Content-based indexing of MPEG-4 video on relational DBMS

E. Ardizzone, M. La Cascia, U. Maniscalco, D. Peri, and R. Pirrone
Dipartimento di Ingegneria Automatica e Informatica
Università di Palermo
Palermo, ITALY

Abstract *In this paper we show how it is possible to store the content of audiovisual MPEG-4 data on conventional relational DBMS. We propose a data model whose structure is based on the subdivision of a video in VideoObjects (VOs) and VideoObjectPlanes (VOPs) as defined in MPEG-4 and whose descriptors are compliant to the upcoming MPEG-7 standard. The paper represents a contribution toward a fully MPEG-7 compliant video indexing and retrieval system.*
We ran several experiments to evaluate the effectiveness of the DBMS in querying by example using vector data. Vectors are used only for visual features of VOPs and are intended only to refine queries based on classical features (like author name, duration, type of VO, etc…).
The proposed data model is described in full detail and some preliminary results are reported.

*Keywords:*Content-based video indexing, MPEG-4, MPEG-7, video data model

1 Introduction

Nowadays, more and more audiovisual information is becoming available. People willing to use audiovisual data are starting to face the problem of finding what they are looking for in a fast and effective way. In other words it is becoming evident the need for tools and techniques allowing to search for audiovisual data in a similar fashion to what we currently do on textual data. This situation is taken in such serious account that MPEG, the committee that developed standards like MPEG-1, MPEG-2 and MPEG-4 [10], is close to release MPEG-7 [12], a multimedia content description standard. The goal of MPEG-7, that is closely related to the extensible markup language XML, is to define how people search and use multimedia data. One of more interesting aspects of this convergence is the possibility of using XML language and tools for facing with description, definition and query problems at higher levels of database interaction [2].

In this paper we show how it is possible to store the content of audiovisual MPEG-4 data on conventional relational DBMS rather than on *home made* systems and how this choice makes it fast and effective for users to find the information needed. We propose a data model whose structure is based on the visual part of the MPEG-4 stream and whose descriptors are compliant to the upcoming MPEG-7 standard. Most of these descriptors may be computed automatically [5, 11, 16]. Extension of our model to the audio stream would be straightforward but it is beyond the scope of this paper.

The outline of the paper is as follows: in section 2 we will describe the proposed data model, in section 3 details on its implementation and the motivations of the choices we made are provided. In section 4 some preliminary experimental results are reported and finally section 5 contains the discussion and future directions.

2 Data model

In recent years several video indexing models have been proposed. In [20] video data are layered in video sequences, video scene, video shots, key-frames and finally objects. In [17] the video data model presented distinguishes four layers: the raw-data layer, the feature layer, the object layer and the event layer. The MAVIS 2 system [4] uses a data architecture for manipulating object that can be represented by a 4-layer model: raw mwdia layer, selection layer, selection expression layer and conceptual layer. Another hierarchical representation of video objects is presented in [19]. A more complete survey on data models for content-based image and video indexing can be found in [16].

In this paper we propose a hierarchical data model and a set of descriptors to represent video and images. The proposed model is based on the MPEG-4 subdivision of a Video in VideoObjects (VOs) and VideoObjectPlanes (VOPs). This choice, as well as the particular set of descriptors, makes our model perfectly suited to handle MPEG-7 descriptions of MPEG-4 data. Note that the data model we present in this section does not account for all of the about one hundred descriptors proposed by the MPEG commitee.

We will show in the next section how this model can be implemented on a standard commercial DBMS.

At the top level of our hierarchy is the Video. A Video has an Author, a Title, a Type and some structural information like Duration and Date. Again, we want to stress that the descriptors we chose are only a subset of the descriptors defined in the MPEG-7 specification but, at this point, we aimed at showing as MPEG-7-like description of a video can be mapped onto conventional DBMSs and how effectively the information can be retrieved. At a finer level a Video is composed of one or more VOs.

A VO is described in terms of the associated closed caption text, if any, the kind of camera operation (fixed, panning, tracking, zooming, etc…) and its dynamic. The descriptors we chose for VOs derive from the analogy we pushed between the hierarchical representation based on Video, VOs, and VOPs and the classical representation based on Video, Shots and R-frames [6]. Obviously, in a fully MPEG-7 compliant implementation of the data model we should also account for descriptors more suited for other types of VOs. At the lower level a VO is composed of one or many VOPs.

The VOP, depending on the corresponding VO from which has been extracted, is in general the object of interest and can be characterized in terms of its visual features in an automatic way. In particular, based on our experience in the field of image and video retrieval [3, 12], we chose the HSV histogram and the dominant color as color descriptors. The edge histogram and the nondirectional edge density were chosen as texture descriptors. The motion of the VOP is described in terms of its spatial coordinates in the image plane and its geometry in terms of area, perimeter, and aspect ratio. Finally a VOP may be marked as belonging to the set of VOP of a *semantical object*. This last feature, that in general cannot be inferred automatically, allows for queries like: “Show me all the VOPs where the *semantical object* Zinedine Zidane appears”.

The logical scheme of the proposed data model is shown in Fig.1.

3 System implementation

Almost all the image and video databases presented in literature in the last years use *home made* databases and query engines. Just a few systems (see for example [5, 7]) use some sort of standard DBMS and, at the best of our knowledge, only in [13] a relational model and a commercial DBMS was used.

The implementation of the data model we present, as shown in Fig. 1, on a relational DBMS is straightforward. Using a commercial ORDBMS (Object Relational DataBase Management System) it is possible to implement our data model in several ways. In fact, if it is obvious how to store in tables textual or numerical data, there are several options to store vector data (for example the color histogram). We analyzed the retrieval performance storing the n-dimensional vectors a) as rows of a table with n columns, b) as rows of a table of elements of a user defined type *vector*, and c) as rows of a table of user defined objects encapsulating the vector. In our tests, vectors made by 64 elements have been used.

We ran several experiments to evaluate the effectiveness of the DBMS. In particular the kind of queries we are mainly interested in consists of finding all the vectors in the DB whose distance from a given vector (not necessarily present in the DB) is less than a fixed value. This is the typical image retrieval query, when we have an image and look for similar images in the DB. The outcome of the experiments was that storing vector data in tables where each row is a feature vector is leads to the smallest query execution time..

Execution times for the retrieval query, using populations made by respectively 1000, 10000, and 100000 vectors are reported in table 1. Queries have been performed using the three data structures described above, and each of them has been repeated more times on the same data set. Iteration of the same query is used to take into account the caching effect performed on data by the DBMS and/or the OS, so minimum, maximum, and average time are reported.

Absolute values of the execution time strongly depend on the hardware architecture we choose, that is a Pentium II PC at 200 MHz, with 64 Mbytes of RAM, running Windows 98. Using a system with very low computational resources, allows us to consider the obtained results as a sort of superior limit to the actual execution time of the working system. The main results of the experiments are shown in Fig. 2.

Note that we analyzed the speed of the system using tables containing a few tens of thousand vectors while a real system would have tables containing millions of vectors. This is not a limitation as vectors are used only for visual features of VOPs and visual features may be intended only to refine queries based on classical features (like author name, duration, type of VO, etc…). In practice the non visual part of the query and the visual part that does not use vector features are executed taking full advantage of the indexing and optimization techniques of modern commercial DBMS and a set of a few thousands VOPs is in general obtained as result. The visual part of the query based on vector features is then executed on this resulting set of VOPs. Moreover, it has been noted [14] that querying a database of million of images using only visual features is not effective. In very large databases, the vector space of image features like color histogram tend to be very dense and a typical query by example leads to obtain a huge amount of data, the most part of them being visually uncorrelated to the query image.

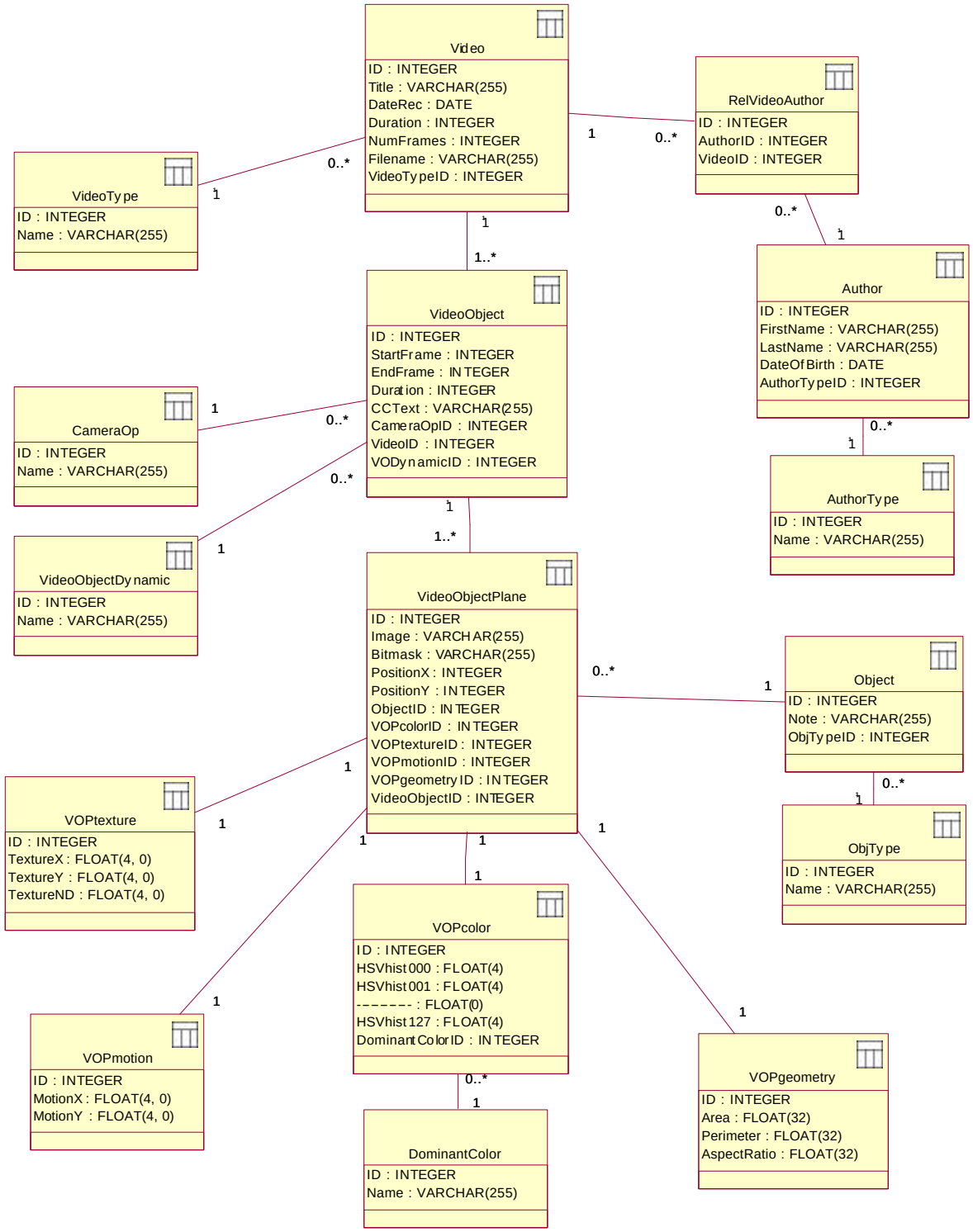


Figure 1: Logical scheme of the proposed data model.

Exec time	Data as rows of a table with 64 numerical fields			Data as rows of a table with a single user-defined *vector* field			Data as rows of a table with a single *object* field encapsulating a *vector* type		
	Population samples			Population samples			Population samples		
	1000	10000	100000	1000	10000	10000	1000	10000	100000
Max.	0,800	9,150	149,960	5,400	34,250	282,600	7,280	30,560	248,280
min.	0,460	4,410	79,300	1,830	18,300	217,760	1,860	18,810	222,000
avg.	0,518	5,025	91,490	2,223	20,501	226,807	2,435	20,486	226,486

Table 1: Execution time report for the test populations, using different data structures.

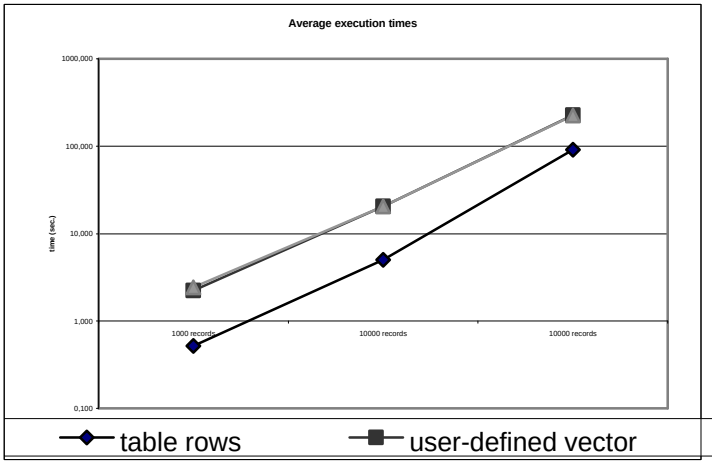


Figure 2: Performance vs. the number of records in the population for the three data structures (log axes).

4 Experimentation

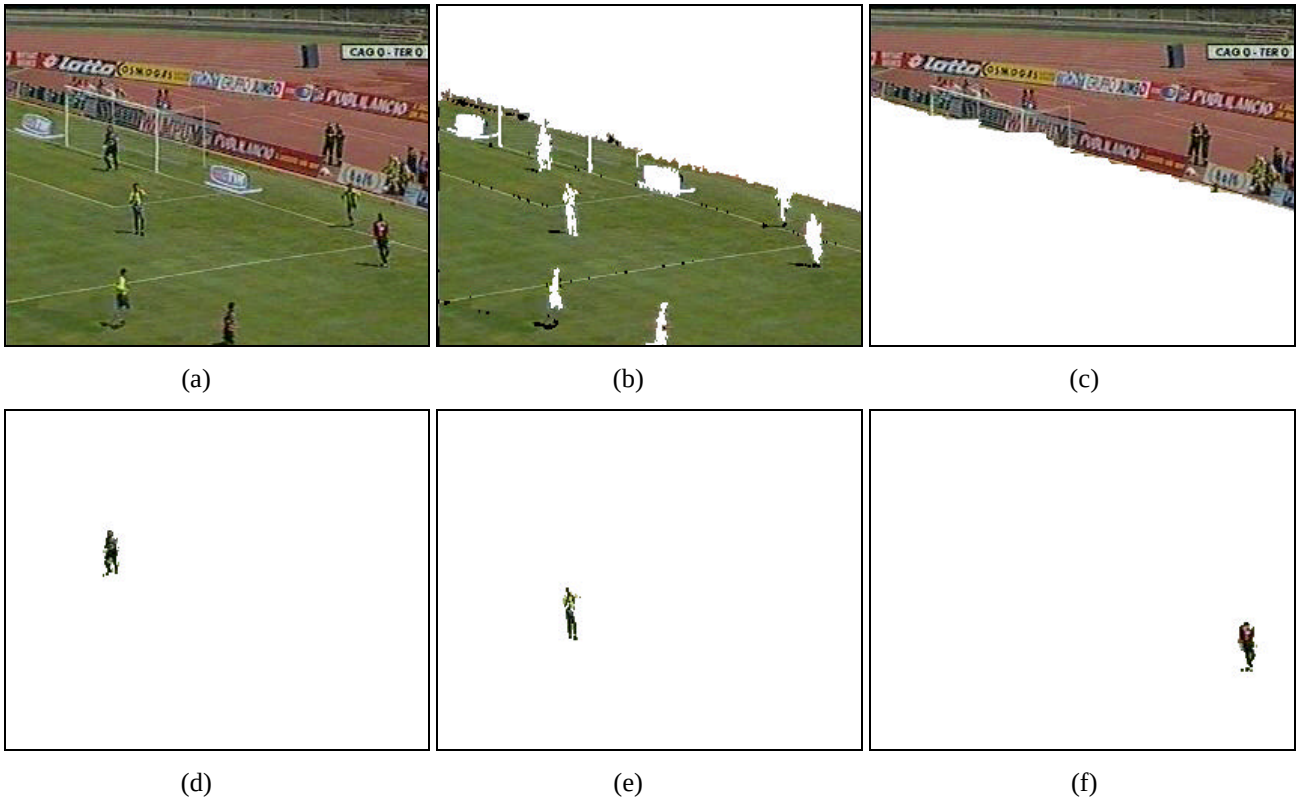
To better understand the performance of our system we are currently considering the specific domain of soccer games. This domain is not only of practical interest, given the inportance of soccer in the European multimedia market, but is characterized by visual properties that make it suited to analyze the performance of automatic indexing. Other examples of video analysis in the soccer domain may be found in [8, 17, 18].

In order to implement our data model VOs have to be extracted from videos. For general videos this task must be accomplished in manual or semiautomatic way. In the specific domain of soccer videos we were able to use a fully automatic technique [1] grounded on a color-based segmentation algorithm for the extraction of video objects. In particular this technique is based on the assumption that in each frame there are three classes of pixels aggregate: the playing field, exterior part of the field and players.

Examples of VOs automatically extracted from a soccer video are reported in Fig. 3. In (a) is reported a frame from the video sequence, in (b) and (c) are reported the VOPs of automatically extracted VOs representing respectively the playing field and the exterior part of the field. In (d), (e) and (f) are shown VOPs of VOs representing players.

To test how effective is the proposed data model we analyzed a video fragment containing 36 shots. The automatic analysis of the video, considering only no more than three players for image, resulted in about 150 VOs. For the sake of simplcity we assumed that only one VOP was able to represent the VO. Each VOP was automatically characterized on the basis of low level image features.

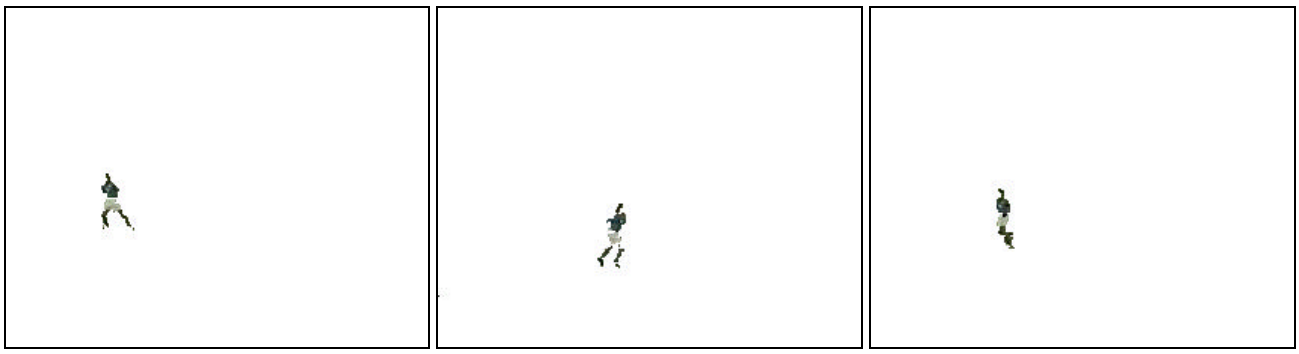
A number of query were put to test the discrimination ability of the used descriptors and results were very promising. For example a query combining the size of VOPs (less than 0.5% of total area of the image) and the color histogram (dominant blue component) allowed to retrieve all the VOPs containing Italian national team soccer players that were correctly segmented. In Fig. 4 are reported some of the VOPs retrieved.



(a) (b) (c)

(d) (e) (f)

Figure 3: Original image (a); VOPs representing VOs automatically extracted (b, c, d, e, f).



(a) (b) (c)

Figure 4: Sample VOPs retrieved querying for small object with a dominant blue component.

5 Discussion

We proposed a data model and an implementation for content-based indexing of MPEG-4 data using MPEG-7 descriptors. In principle, it would be possible to store any MPEG-7 description of a MPEG-4 stream by simply extending, if necessary, the set of basic descriptors already defined in our data model.

Even if, in general, it is not possible to map MPEG-7 descriptions to a predefined data model, as MPEG-7 is based on XML it would be possible to develop techniques (see for example [7]) for using relational DBMS to store and query the represented data. However, these techniques are likely to be effective only in some cases [15].

We believe that, in many cases, the content will be provided in a well structured form so to allow for the application of semi-custom data models, like the one we have presented in this paper.

References

[1] E. Ardizzone, and M. La Cascia, “Automatic Extraction of Video Objects in Soccer Video”, *University of Palermo, DIAI Technical Report*, May 2001.

[2] E. Ardizzone, and M-S. Hacid. “A Knowledge Representation and Reasoning Support for Modeling and Querying Video Data”, *Proc. of 11th IEEE International Conference on Tools with Artificial Intelligence*, Chicago IL, USA, November 1999.

[4] M. Dobie et al. “MAVIS 2: a new approach to content and concept based navigation”, *IEE Colloquium on Multimedia Databases and MPEG-7* (Ref. No. 1999/056), 1999.

[5] D.F. Dunn, and W.E. Higgins. “Optimal Gabor Filters for Texture Segmentation”, *IEEE Trans. on Image processing*, Vol. 4, No. 7, July 1995.

[6] M. Flickner et al. “Query by Image and Video Content: The QBIC System”, *IEEE Computer*, 1995.

[7] D. Florescu and D. Kossman. “Storing and Querying XML Data Using an RDBMS”, *Bullettin of the Technical Committee on Data Engineering*, Vol. 22, No. 3, September 1999

[8] Y. Gong, L.T. Sin, C.H. Chuan, H-J Zhang, and M. Sakauchi. “Automatic Parsing of TV Soccer Programs”, *IEEE International Conference on Multimedia Computing and Systems*, Washington D.C., 1995

[9] A. Gupta and R. Jain. “Visual Information Retrieval”, *Communications of ACM*, Vol. 40, No. 5, 1997.

[10] R. Koenen (Editor). “Overview of the MPEG-4 Standard”, *ISO/IEC JTC1/SC29/WG11 N4030*, March 2001.

[11] M.S. Lew (Editor). “Principles of Visual Information Retrieval”, Springer, 2001

[12] J.M. Martínez (Editor). “Overview of the MPEG-7 Standard”, *ISO /IEC JTC1/SC29/WG11 N4031*, March 2001.

[13] V.E. Ogle, and M. Stonebraker. “CHABOT: Retrieval from a Relational database of Images”, *IEEE Computer*, 1995.

[14] S. Sclaroff, M. La Cascia, L. Taycher, and S. Sethi, “Unifying Textual and Visual Cues for Content-Based Image Retrieval on the World Wide Web”, *Computer Vision and Image Understanding*, (CVIU), 75(1), July 1999.

[15] J. Shanmugasundaram et al. “Relational Databases for Querying XML Documents: Limitations and Opportunities”, *Proc. of 25th International Conference on Very Large Data Bases (VLDB)*, Edinburgh, Scotland, 1999.

[16] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. “Content-Based Image Retrieval at the End of the Early Years”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, (PAMI). Vol. 22, No. 12, December 2000.

[17] M. Petkovic, W. Jonker. “A Framework for Video Modelling”, *18th IASTED Conference on Applied Informatics*, Innsbruck, Austria, 2000

[18] A. Woudstra et al. “Modelling and Retrieving Audiovisual Information - A Soccer Video Retrieval System”, *4th International Workshop on Multimedia Information Systems*, Istanbul, Turkey, 1998.

[19] D. Zhong, and S.F. Chang. “Video Object Model and Segmentation for Content-Based Video Indexing”, *IEEE International Conference on Circuits and Systems*, Hong Kong, 1997.

[20] W. Zhou, A. Vellaikal, C.-C.J Kuo. “Video analysis and classification for MPEG-7 applications”, *International Conference on Consumer Electronics*, 2000.