decompositions without gap nor overlap; (c) and (d) segment decompositions with gap or overlap.

instantiations in frames is spatially connected. (Note that this is not the classical connectivity in a 3D space.)

Figure 4 illustrates several examples of temporal or spatial segments and their connectivity. Figures 4a and 4b illustrate a temporal and a spatial segment composed of a single connected component. Figures 4c and 4d illustrate a temporal and a spatial segment composed of three connected components. Note that, in all cases, the descriptors and DSs attached to the segment are global to the union of the connected components building the segment. At this level, it is not possible to describe individually the connected components of the segment. If connected components have to be described individually, then the segment has to be decomposed into various subsegments corresponding to its individual connected components.

The segment DS may be subdivided into subsegments, and thus may form a hierarchy (tree). The resulting segment tree is used to describe the media source, the temporal and/or spatial structure of the AV content. For example, a video program may be temporally segmented into various levels of scenes, shots, and micro-segments. A table of contents may thus be generated based on this structure. Similar strategies can be used for spatial and spatio-temporal segments.

A segment may also be decomposed into various media sources such as various audio tracks or viewpoints from several cameras. The hierarchical decomposition is useful to design efficient search strategies (global search to local search). It also allows the description to be scalable: a segment may be described by its direct set of descriptors and DSs, but it may also be described by the union of the descriptors and DSs that are related to its subsegments. Note that a segment may be subdivided into subsegments of different types, for example, a video segment may be decomposed in moving regions that are themselves decomposed in still regions.

The decomposition is described by a set of attributes defining the type of subdivision: temporal, spatial, spatio-temporal, or media source. Moreover, the spatial and temporal subdivisions may leave gaps and overlaps between the subsegments. Several examples of decompositions are described for temporal segments in Figure 5. Figures 5a and 5b describe two examples of decompositions without gaps nor overlaps (partition in the mathematical sense). In both cases the union of the children corresponds exactly to the temporal extension of the parent, even if the parent is itself nonconnected (see the example of Figure 5b). Figure 5c shows an example of decomposition with gaps but no overlaps. Finally, Figure 5d illustrates a more complex case where the parent is composed of two connected components and its decomposition creates three children: the first one is itself composed of two connected components, the two remaining children are composed of a single connected component. The decomposition allows gap and overlap. Note that, in any case, the decomposition implies that the union of the spatio-temporal space defined by the children segments is included in the spatio-temporal space defined by their ancestor segment (children are contained in their ancestors).

As described above, any segment may be described by creation information, usage information, media informa-

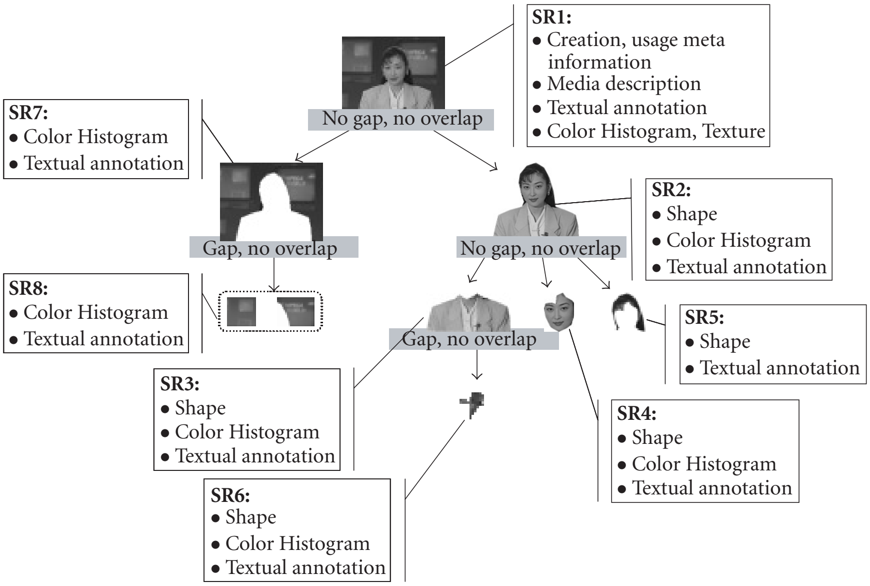


Figure 6: Examples of image description with Still Regions.

tion, and textual annotation. However, specific low-level features depending on the segment type are also allowed. An example of image description is illustrated in Figure 6. The original image is described as a still region, SR_1, which is described by creation (title, creator), usage information (copyright), media information (file format) as well as a textual annotation (summarizing the image content), a color histogram and a texture descriptor. This initial region can be further decomposed into individual regions. For each decomposition step, we indicate if gaps and overlaps are present. The segment tree is composed of eight still regions (note that SR_8 is a single segment made of two connected components). For each region, Figure 6 shows the type of feature, that is, instantiated. Note that it is not necessary to repeat in the tree hierarchy the creation, usage information, and media information, since the child segments are assumed to inherit their parent value (unless re-instantiated).

The description of the content structure is not constrained to rely on trees. Although, hierarchical structures such as trees are adequate for efficient access, retrieval and scalable description, they imply constraints that may make them inappropriate for certain applications. In such cases, the segment relation DS has to be used. The graph structure is defined very simply by a set of nodes, each corresponding to a segment, and a set of edges, each corresponding to a relationship between two nodes. To illustrate the use of graphs, consider the example shown in Figure 7.

This example shows an excerpt from a soccer match. Two video segments, one still region and three moving regions are considered. A possible graph describing the structure of the content is shown in Figure 7. The Video Segment: dribble & kick involves the ball, the goalkeeper, and the player. The ball remains close to the player who is moving toward the goalkeeper. The player appears on the right of the goalkeeper. The goal score video segment involves the same moving regions plus the still region called goal. In this part of the sequence, the player is on the left of the goalkeeper and the ball moves toward the goal. This very simple example illustrates the flexibility of this kind of representation. Note that this description is mainly structural because the relations specified in the graph edges are purely physical and the nodes represent segments (still and moving regions in this example). The only explicit semantic information is available from the textual annotation (where keywords such as ball, player, or goalkeeper can be specified).

2.1.4 Content description: conceptual aspects

For some applications, the viewpoint described in the previous section is not appropriate because it highlights the structural aspects of the content. For applications where the structure is of no real use, but where the user is mainly interested in the semantic of the content, an alternative approach is provided by the semantic DS. In this approach, the emphasis is not on segments but on events, objects in narrative worlds, concepts and abstractions. As shown in Figure 8, the semantic base DS describes narrative worlds and semantic entities in a narrative world. In addition, a number of specialized DSs are derived from the generic semantic base DS, which describe specific types of semantic entities, such as narrative worlds, objects, agent objects, events, places, time, and abstractions.

As in the case of the segment DS, the conceptual aspects of description can be organized in a tree or in a graph. The graph structure is defined by a set of nodes, representing semantic notions, and a set of edges specifying the relationship between the nodes. Edges are described by the semantic relation DSs.

Finally, as an example of combination of structural and conceptual aspects, Figure 9 illustrates the description of a video sequence inspired from the classical way of describing

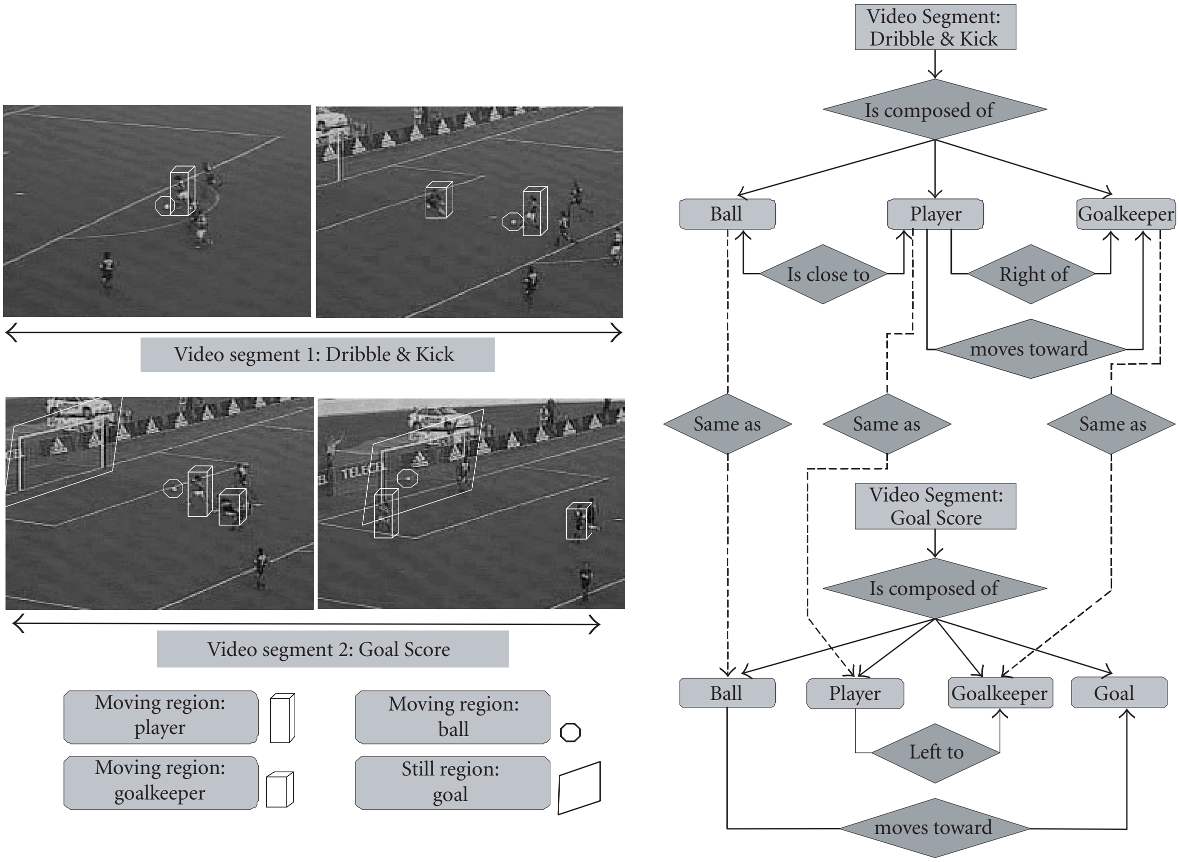


Figure 7: Examples of segment graph.

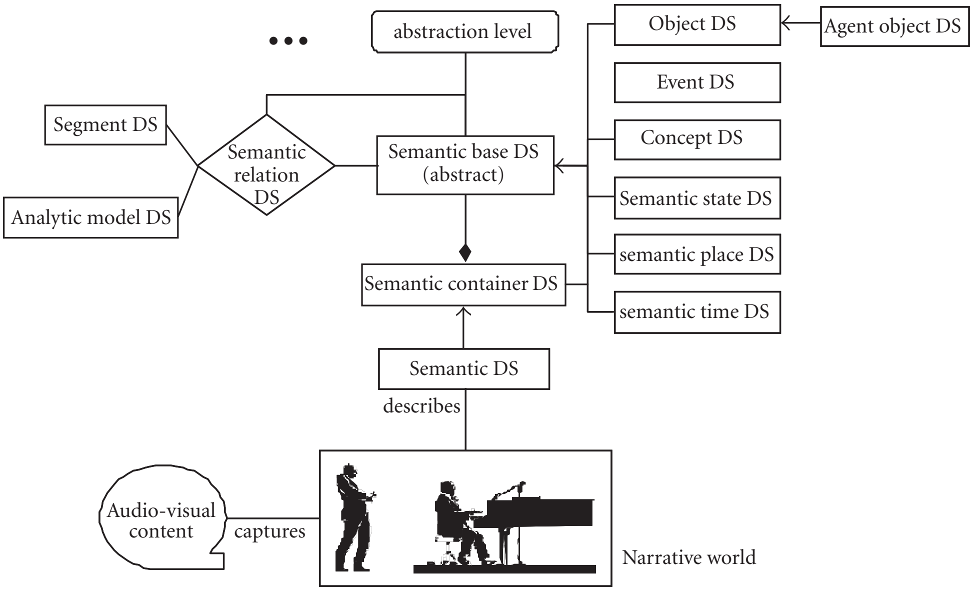


Figure 8: Tools for the description of conceptual aspects.

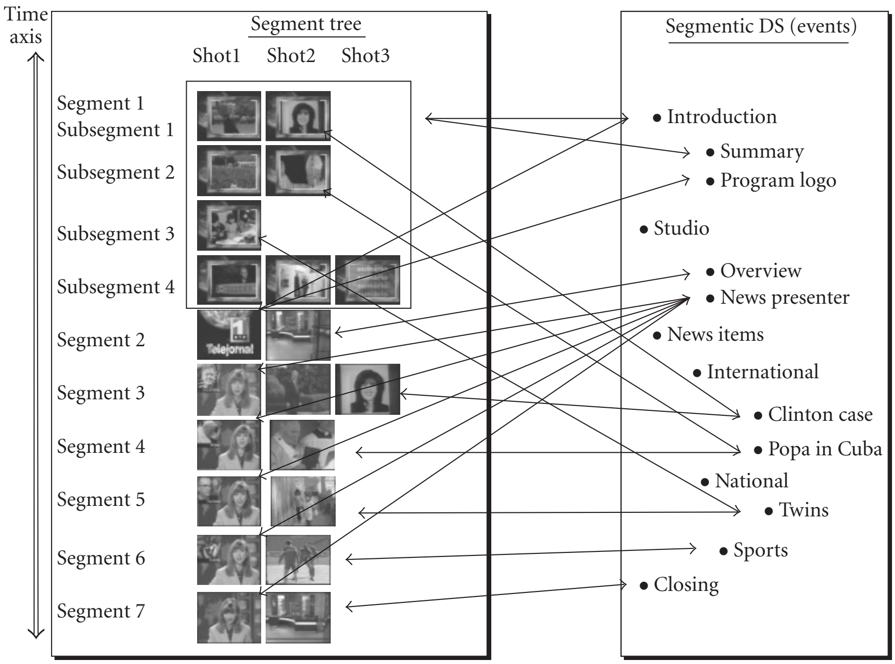


Figure 9: Example of table of content and index combining structural and conceptual aspects.

the content of written documents such as books: the table of contents and the index [4]. The table of contents is a hierarchical representation that splits the document into elementary pieces (chapters, sections, subsections, etc.). The order in which the items are presented follows the linear structure of the book itself. As a result, the table of contents is a representation of the linear, one-dimensional structure of the book. The goal of the index is not to define the linear structure of the book, but to define a set of potentially interesting items and to provide references to the book sections where these items are discussed. In order to be of practical interest to human users, the items are selected based on their semantic value. In many cases, the Index is also presented in a hierarchical fashion to allow fast access to the item of interest for the user.

Figure 9 shows an example of a video description. The description also involves two hierarchical structures represented by trees. The first one is devoted to the structural aspects and is based on a segment tree whereas the second one describes what is happening, that is, the conceptual aspects, and is termed the event tree. The links from the Event Tree to the Segment Tree relate semantic notions (events) with one or several occurrences of these notions in time. As a result the description is itself a graph built around two trees.

2.1.5 Navigation and access

MPEG-7 facilitates navigation and access of AV content by describing summaries, views, and variations. The summary DS describes semantically meaningful summaries and abstracts of AV content. The summary descriptions allow the AV content to be navigated in either a hierarchical or sequential fashion. The hierarchical summary DS describes the organization of summaries into multiple levels of detail. The main navigation mode is from coarse to fine and vice-versa. Note that the hierarchy may be based on the quantity of information (e.g., a few key-frames for a coarse representation versus a large number of key-frames for a fine representation) or on specific features (e.g., only the most important events are highlighted in the coarse representation whereas a large number of less important events may be shown in the fine representation).

The sequential summary DS describes a summary consisting of a sequence of images or video frames, which is possibly synchronized with audio and text. The sequential summary may also contain a sequence of audio clips. The main navigation mode is linear (forward-backward).

The view DS describes structural views of the AV signals in the space or frequency domain in order to enable multiresolution access and progressive retrieval.

Finally, the Variation DS describes relationships between different variations of AV programs. The variations of the AV content include compressed or low-resolution versions, summaries, different languages, and different modalities, such as audio, video, image, text, and so forth. One of the targeted functionalities is to allow a server or proxy to select the most suitable variation of the AV content for delivery according to the capabilities of terminal devices, network conditions, or user preferences.

2.1.6 Content organization

The content organization is built around two main DSs: the collection DS and the model DS. The collection DS includes tools for describing collections of AV material, collections of AV content descriptions, collections of semantic concepts, mixed collections (content, descriptions, and concepts) and collection structures in terms of the relationships among collections.

The model DS describes parametrized models of AV content, descriptors, or collections. It involves two important DSs: the probability model and the analytic model DSs. The probability model DS describes different statistical functions and probabilistic structures, which can be used to describe samples of AV content and classes of Descriptors using statistical approximation. The analytic model DS describes a collection of examples of AV content or clusters of descriptors that are used to provide a model for a particular semantic class. For example, a collection of art images labeled with tag indicating that the paintings are examples of the impressionist period forms an analytic model. The analytic model DS also optionally describes the confidence in which the semantic labels are assigned.

2.1.7 User Interaction

The user interaction DS describes preferences of users pertaining to the consumption of the AV content, as well as usage history. The MPEG-7 AV content descriptions can be matched to the preference descriptions in order to select and personalize AV content for more efficient and effective access, presentation and consumption. The user preference DS describes preferences for different types of content and modes of browsing, including context dependency in terms of time and place. The usage history DS describes the history of actions carried out by a user of a multimedia system. The usage history descriptions can be exchanged between consumers, their agents, content providers, and devices, and may in turn be used to determine the user's preferences with regard to AV content.

2.2. Visual features

The low-level visual features described in MPEG-7 are color, texture, shape and localization, motion, and low-level face characterization. With respect to the Multimedia DSs described in Section 2.1, the descriptors or DSs that handle low-level visual features are to be considered as a characterization of segments. Not all descriptors and DSs are appropriate for all segments and the set of allowable descriptors or DSs for each segment type is defined by the standard. This section summarizes the most important description tools dealing with low-level visual features.

2.2.1 Color feature

MPEG-7 has standardized eight color descriptors: color space, color quantization, dominant colors, scalable color histogram, color structure, color layout, and GoF/GoP color. The first two descriptors, color space, and quantization, are intended to be used in conjunction with other color descriptors. Possible color spaces include $\{R, G, B\}$, $\{Y, C_r, C_b\}$, $\{H, S, V\}$, monochrome and any linear combination of $\{R, G, B\}$. The color quantization supports linear and nonlinear quantizers as well as lookup-tables.

The dominant color descriptor is suitable for representing local features where a small number of colors are enough to characterize the color information in the region of interest. It can also be used for whole images. The descriptor defines the set of dominant colors, the percentage of each color in the region of interest and, optionally, the spatial coherence. This descriptor is mainly used in retrieval by similarity.

The scalable color histogram descriptor represents a color histogram in the $\{H, S, V\}$ color space. The histogram is encoded with a Haar transform to provide scalability in terms of bin numbers and accuracy. It is particularly attractive for image-to-image matching and color-based retrieval.

The color structure descriptor captures both color content and its structure. Its main functionality is image-to-image matching. The extraction method essentially computes the relative frequency of 8 × 8 windows that contain a particular color. Therefore, unlike a color histogram, this descriptor can distinguish between two images in which a given color is present with the same probability but where the structures of the corresponding pixels are different.

The color layout descriptor specifies the spatial distribution of colors for high-speed retrieval and browsing. It targets not only image-to-image matching and video-clip-to-video-clip matching, but also layout-based retrieval for color, such as sketch-to-image matching which is not supported by other color descriptors. The descriptor represents the DCT values of an image or a region that has been previously partitioned into 8 × 8 blocks and where each block is represented by its dominant color.

The last color descriptor is the GroupofFrames/GroupofPicturesColor descriptor. It extends the ScalableColorHistogram Descriptor defined for still images to video sequences or collection of still images. The extension describes how the individual histograms computed for each image have been combined: by average, median, or intersection. It has been shown that this information allows the matching between video segments to be more accurate.

2.2.2 Texture feature

There are three texture descriptors: homogeneous texture, texture browsing, and edge histogram. Homogeneous texture has emerged as an important visual primitive for searching and browsing through large collections of similar looking patterns. The homogeneous texture descriptor provides a quantitative representation. The extraction relies on a frequencial decomposition with a filter bank based on Gabor functions. The frequency bands are defined by a scale parameter and an orientation parameter. The first and second moments of the energy in the frequency bands are then used as the components of the descriptor. The number of filters used is $5 \times 6 = 30$ where 5 is the number of scales and 6 is the number of orientations used in the Gabor decomposition.

The texture browsing descriptor provides a qualitative representation of the texture similar to a human characterization, in terms of dominant direction, regularity, and coarseness. It is useful for texture-based browsing applications. The descriptor represents one or two dominant directions and, for each dominant direction, the regularity (four possible levels) and the coarseness (four possible values) of the texture.

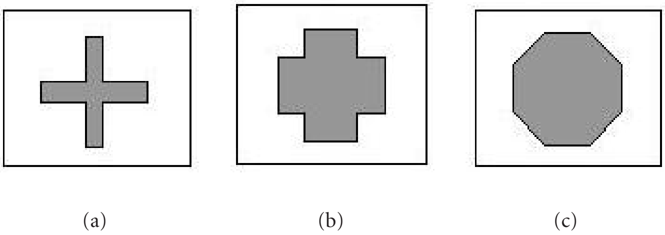

(a) (b) (c)

Figure 10: Illustration of region and contour similarity.

The edge histogram descriptor represents the histogram of five possible types of edges, namely four directional edges and one nondirectional edge. The descriptor primarily targets image-to-image matching (query by example or by sketch), especially for natural images with nonuniform edge distribution.

2.2.3 Shape and localization features

There are five shape or localization descriptors: region-based shape, contour-based shape, region locator, spatio-temporal locator, and 3D shape.

The region-based and contour-based shape descriptors are intended for shape matching. They do not provide enough information to reconstruct the shape nor to define its position in the image. Two shape descriptors have been defined because, in terms of applications, there are at least two major interpretations of shape similarity. For example, the shapes represented in Figures 10a and 10b are similar because they correspond to a cross. The similarity is based on the contours of the shape and, in particular, on the presence of points of high curvature along the contours. This type of similarity is handled by the contour-based shape descriptor. Shapes illustrated in Figures 10b and 10c can also be considered as similar. However, the similarity does not rely on the contours but on the distribution of pixels belonging to the region. This second similarity notion is represented by the region-based shape descriptor.

The contour-based shape descriptor captures characteristics of a shape based on its contour. It relies on the so-called curvature scale-space [5] representation, which captures perceptually meaningful features of the shape. The Descriptor essentially represents the points of high curvature along the contour (position of the point and value of the curvature). This representation has a number of important properties, namely: it captures characteristic features of the shape, enabling efficient similarity-based retrieval. It is robust to nonrigid deformation and partial occlusion.

The contour-based shape descriptor captures the distribution of all pixels within a region. Note that, in contrast with the contour-based shape descriptor, this descriptor can deal with regions made of several connected components or including holes. The Descriptor is based on an *Angular Radial Transform (ART)* which is a 2D complex transform defined with polar coordinates on the unit disk. The ART basis functions are separable along the angular and radial dimensions. Twelve angular and three radial basis functions are used. The descriptor represents the set of coefficients resulting from the projection of the binary region into the 36 ART basis functions.

The region locator and the spatio-temporal locator combine shape and localization information. Although they may be less efficient in terms of matching for certain applications, they allow the shape to be (partially) reconstructed and positioned in the image. The region locator descriptor represents the region with a compact and scalable representation of a polygon. The spatio-temporal locator has the same functionality but describes moving regions in a video sequence. The descriptor specifies the shape of a region within one frame together with its temporal evolution based on motion.

3D shape information can also be described in MPEG-7. Most of the time, 3D information is represented by polygonal meshes. The 3D shape descriptor provides an intrinsic shape description of 3D mesh models. It exploits some local attributes of the 3D surface. The descriptor represents the 3D mesh shape spectrum, which is the histogram of the shape indexes [6] calculated over the entire mesh. The main applications targeted by this descriptor are search, retrieval and browsing of 3D model databases.

2.2.4 Motion feature

There are four motion descriptors: camera motion, object motion trajectory, parametric object motion, and motion activity. The camera motion descriptor characterizes 3D camera motion parameters. It supports the following basic camera operations: fixed, tracking (horizontal transverse movement, also called traveling in the film industry), booming (vertical transverse movement), dollying (translation along the optical axis), panning (horizontal rotation), tilting (vertical rotation), rolling (rotation around the optical axis), and zooming (change of the focal length), the descriptor is based on time intervals characterized by their start time, and duration, the type(s) of camera motion during the interval, and the focus-of-expansion (FoE) (or focus-of-contraction FoC). The descriptor can describe a mixture of different camera motion types. The mixture mode captures globally information about the camera motion parameters, disregarding detailed temporal information.

The motion trajectory descriptor characterizes the temporal evolution of key-points. It is composed of a list of keypoints (x, y, z, t) along with a set of optional interpolating functions that describe the trajectory between key-points. The speed is implicitly known by the key-points specification and the acceleration between two key-points can be estimated if a second order interpolating function is used. The key-points are specified by their time instant and their 2D or 3D Cartesian coordinates, depending on the intended application. The interpolating functions are defined for each component $x(t)$, $y(t)$, and $z(t)$ independently. The description is independent of the spatio-temporal resolution of the content (e.g., 24 Hz, 30 Hz, 50 Hz, CIF, SIF, SD, HD, etc.). The granularity of the descriptor is chosen through the number of key-points used for each time interval.

Parametric motion models have been extensively used within various image processing and analysis applications. The parametric motion descriptor defines the motion of regions in video sequences as a 2D parametric model. Specifically, affine models include translations, rotations, scaling and combination of them. Planar perspective models make possible to take into account global deformations associated with perspective projections. Finally, quadratic models makes it possible to describe more complex movements. The parametric model is associated with arbitrary regions over a specified time interval. The motion is captured in a compact manner as a reduced set of parameters.

A human watching a video or animation sequence perceives it as being a "slow" sequence, a "fast paced" sequence, an "action" sequence, and so forth. The motion activity descriptor captures this intuitive notion of "intensity of action" or "pace of action" in a video segment. Examples of high activity include scenes such as "scoring in a basketball game," "a high speed car chase," and so forth. On the other hand, scenes such as "news reader shot" or "an interview scene" are perceived as low action shots. The motion activity descriptor is based on five main features: the intensity of the motion activity (value between 1 and 5), the direction of the activity (optional), the spatial localization, the spatial and the temporal distribution of the activity.

2.2.5 Face descriptor

The face recognition descriptor can be used to retrieve face images that match a query face image. The descriptor is based on the classical eigen faces approach [7]. It represents the projection of a face region onto a set of basis vectors (49 vectors) which span the space of possible face vectors.

3. CHALLENGES FOR VISUAL INFORMATION ANALYSIS

As mentioned in the introduction, the scope of the MPEG-7 standard is to define the syntax and semantics of the DSs and Descriptors. The description generation and consumption are out of the scope of the standard. In practice, this means that feature extraction, indexing process, annotation, and authoring tools as well as search and retrieval engines, filtering and browsing devices are non-normative parts of the standard and can lead to future improvements. It has to be mentioned however, that, for low-level features, the distinction between the definition of the semantics of a tool and its extraction may become fuzzy. A typical example is represented by the homogeneous texture descriptor (see Section 2.2.2). In order to support interoperability, MPEG-7 has defined the set filters to be used in the decomposition (Gabor filters and their parameters). Beside the implementation, this leaves little room for future studies and improvements. A similar situation can be found for most visual descriptors described in Section 2.2: the definition of their semantics defines partially the extraction process. The main exceptions are the texture browsing and the motion activity descriptors. Indeed, the characterization of the "texture regularity" or of the "motion intensity" is qualitatively done. The camera motion descriptor is a special case, because either one has access to the real parameters of the camera or one has to estimate the camera motion from the observed sequence.

The definition of a low-level descriptor may also lead to the use of a natural matching distance. However, the standardization of matching distances is not considered as being necessary to support interoperability and the standard only provides informative sections in this area. This will certainly be a challenging area in the future.

Most of the descriptors corresponding to low-level features can be extracted automatically from the original content. Most of the time, the main issue is to define the temporal interval or the region of interest that has to be characterized by the descriptor. This is a classical segmentation problem for which, a large number of tools have been reported in the literature (see [8, 9, 10] and the references herein). An area which has been less worked out is the instantiation of the decomposition involved in the segment DS. It can be viewed as a hierarchical segmentation problem where elementary entities (region, video segment, etc.) have to be defined and structured by inclusion relationship within a tree. This process leads, for example, to the extraction of *Tables of Contents* or *Indexes* from the AV content as illustrated in Figure 9. Although some preliminary results have been reported in the literature, this area still represents a challenge for the future.

One of the most challenging aspects of the MPEG-7 standard in terms of application is to use it efficiently. The selection of the optimum set of DSs and descriptors for a given application is an open issue. Even if the identification of the basic features that have to be represented is a simple task, the selection of specific descriptors may not be straightforward: for example, dominant color versus scalable color histogram or motion trajectory versus parametric motion, and so forth. Moreover, the real power of the standard will be obtained when DSs and Descriptors are jointly used and when the entire description is considered as a whole, for example, taking into account the various relationships between segments in trees or graphs.

In terms of research, one of the most challenging issues may be the mapping between low-level and high-level descriptions. First we discuss the relation between low-level, high-level descriptions and recognition processes. Consider the two situations represented in Figure 11: on the top, the description is assumed to rely mainly on high-level features. This implies that the automatic or manual indexing process has performed a recognition step during description genera-

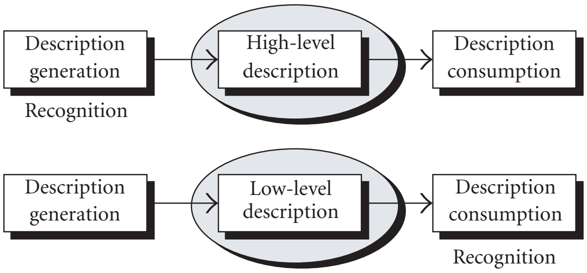


Figure 11: Localization of the recognition process depending on the feature types.

tion. This approach is very powerful but not very flexible. Indeed, if during the description generation, the high-level feature of interest for the end user has been identified, then the matching and retrieval will be very easy to do. However, if the end user wants to use a feature that has not been recognized during the indexing phase, then it is extremely difficult to do anything. The alternative solution is represented in the lower part of Figure 11. In this case, we assume that the description relies mainly on low-level features. No recognition process is required during the description generation. However, for many applications, the mapping between low-level descriptions and high-level queries will have to be done during the description consumption. That is, the search engine or the filtering device will have to analyze the low-level features and, on this basis, perform the recognition process. This is a very challenging task for visual analysis research. Today, the technology related to intelligent search and filtering engines using low-level visual features, possibly together with high-level features, is still very limited. As a final remark, we mention that this challenging issue has also some implications for the description generation. Indeed, a major open question is to know what are the useful set of low-level descriptors that have to be used to allow a certain class of recognition tasks to be performed on the description itself.

REFERENCES

[1] D. C. Fallside, Ed., *XML Schema Part 0: Primer. W3C Recommendation*, May 2001, http://www.w3.org/TR/xmlschema-0/.

[2] T. Bray, J. Paoli, C. M. Sperberg-McQueen, and E. Maler, Eds., *XML: Extensible Markup Language 1.0*, 2nd edition, October 2000, http://www.w3.org/TR/REC-xml.

[3] H. Sawhney and S. Ayer, "Compact representations of videos through dominant and multiple motion estimation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 814–830, 1996.

[4] Y. Rui, T. S. Huang, and S. Mehrotra, "Exploring video structure beyond the shots," in *Proc. IEEE International Conference on Multimedia Computing and Systems*, pp. 237–240, 28 June–1 July 1998, Austin, Tex, USA.

[5] F. Mokhtarian and A. K. Mackworth, "A theory of multi-scale, curvature-based shape representation for planar curves," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 8, pp. 789–805, 1992.

[6] J. J. Koenderink and A. J. van Doorn, "Surface shape and curvature scales," *Image and Vision Computing*, vol. 10, no. 8, pp. 557–565, 1992.

[7] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696–710, 1997.

[8] B. S. Manjunath, T. Huang, A. M. Teklap, and H. J. Zhang, Eds., "Special issue on image and video processing for digital libraries," *IEEE Trans. Image Processing*, vol. 9, no. 1, 2000.

[9] K. N. Ngan, S. Panchanathan, T. Sikora, and M. T. Sun, Eds., "Special issue on segmentation, description and retrieval of video content," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 521–524, 1998.

[10] F. Pereira, S. F. Chang, R. Koenen, A. Puri, and O. Avaro, Eds., "Special issue on object-based video coding and description," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 9, no. 8, pp. 1144, 1999.

Philippe Salembier received a degree from the École Polytechnique, Paris, France, in 1983 and a degree from the École Nationale Supérieure des Télécommunications, Paris, France, in 1985. He received the PhD from the Swiss Federal Institute of Technology (EPFL) in 1991. He was a Postdoctoral Fellow at the Harvard Robotics Laboratory, Cambridge, MA, in 1991. From 1985 to 1989 he worked at Laboratoires d'Electronique Philips, Limeil-Brevannes, France, in the fields of digital communications and signal processing for HDTV. In 1989, he joined the Signal Processing laboratory of the Swiss Federal Institute of Technology in Lausanne, Switzerland, to work on image processing. At the end of 1991, after a stay at the Harvard Robotics Lab., he joined the Polytechnic University of Catalonia, Barcelona, Spain, where he is lecturing on the area of digital signal and image processing. His current research interests include image and sequence coding, compression and indexing, image modeling, segmentation problems, video sequence analysis, mathematical morphology, and nonlinear filtering. In terms of standardization activities, he has been involved in the definition of the MPEG-7 standard ("Multimedia Content Description Interface") as chair of the "Multimedia Description Scheme" group between 1999 and 2001. He served as an Area Editor of the Journal of Visual Communication and Image Representation (Academic Press) from 1995 until 1998 and as an AdCom officer of the European Association for Signal Processing (EURASIP) in charge of the edition of the Newsletter from 1994 until 1999. He has edited (as guest editor) special issues of Signal Processing on Mathematical Morphology (1994) and on Video sequence analysis (1998). He has also co-edited a special issue of Signal processing: Image Communication on MPEG-7 proposals (2000). Currently, he is associate editor of IEEE Transactions on Image Processing and Co-Editor-In-Chief of Signal Processing. Finally, he is member of the Image and Multidimensional Signal Processing Technical Committee of the IEEE Signal Processing Society.