Proceedings of SPIE Conference on Visual Communications and Image Processing, VCIP'02, vol. 4671, pp. 947-958, San Jose, CA, USA, Jan. 21-23, 2002.

MPEG-7 Description of Generic Video Objects for Scene Reconstruction

Olivier Steiger, Andrea Cavallaro and Touradj Ebrahimi

Signal Processing Laboratory, Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland

ABSTRACT

We present an MPEG–7 compliant description of generic video sequences aiming at their scalable transmission and reconstruction. The proposed method allows efficient and flexible video coding while keeping the advantages of textual descriptions in database applications. Visual objects are described in terms of their shape, color, texture and motion; these features can be extracted automatically and are sufficient in a wide range of applications. To permit partial sequence reconstruction, at least one simple qualitative as well as a quantitative descriptor is provided for each feature. In addition, we propose a structure for the organization of the descriptors into objects and scenes and some possible applications for our method. Experimental results obtained with news and video surveillance sequences validate our method and highlight its main features.

Keywords: MPEG–7, XML, content-based coding, descriptors, video objects, video reconstruction.

1. INTRODUCTION

For the past few years, new algorithms and standards such as H.261/263 and MPEG–1, 2 and 4 have been developed for the compression of digital visual data. But along with these storage solutions, the need for efficient multimedia database management has stimulated new technologies for search, indexing and content-based retrieval of images and video. To fulfill these goals, extensive research has been carried out into finding adequate descriptors for the data. The traditional approach of keyword annotation has the drawback that it requires huge manual efforts and cannot characterize the rich visual content efficiently; therefore, effective representation schemes had to be developed. High-level features such as content summaries, timing information or object names are usually represented in text form and often have to be edited manually. On the other hand, low-level features like shape, texture or color can be extracted (semi-)automatically and described based on histograms, filter bank coefficients,[1] polygon vertices[2] or other indirect representations as opposed to simple text.

To insure the interoperability and continuation of different data sets, the need for a uniform description framework for multimedia data arose. To meet this challenge, ISO's Moving Pictures Experts Group (MPEG) has undertaken the standardization activity for a "Multimedia Content Description Interface" called MPEG–7.[3] MPEG–7 is structured into *Descriptors* (D) and *Description Schemes* (DS), which are themselves made of other descriptors or description schemes. To take advantage of its flexibility and interoperability, XML has been used as the core language of MPEG–7. The features that have been standardized range from high-level down to low-level features of images and video as well as audio and text. System tools for the transmission and storage of the data and a *Description Definition Language* (DDL) permitting the creation of new D and DS are also supplied. On the other hand, data search or retrieval algorithms have not been standardized, still allowing interoperability without blocking further enhancements in these domains.

Since the time when the first MPEG–7 drafts were released, significant efforts have been deployed to study potential applications of the new standard. In the applicative part of their paper on object-based multimedia content description,[4] the authors propose diverse image, video and multimedia description schemes for image classification (*The Visual Apprentice*), video indexing and searching (*AMOS-search*) and multimedia search

E-mail addresses of the authors:
{olivier.steiger, andrea.cavallaro, touradj.ebrahimi}@epfl.ch

engines (*MetaSEEK*). The matching of individual user profiles with MPEG–7 content descriptions for digest video delivery has been studied by Echigo et al.,[5] and a low-level description scheme based on color, texture, shape and motion for image retrieval has been proposed[6] by Ohm et al. Many other documents about description schemes for video indexing and content-based retrieval, some of which stem from the standardization process, can be found.[7–10] But while the use of textual descriptors in databases has been widely studied, no work about reconstruction applications is available so far. Since MPEG–7 descriptors are human-readable, compact, interoperable and evolutionary, they permit very flexible video coding while being usable as they are in existing database applications. This is the topic of this paper.

In Section 2, we propose an original description method of generic video scenes for their reconstruction. Its central components are an MPEG–7 descriptor set for the shape, color, texture and motion of visual objects, and tools to organize the descriptors. Some features of our description method are then explored and possible applications are given. The third Section shows experimental results obtained when reconstructing video surveillance and news sequences. Chapter 4 concludes this work.

2. DESCRIPTION OF GENERIC VIDEO SEQUENCES

Current video coding techniques operate according to two main lines: first generation coding (e.g., MPEG–1 and 2) minimizes spatial and temporal redundancies by a combination of *pixel-based* techniques like transform coding, predictive coding, vector quantization or subband coding. But even though compression ratios of 200 can be achieved, the image is assumed to be made of pixels without any semantic meaning. *Object-based* techniques like those used in MPEG–4 on the other hand process semantic objects individually, adding compression efficiency and new editing possibilities to the former approach. But even though MPEG–4 is content-based, pixels remain the fundamental image reconstruction unit. To facilitate certain video application types while keeping the flexibility of semantic coding, we would like to replace the pixel-based approach with a textual scene description similar to those used in database applications. Since we do not want to loose the assets of descriptors for databases, we need to associate some high-level descriptors with the low-level features required for reconstruction. In this Section, we first present a descriptor set for visual objects and then describe how to group these descriptors into video scenes (Figure 1) and show their features. Large parts of this method are based on the sequence description we devised for an MPEG–7 camera.[11]

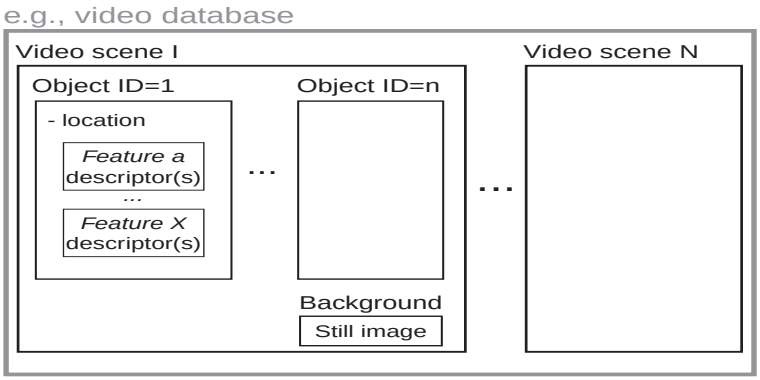


Figure 1: Framework for the encoding of video scenes.

2.1. Descriptor set for generic video objects

From the start, we decided to build our set with the XML-based MPEG–7 descriptors. Unlike customized descriptors, those derive benefit from the MPEG expertise and can be used within other compatible applications. The description is restricted to natural 2–D objects, thus descriptors for graphics, texts or 3–D objects are discarded but can be added easily if necessary. The most relevant low-level features that are directly related to physical properties and easily extractable from video signals are shape, color, texture and motion. An additional

high-level descriptor is needed to relate the objects to the general content (e.g., video file). In order to support *scalable* descriptions, that is, descriptions ranging from coarse to fine, we provide at least one qualitative and one quantitative descriptor for each low-level feature (table 1).

Feature	Descriptor	Purpose
Location	`Media Locator`	Relate descriptions to content
Shape	`Region locator`	Box or polygon shape
	`Contour shape`	Closed contour shape
Color	`Dominant color`	Set of up to 8 dominant colors
	`Color layout`	Spatial distribution of colors
Texture	`Texture browsing`	Perceptual texture description
	`Homogeneous texture`	Structural texture description
Motion	`Motion activity`	Perceptual motion descriptor
	`Motion trajectory`	Single-point motion
	`Parametric motion`	Describe moving regions

Table 1: MPEG-7 descriptor set for scene reconstruction.

We now explain the function of these descriptors and motivate our choices. The syntax of the descriptors is specified in the *Visual* (V)[12] and in the *Multimedia Description Schemes* (MDS)[13] texts of the MPEG–7 standard, and an overview of their structure is also given in appendix A. Non-normative extraction examples as well as the conditions of usage can be found in the corresponding experimentation models.[14, 15] Since MPEG–7 is an evolving standard, some minor changes may still be made, but they should not in any way affect the method described here.

2.1.1. Location

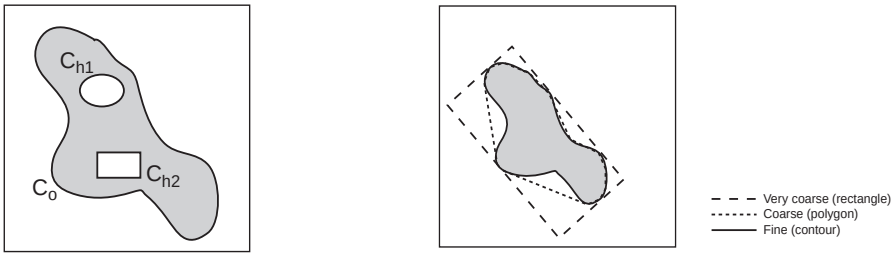
The `Media Locator` (MDS[13] §6.4) relates a description to content in general. The referencing from the description to the media can be done in two ways: `MediaUri` locates external content whereas `InlineMedia` includes the media content itself. In addition, different mechanisms are provided to locate media chunks within the content addressed by the basic MediaLocator type. In our application, solely `MediaUri` will be used. Many other high-level descriptors exist in MPEG–7, but non of them is essential to reconstruction.

2.1.2. Shape

`Region Locator` (V[12] §10.1) specifies regions within images with a brief and scalable representation of a box or polygon. `Contour Shape` (V §8.2) describes a closed contour of a 2D object in the Curvature Scale Space (CSS). Since holes are not supported, they have to be encoded as inner contours (Figure 2(a)), that is, each contour C which is enclosed in another contour C_o is considered a hole C_h of the object delimited by C_o. `Region Shape` (V §8.1) has been disregarded because it is more complex than the contour shape but does not add to the scalability of the reconstruction (Figure 2(b)).

2.1.3. Color

`Dominant Color` (V §6.3) specifies a set of up to 8 dominant colors in an arbitrarily-shaped region. `Color Layout` (V §6.5) specifies the spatial distribution of colors with the DCT coefficients of an 8x8 array of local dominant colors. `Scalable Color` (V §6.4) and `Color Structure` (V §6.6), which are both based on color histograms, are similar to dominant colors except that they specify every color of an object instead of just eight. When this is desirable, they can easily be used in place of dominant colors. The `GoF/GoP Color` descriptor (V §6.7) defines some average color for a group of frames; this is not useful here because we use *intra-frames* (see §2.2) to update description values.



(a) Encoding holes.

(b) Scalable shape representation.

Figure 2: Description of shapes.

2.1.4. Texture

`Texture browsing` (V §7.2) characterizes textures perceptually in terms of regularity, coarseness and direction. `Homogeneous texture` (V §7.1) specifies a region texture using its energy and energy deviation in a set of frequency channels. `Edge histograms` (V §7.3) are very efficient for natural image mapping based on texture, but since they are calculated for square sub-blocks, they fulfill the same function than the simpler homogeneous texture in our case.

2.1.5. Motion

`Motion activity` (V §9.4) captures the notion of "intensity of action" or "pace of action" in a video scene. `Motion trajectory` (V §9.2) is defined as the spatio-temporal localization of the representative point (e.g., gravity center) of a moving region. `Parametric motion` (V §9.3) characterizes the evolution of arbitrarily shaped regions over time in terms of a 2–D geometric transform (translation, rotation/scaling, affine transformation, perspective models or quadratic models). If a still image is used as static background, there is obviously no `Camera Motion` (V §9.1). However, when more complex background encoding schemes, for example using panoramic pictures, are preferred, camera motion should be added to the present set.

2.2. Organization of the description

Video reconstruction asks for descriptors of visual objects, but also for a structure reflecting the semantic structure of the scene. Thus we will next show how build a scalable *object* with our descriptors, and how to group such objects to form scenes.

2.2.1. Building an object

The `Object` DS is defined in §12.4.3 of the MDS[13] document. The standard permits the recursive usage of objects, but we recommend to group objects by linking them to the same content instead; this way, semantic units are kept separated, which facilitates video editing notably. Inside of an object, there has to be at least a media locator and one low-level feature, but more than one descriptor can be used simultaneously for each feature. This adds flexibility to the transmission and decoding process, because descriptors can then be sent in a **scalable** fashion, that is, compact qualitative descriptors (*base layer*) are sent first, quantitative descriptors (*enhancement layer*) later. The decoder can exploit the same property to display only parts of the description. Some descriptors are further **scalable**: the number of vertices of a `region locator` is variable, and so is the number of `dominant colors`. In a `color layout`, the number of DCT coefficients can be augmented. With `texture browsing` and `motion activity`, it is not necessary to transmit all the perceptual properties simultaneously, and the interpolation models in `motion trajectory` and `parametric motion` can be refined (e.g., first linear, then quadratic). Alone `contour shapes` are not scalable. To help setting up processing priorities, a **reliability** measure can be associated with each parameter; more reliable parameters are then sent and/or decoded first. When no such measure is given, the parameters are considered to be sorted by decreasing importance.

Since video is a process taking place in time, some features other than motion, which is anyway a chronological process, may have to be updated. To reflect this in the description, we use ***intra-frames***. Whenever the shape (only with perceptual or single-point motion), color or texture have to be updated, the corresponding descriptor with the new values is inserted right after the last valid motion field. The motion description then continues where it stopped. Static **backgrounds** can easily be encoded as still images, possibly using some compression algorithm (e.g., JPEG). Alternatively, a panoramic background shot combined with the `camera motion` descriptor may also be considered.

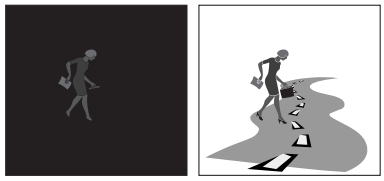
2.2.2. Objects in video scenes

Video scenes normally contain many objects, and diverse scenes may be stored in a video database. Thus objects have to be labelled and located uniquely. To specify the **location** of an object, a `media locator` must be present in its description. This descriptor specifies the Uniform Resource Identifier (URI) of the video file or stream the object belongs to. To **label** the objects, any name or number can be used as long as it is unique for a specific location. But when such data is available, we propose to include some information about past object interactions in the label. A way to do so is by adding a merging `_m` or a splitting `_s` flag after the objects name. Associated with this is an argument list `(L1...Ln, #t)` with Lx the labels of the interacting object and #t the moment of the interaction. This labelling scheme, used recursively, permits to know the full history of an object without reconstructing the other objects of the scene. For example, an object 1 merging with some previously split object 3 (from object 2 in frame 10) in frame 50 would be labelled `o1_m(o1,o3_s(o2,#10),#50)`. The drawback of this technique is the increasing object name length.

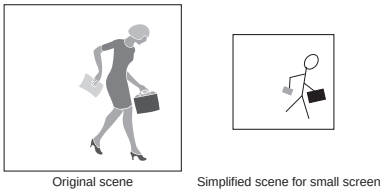
2.3. Description features

The description method we set out in the previous Section shows a number of attractive properties for diverse video applications. Some of them are inherent to the content-based coding approach, others rise directly from the textual description. In this Section, three central features, namely *video manipulation* (*scene visualization*, *image simplification*), *scene description* and *compactness* (Figure 3) are explained and illustrated using video surveillance and news server examples. Experimental results obtained with these scenarios are shown in Section 3.

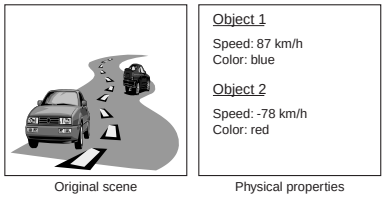
(a) Video Manipulation.



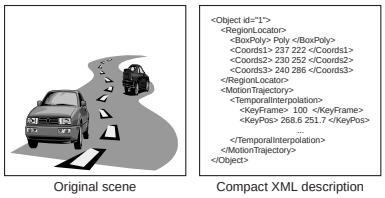
(b) Scene Visualization.



(c) Image Simplification.



(d) Scene Description.



(e) Compactness.

Figure 3: Description features.

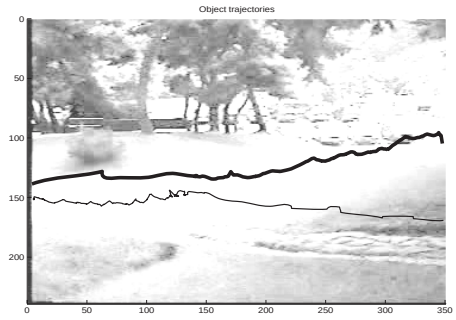
2.3.1. Video Manipulation

Video manipulation is the most fundamental feature of our description method, as its principles are inherent to any content-based video system. The traditional application for video manipulation is compositing, where scenes are constructed out of visual elements (objects). Any property of the objects, that is, their position, size, color, and so on, can be modified and that objects from different scenes can be inserted into the same scene. However, as compared to pixel-based techniques like MPEG–4, there is no need for any image processing here, only the description parameters have to be changed. ***Scene Visualization*** is a specific type of video manipulation with the aim of representing an information so as to make it easily understandable by a human observer. A common example stems from video surveillance: infrared cameras deliver night-shot images containing essential information. By substituting a daylight shot to the background and by replacing the grayscales of foreground objects (e.g., people) with natural colors, the image can be made more understandable. The purpose of ***Image Simplification*** is to adapt the image content to transmission or end-user devices by removing superfluous information in order to provide Universal Multimedia Access (UMA). As an example, we may cite a news sever that has to send the same story to high-definition TV's (HDTV) and cell phones. Since there is no space for the complete HDTV image on the phone screen, it could be simplified by displaying only object positions. Moreover, color information must not be sent if the phone has a black & white screen, thus saving bandwidth.

2.3.2. Scene Description

In XML-based descriptions, the feature values of the objects are textually available and can often be translated into a physical description of the original objects using simple calculations (e.g., perspective to get the speed). In video surveillance, this allows to detect objects entering a restricted area just by comparing their positions with the area coordinates. An example for this is shown in Figure 4: the original sequence (Figure 4(a)) shows two people walking respectively on and out of a path. In Figure 4(b), the gravity center trajectories for both people have been plot. To get this graph, no sequence reconstruction was necessary since the gravity centers of objects are encoded textually in the `motion trajectory` descriptor.

(a) Original video surveillance sequence (frames 10-110-160).



(b) Path of both people (background brightness reduced for readability).

Figure 4: Description of object positions in video surveillance.

2.3.3. Compactness

Compactness has a somewhat different meaning here than with traditional coding techniques. MPEG–1, 2 and 4 mainly remove redundant information from macroblocks or objects to get data compression. Textual description rather removes unimportant visual features. Here the goal consists in providing a compact representation of a scene as close as possible, from a semantic point of view, to the original. We will discuss this further in Section 3.

2.4. Possible applications

The proposed description notably facilitates certain applications which are complex when done with traditional coding approaches. Thanks to its scalability associated with some database overhead, our method can easily be used for **Universal Multimedia Access** (UMA). UMA refers to the delivery of rich multimedia content to diverse client devices over various channels. The classical "info-pyramid" approach to this problem is to store a data stream for each possible channel/receiver (*user*) and to transmit the adequate version on demand. However, it is more storage efficient and flexible to store only one version of the content, which is then adapted to the user "on-the-fly". With MPEG–7 encoded scenes, this can be achieved by selecting and transmitting the features or parameters that fit the user best. For example, the position of objects rather than their shape would be sent to end-user devices with small displays, while one would make sure to send the principal dominant color before transmitting a color layout over slow channels. To help filtering the adequate information out of the description, a *priority list* containing the priority of the parameters or descriptors for diverse users can be associated with the MPEG–7 content database (Figure 5).

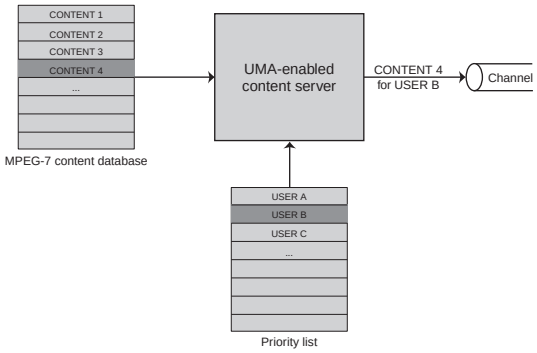


Figure 5: Using MPEG–7 descriptions for Universal Multimedia Access

Another application that has already been investigated[11, 17] but asks for further developments are MPEG–7 compliant **smart cameras**. These devices output a XML description of visual scenes without any user intervention. Surveillance tasks like people counting or intrusion detection can be automated by comparing specific description parameters, for example the number of objects and their positions, with threshold values. Since incidents are detected automatically, the anonymity of supervised people can be fully preserved. However, the performance of smart cameras depends entirely on the image processing possibilities (segmentation and object tracking), and no system is presently capable of describing complex scenes involving multiple interacting objects without errors. **Generic MPEG–7 codecs** for image storage and reconstruction would also benefit of progresses in semantic image processing. If they were available, content providers could archive a unique description for database browsing and reconstruction, instead of associating a separate video stream (e.g., MPEG–4) to the description.

3. EXPERIMENTAL RESULTS

In the previous Section, we have proposed a method for the description of generic video scenes using MPEG–7 descriptors and set out its features. To test the description, we use a *highway surveillance* and a *news* sequence from the MPEG–7 Content Set.[16] The *highway* sequence (MPEG–7 Content set item V29, 425 frames) has been XML-encoded by a method[11] which automatically detects moving objects in video sequences and describes them using MPEG–7 descriptors. The *news* sequence (Content Set Item V3, 28 frames) on the other hand had to be segmented manually in order to get semantically meaningful but immobile objects (hair, neck, body). In our testbed, shapes are encoded using N-sided polygons, where $N \in [1; N_{contourPixels}]$; `contour shapes` are approximated by many-sided polygons. The maximum in an object's quantized color histogram is its `dominant color`; `color layouts` are not used in our experiments. Textures are also unsupported, and gravity centers are used for `motion trajectory`.

3.1. Highway surveillance

Highway is a typical video surveillance scene with a fix camera filming cars driving down a highway. Figure 6 shows some frames of this sequence together with two possible reconstructions. The sizes of the MPEG–7 files which served to generate those reconstructions are compared with the original MPEG–1 video file in Figure 7. The *XML* sizes were obtained by adding the description ASCII file sizes together with a JPEG encoded static background (medium quality, 72dpi). *Binary* sizes are those of the code generated by the MPEG–7 Binary Encoding (BiM) reference software. The *position* reconstruction shown in Figure 6(b) is the most compact one since it requires neither shapes nor colors or textures to be encoded. It nevertheless caries most essential information of such surveillance scenes, because positions permit to make statistics about object movements. From this, incidents can be detected automatically. When more information about certain objects is needed, `dominant colors` and a `region locator` can be added, as in Figure 6(c). The difference in file size between these two detail-levels is primarily due to the fact that we re-encode the shape each time it differs from the intra-shape by more than 10%, thus generating many `region locators`. However, the MPEG–7 description size stays much below the original file size while it shows the *image simplification* (displaying only object positions), *scene description* (knowing car positions without any reconstruction) and *compactness* features.

3.2. News sequence

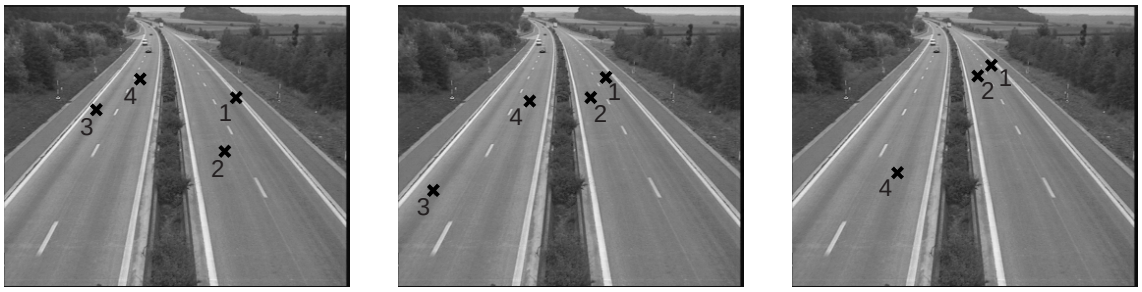
The *news* sequence shows a newsreader in front of a studio background (Figure 8(a)). For reconstruction, we divided the speaker into 12 objects* which were mainly chosen because of their color homogeneity, leading to visually satisfying reconstruction results. This sequence has again been reconstructed using only object positions, but this is not sufficient in a sequence where object forms are crucial for its understanding; therefore, this result is not shown in Figure 8. On the other hand, 6-sided polygons which are updated each two frames as in Figure 8(b) lead to a good approximation of the scene. However, the "full reconstruction" with 30-sided polygons updated each frame shown in Figure 8(c) comes much closer to the original sequence while being about of the size than the 6-sided variant. This is because the `region locator` overhead, which is present in both cases, needs much more space than the polygon vertices themselves. When polygon shapes have to be updated often, it is therefore recommendable to use `parametric motion` instead.

4. CONCLUSIONS

We have developed a new method for the MPEG–7 coding of video scenes aiming at their reconstruction. Particular attention was given to the flexibility of the resulting description by providing a scalable descriptor set together with an organization permitting reconstruction while keeping the advantages of MPEG–7 in databases. Three main description features have been identified and the advantages of our method as compared to traditional coding have been demonstrated experimentally using video surveillance and news server scenarios. Also, some possible applications for our description method have been investigated. Since semantic image processing and MPEG–7 are emerging technologies and because our description has on purpose been kept as open as possible, there is plenty of room for further improvements and developments of this topic.

*Hair, left/right eye, mouth, face, neck, body, left/right arm, left/right hand, legs.

(a) Original sequence.



(b) Reconstruction of object positions and labels.

(c) Reconstruction with 10-sided polygons (pseudo-colors for visualization).

Figure 6: Scalable reconstruction of the *highway surveillance* sequence (frames 110-130-145).

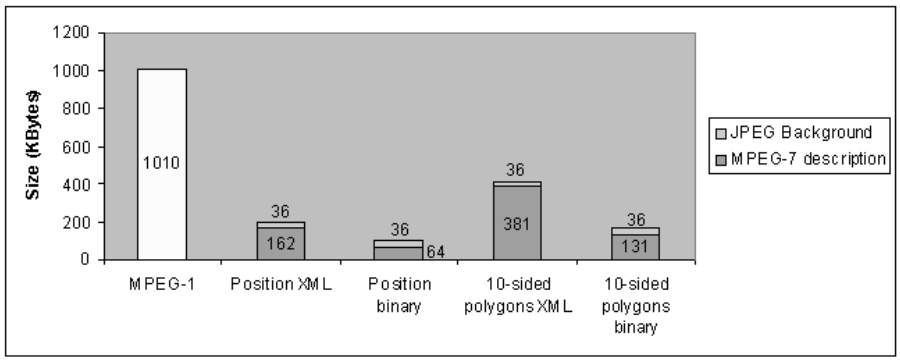
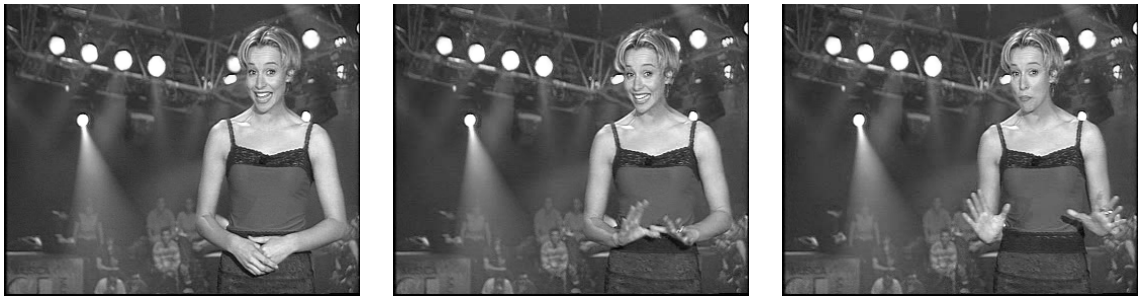
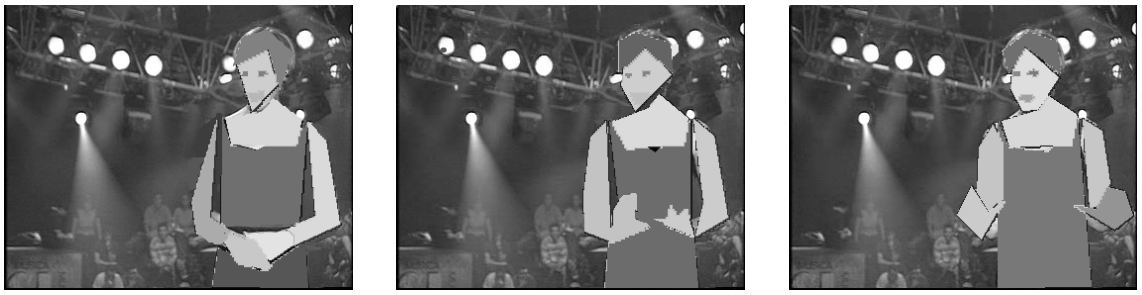


Figure 7: File sizes of diverse encoding methods (*highway* sequence).



(a) Original sequence.



(b) Reconstruction with 6-sided polygons.

(c) Reconstruction with 30-sided polygons.

Figure 8: Scalable reconstruction of the *news* sequence (frames 1-11-18).

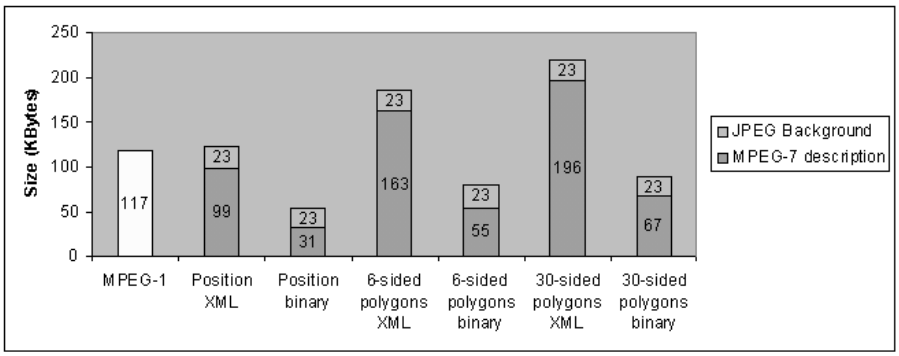


Figure 9: File sizes of diverse encoding methods (*news* sequence).

APPENDIX A. STRUCTURE OF THE MPEG–7 DESCRIPTORS

(Optional parameters are typeset in *italic*).

MediaLocator

```
{  MediaURI; };                          % Location of the media
```

RegionLocator

```
{  BoxPoly;                              % Select box/polygon representation
   Coords[numOfVertices];                % Coordinates of the box/polygon vertices
   UnlocatedRegion; };                   % Specify if inner/outer shape region is located
```

ContourShape

```
{  GlobalCurvatureVector;                % Eccentricity/circularity of original contour
   PrototypeCurvatureVector;             % Eccentricity/circularity of prototype contour
   HighestPeakY;                         % Highest CSS peak
   Peak[numOfPeaks-1]; };                % Remaining peaks
```

DominantColor

```
{  ColorSpace;
   ColorQuantization;                    % Uniform quantization of the color space
   ColorValueIndex[numOfDomCols];        % Index of each dominant color
      Percentage;                        % Percentage of pixels with associated dominant color
      Variance;                          % Variance of dominant color
   SpatialCoherency; };                  % Dominant color pixels coherency
```

ColorLayout

```
{  DCCoeff[3];                           % First quantized DCT coefficient of Y/Cb/Cr component
   ACCoeff[3][numOfCoeffs-1]; };         % Successive DCT coefficients of each color component
```

TextureBrowsing

```
{  Regularity;                           % Periodicity of underlying basic texture elements
   Direction;                            % Dominant direction of the texture
   Scale; };                             % Coarseness of the texture
```

HomogeneousTexture

```
{  Average;                              % Average of image pixel intensity
   StandardDeviation;                    % Standard deviation of image pixel intensity
   Energy[30];                           % Energies from each frequency channels
      EnergyDeviation[30]; };            % Corresponding energy deviations
```

MotionActivity

```
{  Intensity;                            % Motion intensity
   DominantDirection;                    % Motion direction
   SpatialDistributionParameters[3];     % Number and size of active objects in the scene
   SpaLocNumber;                         % Subdivide frame into SpaLocNumber rectangles
      SpatialLocParameters[SpaLocNumber];% Relative activity of each rectangular region
   TemporalParameters[5]; };             % Activity histogram
```

MotionTrajectory

```
{  CameraFollows;                        % Specifies whether camera follows object
   TrajParams[numOfKeyPoints]; };        % Specifies key-points and interpolation of moving point
```

ParametricMotion

```
{  MotionModel;                          % Translational/rotation/affine/perspective/quadratic
   CoordRef                              % 2D coordinate system
   Duration                              % Length of described motion time interval
   Parameters[numKeyPoints]; };          % Motion parameters for each key point
```

REFERENCES

1. J. R. Smith and S.-F. Chang, "Local Color and Texture Extraction and Spatial Query," *Proc. Int. Conf. Image Processing*, IEEE, Vol. **3**, pp. 1011–1014, 1996.
2. A. Kaup and J. Heuer, "Polygonal Shape Descriptors - An Efficient Solution for Image Retrieval and Object Localization," *Conf. Records of the 34th Asilomar Conf. on Signals, Systems and Computers*, IEEE, Vol. **1**, pp. 59–64, 2000.
3. José M. Martinez, "Overview of the MPEG–7 standard," Tech. Rep. N4031, ISO/IEC JTC1/SC29/WG11, Singapore, SG, March 2001.
4. A. B. Benitez, S. Paek, S.-F. Chang, A. Puri, Q. Huang, J. R. Smith, C.-S. Li, L. D. Bergman, C. N. Judice, "Object-Based Multimedia Content Description Schemes and Applications for MPEG–7," *Signal Processing: Image Communications*, EURASIP, pp. 235–269, 2000.
5. T. Echigo, K. Masumitsu, M. Teraguchi, M. Etoh, S.-I. Sekiguchi, "Personalized Delivery of Digest Video Managed on MPEG–7," *Int. Conf. Information Technology, Coding and Computing*, IEEE, pp. 216–219, 2001.
6. J.-R. Ohm, F. Bunjamin, W. Liebsch, B. Makai, K. Müller, A. Smolic, and D. Zier, "A Set of Visual Feature Descriptors and their Combination in a Low-level Description Scheme," *Signal Processing: Image Communications*, EURASIP, pp. 157–179, 2000.
7. A. M. Ferman, A. M. Tekalp, and R. Mehrotra, "Effective Content Representation for Video," *Proc. Int. Conf. Image Processing*, IEEE, pp. 521–524, 1998.
8. P. Salembier, R. Qian, N. O'Connor, P. Correia, I. Sezan, and P. van Beek, "Description Schemes for Video Programs, Users and Devices," *Signal Processing: Image Communications*, EURASIP, pp. 211–234, 2000.
9. J. M. Martinez, J. Cabrera, J. Bescós, J. M. Menéndez, G. Cisneros, "Description Schemes for Retrieval Applications Targeted to the Audiovisual Market," *Int. Conf. Multimedia and Expo*, IEEE, pp. 793–796, 2000.
10. A. D. Doulamis, N. D. Doulamis, S. D. Kollias, "A Fuzzy Video Content Representation for Video Summarization and Content-based Retrival," *Signal Processing*, EURASIP, pp. 1049–1067, 2000.
11. O. Steiger, "Smart Camera for MPEG-7," Tech. Rep., Swiss Federal Institute of Technology, Lausanne, Switzerland, February 2001.
12. A. Yamada, M. Pickering, S. Jeannin, L. Cieplinski, J.-R. Ohm, M. Kim, "Text of 15938-3/FCD Information Technology – Multimedia Content Description Interface – Part 3 Visual," Tech. Rep. N4062, ISO/IEC JTC1/SC29/WG11, Singapore, SG, March 2001.
13. P. van Beek, A. B. Benitez, J. Heuer, J. Martinez, P. Salembier, Y. Shibata, J. R. Smith, T. Walker, "Text of 15938-5/FCD Information Technology – Multimedia Content Description Interface – Part 5 Multimedia Description Schemes," Tech. rep. N3966, ISO/IEC JTC1/SC29/WG11, Singapore, SG, March 2001.
14. A. Yamada, M. Pickering, S. Jeannin, L. Cieplinski, J.-R. Ohm, M. Kim, "MPEG–7 Visual part of eXperimentation Model Version 10.0," Tech. Rep. N4063, ISO/IEC JTC1/SC29/WG11, Singapore, SG, March 2001.
15. P. van Beek, A. B. Benitez, J. Heuer, J. Martinez, P. Salembier, Y. Shibata, J. R. Smith, T. Walker, "MPEG–7 Multimedia Description Schemes eXperimentation Model Version 7.0," Tech. rep. N3964, ISO/IEC JTC1/SC29/WG11, Singapore, SG, March 2001.
16. S. Paek, "Description of MPEG–7 Content Set," Tech. Rep. N2467, ISO/IEC JTC1/SC29/WG11, Atlantic City, USA, October 1998.
17. T. Ebrahimi, Y. Abdeljaoued, R. M. Figueras i Ventura, O. Divorra Escoda, "MPEG–7 Camera," *Proc. Int. Conf. Image Processing*, IEEE, Vol. **3**, pp. 600–603, 2001.