ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

SIMILARITY-BASED VIDEO RETRIEVAL: MODELING AND PROCESSING

BY

DESSALEGN MEQUANINT

JUNE 2004

SIMILARITY-BASED VIDEO RETRIEVAL: MODELING AND PROCESSING

BY

DESSALEGN MEQUANINT

A THESIS SUBMITTED TO THE SCHOOL OF GRADUATE STUDIES OF ADDIS ABABA UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN COMPUTER SCIENCE

JUNE 2004

Acknowledgement

First of all, my deep and sincerely appreciation goes to my advisor Dr. Solomon Atnafu for his critique, supervision of the thesis work, friendly approach and the good time that I had with him while I was working on this thesis.

Special thanks goes to all the instructors of the graduate program of the Computer Science department who have been always working tirelessly for the success of the graduate program.

Acknowledgment also goes to friends: Mastewal Getu, Basazin G/Iyesus, Fikre Ayele, Ajire Tedla, Senait Berihu, Hellen Tadesse and Huluanten Abate who have been providing me the necessary support for the successful completion of the thesis work.

Last but by no means the least I am grateful to my parents who have been doing for me more than all they could.

Table of Contents

List of Figures

List of Tables

ABSTRACT

Nowadays, digital video data is increasingly available for public accesses. However, the rate at which access to this key element of multimedia computing is growing is unparalleled with the tools and techniques that have been developed for retrieving and browsing it. In recent years, a lot of research efforts have been put into video data management, amongst other active research topics, video data modeling has been extensively researched, but nearly no work has been done in the area of video data modeling that reveals the role of metadata in enhancing video retrieval systems, and only little work has been done in the area of video data repository modeling. In many cases the motivation of the research efforts has been the accessibility problem.

In this thesis, a video data model with two separate representation schemes is proposed. The proposed video data model distinguishes three parts: the frame-based scheme, the object-based scheme and the external description block. We focus our attention on the frame-based scheme and the external description block. In addition, a video data repository model that can be conveniently used under OR-DBMSs environment is proposed. On the basis of the proposed video data repository models, a similarity-based video retrieval technique in the context of OR-DBMS is also proposed.

In an effort towards the design of effective and interoperable video retrieval systems, standards play a major role, in this thesis through the video data model we proposed the benefits of MPEG-7 are directed to reach OR-DBMSs. Furthermore, to demonstrate the practicality of our proposals a prototype system for a soccer application domain which is called SVAMS is developed.

Key words: Video data model, Video data repository model, Frame-based representation, Object-based representation, Video databases, Similarity-based video retrieval.

CHAPTER 1

INTRODUCTION

In this chapter, we will first describe the problem statement, motivation and the current status of video data modeling and retrieval. We then introduce our research scope, research objectives and the main contributions of this thesis work.

1.1 Problem Statement

Nowadays, digital data of audiovisual nature is becoming available to public access increasingly. According to a study conducted to measure how much information is produced annually in the world [30], the rate at which digital data is produced, particularly video and image data is reported to be rapidly increasing. The rapidity with which digital information, such as image and video generated is growing in unprecedented rate as compared to the tools available to manage these digital media efficiently and systematically.

As observed in the study,

> *"There are over 2700 photographs taken every…*
>
> *Apart from still photography, film is also used to store moving pictures. In the years from 1990 to 1995, UNESCO reports that there were 4,250 films produced annually throughout the world. The Motion Picture Association of America reports that for the year 1998, its members released 221movies (compared to 219 in 1997), while*

releases by all U.S. companies, including independent film companies, rose from 461 in 1997 to 490 in 1998.

It takes approximately 2 gigabytes to store an hour of motion picture images in digital form using the MPEG-2 compression standard. If the images in 4500 full length movies were converted into bits, the world's annual original cinematic production would, therefore, consume about 16 terabytes." [30, p.17-18].

Table 1.1 shows sample figures on the rate at which image and video data are produced annually and the corresponding storage space requirements as indicated in [30].

Table 1.1: Original Data On Film Annually Worldwide.

	Units	**Digital Conversion**	**Total Petabytes**
Photography	82,000,000,000	5 MB per photo	410
Motion Pictures	4,000	4 GB per movie	0.016
X-Rays	2,160,000,000	8 MB per radiograph	17.2
Total:			**427.216**

On the study, hours per year spent on various medias in US households in 1992 and 2000 have also been assessed. Table 1.2 shows samples figure on the percentage rate at which access to a media is changing and the storage space requirements of the year 2000.

Table 1.2 Summary of yearly media use by US households in hours per year, with estimated megabyte equivalent. (Hours from Statistical Abstract of the United States, 1999, Table 920, (projected)).

Item	In 1992 Hours Spent	In 2000 Hours Spent	In 2000 Storage in MB	%Change
TV	1510	1571	3,142,000	4
Radio	1150	1056	57,800	-8
Recorded Music	233	269	13,450	15
Newspaper	172	154	11	-10
Books	100	96	7	-4
Magazines	85	80	6	-6
Home video	42	55	110,000	30
Video games	19	43	21,500	126
Internet	2	43	9	2,050
Total:	**3,324**	**3,380**	**3,344,783**	**1.7**

The above two tables Table 1.1 and Table 1.2 clearly show how video is becoming a key element of multimedia computing.

By considering the size of video collections that have been already produced since a decade or two, including this annual rate of production of videos as shown in table 1.1 and the possible future increase in the annual rate of production of videos due to advance in technology (i.e. ease

of capturing and encoding of digital video), the need to have an efficient video and video related data management system is critical. However, video is content rich, complex, lengthy and unstructured format, which in many cases is not amenable to the traditional text and alphanumeric data management techniques. This calls for the development of a better data management techniques tailored to the behavior of video data.

This thesis tries to address some of the difficulties of video data management techniques. In particular it addresses problems of video data modeling, video data repository modeling and multi-criteria video retrieval.

1.2 Motivation and State-of-the-Art Status

Motivated by the drawbacks of traditional data management techniques in managing video data, a lot of research efforts have been put into the development of efficient video data management techniques aimed at making video data as searchable as text data [1,2,3,4,6,10,28]. In this regard, in the field of video retrieval there are many areas of active research interest. Some of these are segmentation of a video, extraction of a video unit, representation of a video and its constituent units, and modeling the perceptual and the semantic features of a video. Despite the tremendous amount of research efforts put into this area, video retrieval is still at the level of infancy, the achievements obtained in the field have indicated a long way to go. This is because video is a complex media, and the complexity of dealing with video data stems from the media itself (i.e. video is much more information-rich than text data). Thus, any research endeavors that contribute to the field in any manner are highly needed.

The research activities concentrated on video retrieval have come up with video data models that model the video data only [2, 20,21]. The importance of metadata in enhancing retrieval systems

has got due attention in many areas such as in traditional databases and geographic information systems [42], there is a similar need to exploit these features of metadata in video data management. However, in existing research prototype and commercial video retrieval systems, metadata is not used to its full advantages, this is mainly attributed to lack of a video data model that can model, classify and explicitly represent metadata of a video along with the video content. Moreover, a video repository model that can conveniently be used under OR-DBMS environment is still lacking. Likewise a similarity-based video retrieval technique that exploits the features of an OR-DBMS is also lacking.

Thus, motivated by the drawbacks of existing video data management techniques, in this thesis a video data model tailored to the behavior of video data is proposed. Moreover, a video data repository models associated to a video and its constituent units are proposed, the video repository models are mirror images of the proposed video data model that can conveniently be used under OR-DBMSs environment. The features that an OR-DBMS offers are fully exploited in an effort towards the development of a new similarity-based retrieval technique proposed in this thesis.

1.3 Research Objectives and Main Contributions

This thesis is aimed at solving some of the problems that surround video data management. Particularly, it addresses problems of:

- video data modeling,
- video data repository modeling

In line with the above objectives, major contributions of this thesis are the following.

- Introduces a video data model, which distinguishes three parts: the frame-based scheme (or frame-based representation scheme), the object-based scheme (or object-based representation scheme) and the external description scheme.
- Introduces a schematic video data repositories models defined at the video and its constituent elements, which can be conveniently used in the management of video collections under OR-DBMSs environment,
- Introduces metadata as first class citizen of a video data model[1]: metadata embedded in a video is modeled, classified and its role in enhancing retrieval efficiency is fully revealed.
- Introduces a similarity-based video retrieval technique in the context of OR-DBMS,
- Identifies the important components of a video data that need to be considered in order to fully represent and describe a video and its metadata,
- A prototype system for a soccer application domain, which is called SVAMS that demonstrated the workability of the proposals made in this thesis, is developed.

1.4 Thesis Organization

The rest of this thesis is organized as follows. Chapter 2 gives a detailed review of related work in video data management. In Chapter 3, the proposed video data model is presented. In Chapter 4, the proposed video data repository models defined at the video and its constituent units and their management under OR-DBMSs environment is presented. In Chapter 5, SVAMS, a simple soccer video archive prototype system that demonstrates the practicality of our proposals (particularly the frame-based representation scheme of the proposed video data model) is

[1]In the proposed video data model, in addition to the content of a video, metadata of a video is also modeled, classified and represented.

presented. In Chapter 6, discussion on issues that have been addressed in this thesis is presented. Finally, we present conclusions and future perspectives of our work in Chapter 7.

CHAPTER 2

RELATED WORK

Multimedia applications are rapidly increasing at an ever-amazing rate, however the technologies or tools that enable applications to use these digital medias effectively and efficiently are still in their infancy [9]. One of the facets of problems that surround such systems is finding efficient ways to summarize the huge amount of data involved. The problem is acute for video data. This is because video is sheer volume, complex, unstructured format and content rich, due to this content-based access to a video data is often impeded. Therefore, efficient and effective methodologies for organizing and manipulating the vast volume of digital video data are highly needed.

Despite the diversity in modeling and application of CBVR (Content Based Video Retrieval) systems, most systems rely on similar techniques or methodologies in manipulating and organizing video data. Therefore, in this chapter the prevalent techniques and methodologies that are highly useful for effective management of video data are presented.

In this chapter a review of several related works that are fundamental to the theme of this thesis are presented. First video data modeling approaches are presented. Then we review: video content analysis techniques, video perceptual features representation, video semantics content representation, video browsing, video retrieval, content-based retrieval methods, CBVR systems that are either research prototype or commercial, standards that are relevant to video data

management, content-based retrieval systems, and database management systems that support content-based video retrieval and finally, a succinct summary of the shortcomings of existing systems and the usefulness of our work in solving some of these shortcomings in the area are presented.

2.1 Video Data Modeling Techniques (or Approaches)

Modeling the video data is one of the principal activities that must be carried out in an effort towards the design of effective video retrieval systems. It is an important tool that can be used:

- to portray the requirements of video retrieval systems,
- to facilitate the understanding of the characteristics of video data,
- in the specifications of similarity-based query that the retrieval system can answer,
- in the specifications of video operations that can be performed on the video data.

The success of conventional databases is partly attributed to the data modeling tools and techniques they use. There is a need to have similar achievements in video data management so as to have effective and efficient video retrieval systems. Thus, the place of video data modeling for retrieval efficiency cannot be overemphasized, without the help of this indispensable tool designing of an effective retrieval system is impossible.

However, unlike conventional alphanumeric data, video data, due to its length, unstructured format and complexity, is not easy to model. Modeling a video data is much more complex than modeling conventional alphanumeric data. This calls for the development of data modeling technique or approach amenable to the behavior of video data. The development of a video data model should take into account factors that are specific to the behavior of video data.

Nevertheless there are properties that a video data model shares in common with the traditional data models such as expressiveness and semantic realism [46]. The expressive power of any video model partly depends on the granularity, the range and variety of video units: frames or portions of a frame (sub-frames), structures, relationships, and behaviors that can be represented with it [9].

Thus, video data modeling poses new demands that are not present in traditional data modeling techniques. This is because in modeling a video data, in addition to modeling the structural properties of a video, the concepts and events that represent the content of a video should also be modeled [9]. That is modeling video involves making a distinction between two important things that should be modeled: modeling the structure of a video and modeling the content of a video.

In the subsections that follow video data modeling approaches that have been proposed in the literature are presented.

2.1.1 Modeling the Video Structure

Video is content rich, characterized by voluminous and unstructured format. It conveys large amount of information. To manage video data effectively, the huge amount of video data must be structured into constituent units of the video such as scenes, shots and key frames. To this effect, the elemental video units as well as the segmentation criteria (boundaries detecting techniques) to be used in detecting boundaries between consecutive video units should be identified [9]. Moreover, video is a time-based media; therefore besides video structure modeling, temporality is one of the most important factors that a video model should be able to express [10].

The most important issue that arises in the design of video retrieval systems is the description of structure of video data in a form appropriate for querying. To this end, the structural elements of

a video data must be obtained by segmenting the video into smaller and manageable units. The video segmentation criteria commonly employed in research prototype and commercial retrieval systems can be classified into two: syntactic and semantic [11]. Syntactic criteria segments a video into sequences of frames generated during a continuous camera operation. In this approach a video is divided into fixed segments (shots) and every segment is described independently using free text or keywords annotation as the semantic content description of the video segment. This segmentation approach is inefficient for large collections of video. Moreover, the semantics or important contextual information about the video sequence is lost during segmentation [9]. This in turn results in the predetermination of the type of query that the retrieval system is going to answer at the outset, and hence it is difficult for the user to retrieve part of a video shot and to access the same video shot for different purposes other than it was intended. Therefore, a segmentation criteria based on semantic is required for organizing video information effectively. The semantic criteria segment contextual information of a video instead of simply partitioning a video into sequences of frames. A video model, which uses this technique such as the stratification-based approach, is proposed in [12]. In this approach a video sequence is divided into overlapping parts, which are known as strata, and annotated video elements are organized hierarchically as nodes in a tree, and based on the nested relationships between the nodes it is possible to determine the context in which a node appears.

2.1.2 Modeling the Video Content

The next logical step that follows temporal segmentation in modeling video data is, modeling of the content of each video unit obtained as a result of the use of segmentation. This phase, that is modeling the video content, facilitates similarity-based matching between video units at query

time. It is achieved in two ways: through perceptual features-based modeling and semantic-based modeling [9].

2.1.2.1 Feature-based modeling

Though video is a time-based media, most of the techniques that model visual content rely on extracting image-like features from the video sequence [13,44]. In features-based modeling of video data, two categories of features are considered: visual features that can be extracted from key frames or the sequence of frames after the video sequence has been segmented into shots and temporal motion features that can be extracted from raw video streams. In this section we will review the low-level perceptual features that are specific to video data, and a general overview of image like features of a video. Detailed descriptions of image like features of a video can be looked up in [18] and the abundant material cited in it.

2.1.2.1.1 Temporal Motion Features

Video is a dynamic media, which is also rich in content. The dynamism of a video stems from its motion attribute, which stands out as its most distinguishing feature to effectively managing it [13]. There have been works concentrated on the description of a video object activity using motion [14,15]. In [14], macroblock tracing and clustering is used to derive trajectories and then compute similarity between these raw trajectories. In [15], a Video Tracking and Retrieval System known as VORTEX is developed to track an object in the compressed video stream using a bounding box. Object tracking is facilitated through motion vector information embedded in the coded video stream.

Once the task of extraction of object trajectory of video objects is accomplished, modeling of the motion trail is essential for indexing and retrieval systems.

2.1.2.1.2 Spatial Image Features

To efficiently model the visual content of a video, its low-level image representation features must be extracted from the video sequence such as from the key frames. We then can use techniques that have been developed for image data on the extracted key frames. The obvious candidates for representing the visual feature space of a video are color, texture, and shape. Thus, features used to represent the visual content of a video have conventionally been the same ones used for images, extracted from key frames of the video sequence [13]. A visual features-based approach that uses spatial image features to represent or model the visual content of key frames is proposed in [16]. In this approach, first the video is segmented spatio-temporally to obtain regions in each shot. Each region is then processed for feature extraction. After spatio-temporal segmentation, the features are extracted from objects detected and tracked in a video sequence.

2.1.2.2 Semantic Modeling

Once feature level modeling is done, semantic level modeling based on conceptual models is required to facilitate both browsing and retrieval of video data.

Video contains a rich set of information about objects and events being depicted. The major drawback of feature-based models is their inability of portraying or conveying the semantic interpretations embedded in the video [9]. Thus, the need to have a model that bridges the semantic gap is quite apparent. There are times where querying video data using perceptual features may not be feasible. This limitation may stems from the query paradigm itself. The QBE (Query By Example) is the paradigm dominantly used in research prototype and commercial retrieval systems, in this query paradigm the user is required to submit a key frame or an image that specifies the desired features. Its drawback is manifested at times when there is no more

example key frame or image to use. This is where Query By Keyword (QBK) an alternative to QBE comes in. It is a querying paradigm that enables users to retrieve video data that have been annotated using high-level concepts or domain specific keywords. Even though QBK is better than QBE in returning query responses that are semantically meaningful, nevertheless, it suffers from problems such as subjectivity of describing objects and events of a video and the difficulty associated to describing the huge volume of video data, which in many cases could be prohibitive in terms of the time it takes. In general current semantic-based modeling approaches can be classified into two: segmentation-based and stratification-based [9]. As we mentioned in the previous subsection, the drawback of the former approaches is lack of flexibility, and its inability of representing semantics residing in overlapping segments. The latter approaches, however, segment contextual information of video instead of simply partitioning it. In the literature there have been approaches proposed for bridging the semantic gap through the use of semantic-modeling. One such approach is found in [16]. In this approach the concept of "Multiject", a Multimedia Object is used. A Multiject is the high-level representation of a certain object, event, or site having features from audio as well as video. It has a semantic label, which describes the object in words. Other approaches proposed for semantic modeling are found in [16,17]. In [16], semantic content having unrelated time information are modeled as ones that do. In [17], a graphical model, VideoGraph, that supports not only the event description, but also inter-event description that describes the temporal relationship between two events.

Modeling the semantic content of a video is much more difficult than modeling the video structure or the low-level perceptual content of a video [9]. However, from users pint of view semantic-based modeling are more appealing. In this context, in the video data model we

proposed this desire of users is addressed, that is, for effective semantic-based query metadata of a video have been modeled and classified in conjunction with the content of the video.

2.2 Video Data Models

Video data models proposed in the literature can be classified into generalized models and domain-specific models [18]. Generalized models do not take into account the peculiarities of videos, but concentrate on the conceptual level at which visual content can be represented. Domain-specific models are, instead, tailored to the peculiarities of the application domain.

In this section in addition to some of the known video models proposed in the literature, the generic image data model proposed by R. Chbeir et al. [19] is also presented. The chosen image model is generic, and it also complies the requirement of our modeling approach. Therefore, a key frame in the scene-shot-key frame representation scheme of the video data model we proposed can be conveniently represented using image models.

Video data modeling approaches proposed in the literature [2,20,21] have only focused on modeling of the video structure and video content only, no work has been done in the are of modeling, classifying and representing metadata of a video as first class citizen[2] of a video model. That is, a video data model focusing on the three aspects of a video such as the video structure, the video content and metadata of the video data at the constituent units of the video was lacking. Our modeling approach differs from those video models proposed in the literature in that it has addressed the key role that metadata play for video data management. It has also explicitly addressed differences in the representations of video data thereby exploiting object-based characteristics of video data available in MPEG-4 compressed bitstreams. Moreover,

[2]First class citizen: in the video data model we proposed, the key role that metadata can play for video retrieval efficiencies is realized by modeling, classifying and representing it along with the video data, which in many cases is overlooked by existing video data models proposed in the literature

metadata embedded in the video is modeled, classified and its accurate and consistent representation is managed using the standard content descriptor interface MPEG-7.

In the following paragraphs we have succinctly presented some of the generic models that have been proposed in the literature.

2.2.1 The DISIMA Video Model

This video model is an extension of the DISIMA image model proposed previously [29], it is a generic video model, which is based on segmentation-based and salient objects-based approaches. It uses a hierarchical representation of video data [2].

The video data model captures the structural characteristics of video data and the spatio-temporal relationships among salient objects that appear in the video.

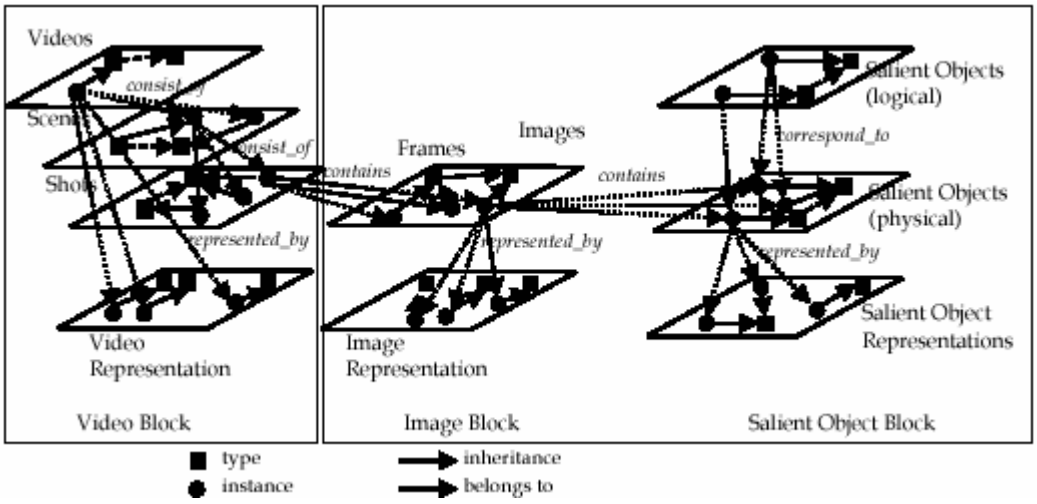

Figure 2.1 The DISIMA Video Model

In this model a video block is introduced to model video data. A block represents a group of semantically related entities. As has been depicted in the figure the video block has four layers: video, scene, shot and video representation. In the model only the key frames are used to

represent the contents of a shot. The relationship between key frames and shots establishes the connection between a video block and a DISIMA image block. This model is meant to the salient-object of video key frames.

2.2.2 The VIMSYS Video Model

Visual Information Management System (VIMSYS) is a generic video data model that allows explicit representation of abstract levels in visual content [20, 21]. Like DISIMA it also uses a hierarchical representation of video data.

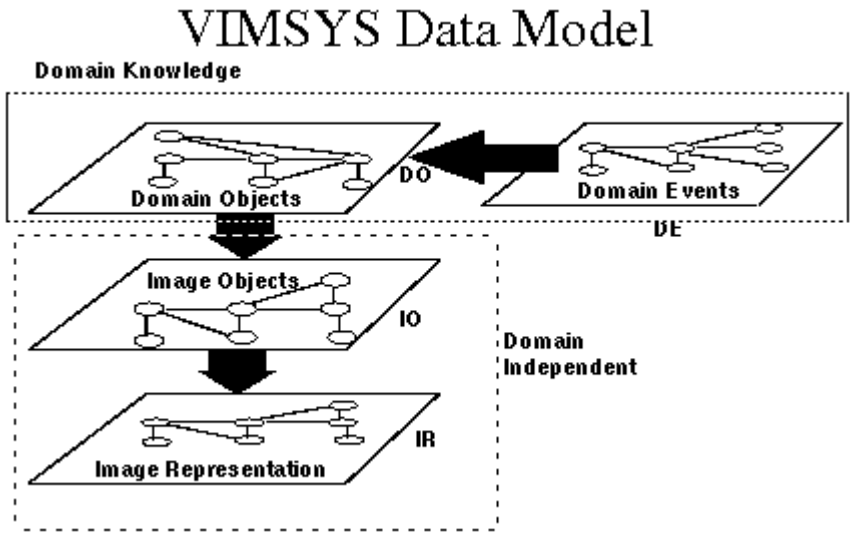

Figure 2.2 The VIMSYS Video Data Model

It has three layers: Domain Objects or Domain Events, Image Objects and Image Representation. The image representation level, which stores raw image data. The image object level, which stores spatial entities like points, lines and regions that are extracted from images, these entities are regarded at a syntactical level with no interpretation. The domain object level, which associates entities, identified at the image object level, with objects, provides semantic interpretation. The domain event level, defines relations between objects. Each object at the

domain object level can be associated with any other object using a domain entity in the domain event level. Spatial or temporal relationships between objects are represented at this level.

This model is suited for both image and video. The first two levels are domain independent and only account for syntactical aspects of visual data. The other two account for semantics. Using this data model, high-level semantic concepts can easily be mapped into low-level content features [18].

2.2.3 The Strata Video Model

A stratified model for video information is proposed in [22]. In this model the shot as the basic component of a video stream is identified. Then for each shot a stratum is assigned, which are a set of textual descriptors: the frame number, and a number of keywords referring to the salient elements or actions represented in the shot [23]. This video modeling approach differs from those approaches presented above in that the shot video unit is not identified merely based on physical boundaries instead it is identified based on the context it represents.

2.2.4 Image Data Model

Here we presented an image data model that can be used to further describe the content of a key frame in the scene-shot-key frame representation scheme of the video data model we proposed. Images are of prime importance among the various media types, not only because images are the most widely used media types besides text, but also because images are the basic components for video data [45]. Owing to the fact that a key frame is special type of image, image data models proposed in the literature can be conveniently used to describe and represent the content of a key frame. To this effect, we have chosen a generic image model proposed by R. Chiber et. al [19]

for the description and representation of key frames of a video that can be captured in the frame-based representation scheme of the proposed video data model.

Therefore, by establishing a relationship between key frames of a video and images data, it is possible to employ the tried and tested techniques of image data management for the management of video key frames. The descriptions of the components involved in the image data model can be found in [19]. Below, is the generic image data model proposed by R. Chiber et. al [19].

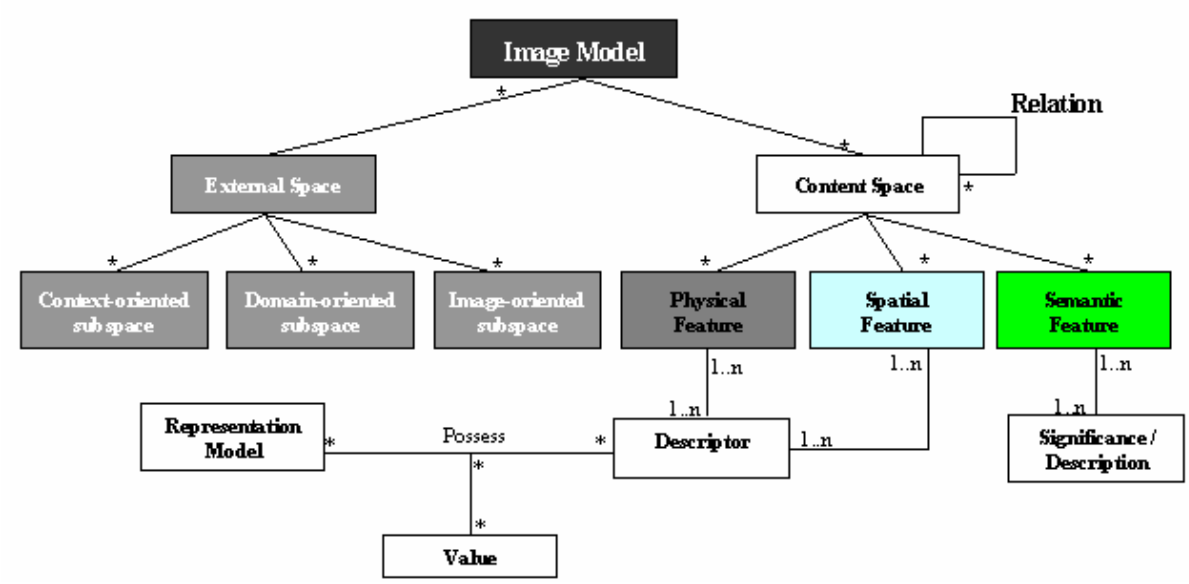

Figure 2.3 An image data model in UML notation

2.3 Video Analysis

The primary work of any video retrieval system is the analysis of video streams. Video analysis deals with the signal processing part of video streams. It is a video preprocessing activity that involves two stages: shot boundary detection and key frame extraction. The first stage, shot boundary detection, is carried out to segment a video stream into a set of manageable video units such as shots. The second stage, key frame extraction, is carried out to select representative frames of a video shot that aims to abstract a shot using a frame or more. Video content analysis techniques are also important in presenting the content of a video to users in a manner that enables rapid browsing and visualization

In this section a review of the various video content analysis techniques proposed in the literature are presented.

2.3.1 Temporal Segmentation

The first and most important task in video content analysis phase is the determination of the video unit, in which the video temporal sequence may be organized [24]. Once a unit of interest is determined, video content analysis techniques such as segmentation will be employed to decompose a video stream into manageable units such as shots represented through key frame(s), and their high level compositions, such as scenes. Therefore, through the use of segmentation techniques a video can be decomposed into constituent elements such as shots, on top of which a set of bookmarks that facilitates content-based access to video data can be created [5].

Existing shot boundary techniques have problems in detecting real shot boundaries, they sometimes fail to detect shots that must be detected (i.e. false negative) and they also sometimes detect shot boundaries that are not (i.e. false positive) As a result automatic detection of video

elements such as shots is an important topic of research. Shot boundary detection techniques are algorithms that compute the frames that lie on the boundaries (start and end) of each shot. Most of the research activity concentrated on the detection of shot boundaries use, both sharp or abrupt transitions (cuts) and gradual shot transitions (fades, dissolves and wipes) in order to automatically detect shots [18]. Early days segmentation techniques have focused on cut detection, but there has been increasing research activities on gradual boundary detection as well. In the literature several shot boundary detection algorithms have been proposed, however, in this section, for the sake of brevity we discussed only some of the algorithms that are dominant both in the research prototype and commercial video retrieval systems.

The process of determining shot boundaries through cut (sharp or abrupt) transition detection approach is quite simple in that a shot boundary is declared by determining the boundaries between consecutive camera shots and checking for camera break [18].

The prevalent shot boundary techniques reported in the literature detect shot boundary by extracting some form of features for each frame in the video sequence, then evaluating a similarity measure on features extracted from successive pairs of frames in the video sequence, and finally declaring the detection of a shot boundary if the feature difference conveyed by the similarity measure exceeds a threshold [13]. One such approach is found in [47], in this approach two difference-based metrics such as Histogram Distance Metric (HDM) and Spatial Distance Metric (SDM) are proposed. These metrics are computed for every frame pair. HDM is defined in terms of 3-channel linearized histograms computed for successive frame pairs as a difference of their histograms. SDM is defined in terms of the difference in intensity levels between successive frames at each pixel location. To this effect, a spatial distance operator is defined to compute difference in intensity levels of frames. Then these two distances are treated as 2-D

feature vector and an unsupervised K-Means clustering algorithm is used to group shot boundaries into one cluster.

There has been also shot boundary detection techniques proposed for gradual transitions detection. Gradual transitions, such as fades, dissolves and wipes, cause more gradual changes, which evolve during several frames [18]. Traditional approaches for detecting gradual transitions are Twin-Comparison and Edge detection. In the Twin-Comparison algorithm [24], the first and last transition frames are treated as quite different, while consecutive frames remain very similar. It detects gradual transitions and distinguishes them from cuts. In the Edge detection algorithm [48], criteria for detecting gradual transitions have been used. In this approach the number of edge changes measured are compared against a threshold value, accordingly gradual transition detection is declared when the measured edge changes exceeds a threshold value.

A shot boundary detection technique based on the spatio-temporal relationship of objects is proposed in [49]. It is a rule-based approach for a high-level segmentation of a video into shots based on spatio-temporal relationships between objects in video frames. In this approach a segmentation of video clips (an interval of video frames) into shots occurs whenever the current set of relations between objects changes.

While shot-based video analysis approaches provide users with better access than unstructured raw video stream, they are still not sufficient for meaningful video browsing and retrieval. This is because shots are not marked by semantic boundaries. From users' point of view a video unit at a coarse grain such as a scene is needed, however, users' desire of having a coarse level video unit is impeded by the limitations of existing video content analysis techniques. In the literature segmentation techniques beyond the shot unit have also been proposed [50,51]. In [50], a video is segmented into scenes based on a pesudo-object-based shot correlation analysis. In [51], video

scenes are located by analyzing inter-frame similarity matrices. However, most of the research efforts concentrated on scene boundary detection including those mentioned earlier on have not proven to be successful to be used in unconstrained domains. This is because, the raw video is provided in terms of a sequence of frames, in the literature even detecting the boundary of a shot from a video with a sequence of raw frames has been reported to be a challenging task, and as a result shot boundary detection is still active research topic. Therefore, detecting a boundary at a coarse grain beyond a shot such as a scene which is marked by semantic boundary is much more challenging [13,18].

2.3.2 Video Summarization

The next logical step that follows shot boundaries detection is the selection and extraction of the corresponding key frames from the shots identified during video segmentation. Abstracting the content of a large video streams using key frames is the simplest and widely employed method. The simplest and intuitive approach to abstract large video streams is to use the first frame of each shot as a key frame [13]. Although the approach is simple, it is more restrictive in that each shot is represented in a single frame regardless of its content complexity. Also, the choice of the first frame over the other frames in the shot is arbitrary. The other intuitive approach is to select the first and last frames of a shot as key frames [52]. Although this approach is better than the previous approach, it is again blamed for the flexibility it lacks in representing a shot with complex content with sufficient number of key frames. More complex key frame extraction techniques are proposed in [53,54,55]. In [53], visual content complexity is considered, and in [54], shot activity indicator is considered, and in [55], shot motion indicator is considered. In [65] a shot with significant content changes is segmented into several subshots that are of coherent content and shot similarity measure for video retrieval is computed from similarity

between corresponding sub-shots. Key frame extraction or content abstraction technique is performed using two descriptors such as Dominant Color Histogram (DCH) and Spatial Structure Histogram (SSH) in order to characterize the temporal content variation. In [67] the notion of Perceived Motion Energy (PME) feature of a shot is used to determine the number and location of key frames for abstracting the content of a video. In this approach a triangle model is developed to describe the motion pattern and the segmented video sequences, the frames at the start point and end point of motion acceleration are selected as key frames candidates. An approach based on two adaptive algorithms [66], are proposed in order to effectively select key frames from segmented video shots and both apply a two-level adaptation mechanism. The first level is based on the size of input video file while the second level is performed on a shot-by-shot basis in order to account for the fact that different shots have different levels of activity.

Another video summarization approach, which uses spatio-temporal relationships between objects of video frames, is proposed in [49]. In this approach whenever the current spatio-temporal relationships between objects of video frames changes, the video frames that exhibit the change will be selected as key frame(s).

The video summarization techniques discussed in the previous paragraphs are the traditional ones in that they all use key frame(s) as a unit to represent a shot. There is an emerging content abstraction scheme in the compressed domain, which is the result of the use of the MPEG-4 compression standard in videos, which is known as object-based summarization [4]. In conventional summarization techniques key frames are used to represent a shot whereas in object-based video coding paradigm, key VOPs are used to abstract the content of a video that views a scene as composition of video objects, corresponding to the bit stream that a user can access and manipulate in a video data. In the literature a key VOP selection algorithm has been

proposed [56]. In this approach, the shape of the VOP is used as a cue for selecting key VOP. The first VOP is selected as the key VOP, and a new key VOP is declared whenever a significant change occurs in the shape of video object. By taking into account the growing availability of video in MPEG-4 compressed bitstream, in the video data model we proposed, the two possible representations schemes such as scene-shot-key frame and scene-VO-VOP are supported. Thus, using the object-based representation scheme of the proposed video data model, the object–based coding of MPEG-4 can be exploited.

2.3.3 Video Representation

Central to visual information retrieval is the representation of perceptual features of images and video such as color, texture, shape, image structure, spatial relationships and motion [18]. Image analysis and pattern recognition algorithms provide the means to extract numeric descriptors, which gives a quantitative measure of these features and computer vision enables object and motion identification by comparing extracted patterns with predefined models.

The representations of visual features such as color, texture and shape of an image have been purposely skipped because it can be looked up in [18] and other abundant references which has been cited in this book. Thus, in this section we only discussed the representation of semantic features of a video and its perceptual feature such as motion, which stands out the most distinguishing feature of video data.

2.3.3.1 Motion Representation

While visual features characterize the content of a still image, in processing video data it is also essential to account for the temporal dimension.

Motion is an important content descriptor of a video that distinguishes it from a still image. It is the main characterizing element in a sequence of frames, directly related to a change in the relative position of spatial entities or to a camera movement [18].

It is obvious that the motion content in a video is the result of either camera motion such as pan, zoom, tilt or object and background motion. Camera motion is known as global motion whereas the second type of motion, which is caused by the movement of an object, is known as local motion. The major concern in motion content-based video indexing and retrieval is mostly the object's motion and not the camera effect [13]. This is because while querying video retrieval systems, users tend to be more interested in change of direction of the objects in the scene, and not in the way camera is being titled, rotated or zoomed with respect to the object. This poses a problem on the assessment of object motion that is true motion of an object cannot be assessed unless the camera motion is compensated for. To this end there are motion analysis and representation techniques developed for the estimation of motions from the camera and an object in the video.

Motion analysis and representation is a widely addressed subject in pattern recognition and computer vision. Methods that compute an approximate estimation of motion follow two distinct approaches [18]. One method takes into account temporal changes of grey-level primitives, from one frame to the following one, and computes a dense flow field usually at every pixel of the image. The second approach is based on the extraction of a set of sparse characteristic features of the objects, such as corners or salient points, and their tracking in subsequent frames.

An approach, which uses the temporal changes of grey-level primitives, is found in [58]. In this approach different motions are expressed accurately by estimating motions from the camera and object. Firstly, the total motion (TM) in a shot is measured; this is achieved by computing the

accumulated quantized pixel differences on all pairs of frames in the shot. Before computing pixel differences, the color at each pixel location is quantized into 32 bins to reduce the effect of noise. Once the total motion has been estimated, each frame in the shot is checked for the presence of camera motion. Pan and tilt of a camera is detected, if present, the amount and direction of camera motion is computed. Thus, Object Motion (OM) is computed by a technique similar to the computation of TM, after compensation of camera motion.

The camera motion can be extracted from either uncompressed or compressed video streams [18]. The former approach is based on the evaluation and analysis of a flow field approximating the motion field pixels between two consecutive frames. The latter is concerned with the analysis of frames of the compressed streams which encode motion vectors.

In the video model we proposed, temporal information is represented at shot level for the scene-shot-key frame representation scheme whereas in the case of scene-VO-VOP representations scheme temporal information of a video is embedded in the VOP.

2.3.3.2 Representation of Content Semantics

Besides the representations of the perceptual features of a video, representing the semantic features of a video is critical for managing video data effectively. The traditional perspective of representing the semantic interpretations or meanings of a video is the identification of semantic primitives such as objects, concepts and events as abstractions of visual content, which can be either through recognition or interpretation [18]. Semantic primitives are high-level concepts or interpretations associated to the content of a video. Solutions for identification of semantic primitives are usually domain dependent [9,13].

As it produces query responses often that are irrelevant, in the literature, formulating a query using perceptual features has been blamed for its inefficiencies. This is because good match in the metrics space may not be interpreted as good match in terms of semantics [9]. This calls for a better mechanism or technique of formulating a query using high-level concepts such as objects and events of the video that users' can easily understand and recall. Motivated by the drawback of queries based on low-level visual features, semantic-based retrieval using keywords like annotations was proposed. At this stage, automatic detection of semantic concepts or events is not possible, thus, it is only possible to have video retrieval systems that support semantic-based retrieval using manual (or hand-coded) or semi-automatic content descriptions. Since hand-coded descriptions or manual annotations of video content are often imprecise and time taking, this impacts the efficiency of semantic-based retrieval. Nevertheless, semantic features are essential in augmenting content-based or perceptual features-based retrieval of video data. Though, automatic mapping from the raw video data to visual features has been achieved, automatic mapping from the feature to semantics or high-level concepts is not yet achieved and remains as a challenging problem [10].

2.4 Video Browsing

As video is rich and complex media, formulating visual feature-based query to retrieve a portion of a video, which is of subject of interest, is most of the time difficult. Thus, in a multimedia environment browsing has additional importance and queries can be based not only on exact matching but also on degrees of similarity and relevance feedback [9]. However, structuring the content of a video into meaningful units for efficient browsing and retrieval is not an easy task. This is mainly attributed to the nature of video data- video is lengthy and unstructured in format. The key frame or shot-based browsing or accessing of video data cannot discern semantic meanings unless they are purposely organized into scenes. Thus, the significance of a coarse video data unit beyond a key frame or a shot is just apparent for efficient retrieval and browsing. However, the research efforts [25,50,51] towards creating a better access unit beyond a shot such as a scene have proved to be successful for constrained domains only.

In the literature depending on the video unit considered, video browsing has been approached at four different levels such at key frame level, at shot level, at group-based level and at scene level [5]. In the shot and key frame based approach the raw video stream is first segmented into a sequence of shots with the help of automatic shot boundary detection techniques. Accordingly, key frames that are representative of shots of the video are extracted from each shot. In this approach a user can access the content of the video by browsing through a sequence of key frames. While this approach in contrast to the raw or unstructured video data offers much better accessibility option for a user, the semantic meaning of the video data cannot be fully discerned from the collection of key frames. The group-based approach is a bit at a higher coarse grain than the key frame or shot-based video structuring; in this approach semantically related shots are merged into groups, and a browsing hierarchy can be constructed based on the newly derived

unit [59,60]. The proposed key frame or shot-based and group-based hierarchical structuring of video are not the unit of users' interest, this calls for a better video structuring unit which is of at a higher coarse grain, such as a scene-based approach. In this approach video browsing is based on at a higher coarse grain of video unit, such as a scene. In [5], scene-based browsing is proposed, to this effect a scene structure construction algorithm using a two steps process is carried out. In this approach similar shots are collected into groups using time-adaptive grouping and semantically related groups are merged into scene. The advantages of the scene-based approaches proposed in [5] over the other approaches discussed in this section are: in the scene-based approach the video elements or entries generated or composed for presentation are not huge in number as compared to shots, key frames, and groups-based approaches; and shots, key frames, and even groups mark only physical boundaries of a video (convey only physical discontinuity), whereas scenes mark semantic boundaries of a video (convey semantic discontinuity such as a scene change in time and in location).

2.5 Video Retrieval

Video is characterized by voluminous, thus, at this stage, retrieving and browsing of a portion of the video is only possible by processing the information contained in a video data and creating an abstraction of its content in terms of visual attributes. Thereafter, any query operations deal solely with this abstraction rather than with the raw video file.

In the literature content-based video retrieval has been approached in two ways: the first approach is using perceptual features of a video such as color, texture, shape, and motion; and the second approach is based on free text descriptions. Most video retrieval systems support retrieval according to manually annotated descriptions. The first approach imposes new requirement in querying a video data: similarity-based matching whereas querying video data

using the second approach is amenable to traditional data retrieval and data manipulation techniques, simply put just it is text processing.

In this section we present the requirements of multimedia environment pertinent video retrieval.

2.5.1 Content-based Retrieval Methods

Though still image-based queries are not appealing, due to the limitations of existing tools and techniques, content-based video retrievals based on still image is prevalent in most research prototype and commercial video retrieval systems. Therefore, we limit our discussion of CBVR methods in this framework only.

In CBVR the notion of exact match is not well defined, instead similarity-based matching is more meaningful. In similarity-based retrieval for a given query key frame (image), a CBVR algorithm searches a key frame or a set of key frames in the target database, based on a similarity metrics. In order this approach to work the feature vector representation of the images, the similarity metrics, and the similarity search algorithm to be used must be determined first. The feature vector representation of the query key frame and the set of key frames to be searched for similarity in the target database are required to be formed with compatible feature measures. During query time, the similarity of two key frames (images) is defined as the proximity of their feature vector representations in a feature space. As the vector representation of key frames (image) is high dimensional, first the features vector dimension must be reduced for the purpose of retrieval efficiencies and avoiding computation complexities. The most common retrieval methods used are the k-NN (k-Nearest Neighbor)[61,62,64] search and the Range Query search [63]. The former method searches the k similar images (or key frames), where k is a positive

integer. The latter method searches all the images (or key frames) in the target database that are within a given radius to the feature vector space of a query image (or key frame).

2.5.2 Querying a Video Retrieval System

Query formulation in a multimedia environment brings additional challenges that are not present in conventional database managements systems. This complexity stems from the nature of the media itself. Moreover, conventional SQL-based languages are not capable of offering adequate support for query formulation. Therefore an appropriate query language that exceeds the power of these traditional languages is needed.

A retrieval system is only effective if it can return a query response of a user's subject of interest. To effectively query a video data several query paradigms have been researched, the prevalent ones are: Query By Example (QBE), Query By Keyword (QBK) and Query By Sketch (QBS). Query By example could be either the system generates a random set of examples and the user selects one out of these examples or the user supplies the system with an example directly. QBE will no more be of helpful if a user has no a seed (an example) image. Query By Sketch enable a user to formulate a query by producing a sketch of the key frame (image) that is going to be searched in the target database. The two types of queries such as QBE and QBS are not efficient in terms of returning query results, which are semantically meaningful [9]. Therefore, semantic-based query such as Query By Keyword is used to augment queries based on perceptual features. This query paradigm enables a user to formulate a query using domain specific keywords or vocabularies, which is amenable to text manipulation. However, describing a complex media such as a video using keywords is time consuming and prone to imprecision. Thus, the design of a CBVR system involves the inclusion of a rich set of tools so as to facilitate the formulation of the types of queries described above.

2.5.3 Relevance feedback

Query formulation in a multimedia environment is characterized by uncertainties. Since query is formulated based on similarity metrics, specifying a one shot lengthy query to retrieve video data is difficult if not impossible. Thus, there should be a relevance feedback mechanism to refine queries iteratively based on query responses. Therefore, for effective CBVR relevance feedback mechanisms are critical.

2.6 Standards Relevant to Video Data Management

A raw video file is very large which presents us two major problems: storage and transmission. Moreover, processing a raw video file, which is huge in content is a time taking and resource demanding activity. Thus, using compression standards the size of a raw video file can be reduced into much more manageable size; in essence the same (or nearly the same) information is represented using fewer data bits. Furthermore, there is a growing trend to use compressed video data as first hand data in application areas such as in video analysis and retrieval, without involving the overhead cost of decoding and encoding. Thus, compressions standard are indispensable tools for effective video data management.

Query standards that enable the management of multimedia data are emerging, not all SQL standards are capable of offering adequate support for querying multimedia data. It is only the recent versions of SQL that possesses multimedia data management capability.

In the subsections that follow we have presented some of the existing and emerging standards that are relevant to video data management.

2.6.1 The MPEG Standards

Video is sheer volume, its storage space requirement is commensurable to its duration, even for a video clip (an interval of video frames) that lasts in a small duration there is high storage space demand. Transmissions and further processing of video data is also constrained by its voluminous. Thus, the use of standards to address problems of video data management is critical. To this effect, the MPEG group has developed several standards such as: MPEG-1, MPEG-2, MPEG-4, MPEG-21 and MPEG-7. The goal of MPEG-1 was to produce VCR NTSC (352 x 240) quality video compression to be stored on CD-ROM using a data rate of 1.2 Mbps [37]. This approach is based on the arrangement of frame sequences into group of pictures (GOP) consisting of four types of pictures: I-Picture (Intra), P-Picture (Predictive), B-Picture (Bidirectional), and D-Picture (DC). It is intended for content-based access of video data. Video data available in MPEG-1 compressed bitstream are frame-based and they can be managed under the scene-shot-key frame representation scheme of the video data model we proposed. The aim of the MPEG-2 was to produce broadcast-quality video compression and was expanded to support higher resolutions including High Definition Televisions (HDTV). The MPEG-2 compressed video data rates are in the range of 3 to 100 Mbps [38]. Similar to the MPEG-1 compression standard, video data available in MPEG-2 bitstreams are frame-based and they can be managed under the scene-shot-key frame representation scheme of the video data model we proposed. MPEG-4 (Content-based Video Coding) covers a wide range of applications and allows object-based coding [39]. One of the main features of MPEG-4 is the provision of a standard ways to represent units of audiovisual content, called media objects and composition of these objects to create compound media objects that form audiovisual scenes [32,34]. It provides support for the composition of audiovisual information and representation of synthetic media in

multimedia environments [18,32]. Moreover, MPEG-4 is a genuine multimedia compression standard that supports audio and video as well as synthetic and animated images, text, graphics, texture, and speech synthesis [32,34]. Thus, in order to exploit the multi-modal behavior of video data, MPEG-4 is the convenient compression standard to use. Its foundation on the hierarchical representation and composition of Audio-Visual Objects (AVO) made it attractive for content-based access. Therefore, in the video data model we proposed a representation scheme such as scene-VO-VOP that takes advantage of the object-based representation of MPEG-4 is incorporated. The MPEG-7, formally named "Multimedia Content Description Interface", is a standard for describing the multimedia content data that supports some degree of interpretation of the information [40]. It is the standardization effort towards setting a standard framework to describe all aspects of the content of multimedia items, including visual features and high-level descriptions of multimedia objects. To this end, it offers a rich set of standardized tools that can be used to describe various types of multimedia content. The standard is organized into eight components [40,69]: reference framework for storage and transport of documents, Description Definition Languages, low-level visual description tools (Descriptors for image and video content), low-level audio description tools, Multimedia Description Schemes (MDS, high-level description schemes and descriptors independent of the media types), reference software, guideline for testing the validity of the MPEG-7 decoder, and finally the extraction and use of descriptions. However, it doesn't cover the methods by which those features will be extracted, or the way in which search engines make use of the features for retrieval.

In summary, video retrieval systems such as research prototype and commercial retrieval systems need to consider features representation compatibility with the MPEG-7 standard of video or video units descriptions. An MPEG-7 instance must be structured according to the rules

of the Data Definition Language (DDL). The DDL is based on XML schema; therefore it is possible to transform video or video unit descriptions data, which are in XML format to a format that can be managed under DBMSs. Thus, by recognizing the key role that MPEG-7 can play in video retrieval efficiency, we have incorporated it as a content descriptor interface for the accurate and consistent representation of the external description or metadata of a video and its constituent units. MPEG-21 is a standard focusing on the definition of a normative open framework for multimedia delivery and consumption for use by all the stakeholders in the delivery and consumption chain [41]. It is a standardization effort towards providing content creators, producers, distributors and service providers equal opportunities in the MPEG-21 enabled open market. It enables content consumers to gain access to a large variety of content in an interoperable manner.

2.6.2 Query Language Standards

SQL, the Standard Query Language is the initiative work of IBM where its introduction dated back to more than three decades. All standards of SQL including the latest versions of SQL standards such as SQL-92 and SQL-99 are subjected to revisions [68]. In every revision there are enhancements to existing functionalities of SQL, however, the basic conceptual framework such as SQL grammar remains the same. The goal of SQL standard is to enable the portability of SQL applications across conforming products [68]. There are many initiatives targeting the further enhancement of the SQL standards, one such effort is the provision of universal data access to the diversified datasets existing in a distributed manner, such as the Internet. The SQL standard before it reaches to the level what is today it has undergone several revisions, the series of revisions that the SQL standard has undergone can be found in [68]

In the paragraph that follows an overview of the SQL 99 standard (also known as SQL 3) is presented.

Among the various features the SQL 99 (SQL 3) standard offers, the accommodation of complex data types powers database management system vendors to increase the range of application that their product support. The SQL 99 is the superset of SQL 92 and offers upward compatibility support [68]. Its major features are: Object-Relational Extensions such as user-defined data types, reference types (that is, support for object identity), collection types such as arrays and large object support such as LOBs (intended for multimedia data management support), triggers, stored procedures and user-defined functions, recursive queries, OLAP extensions such CUBE and ROLLUP, SQL procedure constructs, Expressions in ORDER BY, Save points and Update through unions and joins.

The object relational extension feature of SQL 99 has greatly increased the modeling power of database management systems while increasing the range of applications supported by them, and it also enables the integration of object oriented and relational concepts in a single language.

Thus, it is the object relational extensions of the SQL 99 standard that makes possible the storage and content-based access of complex data types such as image, audio and video under a DBMS environment.

SQL-99 standard is currently integrated into commercial database systems such as Oracle, DB2 and Informix. It is this feature of the SQL-99 standard that enabled us to demonstrate our proposals in the Oracle database server, that is, SVAMS (Soccer Video Archive Management System), the prototype system we developed, uses Oracle as a video database server.

The Object Query Language (OQL) was introduced by the Object Database Management Group's (ODMG). It is a query language standard that supports the Object Database Management Group's data model [70]. OQL is a seamless extension of SQL. In other words a query expressed in SQL is also a valid OQL query [8]. However, SQL queries can only access "flat" relational tables. In contrast, objects may have a nested structure, as well as include fields that contain the collection types-sets, lists, and bags. OQL provides facilities to access such data types as well.

2.6.3 Content-Based Retrieval System

Video retrieval systems that support content-based retrieval can be classified into: commercial and research prototype systems. However, nowadays, as compared to the research prototype retrieval video systems, only a few content-based video retrieval systems are commercially available. The most known ones are IBM's QBIC, Virage Video Engine and Visual Retrievalware by Excalibur Technologies Corporation [18]. QBIC offers both content-based retrieval for still images and visualization for a video. The functions it offers for video are: video content analysis such as automatic segmentation and extraction of shot key frames; querying by example: an image in a query is compared with key-frames in the target database; querying by content; shot content visualization using mosaic. The QBIC retrieval engine currently runs under IBM OS/2 and UNIX.

The Virage Video Retrieval Engine supports querying of still images and video streams. For a video, it offers functions such as video content analysis (i.e. automatic segmentation and extraction of shot key frames), querying by content using sound, scripts and captions. Currently Informix and Oracle have integrated the Virage retrieval engine into their database systems [18].

Some of the most known content-based research prototype video retrieval systems are VideoQ [74], Jacob [72,73], and OVID system [71].

The VideoQ system is developed at the Columbia University Center for Telecommunication Research, it supports retrieval of still images based on visual features such as color, texture, shape and spatio-temporal relationships. The Jacob system is developed at the university of Palermo, which allows retrieval of video segments, by motion similarity. In the OVID prototype system, video is structured as a set of video objects according to object-oriented concepts.

2.6.4 Database Management Systems that Support CBVR

Due to the introduction of the recent SQL standard SQL99 (or SQL-3) and the possibility of integrating third party CBVIR (Content Based Video and Image Retrieval) modules into DBMSs, currently commercial DBMSs have started providing image and video data management under a DBMS environment. Among the widely known commercial DBMSs, DB2 of IBM, Oracle, and Informix have integrated CBVIR capability into their DBMSs. Our prototype system which we call **SVAMS** has been developed under Oracle OR-DBMS environment, thus we will limit our discussion on functions that are available for video data management in the Oracle OR-DBMS. As it has been indicated in the video data repository model we proposed (see chapter 4), Oracle supports additional data types that enables the storage of multimedia objects such as audio, video and image under the transaction control of Oracle DBMS or as reference to external file that can be managed by the underlying operating system. Oracle starting from its 8i version has introduced multimedia data management features into its DBMS, and in its latest release it has incorporated the multimedia data management module, which is known as ”interMedia”, which is the product of Virage Corporation. The interMedia module of Oracle uses automated image features extraction, object recognition, and similarity-

based image comparison techniques to manage images by their content in addition to by textual descriptions. Though Oracle's interMeida support for image data managements has reached to a reasonable level of maturity, the functions it offers for video data management are still at infancy level. For instance its annotator supports limited video compression formats and content-based video retrieval is restricted to similarity-based retrieval based on still image.

2.7 Summary

Video is complex and content rich media. Knowing the behavior of video data is prime importance towards the design of efficient and effective video retrieval systems. Video data modeling deals with the issue of representing the video data, that is, it deals with the design of the high-level abstractions of the raw video data that facilitates various video operations such as video retrieval and browsing. Traditional data models are proved to be insufficient in representing and describing video data. Thus, a robust video data model that surpasses the power of traditional data modeling techniques is highly needed. Video data models proposed in the literature can be classified into two categories such as generalized and domain-specific. Generalized models can be extended to the context to be represented whereas domain-specific models are only tailored to a specific application domain. Motivated by the drawbacks of existing video data models, in this thesis a generic video data model that entertains both the traditional frame-based and the object-based representation schemes is proposed. In the proposed video model, metadata embedded in a video, which in many cases overlooked by existing video models has also been modeled, classified and represented along with the video data. Thus, revealing the key role that metadata can play for retrieval efficiency is one of the pecularities of the video data model we proposed. Video data repository models, which are mirror images of the proposed video data model, which can be used in the context of OR-DBMS is also proposed. In

addition, on the basis of the proposed video repository models, a similarity-based video retrieval technique, which exploits the features of OR-DBMS, is proposed.

The next logical phase that follows video data modeling is video content analysis. Video content analysis deals with the signal processing part of video streams. To effectively retrieve video data, the video stream must be segmented into meaningful and manageable units. To this end, segmentation techniques are employed. Existing segmentation techniques allows automatic detection of shots from raw video streams. There have been several shot detection techniques proposed in the literature. The traditional and prevalent techniques are cut (sharp or abrupt) transition detection and gradual transition detection. Cut (sharp or abrupt) detection approach is quite simple in that a shot boundary is declared by determining the boundaries between consecutive camera shots, that is, a shot boundary is declared by checking for camera breaks only. However, it is not the ideal approach. For instance in a soccer application domain, which is mainly characterized by the existence of intense motion, the dribbling of a given player may only results in a single shot as long as the player is not out of the camera view. Traditional segmentation techniques have focused on cut detection, but there has been increasing research activities on gradual boundary detection as well. After a video is segmented into manageable units such as shots, key frames are extracted from each shot to abstract the raw video stream. Key frame is the simplest and widely employed method for abstracting the content of a large video streams. Determining the type and number of key frames that should be captured in order to represent or abstract the content of a shot is a challenging problem, as a result key frame extraction is still active research topic. For video data retrievals to be effective, both perceptual and semantic features must be represented in a feature vector space. Perceptual features such as color, texture, shape and motion of a video are the most widely used content descriptors for

content-based retrieval. Queries based on perceptual features of a video data have problems of returning a query result semantically meaningful. To augment the limitations of perceptual features-based query, queries based on high-level concepts that is semantic-based query is used. As queries in multimedia systems are based on degree of similarity and relevance feedback, browsing has additional importance.

Content-based image or key frame retrieval involves the notion of inexact match. Thus, the notion of exact match is not well defined. The common practice is, for a given query image (key frame) a certain number of similar objects in a target image or key frame database are searched. The most common methods of content-based image (key frame) retrieval methods are the k-NN and the Range Query search.

The manner in which content-based video retrieval and video data management systems operate greatly relies on existing and emerging standards related to multimedia data. We thus made a review of some of the relevant standards such as video compression standards, content descriptor standard and query language standards.

We have also seen some of research prototype and commercial retrieval systems that support content-based video access to video data. Currently there are a few numbers of commercial systems that support CBVR. The functions available for video data management in both research prototype and commercial video retrieval systems are still at the level of infancy. Thus, any contribution in the field that addresses problems of video data management is highly needed.

There are also efforts towards integrating video and image data management into DBMS, currently DBMS vendors such as Oracle, Informix and IBM have integrated these functionalities into their database management systems.

CHAPTER 3

VIDEO DATA MODEL

There have been considerable researches conducted on video data modeling and retrieval [1,2,5,6, 10,12,15,27,28,33]. The dominant approaches that have been employed on video data modeling and retrieval can be classified into three [2], such as segmentation-based approaches (i.e. structuring the content of a video hierarchically into scene-shot-keyframe), annotation-based approaches (i.e. annotating the content of a video using domain specific keywords) and object-based approaches (i.e. representing the content of a video using semantically meaningful video objects). Most of the video models proposed in the literature use a combination of either of the two approaches. One such approach is found in [2]. In this approach a video model that captures the structural characteristics of video data and the spatio-temporal relationships among salient video objects that appear in the video is proposed. Our approach differs from those video models proposed in the literature [2, 20, 33, 36] in that it can accommodate video data available in raw bitstreams and MPEG-4 compressed bitstreams thereby exploiting the object-based coding of MPEG-4.

3.1 The Proposed Video Data Model

Representing a video data hierarchically become a norm in the research community, most if not all video models proposed in the literature are built on top of the hierarchically organization of a video [2, 20, 23]. Structuring a video data hierarchically into its constituent elements smooth out

problems of describing the behavior of video data, in effect the operations that can be performed on the video can be determined (i.e. it generally provides more control of the query formulation). Owing to these advantages, in our video data model we have used the hierarchical organization of a video. In addition to the traditional approach (i.e. the scene-shot-keyframe representation of video data), the object-based representation of a video data is also considered. Thus, the video data model we proposed is organized hierarchically as Scene-Shot-Keyframe for video data available in raw bitstreams, and Scene-AVO-VOP for video data available in MPEG-4 compressed bitstream: taking advantages of the object-based representation of MPEG-4 compressed bitstream.

In other words, in the proposed video model situations such as achievements in MPEG-4 and the recent trend to use compressed video data as first hand data are recognized. Moreover, representation of video data available in the conventional frame-based coding scheme is also one of the main features of the proposed video data model. In the conventional structuring of video data, scene is viewed as a compositions of video shots, whereas in MPEG-4, a scene is viewed as a composition of Video Objects (VO) with intrinsic properties such as shape, motion, and texture [4].

The video data model we proposed is a generic one in that it can be used by unconstrained domains. It is different from those video models proposed in the literature [2, 20, 23, 33], in that it represents external descriptions of a video and its constituent units as its first class citizen (i.e. in the proposed video data model, in addition to the content of a video, its metadata is also modeled, classified and represented), which in many cases overlooked by many existing video data modeling schemes. Moreover, standards which offer a wealth of features such as MPEG-4 and MPEG-7 are introduced in the model so as the object-based coding of MPEG-4 can be

exploited and metadata embedded in the video can be consistently and accurately represented. The foundation of MPEG-4 is on the hierarchical representation and composition of Audio-Visual Objects (AVO) [8,34], it is this aspect of the MPEG-4 feature exploited in the proposed video data model. It provides a standard way to represent units of audiovisual content called "media objects" [39], it also provides a standard way to describe the composition of these objects to create compound media objects that form audiovisual scenes, which are users' unit of interest.

It has been long ago since the role of metadata in multimedia data management such as image and video has been recognized [42]. However, no video data modeling scheme was proposed to reveal the key role that metadata can play in enhancing the performance of video retrieval systems, as a result metadata in video retrieval systems has been used on ad-hoc basis. The proposed video data model addresses this limitation of existing video data models by modeling, classifying and consistently representing metadata of a video in conjunction with the video content. Video is complex; a query formulated using the perceptual features of a video most of the time may not return a query response of user's subject of interest [9]. Moreover, even when content-based search is possible, it cannot be used frequently for performance reasons [42]. This is where metadata model comes in. To get suitable abstractions aids in exploiting metadata, modeling and classifying the metadata is important. In this thesis the metadata classification found in [35] is adopted in the context of video data. Metadata, which typically consists of text-based information that describes the video, is usually embedded within the video using a proprietary format, and is therefore not always easily accessible. To be able to efficiently manage and use metadata, it must be extractable from the video. After extraction there must be a consistent and accurate representation of the metadata, regardless of the original video. This is where the standard content descriptor interface such as MPEG-7 comes in. In the proposed video

model, MPEG-7 is used to alleviate problems of representation (i.e. representation issues such as inconsistencies problems are addressed through the use of MPEG-7 content descriptor interface).

Since February 2002 MPEG-7 has become internationally accepted standard for describing the content of multimedia that supports some degree of interpretation of the media content [40]. This feature of MPEG-7 is considered in the proposed video model for describing the content-dependent, content-independent and video-oriented metadata of a video and its constituent units in a consistent manner.

In the literature it has been reported that retrieval systems that take advantage of the multi-modal characteristics of video perform much better than those that do not [13]. In this context, MPEG-4 is a genuine multimedia system [32], it can be conveniently used to exploit the multi-modal nature of video data: video comprises a sequence of images along with associated audio and in most of the time, text captions. Thus, the proposed video model possesses one of the features of a good data model, that is extensibility [46]. Therefore, the proposed video model can be extended further to exploit the multi-modal behavior of video data; this is because video data available in MPEG-4 compressed bitstreams allows the representation of a video into a set of primitive media objects that can collectively create a set of Audio Visual Objects (AVO) which are of multi-modal by themselves [13].

The anatomy of the proposed video model is presented on the page that follows.

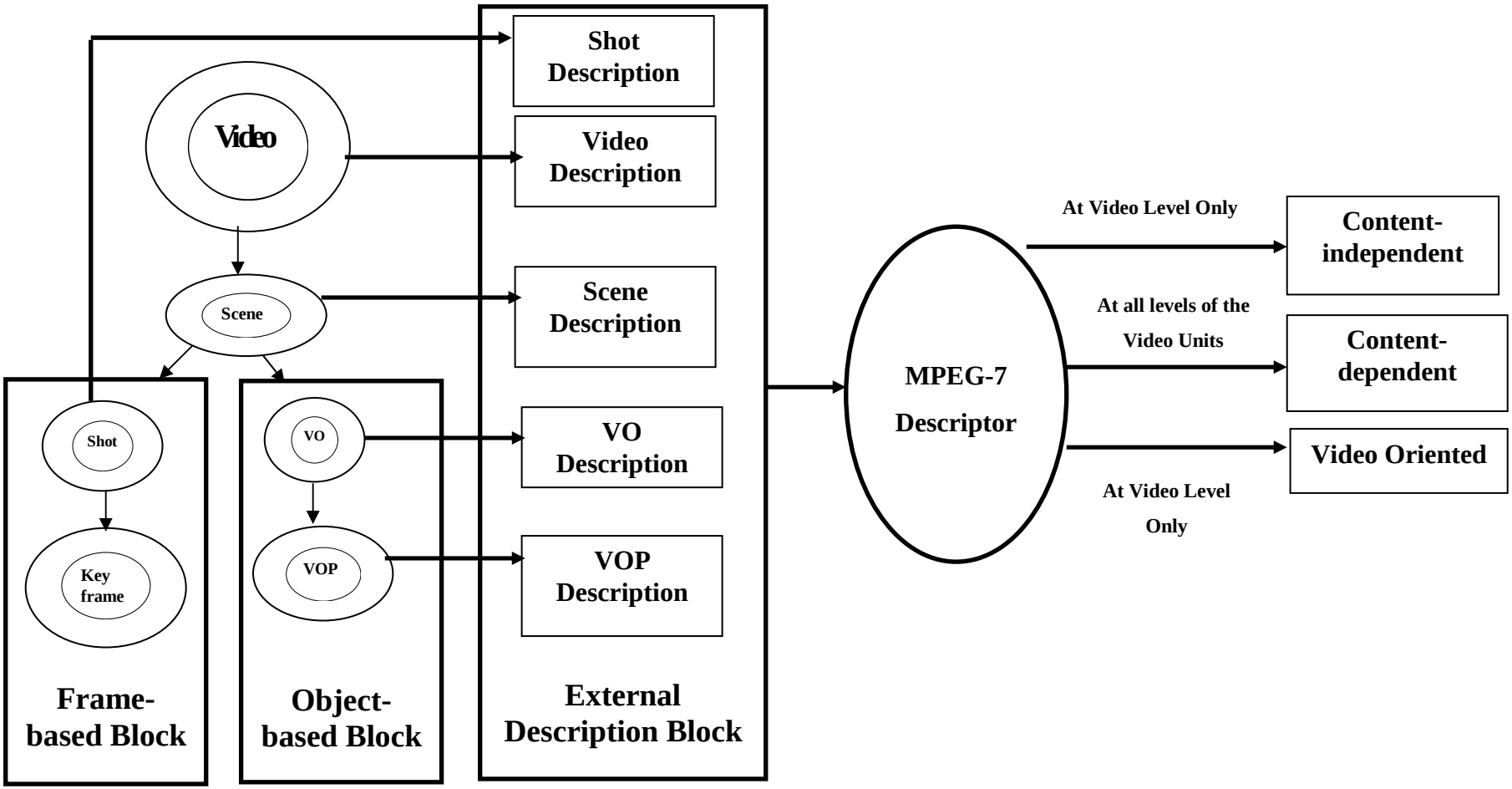

Figure 3.1 The Proposed Video Data Model

Note:

- Double border ellipses are used to signify the multiplicity of a video and its constituent units
- The frame-based scheme is meant to frame-based representation of a video
- The object-based scheme is meant to the object-based representation of a video
- The External description block is meant for a metadata description of video and its constituent units

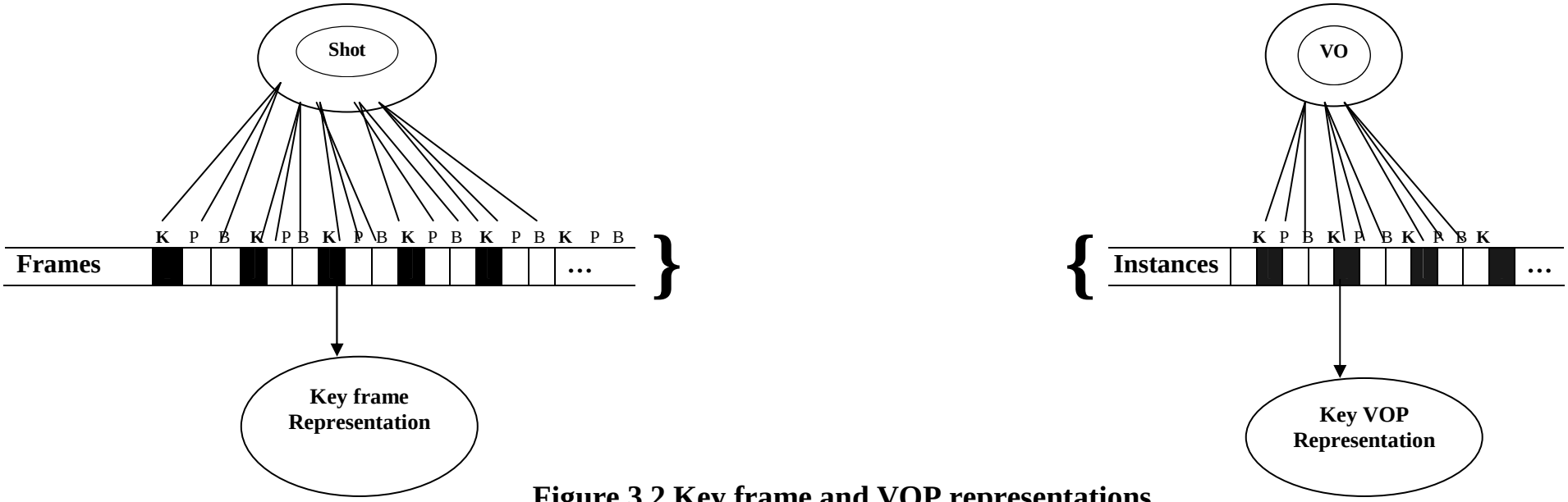

Figure 3.2 Key frame and VOP representations

.

K: designates the key frame(s) used to abstract the content of a shot. **P** and **B** designate the P-frame and B-frame frame types of a shot respectively. A key frame can be represented using an image data model. To this end, the image data model proposed in [19] can be used.

K: designates the key video object plane(s) used to abstract the content of a video object. **P** and **B** designate P-VOP and B-VOP video object plane types of a video object respectively.

Note:

- Double border ellipse signifies multiplicities (i.e. it represents two or more shots, and two or more video objects)

Before we give a detailed description of the proposed video data model, it would be good just to explain a few of the fundamentals, that is, let us first define terms and we then describe the components of the video data model.

Video Objects (VOs): VOs in the model correspond to entities in the bitstream that the user can access and manipulate (cut, copy, paste).

Video Object Planes (VOPs): These are instances of Video Objects (VOs) at a point in time.

As it has been mentioned in the previous paragraphs, the proposed video data model entertains both types of representations schemes: the conventional video representation scheme (or the frame-based scheme) the case where the content of a video is abstracted through key frames and the object-based representation scheme (or the object-based scheme) the case where the content of a video is abstracted through Video Object Planes (VOPs). Key frame is a video unit, which is used to represent or abstract the content of a shot. It is a special type of image that can be used as an index or table of contents of a video. The number of key frames that must be captured to abstract the content of a video shot varies depending on the content complexity of the shot. Thus, for content-based retrieval of video data to be effective, key frames of a video have to be managed effectively and systematically. To this end, data management techniques that have been developed for image data can be used (with or without some extensions) for the management of video key frames. In our video data model of the frame-based scheme (i.e. the scene-shot-key frame representation scheme), key frames can be described and represented using the image model proposed by R. Chbeir et al. [19]. Thus, the physical and semantic features of a key frame can be captured under the context of this image data model. In Chapter 2 (related work), we have discussed the significance of this image data model in relation to the proposed video data model.

In the proposed video model for every video unit an external description is associated, and each external description is classified into content-dependent, content-independent and video-oriented metadata as depicted in the video data model (see the external description block of Figure 3.1).

In the paragraphs that follow a brief description of the components of the proposed video model is presented.

In the proposed video data model, by explicitly representing external descriptions of a video and its constituent units, metadata that should be captured for effective video retrieval is identified. Accordingly, external descriptions such as “Video Description”, “Scene Description”, “Shot Description”, “Video Object Description”, and “Video Object Plane Description” are identified and their consistent representation is addressed through the standard content descriptor interface MPEG-7, which is one of the major components of the proposed video data model. The data of these descriptions are all alphanumeric. Thus, in this regard the proposed video data model can also be used as an important tool in revealing the three classes of metadata of a video that need to be captured for effective and efficient video data retrieval.

A summary of the classifications of metadata that needs to be captured at a video and its constituent units is presented in Table 3.1.

Based on the classification of metadata adopted from [35], a description of the classifications such as content-dependent, content-independent and video oriented metadata (i.e. a description of the external description block of the proposed video data model) is given below.

Content-dependent metadata: consists of alphanumeric data describing the video in a manner specific to the application or subject domain. As the terms used to describe the content of the video have to be chosen in a domain specific manner, in this case issues of vocabularies have

prime importance. For example, in a soccer application, it contains information about events such as fouls committed, goal kick, goal scored, etc. In other words the content-dependent metadata consists of data that are directly or indirectly related to the video. It includes semantic interpretations that integrate high-level descriptions of a video unit of the chosen video representation scheme of the proposed video data model. That is, using application domain oriented keywords the video units such as video, scene, shot and key frame need to be described in the case of the frame-based scheme, and video, scene, VO, VOP need to be described in the case of object-based representation scheme. Metadata description is used to represent high-level concepts of a video, thus semantic features of a video are captured through metadata descriptions. Semantic features can then be exploited during query time to answer queries formulated using high-level concepts. For example, in a soccer application domain the semantic feature can be used to answer queries like: “Give me the great Thery Henry’s goal”.

Content-independent metadata: consists of application-specific data that are completely independent of the video content and have no impact on the video content description. For example, in a soccer application domain, it contains information about name of the field where a match was held, name of the teams played, season, date the match was held etc.

Video-oriented description: this description is meant to the video unit only. A description at this level corresponds to the information that are directly associated with the creation and storage of a video. It also includes information about the format, frame per second, maximum bit rate, etc. of a video data. This description is not required for the elemental units of a video (i.e. by identifying the video to which a video unit belongs to, it is possible to associate this description with the video unit).

Structuring a video data hierarchically into Scene-Shot-Key frame is a norm in many of the research works that involve video retrieval and indexing. Based on this scheme of structuring of a video, in the literature many video data models have been proposed [2, 20, 21, 22]. The following table explains the similarities and differences of the proposed video data model and the DISIMA video data model proposed in [2].

Table 3.1: Comparison of the proposed video data model versus the DISIMA video data model:

Video data model	**Description**
DISIMA	The DISIMA model is an extension of the image data model proposed by the same research group in [29]. Like many of the video data models proposed in the literature it structures video data hierarchically into Scene-Shot-Key frame, the model is intended for representing video data available in raw video streams. It is based on two approaches: segmentation and salient video object-based. In this model salient video-objects are extracted out of the key frames (images) using the notion of Minimum Bounding Rectangle (MBR), which is merely an approximation technique. What is common between the DISIMA video data model and the video data model proposed in this thesis is the structuring or segmentation of video data into Scene-Shot-Key frame, which is a standard practice in video data modeling.
The proposed video data model	Unlike the DISIMA video model, the proposed video data model offers or supports two types of representations schemes; in this model, video

	data available in a raster format (raw video streams) fits the frame-based representation scheme, in which a video is structured or organized into Scene-Shot-Key frame whereas object-based coded video data (MPEG-4 bitstreams) fits the object-based representation scheme, in which a video is structured into Scene-VO-VOP. The video data model proposed in this thesis is based on three approaches: segmentation, salient video-objects and annotation. In the proposed video data model, the object-based coding of MPEG-4 is exploited to represent and describe objects that appear in a video. To this effect, a separate representation scheme for video data available in MPEG-4 compression format is introduced. In the case of DISIMA, objects are extracted merely by an approximation technique, which is known as Minimum Bounding Rectangle (MBR). In addition the video data model proposed in this thesis reveals the key role that metadata can play in enhancing video retrieval systems by introducing metadata as part of the video data model. Future trends such as introducing new services like making accessible video data on mobile terminals is possible when we have a video data that can be transmitted with low bit rates, which the proposed video data model has readily an answer by exploiting the MPEG-4 features.

Table 3.2: Summary of external descriptions of a video and its constituent units that need to be captured for effective video retrieval.

Video Unit	Content-dependent Metadata	Content-Independent Metadata	Video Unit-Oriented Metadata	Viewed
Video	High-level description of a video based on the semantic information it conveys.	Descriptions given to a video as a whole that has nothing to do with the actual content of the video. Example, title of a video	Meta information related to the creation, storage and format of the video.	As a sequence of Scenes.
Scene	High-level description of a segment of a video (i.e. a scene), based on the semantic information it conveys.	It inherits the description of the video	It inherits the description of the video	As a sequence of Shots or Video objects
Shot	High-level description of a segment of a video (i.e. a shot), since shot is a physical boundary it may not convey semantically meaningful information	It inherits the description of the video	It inherits the description of the video	A sequence of video frames
Key frame	High-level description of a portion of a shot, which is the smallest unit of a video down in the scene-shot-key frame representation scheme of a video.	It inherits the description of the video	Not required	A video unit extracted from a shot
VO	High-level description of video objects.	It inherits the description of the video	It inherits the description of the video	An object that appears in the video
VOP	High-level description of an instance of a video object.	It inherits the description of the video	It inherits the description of the video	An instance of a video object

After a video content is abstracted with the help of content analysis techniques, the content of a video can be represented through its perceptual features. Thus, perceptual features (physical and temporal features) and semantic features (external or high-level descriptions associated to a video unit) enable video retrieval systems to support the following types of queries.

- The Perceptual Features: key frames in the frame-based scheme of the proposed video data model can be represented and described using the image data model proposed in [19], in this model the content of an image is represented using low-level visual features such as color, shape and texture. Similarly key VOP in the object-based scheme of the proposed video data model can be represented using perceptual features such as low-level visual features (i.e. shape or color) and motion. The obvious color descriptors such as color distribution histogram, dominant color, etc. can be used to describe the color feature of a key frame or a key VOP. The use of perceptual features allows answering CBVR queries like: Give me video objects that are similar in color with this query video object or Give me key frames that are similar in color with this query key frame.
- Spatial features: these are features that concern the geometric and topological aspects of salient video objects such as shape and position. The spatial feature is used to identify relations between video objects such as directional (right, left, above, front, etc.) and topological (touch, disjoint, overlap, equal, etc) relations. For example, in soccer applications the spatial relationship between video objects can be used to answer queries like: Give me all the videos in which Thery Henry appears to the left of Roberto Pirese.
- Temporal Features: describe the temporal aspects of the Video Object. The temporal feature is used to identify the temporal relationships such as (before, after, etc.) between

video objects. For example, in soccer applications the temporal relationship between video objects can be used to answer queries like: Give me the videos in which Thery Henry appears before Roberto Pirese.

- Spatio-temporal features: describe the spatio-temporal relationships among salient video objects. For example in soccer applications, the spatio-temporal relationships can be used to answer queries like: Give me all the videos in which Thery Henry appears on the left of Roberto Pirese before Patric Viera appears to the left of Roberto Pirese.

As it has been depicted in the proposed video model (see Figure 3.1), the standard content descriptor interface MPEG-7 is incorporated in the model for the purpose of describing and representing metadata consistently and accurately. Accordingly, the MPEG-7 descriptions of the metadata (or content), which are captured through the proposed video model provides the semantic features that enable users to formulate queries using high-level concepts. The MPEG-7 descriptions of metadata of a video, which can be captured and classified through the proposed video data model, are presented below:

- Information describing the creation and production processes of the video content (director, title, etc.), this description with respect to the metadata classification shown in the proposed video data model is classified as content-independent metadata,
- Information of the storage features of the video content (storage format, encoding), this description with respect to the metadata classification shown in the proposed video data model is classified as video-oriented metadata,
- Structural information on spatial, temporal or spatio-temporal components of the content, this description with respect to the metadata classification shown in the proposed video data model is classified as content-dependent metadata,

- Conceptual information of the reality captured by the content (objects and events, interactions among objects), this description with respect to the metadata classification shown in the proposed video data model is classified as content-dependent.

All these descriptions can be coded in an efficient way for searching video data effectively and efficiently.

Though the proposed video model is so generic and can be used in unconstrained domain. In this thesis, its practicality is demonstrated in a specific application domain such as in a soccer video archive management application.

3.2 Summary

Video is complex and content rich. It is complex because it can be interpreted differently depending on the viewer's perception, the application domain, and the context in which it is going to be used. Thus, there is a pressing need to have a video data model, which is amenable to the behavior of video data. Unlike traditional data models a video data model should facilitate capturing of the structural characteristics of video data, semantics or high-level concepts of a video. Moreover, it should also facilitate the representation of a video in a convenient manner so as video data can be effectively and efficiently searchable.

A good video data model is required to possess the properties listed below.

- Besides the units that represent the inherent structural properties of the video, it should also enable the capturing of concepts that represent the video content,
- It should be expressive, generic and extensible,
- It should be capable of capturing the relationship among component objects,
- It should be the pillar towards the design of effective and efficient video retrieval systems.

Most video data models proposed in the literature fulfill the aforementioned properties or requirements partially, whereas the video data model we proposed is fully amenable to these properties. In addition to the provision of support to the traditional frame-based representation scheme, support to the object-based representation scheme for video data available in MPEG-4 compressed bitstreams is also offered. Furthermore, video is multi-modal; therefore, both representation schemes of the proposed video data model can be further extended to exploit the multi-modal characteristics of video data.

Therefore, the pecularities of the proposed video data model lies: firstly it is generic in that it can be used in unconstrained domain. Secondly, besides the content of a video, metadata of a video is modeled, classified and represented. Therefore, the key role that metadata can play in retrieval efficiency has been revealed and exploited. Thirdly, it enables or facilitates similarity-based retrieval of video data in both frame-based and object-based representation schemes. In addition, the wealth of features available in MPEG-4 and MPEG-7 standards can be exploited (i.e. the object-based coding of MPEG-4 can be exploited and the metadata embedded in a video can be consistently and accurately represented as a result of the use of MPEG-7) for enhancing the efficiency of video retrieval systems. Thus, the proposed video data model is a good means to direct the benefits of MPEG-4 and MPEG-7 standards towards video databases.

Moreover, many video data models, which more or less reflect the needs of databases users and developers, have been proposed in the literature, however, none of these proposals were directed towards reaching the benefits of MPEG-7 to OR-DBMSs. In this context our approach to video data modeling has addressed this key issue, that is, it is directed towards mapping the MPEG-7 content descriptor standard into database models through the video data repository models

(which are mirror images of the proposed video data model) proposed in this thesis (see chapter 4).

CHAPTER 4

DATA REPOSITORY MODEL

In the proposed video data model the elemental units of the video and their metadata descriptions that must be captured for managing video data effectively and efficiently have been identified. Once the video data is modeled and its behavior is described, the next logical phase that follows the video data modeling activity is data repository modeling, which can be defined as a conceptual representation of a repository that deals with what data is stored in a DBMS.

The Relational Model is the most widely used model to manage alphanumeric data and has proven to be very successful in supporting business data processing applications [7]. Its wide spread use is the result of its mathematical foundation, the simplicity of its data structure, its solid foundation to data consistency and the set-oriented manipulation of relations it allows.

Though the relational model has been the best studied and the most widely implemented data model, it nevertheless is not without flaws [8]. Some of its drawbacks include: data is organized in the form of relatively "flat" tuples, the schemes of relations are relatively static and relationships that might exist between the content of one table and another relational table must be explicitly encoded through the use of constructs such as integrity constraints.

Therefore, due to its inability to cope up with the fundamental needs of new or emerging classes of applications such as multimedia information systems, a new paradigm, which is known as the object-oriented data model has been brought into use out of these fundamental needs. The

Object-Oriented data model addresses the limitations of relational models by allowing explicit storage and manipulation of more abstract data types and representation of structural application objects [7,8]. These new demand of applications resulted in the introduction of a new generation of DBMS, the OODBMS. But, OODBMS is not as dominant as the RDBMS; it is not widely used, which is a manifestation of its inability to penetrate the DBMS market.

Lack of support for complex data types such as image, video and audio as well as user defined data types in relational models, coupled with the failing of the object-oriented data models to live up to the expectations, the need to have a much more newer model such as the Object-Relational (OR) model became apparent [7]. The OR model combines the best of both worlds. It supports both relational systems and object-oriented systems. Though, at present commercial databases which have been extended to object relational offer limited support for video data management, the OR paradigm is an evolving data model that can conveniently support image, video and audio in a DBMS environment. Therefore, we limit the scope of the proposed video data repository model within this framework.

In this thesis a video data repository model amenable to the video-scene-shot-keyframe sequence is proposed. A video repository models amenable to the scene-VO-VOP sequence can be addressed in a latter work. A brief description of the repository contents along with its implementation under an OR-DBMS environment is presented. Furthermore, its association with the proposed video data model is also described.

4.1 Multimedia Repository Models

By definition a table is not a relation, for a table to be a relation, its columns (attributes) must only have atomic values. That is, any relation is at least in first normal form. In this section the

term “table” is synonymously used with a relation in first normal form and the term column is synonymously used with attribute. Traditionally every attribute in a relational table holds alphanumeric information and is an instance of an entity. Most operations on such type of tables involves exact match, whereas a table that contains an image or a video column is quite different from the traditional relational tables that are widely used in many classes of business applications. Here, additional attributes that describe the image or video characteristics either through the use of keywords or low-level visual features are required. Therefore, attributes are not only used to describe the instance of an entity but also to describe the image or video characteristics, this is because image and video are content rich that need to be described in details. Contrary to the traditional representation of data, when image collections or video collections are stored in a DBMS, their content is represented in a table in terms of content descriptors such as color, texture, shape and motion. Thus, techniques for storing, describing and manipulating images and videos data that exceeds the power of traditional data management techniques available in conventional database systems are required. This clearly shows how developing a repository model for a video is challenging, as compared to the traditional alphanumeric data. For effective video retrieval, in a video repository, besides the perceptual feature of the video, metadata representation such as content-dependent (high-level semantic descriptions), content-independent descriptions and meta-information associated to the video need to be captured. Therefore, the design of an effective video repository model should consider the following additional requirements:

a). Video is sheer volume. Thus, storing video data in a table demands huge storage space requirement, which has to be planned in advance.

b). Using video content analysis techniques a video should be structured into its constituent elements such as scene-shot-keyframe. Moreover video content analysis techniques must be employed so as the sheer volume of video data can be represented in a compact and more manageable size.

c). Since video is complex and content rich, video tables must be able to capture features such as image like features: low-level visual features, temporal features and high-level semantic descriptions of the video.

d). The different levels of descriptions of videos require the introduction of new operators and complex data structures for the components (or the attributes) of the tables to be defined.

e). A video repository model should be able to facilitate the requirement for the play back of a video or its abstraction unit such as shot or scene at the time of query response.

In the literature there have been data repository models proposed for multimedia data [2,3,6,36], these repository models need some sort of adjustment to the context they are used to model or represent. The data repository model proposed in [36] is intended for both video and image data whereas data repository models proposed in [2,6] are intended for video data only. However, a generic video data model that can be used under OR-DBMS irrespective of the nature of the problem domain is still lacking. In [36], a repository model for a multimedia object is proposed, accordingly, a multimedia object (MOB) is defined with 6-tuple and can be expressed as $\mathbf{O_{mob} = \{U, F, M, A, O^p, S\}}$, detailed descriptions of the components are found in [36]. In this data repository model, to a limited extent, video operations requirement can be addressed through the $\mathbf{O^p}$ component, which is a data structure or a pointer to multimedia object represented in the hierarchy that can be used for carrying out video operations such as

compositions of the video units or objects at the lower level of the hierarchy. However, besides the support of composition operation which can be to a certain extent associated to a similarity-based selection operation, there is also a similar need to have a repository model that can be conveniently used for more complex operation such as similarity-based joins: similarity-based operation that involves a similarity-based matching of two or more video tables, which this repository model has overlooked.

There has been a repository model proposed for image data [3]. This model is proposed to fulfill the various processing requirements of multimedia applications that need access to images collections hosted in an image repository. It has gone one step further in that besides the identification of the important components of the image data repository model, formalisms of the similarity-based operations that can be used on image databases for performing similarity-based queries and similarity-based query optimization has also been developed.

The image data repository model proposed in [3] is adopted to the video key frames repository model proposed in this thesis. Thus, it is worth mentioning the components that appear in this image data repository model.

In [3], an image data repository model, which is also known as “image table” is defined under an OR paradigm using 5-tuples, which is defined as M = {Id, O, F, A, P). Where:

- Id: is a unique identifier of an instance of M,
- O: is a reference to the image object itself which can be stored as a BLOB internally in the table under the transaction control of the database management system or which can be referenced as an external BFILE,
- F: is a feature vector representation of object O,

- A: is an attribute component that is used to describe the object using key-word like annotations and may be declared as a set of object types,
- P: is a data structure that is used to capture pointer links to instances of other tables associated by a binary operation.

Similar to the multimedia repository model proposed in [36], the above image repository model clearly shows the subject of primary importance for content-based retrieval, which is the image object itself. The image object is described by its feature vector and by all the remaining attributes that are associated to the image.

The image repository model proposed in [3], can be considered as a descendant of the repository model proposed in [36]. This is because both repositories models have components in common, for instance the three components “O”, “F”, and “A” of the image repository model [3], have nearly similar purposes and representations as the components “U”, “F”, and “A” of the multimedia repository model [36] and they can be used to capture sufficient information about multimedia objects. In the case of [3] semantic meanings are captured through its “A” component, whereas in [36], in addition to the “A” component a separate component “M” is included to capture interpretations or semantics of multimedia objects.

Though the repository model proposed in [36] is intended to represent other medias besides image, due to its structural limitation in representing the temporal aspect of a video, in its existing structure it cannot be readily adopted for video data. This is because, representing a video in a repository model needs special treatment, in addition to the image like features temporal aspects of a video must also be captured.

Repository models proposed for video data are found in [2,6], which both are based on the salient features of a frame or a video sequence. In [2], the repository model is constructed based on the hierarchical organization of a video. In this approach a video is first structured hierarchically into scene-shot-keyframe, and the salient contents of a shot is extracted and represented using a keyframe. This video repository model has several components, associated to a video unit. The following are a succinct descriptions of the video repository defined at each level of the video unit [2]:

- The video repository at keyframe level has 6-tupel, and it is defined as

  $KF_i = (Id, R_i, C_i, D_i, SH_i, I_i)$;

- The video repository at shot level has 5-tuple, and it is defined as

  $SH_j = (Id, I_j, KFS_j, SC_i, D_j)$;

- The video repository at scene level has 5-tuple, and is defined as

  $SC_k = (Id, I_k, SHS_k, V_k, D_k)$;

- The video repository at video level has 5-tuple, and it is defined as

  $V_n=(Id, I_n, R_n, SCS_n, D_n)$;

- Furthermore, a motion vector is defined to model the movement of objects that appear in the video. A motion vector has 3-tuple, which is defined as $MV_p = (Rp, Dp, Ip)$.

Where:

- KF stands for a key frame
- SH stands for a shot
- SC stands for a scene
- and V stands for a video

Detailed explanations of the components of the repositories described above are found in [2]. At all levels of the video units the component “D” represents a set of descriptive alpha-numeric data associated to each video unit. However, the model only gives a general description of “D”, it doesn’t clearly identify and classify the metadata that this component is intended to represent at all levels of the video units. In our proposed repository model this issue is fully addressed by modeling, classifying and representing metadata in conjunction with the content of the video at all levels of the video units (see Chapter 3). An object-based video repository model is proposed in [6], in this model objects are identified and classified into static, moving, background and foreground video objects. The properties of foreground video objects are further classified into two categories such as static and dynamic attributes. It defines the video repository at three levels such as at video level, at clip level and at CAI level on the basis of the Common Appearance Interval (CAI) of video objects. Simply put, the video repository is constructed based on three items of interest, such as:

- $Video_i=(VideoId_i, CList_i)$,
- $Clip_j=(ClipId_j, CAIL_j)$,
- and $CAIi=(I, BVOS_i, MVOS_i, SVOS_i, STS_i)$.

Descriptions of the components of the repository models are found in [6]. This video repository model is intended to capture the salient video objects, aimed at enhancing video data retrieval

systems by enabling users to query video data in terms of semantically meaningful objects that appear in a video. This video repository model has no room for capturing metadata of a video. That is, in the repository model a component is lacking for capturing alphanumeric data associated to video units or objects. Thus, in this repository model, the key role that metadata can play for effective video data management has been overlooked.

In both video repository models [2,6], the notion of Common Appearance Interval (CAI) is used to capture the time interval within a video in which video objects appear together. It is intended to answer queries related to spatio-temporal relationships among video objects with in a shot or a video clip. However, in [2,6], video operations are overlooked. Both repository models lack a component that can be used for performing similarity-based selection and similarity-based join operations. In [2], predicates or functions are constructed to carryout video operations, however, these predicates or functions cannot be treated as operators that can be formalized and used for query optimizations. Our approach to the video repository model has addressed issues of video operations using a component in the key frames repository model that can be used for similarity-based selection and similarity-based join operations. In the proposed video key frames repository (which is a rollover of the repository model developed for image data [3]), a data structure “P” is used to fulfill the requirements of video operations such as similarity-based selection and similarity-based join operations.

4.2 The Proposed Video Repository Model

In this thesis a video data repository model that facilitates the management of video data under an Object-Relational (OR) paradigm is proposed. We have only considered the development of a video repository model amenable to the scenes-shots-key frames representation scheme of the

video data model proposed in this thesis. A video repository model amenable to the scene-VO-VOP representation scheme of the proposed video data model can be developed in a latter work.

The video data repository model we proposed is defined at the video and its constituent units, which is a mirror image of the scene-shot-key frame representation scheme of the proposed video data model. That is , the video repository models are constructed at four levels (i.e. they are constructed at the video, at the scene, at the shot and at the key frame levels).

Like the video repository models proposed in [2,6], a repository model that characterizes a video at its constituent units is proposed, unlike in [2,6], in the proposed video repository models, a key frame is captured in its entirety instead of video objects that belongs to a key frame. Though object-based repositories are more appealing than key frame-based repositories, they greatly rely on robust techniques for extracting video objects, which are of subject of interest. The techniques currently employed to extract video objects that appear in a key frame are merely approximations, only the notion of Minimum Bounding Rectangle (MBR) is used. In this context, we have addressed the significance of object-based video data management in the video data model we proposed, by introducing a separate video data representation scheme such as scene-VO-VOP. This representation scheme of the proposed video data model takes into account factors such as the reasonable maturity that compression standards have reached (i.e. MPEG-4 allows object-based coding), and the growing availability of video data in MPEG-4 compression bitstreams which are amenable to object-based manipulation without involving approximation techniques such as Minimum Bounding Rectangle (MBR).

With these considerations, below we propose a video repository through a sequence of definitions:

1. **Key frame**: is a video unit that is selected from a shot to represent the content of a shot. At key frame level, the key frames table is constructed, it has 5-tuple, which is defined as

   KF = (Id, O, F, A, P).

We roll over the repository model developed for image data [3] to the key frames table in its entirety. Here we have used the adage that key frames are special type of images and the tried and tested techniques of image data management can be adopted to key frames with little or no modification. The components of the key frames table have similar description with that of the image repository described in the previous section. Here we highlight the role of "A" and "P" components once again in the context of video data. The "A" component of the key frames table or repository captures all metadata associated to the key frame. According to the metadata classifications shown in the proposed video model (see Chapter 3, Figure 3.1), at the key frame level, the "A" component of the proposed key frames repository model captures content-dependent metadata of a key frame. Although a key frame is not the unit of users' interest, capturing the semantic descriptions of a key frame in the "A" component is important, this is because the key frames table can serve as the video TOC (table of contents), on top of which semantically meaningful portion of a video, which are of subject of interest, can be retrieved and browsed.

Though discussing about the role of the "P" component of the key frame repository is beyond the scope of this thesis, nevertheless, here an overview of the component in general terms is given. The data structure (or the component) "P" in the key frames table or repository has the same

purpose as in the image repository model, here it is introduced for the purpose of carrying out similarity-based selection and similarity-based join operation on videos collections abstracted through key frames extracted from the corresponding video shots using video content analysis techniques. Therefore, operations, which involve comparison of two or more video repositories such as similarity-based join, can be carried out on the key frames tables or repositories that summarize videos collections of video repositories. An example of application domain, which needs this type of video operation is, video surveillance: automatic detection of unauthorized access.

Moreover, this repository, that is, the key frames table is the basis for the similarity-based retrieval technique applied in the prototype system we developed. The similarity-based techniques introduced in this thesis works in such away that a shot or scene that contains a key frame of interest is retrieved using an example query key frame. An example query key frame is looked up in the key frames table, when a matching key frame is found, the shot or the scene containing the key frame can be located in the shot or the scene table and retrieved. Thus, in this regard, the key frames table or repository is used as a video table of contents.

2. **Shot**: is unbroken sequence of frames recorded from a single camera operation, At shot level, the shots table is constructed, it has 4-tuple, which is defined as SH = (Id, KFs, A, I), where:

    - Id: is the unique identifier of a shot;
    - KFs: is the sequence of key frames used to represent the content of a shot,
    - A: is the component which captures content-dependent metadata such as semantic description of a shot and the scene which a shot belongs to; content-independent metadata and video-oriented metadata can automatically be inherited or derived from

the video to which a shot belongs to, thus at shot level, the “A” component is used for capturing content-dependent metadata only,

- I: is the component that captures temporal information such as the start and end time of a shot.

3. **Scene**: is a sequence of shots, which are grouped together to convey the concepts or story. At scene level, the scenes table is constructed, it has 4-tuple, which is defined as

   SC=(Id, SHs, A, I), where:

   - Id: is the unique identifier of a scene;
   - SHs: is the sequence of shots that belong to a scene. Each shot in turn is a sequence of key frames. Shots used to construct a scene,
   - A: is the component that captures content-dependent metadata such as semantic descriptions of a scene, and the video, which a scene belongs to; like in the shots table, the ”A” component of the scenes table is used for capturing content-dependent metadata only. The other classes of metadata can automatically be inherited or derived from the video, which the scene belongs to.
   - I: is the component that captures temporal information such as the start and end time of a scene.

4. **Video**: is a sequence of scenes. At video level, the video collections table is constructed, it has 3-tuple, which is defined as V=(Id, V, A), where:

   - Id: is the unique identifier of a video;

- V: is a reference to the video itself which can be stored as a BLOB internally under the transaction control of the OR-DBMS or which can be referenced as an external BFILE, which is not under the transaction control of the OR-DBMS,
- A: is the component that captures content-dependent metadata such as semantic description of a video, content-independent metadata such as Title of the video, etc. and video-oriented metadata such the compression format of the video, its creation date, etc. In this data repository model all classes of metadata need to be captured.

At video level it is not required to have explicit component in the repository model to capture time information, this is because at video level time information is embedded and its starting time is obvious which is at 0:00.

Although we defined a video as a sequence of scenes, the repository model we proposed at the video unit is not constructed based on this definition. In this regard, our repository model proposed at the video level differs from those proposals made in [2,6]. In [2], a video is constructed as a sequence of scenes, and in [6], a video is constructed as a sequence of video clips which are abstraction of a video scenes or shots. In the repository model we proposed for the video unit, we have used the component "V" to capture a video in its entirety, instead of constructing it from its constituent elements such as from the scenes using composition operations. The proposed video data repository model at a video unit involves much storage space requirement than the other two approaches; a trade off is made on issues such as storage space requirement versus quality, performance and data integrity. Thus, our proposal of a data repository at a video unit differs from those proposed in [2,6] with the following issues:

- Quality issue: quality wise abstracted video is not as good as the original video.

- Performance issue: composing a video from its constituent elements such as from scenes is a time taking and resource demanding activity. Moreover, video operations for composition are emerging they have not fully evolved.
- Enforcing integrity constraints at a video unit is much easier and incurs less cost than on its constituent units such as at scenes and at shots levels.

Thus, the principal axiom of the proposed repository data model at video unit is, give the whole video for a viewer who is interested in it, instead of a video generated from its constituent elements such as from the scenes or shots or key frames which brings up the aforementioned issues.

4.3 Object Types to the Video Repository Models:

Usage Scenario:

In this section a formal definitions of the types of the components or objects of the video repository models are presented. The definitions are independent of the application domain in which the repository models are to be used. The key frames extracted from the video need to be explicitly stored or represented in a feature vector space under OR-DBMS. The key frames repository or table can be used as a video table of contents (video TOC) for locating and playing back of a video scene or shot containing a key frame of interest. As we have explicitly stated it early on in this chapter, in this thesis the video data repository model is proposed for the frame-based representation scheme of the proposed video data model where videos are represented in a raster format. Videos in the frame-based representation scheme are also represented in a raster format under OR-DBMS environment. In addition to the visual low-level features of the video units, the temporal attribute of the shots and the scenes need to be explicitly stored or represented under OR-DBMS. Based on these considerations, we give a sequence of definitions of the object types in the repository model associated to each video unit.

The definition is made on the PL/SQL language, which is compliant with the latest SQL3/SQL99 standard. The types defined in this section can be adjusted to the context they are supposed to represent.

The key frames table can be created using the following SQL statement:

```
CREATE TABLE KF(id INTEGER,
                O Otype,
                F Ftype,
                A Atype,
```

P Ptype,);

Object Types

We present here the object types: Otype, Ftype, Atype, which allow us to illustrate the method of accessing key frames stored in a table. Discussing about the role of the object type Ptype of the key frames repository defined in this thesis is beyond the scope of this thesis.

Otype Object Type

Otype object type that supports the storage and management of image (key frame) data is defined as follows:

```
CREATE OR REPLACE TYPE Otype AS OBJECT
(
        -- TYPE ATTRIBUTES
        ------------------
        source ORDSource,
        height INTEGER,
        width INTEGER,
        contentLength INTEGER,
        fileFormat VARCHAR2(4000),
        contentFormat VARCHAR2(4000),
        compressionFormat VARCHAR2(4000),

        ---------------------
        -- METHOD DECLARATION
        ---------------------
```

---- Only some of the relevant methods associated with the object are listed below

MEMBER FUNCTION getHeight RETURN INTEGER,

MEMBER FUNCTION getWidth RETURN INTEGER,

MEMBER FUNCTION getContentLenght RETURN INTEGER,

MEMBER FUNCTION getCompressionFormat RETURN VARCHAR2,

MEMBER FUNCTION getUpdateTime RETURN DATE,

MEMBER FUNCTION getMimeType RETURN VARCHAR2,

MEMBER FUNCTION getSource RETURN VARCHAR2,

MEMBER FUNCTION getSourceLocation RETURN VARCHAR2,

MEMBER FUNCTION getSourceName RETURN VARCHAR2,

);

Where**:**

- source: the source of the key frame(image) data,
- height: the height of the key frame(image) in pixels,
- width: the width of the key frame(image) in pixels,
- contentLength: the size of the key frame(image) file on disk, in bytes,
- fileFormat: the file type or format in which the raw frame(image data) is stored (JPEG,TIFF, JIFF,and so forth),
- contentFormat: the type of image (monochrome and so forth),
- compressionFormat: the compression algorithm used on the raw frame or image data.

Ftype Object Type

Ftype object is the type that enables content-based retrieval of key frames or image data. The feature vector representation or the signature of the image (key frame) that describes the color, texture, and shape features of the image or the key frame. This data can be stored in a BLOB. Thus, without loose of generality this object type can be defined as follows:

```
CREATE OR REPALCE TYPE Ftype
AS OBJECT
(
F  BLOB,
-----------------------------------------------
-METHOD DECLARATION
----------------------------------------
-Only some of the useful methods associated to the object are listed below:
STATIC FUNCTION init RETURN Ftype,
STATIC FUNCTION evaluateScore(F1 Ftype, F2 Ftype, score VARCHAR2)
                                                RETURN FLOAT,
STATIC FUNTION isSimilar(F1 Ftype, F2 Ftype, score FLOAT, threshold FLOAT)
                                                RETURN INTEGER,
STATIC PROCEDURE generateF(image Otype)
);
```

Where:

- F: holds the feature vector representation of the stored key frame (image) data.

Atype Object Type

Atype can be defined as an object that can capture the semantic description of the key frame using keyword like annotations. It also can capture information about a shot that the key frame belongs to. Its attribute components are standard data types. It is the component that can be tailored to the specific application domain being considered.

Since external descriptions such as content-dependent, content-independent and video-oriented descriptions of a video are textual or alphanumeric data, they can be defined in a separate relational table and can be associated to the table defined based on the video unit using a foreign key component in the Atype object. For the sake of clarity we presented the Atype object structure of both video units such as the Atype object of a shot, a scene and a video here below.

For a key frame, the component “A” can have the following general structure:

```
CREATE OR REPLACE TYPE Atype
AS OBJECT
(
id INTEGER,
cd VARCHAR2(100),
f-key INTEGER,
);
```

Where:

- id: is the unique identifier of the key frame or image object,
- cd: is the content dependent or semantic description of the key frame,
- f-key: is a foreign key element that can be used to establish a relationship between the key frames table and the shots table. This foreign key element enables us to formulate queries like “give me the shot containing the key frame **KF**” or “give me all the key frames contained in the shot **SH**”.

For a shot, the component “A” can have the following general structure:

```
CREATE OR REPLACE TYPE Atype
AS OBJECT
(
id INTEGER,
cd VARCHAR2(200),
f-key INTEGER
);
```

Where:

- id: is the unique identifier of the shot,
- cd: is the semantic description of the shot using domain specific keywords or vocabularies,
- f-key: a foreign key element that can be used to establish a relationship between the shots table and scenes table. This foreign key element enables us to formulate queries like "give me the scene containing the shot **SH**" or "give me the shots contained in the scene **SC**".

For a scene, the component "A" can have the following general structure:

```
CREATE OR REPLACE TYPE Atype
AS OBJECT
(
id INTEGER,
cd VARCHAR2(200),
f-key INTEGER
);
```

Where:

- id: is the unique identifier of the scene,
- cd: is the semantic description of the scene using domain specific key-words or vocabularies,
- f-key: a foreign key element that can be used to establish a relationship between the scenes table and the video table. This foreign key element enables us to formulate queries like "give me the scenes contained in the video **V**" or "give me the video containing the scene **SC** ".

The component “A” can have the following general structure at video level:

```
CREATE OR REPLACE TYPE Atype
AS OBJECT
(
id INTEGER,
cd VARCHAR2(200),
ci VARCHAR2(200),
vo VARCHAR2(100)
);
```

Where:

- id: is the unique identifier of the video object,
- cd: is the content dependent or semantic description of the video using domain specific key-words or vocabularies,
- ci: is the content independent description of a video,
- vo: is meta-information specific to a video such as its compression format (MPEG, AVI, MOV, etc.) .

Since structurally the repository model at the shot and the scene level under the OR-DBMS paradigm are nearly the same, for the sake of brevity we have only presented the object types of the repository model at the shot level.

A PL/SQL statement to create or define the structure of the shot table is presented below.

```
CREATE TABLE SH(id INTEGER,
                KFs KFstype,
                A Atype,
                I  Itype,
               );
```

Object Types

Below we present the object types: **KFsOtype**, **Atype** and **Itype** that allow us to illustrate the method of accessing shots stored in a table.

KFstype Object Type

The KFstype object type that supports the storage and management of a shot, which is composed of sequence of keyrames, it is defined as:

```
CREATE OR REPLACE TYPE KFstype AS OBJECT
(
    -- ATTRIBUTES
    source ORDSource,
    comments CLOB,
    -- SHOT RELATED ATTRIBUTES, ATTRIBUTES THAT CAN BE
    ---INHERITED FROM THE VIDEO TO WHICH A SHOT BELONGS TO ARE ---------
    -----NOT INCLUDED
    numberOfFrames INTEGER,

    -- METHOD DECLARATION
```

---- Only some of the relevant methods associated to a shot are listed below

MEMBER FUNCTION getContentLenght RETURN INTEGER;

MEMBER FUNCTION getUpdateTime RETURN DATE;

MEMBER FUNCTION getSource RETURN VARCHAR2;

MEMBER FUNCTION getSourceLocation RETURN VARCHAR2;

MEMBER FUNCTION getSourceName RETURN VARCHAR2;

MEMBER FUNCTION getNumberOfFrames RETURN INTEGER,

);

Where:

- source: the source where the video data(the video shot) is to be found.
- numberOfFrames: the number of frames in the video data.

Itype Object Type

Itype object is the type that supports access to the time interval at which a shot starts and ends.

This object type is defined as follows.

CREATE OR REPALCE TYPE Itype

AS OBJECT

(

IStart Number,

IEnd Number,

-METHOD DECLARATION

-Only some of the useful methods associated to the object are listed below:

MEMBER PROCEDURE setShotDuration(known ShotDuration IN INTEGER),

MEMBER FUNCTION getShotDuration RETURN INTEGER

MEMBER PROCEDURE setShotStart(known ShotStart IN INTEGER),

MEMBER FUNCTION getShotStart RETURN INTEGER

MEMBER PROCEDURE setShotEnd (known ShotEnd IN INTEGER),

MEMBER FUNCTION getShotEnd RETURN INTEGER

);

Where:

- IStart: holds time information such as the starting time of a shot
- IEnd: holds time information such as the ending time of a shot

A PL/SQL statement to create or define the structure of the video table is presented below.

```
CREATE TABLE V(id INTEGER,
                    V Vtype,
                     A Atype,
                    );
```

Except the differences exhibited in the components of the repository model, the structural definition of a video is similar to the definition that has been made for a shot. In the Vtype object type of a video, time information, that is video duration is embedded, and its starting time is obvious, a video starts at time 0:00. Thus, for a video object no explicit component is required to capture time information.

Vtype Object Type

The Vtype object type that supports the storage and management of a video is defined as follows:

CREATE OR REPLACE TYPE Vtype AS OBJECT

```
(
        -- ATTRIBUTES
        source ORDSource,
        format VARCHAR2(31),
        mimeType VARCHAR2(4000),
        comments CLOB,
        -- VIDEO RELATED ATTRIBUTES
        width INTEGER,
        height INTEGER,
        frameResolution INTEGER,
        frameRate INTEGER,
        videoDuration INTEGER,
        numberOfFrames INTEGER,
        compressionType VARCHAR2(4000),
        numberOfColors INTEGER,
        bitRate INTEGER,

        -- METHOD DECLARATION
        ---------------------
        ---- Only some of the relevant methods associated with the object are listed below
MEMBER FUNCTION getHeight RETURN INTEGER;
MEMBER FUNCTION getWidth RETURN INTEGER;
MEMBER FUNCTION getContentLenght RETURN INTEGER;
MEMBER FUNCTION getCompressionFormat RETURN VARCHAR2;
```

MEMBER FUNCTION getUpdateTime RETURN DATE;

MEMBER FUNCTION getMimeType RETURN VARCHAR2;

MEMBER FUNCTION getFrameResolution RETURN INTEGER,

MEMBER FUNCTION getSource RETURN VARCHAR2;

MEMBER FUNCTION getSourceLocation RETURN VARCHAR2;

MEMBER FUNCTION getSourceName RETURN VARCHAR2;

MEMBER PROCEDURE setFormat(knownformat IN VARCHAR2),

MEMBER PROCEDURE setVideoDuration(knownVideoDuration IN INTEGER)

MEMBER FUNCTION getVideoDuration RETURN INTEGER,

MEMBER FUNCTION getFormat RETURN VARCHAR2,

MEMBER FUNCTION getFrameRate RETURN INTEGER,

MEMBER FUNCTION getNumberOfFrames RETURN INTEGER,

MEMBER FUNCTION getBitRate RETURN INTEGER,

);

Where:

- source: the source where the video data is to be found.
- format: the format in which the video data is stored.
- mimeType: the MIME type information.
- width: the width of each frame of the video data.
- height: the height of each frame of the video data.
- frameResolution: the frame resolution of the video data.
- frameRate: the frame rate of the video data.
- videoDuration: the total duration of the video data stored.

- numberOfFrames: the number of frames in the video data.
- compressionType: the compression type of the video data.
- bitRate: the bit rate of the video data.

4.4 Summary

The video data repository models proposed in this thesis are intended to capture the video itself, its abstraction units such as the key frames, and the constituent units such as the shots and the scenes. Moreover, the perceptual or low-level features representation (in terms of its color, texture, shape, motion), and all alphanumeric data associated to the video units can also be captured. The paradigm shift in the DBMS, which is the OR model has been fully exploited to effectively manage video data under a DBMS environment.

CHAPTER 5

SVAMS (Soccer Video Archive Management System)

Relational databases have proved to be very successful in managing flat data. However, they are not capable of supporting emerging or new classes of applications such as multimedia information systems [8]. This has necessitated a paradigm shift, which is known as the object-relational paradigm. By recognizing this situation a number of traditional commercial DBMSs have been extended to Object Relational DBMSs so as to offer new features such as base type extension, support to complex objects, inheritance, etc. [68]. OR-DBMSs combines the best features of both worlds, it permits systems to be able integrate tools that can be used to manage multimedia data such as image, video, audio, or heterogeneous type of data, in addition to the traditional alphanumeric data management. To demonstrate the practicality of the proposals made in this thesis, we have developed a prototype system called SVAMS (Soccer Video Archive Management System). We have chosen the Oracle OR-DBMS for managing the soccer video archive. The choice is made based on factors such as availability of sufficient resources on the web[3] and the wide spread use of the DBMS product. Starting from its Oracle 8i release, Oracle has introduced multimedia capability under its DBMS. In our prototype system we have used Oracle 9i to manage the soccer video data for the better multimedia features it offers as compared to its predecessors.

[3] URL of Oracle Technology Network (OTN): http://otn.oracle.com/

SVAMS is developed in Java under Oracle 9i environment using the JDBC interface. It uses the video repositories model proposed in this thesis. SVAMS offers an interface that allows the storage and retrieval of a video and its abstraction units with multi-criteria query formulation support. In the sections that follow the functionalities that SVAMS offers and the components used in developing the prototype system are presented.

5.1 The Oracle interMedia Module

Oracle interMedia is a third-party module integrated with Oracle9i in order to store, manage, and retrieve images, audio, video, or other heterogeneous media data in an integrated fashion with other existing enterprise information [31]. It extends the traditional role of Oracle9i such as the safe and efficient management of relational data by adding multimedia data management capability.

Oracle9i provides support for the definition and management of object types. Database applications and plug-in modules written in Java, C++, or in traditional 3GLs can interact with interMedia through class library interfaces, or PL/SQL, and Oracle Call Interface (OCI). The Oracle 9i interMedia provides object types such as ORDImage, ORDVideo, ORDAudio that enables the management of multimedia data under the transaction control of Oracle's DBMS or referencing as external object [31]. The Oracle 9i interMedia provides the ORDSource object type and methods for multimedia data source manipulation. The ORDAudio, ORDDoc, RDImage, and ORDVideo object types all contain an attribute of type ORDSource. Though Oracle's DBMS support to image data management has reached a reasonable level of maturity, the support it provides for video data management is not sufficient. For instance, the metadata annotations facility it offers through its annotator GUI is limited to a few number of compression formats and similarity-based retrieval support is based on still image only. However, in our

prototype system in addition to the metadata-based query support, we have incorporated an enhanced still image similarity-based retrieval support, which enables retrieval or playing back of a video clip, which contains a key frame or a still image of interest. That is in SVAMS, it is possible to retrieve a video shot or scene by formulating a query using an example key frame (or image).

The video data management support that Oracle 9i offers is not matured. Therefore, in the subsections that follow, only the underlying DBMS support for key frame or still image-based retrieval is discussed.

5.1.1 The Similarity-based Comparison

In our prototype system SVAMS, key frames of shots are extracted and stored in the key-frames repository or table proposed in this thesis. Key frames are still images, then the tried and tested techniques of image data management can be readily applicable for the management of key frames [45]. In this regard, what interMedia does behind the scene is, before a similarity-based comparison is performed, every image (key frame) inserted into a repository or table is analyzed and a compact representations of its content is stored in a feature vector, or signature column. The signature contains the global color, texture, and shape information along with their-object based location information to represent the visual attributes for the entire image or key frame. Thus, any query operation deals solely with this abstraction rather than with the image itself. Images are thus compared based on their color, texture, and shape attributes. The positions of these visual attributes in the image are represented by location. Location by itself is not a meaningful search parameter; but in conjunction with one of the three visual attributes, it represents a search scheme where the visual attribute and its location within the image become equally important. Specifically the signature generated by the ORDImage image analysis

contains information about the visual attributes color, texture, shape and location. The feature vector data for all these visual attributes is stored in a signature, whose size typically ranges from 3 to 4 kilobytes. Thus, feature data is a compact representation form of an image data, which is smaller in size than the image data and can easily be managed.

Using one of the querying paradigms such as query by example, images in a database can be retrieved by similarity-based matching. For retrieval of complex objects such as image data, the notion of exact match is not well defined instead similarity-based retrieval is more meaningful. Therefore, matching is based on degree of similarity on the visual attributes and a set of weights for each attribute. The score, which is the relative distance of between two images being compared, is used to determine the degree of similarity when images are compared; with a smaller distance with respect to a threshold value implies a closer match.

5.1.2 How The Comparison Works?

When images are compared for similarity, an importance measure or weight is specified by the user to each of the visual attributes, and interMedia calculates a similarity measure for each visual attribute. Each weight value reflects how sensitive the similarity matching of a given attribute should be to the degree of similarity or dissimilarity between two images. The weight of the attribute determines the relevance or irrelevance of the attribute in the similarity matching operation. A weight value of 0.0 of an attribute implies the irrelevance of the attribute for the similarity matching whereas any value greater than 0.0 implies the relevance of the attribute for the similarity matching. A weight value of 1.0 of an attribute implies similarity matching on that specific attribute only.

The similarity measure for each visual attribute is calculated as the score or distance between the two images being compared with respect to that attribute. The score can range from 0.0 (no difference) to 100.0 (Maximum possible difference). Thus, the more similar two images are in visual attribute, the smaller the score will be for that attribute. In Oracle what is special about image comparison as compared to other image retrieval systems is, the degree of similarity calculation, that is, the degree of similarity is based on not only a single visual attribute instead it is based on a weighted sum, which reflects the weight and distance of all of the three visual attributes [31].

Images comparison operation use a threshold value specified for comparing the weighted sum of the visual attributes against it. If the weighted sum of the distances is less than or equal to the threshold, the images will be treated as similar. Otherwise, the images will be treated as dissimilar. The threshold value is given as a ratio and its value ranges from 0 to 100.

5.2 General Architecture of SVAMS under a DBMS

In order to have a platform independent application, we have implemented SVAMS in Java. Figure 5.1 presents a simplified general architecture of SVAMS. The architecture also shows how a content-based retrieval module can be integrated into existing OR-DBMSs.

SVAMS enable users to get connected into an Oracle 9i database through JDBC. It provides two components: the visual interface and the query manager. The visual interface includes the data-entry and the query interfaces. The details how these components work together is explained right after this section.

Oracle 9i stores rich media in a table along with alphanumeric data using the facility that interMedia offers. A server-side media parser and image processor is supported through the

oracle 9i Java Virtual Machine (JVM). The media parser has object-oriented and relational interfaces, provides format and application metadata parsing, and includes a registry for new formats and extensions. The image processor has additional components such as Java Advanced Imaging (JAI) classes and provides image-processing tools for converting, matching, and indexing images. Through the interMedia Java classes interMedia media objects and alphanumeric data can be managed. Thus, application can work with results sets that contain interMedia columns and alphanumeric data. The interMedia Java classes also enable access to interMedia object attributes and methods invocation. The JMF (Java Media Framework) classes expose interfaces for the acquisitions, processing and delivery of time-based media[4] such as video. A simplified architecture of SVAMS under a DBMS environment in a layered fashion is presented below.

[4] http://java.sun.com

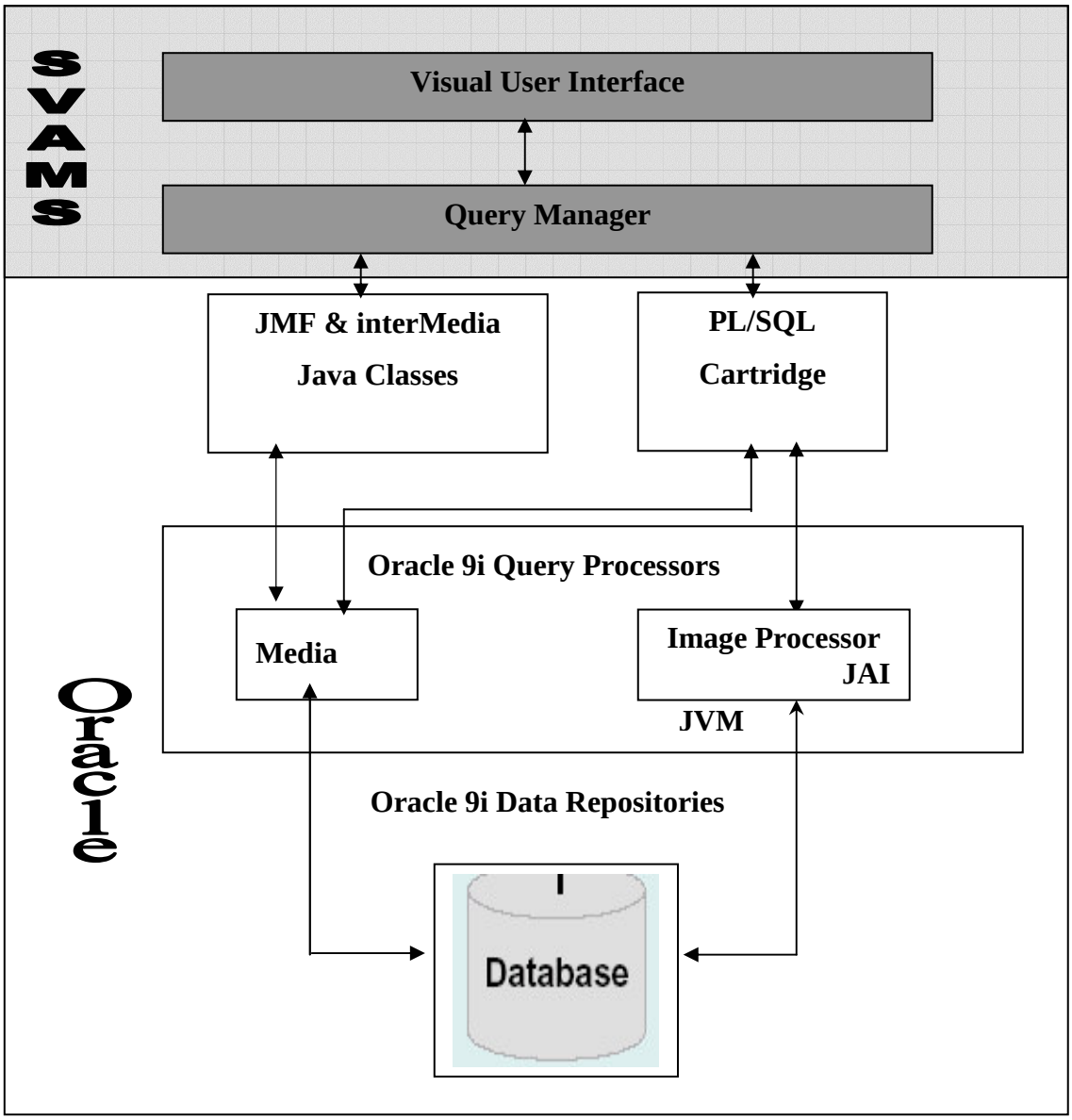

Figure 5.1 The Architecture of SVAMS under a DBMS environment

5.3 The Interfaces of SVAMS

In SVAMS, off-the-shelf automatic segmentation and metadata annotation tool such as VideoAnnex[5] of IBM has been used to segment a video into its constituent elements such as shots, thereby identifying and extracting the corresponding key frames of shots. At this stage, tools that can be used for automatic detection of scenes of a video in unconstrained domain are nearly nonexistent. Therefore, we are forced to identify scenes of a soccer video manually using the video-editing tool, named "AllSplitter"[6]. On the page that follows, the various interfaces that SVAMS exposes is depicted.

[5] VideoAnnext annotation tool available at: http://www.alphaworks.ibm.com
[6] AllSplitter Video editing tool available at: http://www.webmasterfree.com/software/meidatools/Videoeditingtools/

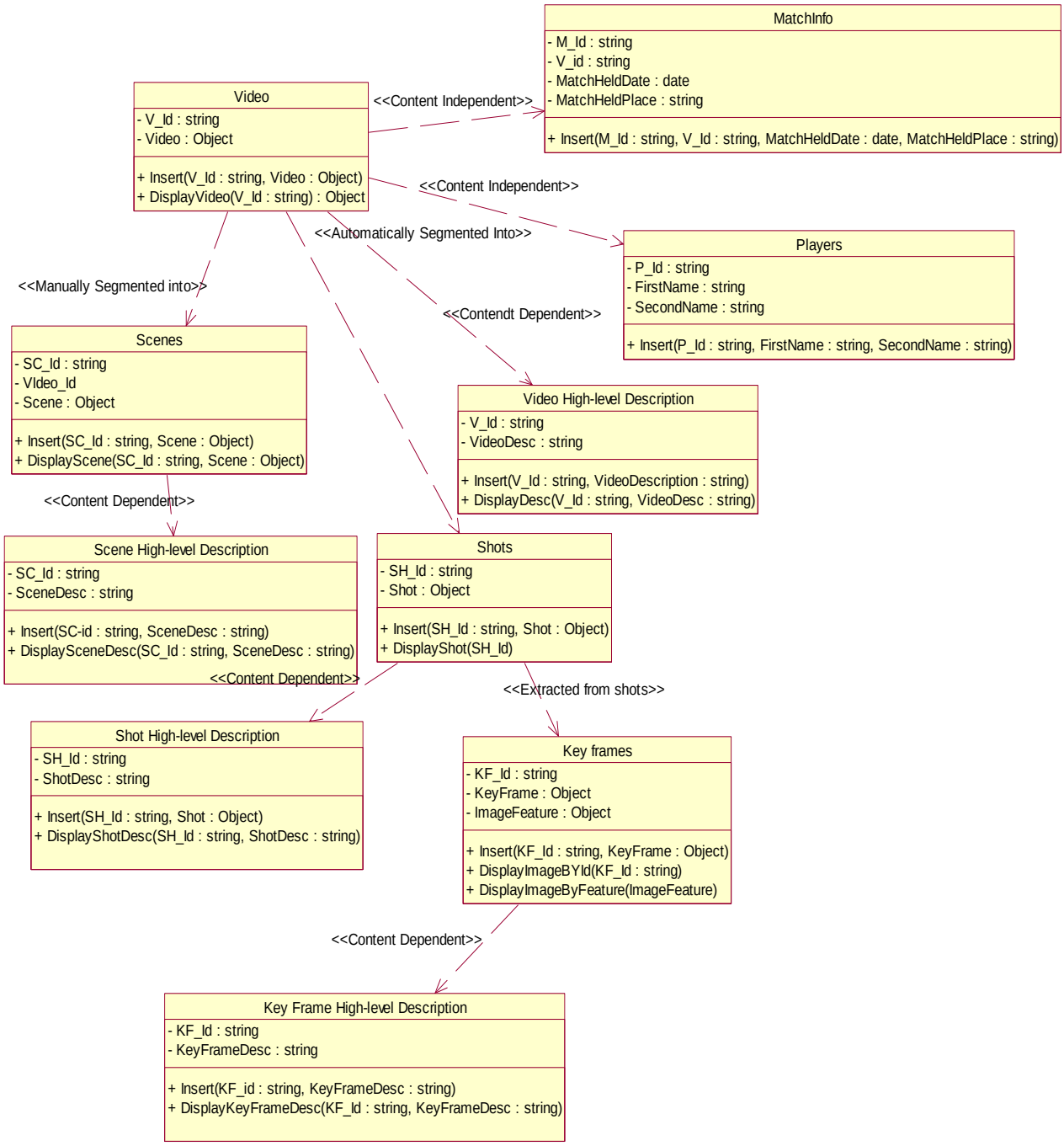

Figure 5.2 The Class diagram of SVAMS in UML

5.4 The Sample Database Used in SVAMS

The video data model and the video repository models proposed in this thesis are generic and can be used in unconstrained domain. However, to demonstrate the practicality of our proposals, we opted a specific application domain such as the soccer application domain. For the sake of simplicity we limit SVAMS to manage soccer funs video data only. The proposed video data repository models defined at a video and its constituent units can also be used in any OR-DBMS environment. A sample database of a soccer application domain, which we used in the prototype system, is presented below.

The tables for the chosen application domain are organized as in the following manner. The tables defined below are associated to the components of the proposed video data repository models defined at a video and its constituent units.

- **Players(P_Id,** FirstName, LastName), information related to a player which is uniquely identified by the Player Id (P_Id) is captured. This table is just traditional alphanumeric data, which is independent of the content of a video.
- **SoccerVideo(V_Id**, S_Video, Format, Title), information related to a video which is uniquely identified by Video Id (V_Id) is captured. This table can be conveniently associated to the video table of the proposed video repository models.
- **VideoDescription(V_Id**, V_Description), information related to the semantic content of a video which is uniquely identified by Video Id (V_Id) is captured. This table holds traditional alphanumeric data, which describes the semantic interpretations of a video. This table can be conveniently associated to the “A” component or object of the video table of the proposed video data repository models.

- **MatchInfo(M_Id**, V_Id, MatchHeldDate, MatchHeldPlace, Season), information related to a match held which is uniquely identified by Match Id (M_ID) is captured. This table holds traditional alphanumeric data, which is independent of the content of a video. This information is required to be captured at video level only; the elemental units of a video inherit this information. This table can be conveniently associated to the "A" component of the video table of the proposed video repository models.

- **Dribbling(SC_Id**, P_Id**,** V-Id, Shots, SceneDuration), information related to a player's dribbling which is uniquely identified by the Scene Id (SC_Id) is captured. Since dribbling conveys semantically meaningful events, this table can be conveniently associated to the scenes table of the proposed video repository models. For example, in this table the scene content of the dribbling's of a known player such as " Ronaldo's dribbling" can be captured.

- **Dribbling_Desc(SC_Id**, Event, SC_Description), information related to high-level descriptions of a player's dribbling which is uniquely identified by the Scene Id (SC_Id) is captured. This table can be conveniently associated to the "A" component of the scenes table of the proposed video repository models.

- **ShortSoccerEvent(SH_Id**, **SC_Id**, KeyFrames, ShotDuration), information related to a shot such as a short soccer event which is uniquely identified by the composite attributes such as Shot Id and the Scene Id (SH_Id, SC_Id) is captured. In this case the primary key is composite; this is because a specific shot may appear in more than one scene depending on the context in which it is viewed. Soccer funs video clips with short duration will be captured under this table. This table can be conveniently associated to

the shots table of the proposed video repository models. For example, in this table the shot content of the penalty shoot of “Thery Henry” can be captured.

- **ShortSoccerEventDesc**(**SH_Id**, SH_Description), information related to high-level description of a short soccer video clip (shot) which is uniquely identified by the Shot Id (SH_Id) is captured. This table can be conveniently associated to the “A” component of the shots table of the proposed video repository models.

- **EventSnapshot**(**KF_Id**,SH_Id, P_Id , keyframe), information related to a snapshot of a specific event which is identified by the Key Frame Id (KF_Id) is captured. Since the “EventSnapshot” table is a collection of still images that portrays the snapshot of a soccer event, this table can be conveniently associated to the key frames table of the proposed video repository models. For example, in this table some emotional aspects of a player in a match held can be captured. This table can also be used as the video’s table of contents in which a scene or a shot with a key frame of interest can be retrieved.

- **EventSnapshot_Desc**(**KF_Id**, KF_Description, DateEventHappend), information related to high-level descriptions of a snapshot of soccer event which is identified by Key Frame Id is captured. This table can be conveniently associated to the “A” component of the key frames table of the proposed video repository models.

Using the database tables described above we demonstrated how effective multi-criteria retrieval support could be realized based on the proposals we made.

5.5 The Visual User Interfaces of SVAMS

The visual user interfaces of SVAMS possess the necessary components, which enables a user to interact with the oracle video server. The current implementation is written to run as Java

application. The visual interfaces that the system offers are: The Soccer Video Data Entry Interfaces designed for the different units of a video such as at video level, at scene level, at shot level and at key frame level, and the Query Interface. For the sake of brevity we have presented selected data entry interfaces of SVAMS. The shot data entry interface is purposely skipped. The specific functionalities that these interfaces offer to a user are described below.

5.5.1 The Soccer Video Data Entry Interface

The Soccer Video Data Entry Interface of SVAMS has three panels such as Video-Oriented data entry panel, Content-dependent data entry panel and Content-independent data entry panel. The designed interface is a reflection of both the video data model and the repository models proposed in this thesis (See section 3.1, Figure 3.1).

5.5.1.1 The Video-Oriented Data Entry Panel

At video unit all classes of metadata are required to be captured for effective video retrieval. Thus, below we presented the Video-Oriented, Content-Dependent and Content-Independent panels of the video visual data entry interfaces.

The Video-Oriented Panel is a visual video data entry interface, used to select a video file and insert into the video table designed to manage soccer videos collections as depicted below.

Figure 5.3 A Screenshot of the Video-Oriented panel of the Soccer Video Data Entry Interface

5.5.1.2 The Content-Dependent Data Entry Panel

The Content-dependent is a visual interface used to enter high-level descriptions or semantics information of a video.

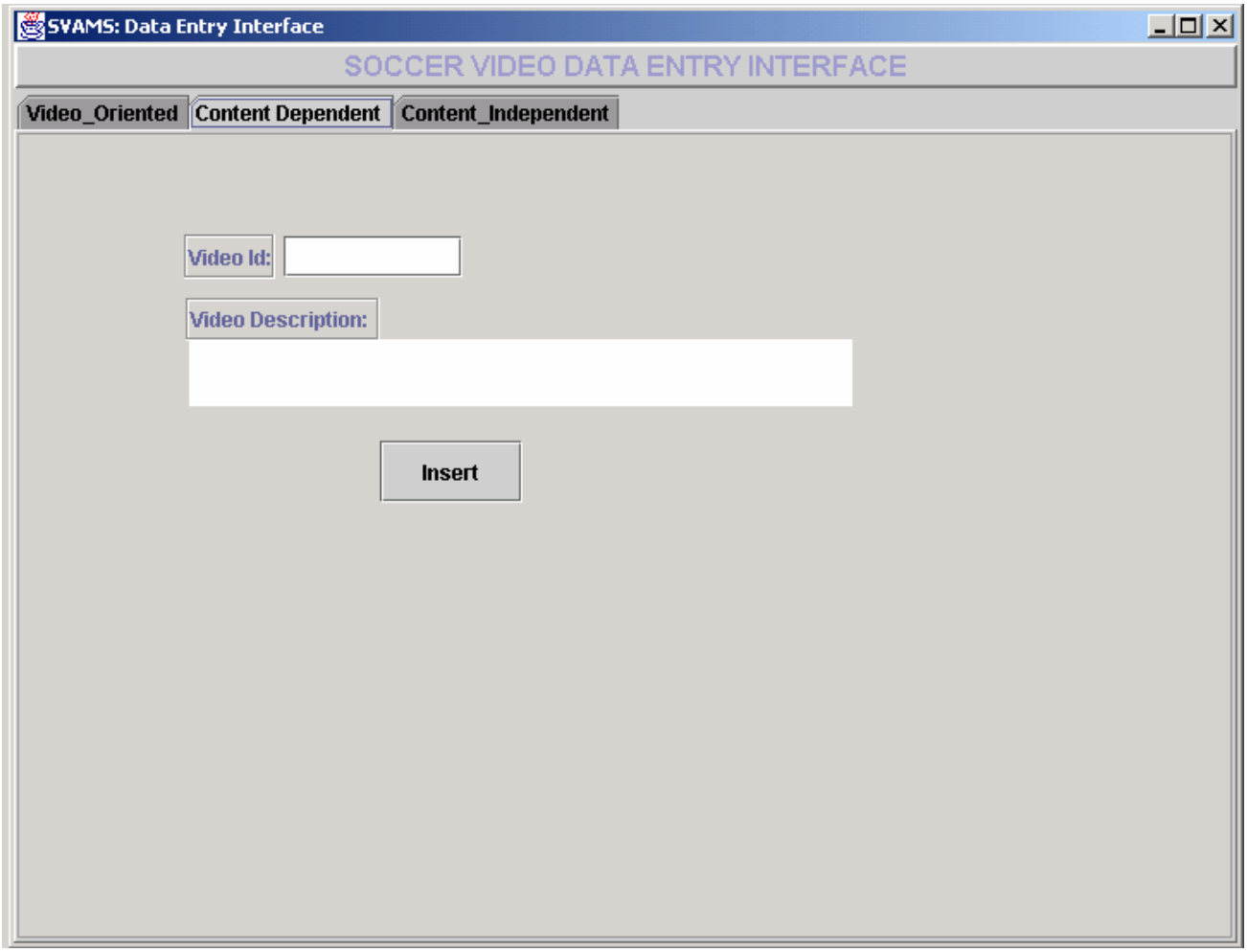

Figure 5.4 A Screenshot of the Content-Dependent panel of the Soccer Video Data Entry Interface

5.5.1.3 The Content-Independent Data Entry Panel

The Content-Independent Panel is a visual interface used to enter information, which are independent of the content of a video.

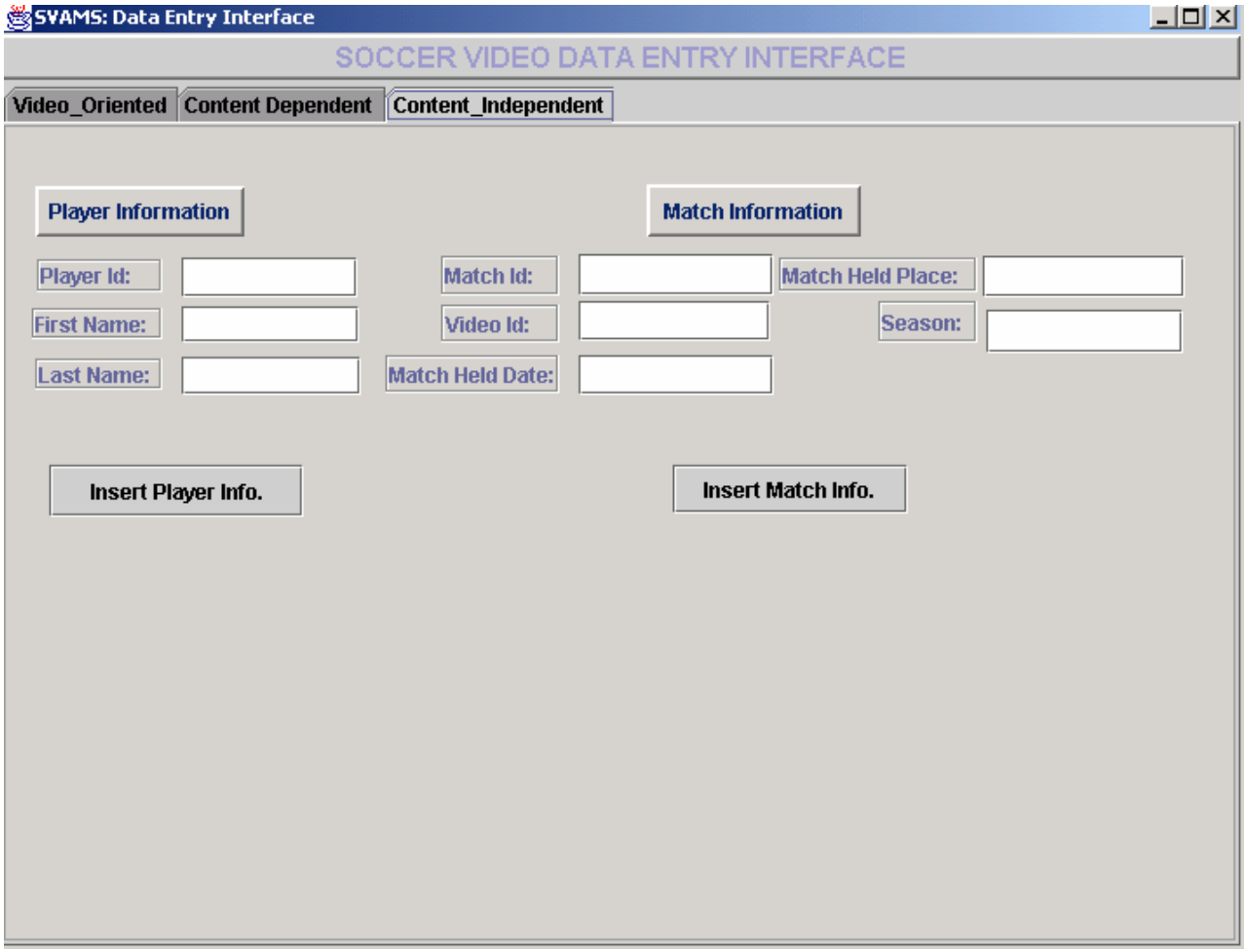

Figure 5.5 A Screenshot of the Content-Independent panel of the Soccer Video Unit Data Entry Interface

5.5.1.4 The Scene-Oriented Data Entry Panel

The Scene-Oriented panel is a visual interface used to browse and select a scene file, which is going to be inserted into the scenes table designed to manage scenes of a video of soccer matches.

Figure 5.6 A Screenshot of the Key frame-Oriented panel of the Soccer Video Key frames Data Entry Interface

5. The Key frame-oriented Data Panel

The Key frame-oriented panel is a visual interface used to browse and select a key frame (or an image) file, which is going to be inserted into the key frames table designed to manage key frames of shot(s) of a video of soccer matches. Key frames in SVAMS play a very important role in that it is used as the table of contents of a video in locating and playing back a video shot or scene of interest, using an example key frame or image.

Figure 5.7 A Screenshot of the Key frame-Oriented panel of the Soccer Video Key frames Data Entry Interface

5.5.2 The Query Interfaces

Our prototype system SVAMS supports two types of queries: QBE (Query By Example) and QBK (Query By Keyword). The QBE interface is a visual panel mainly used to formulate a similarity-based selection query by presenting an example key frame or image as a query key frame or image (see Figure 5.7). It has a browse button to enable a user select a key frame or image stored externally. The QBK interface allows searching by domain specific key word or vocabulary. For the sake of simplicity, in SVAMS we have limited our scope to the management of special events that occur in a football match, below we presented the two main query interfaces of SVAMS.

5.5.2.1 Query By Example Interface

Many commercial video retrieval engines including the Virage Video Engine[7] supports only still-image based video retrieval where the query response may contain only a storyboard (moving or non-moving key frames). In this regard, the query by example interface of SVAMS is different from those commercial video retrieval systems in that it is possible to search for and play back a video scene or shot that contains a specific key frame or image of interest using the OR-DBMSs features. In SVAMS, the following type of query by example is supported. The left pane of the QBE visual query interface displays the key frame or image that can be used as an example image or key frame in order to search for and play back the scene containing the key frame or image in the scenes table of the SVAMS database. The right pane displays the result of the search (a video scene playing), which the key frame or image belongs to. What SVAMS does behind the scene is, it uses the key frame table or repository as a table of contents in order to locate and play back a video shot or scene that contains a key frame of interest.

[7] http://www.virage.com

The query by example works in the following manner, an example key frame is compared with those key frames captured in the target key frame table, and then all similar or matching key frames are retrieved and displayed in a table as shown below. A result set is used to hold the result of the query response which can be saved into another table, depending on the degree of similarity needed the result set can be either empty or can hold several number of rows.

The below query by example can be described textually in the following manner:

“Search and play back a video scene containing the key frame or image displayed on the left pane”

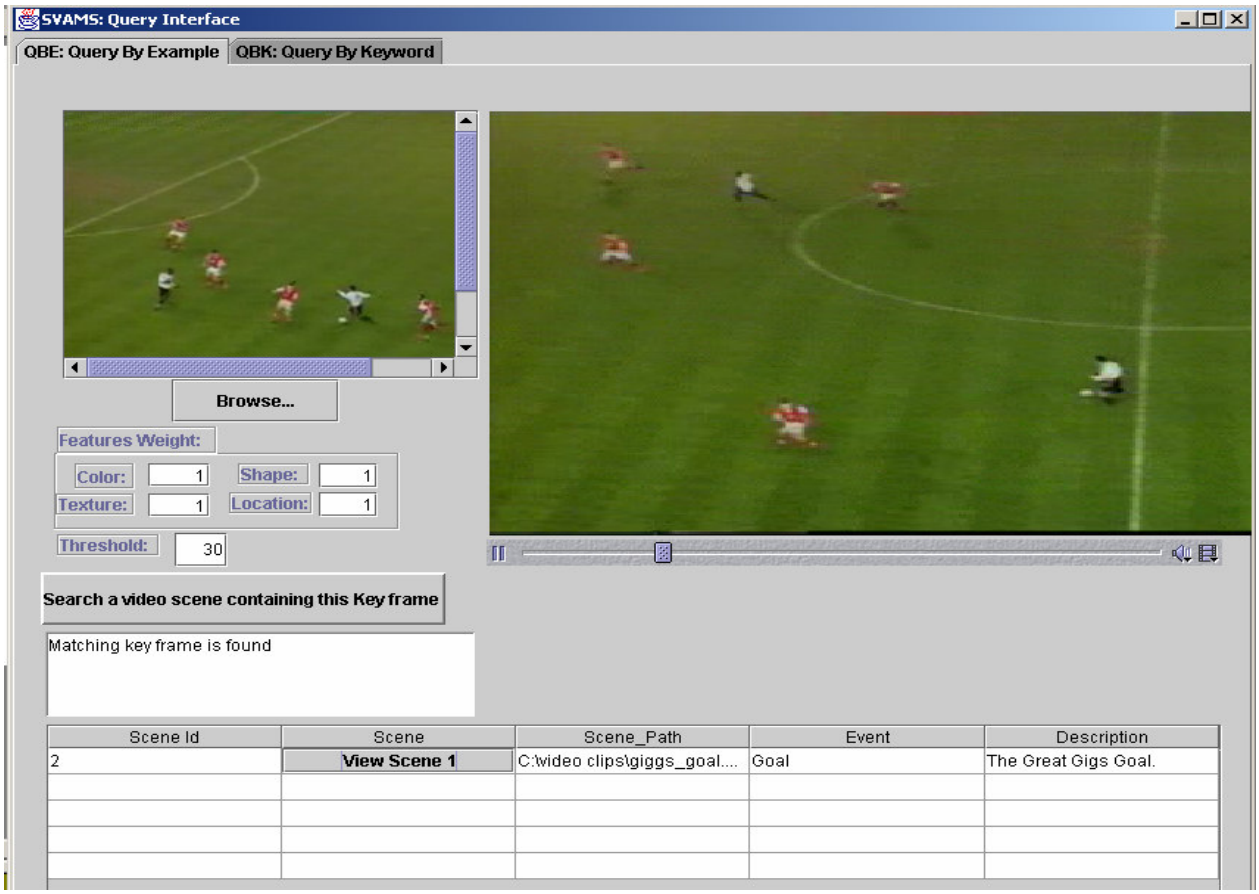

Figure 5.8 A Screenshot of the Query By Example Query Interface

5.5.2.2 **The Query By Keyword Interface**

In this visual query interface a query is formulated using keywords or vocabularies that are specific to the soccer application domain. Thus, beside the similarity-based (or content-based) query, relational query is also supported. That is, the prototype system SVAMS supports multi-criteria query. A query by key word interface is presented on the page that follows.

In SVAMS, an example of the following type of query is supported:

“Search and play back the video scene containing the great goal of Dieago Armando Maradona”

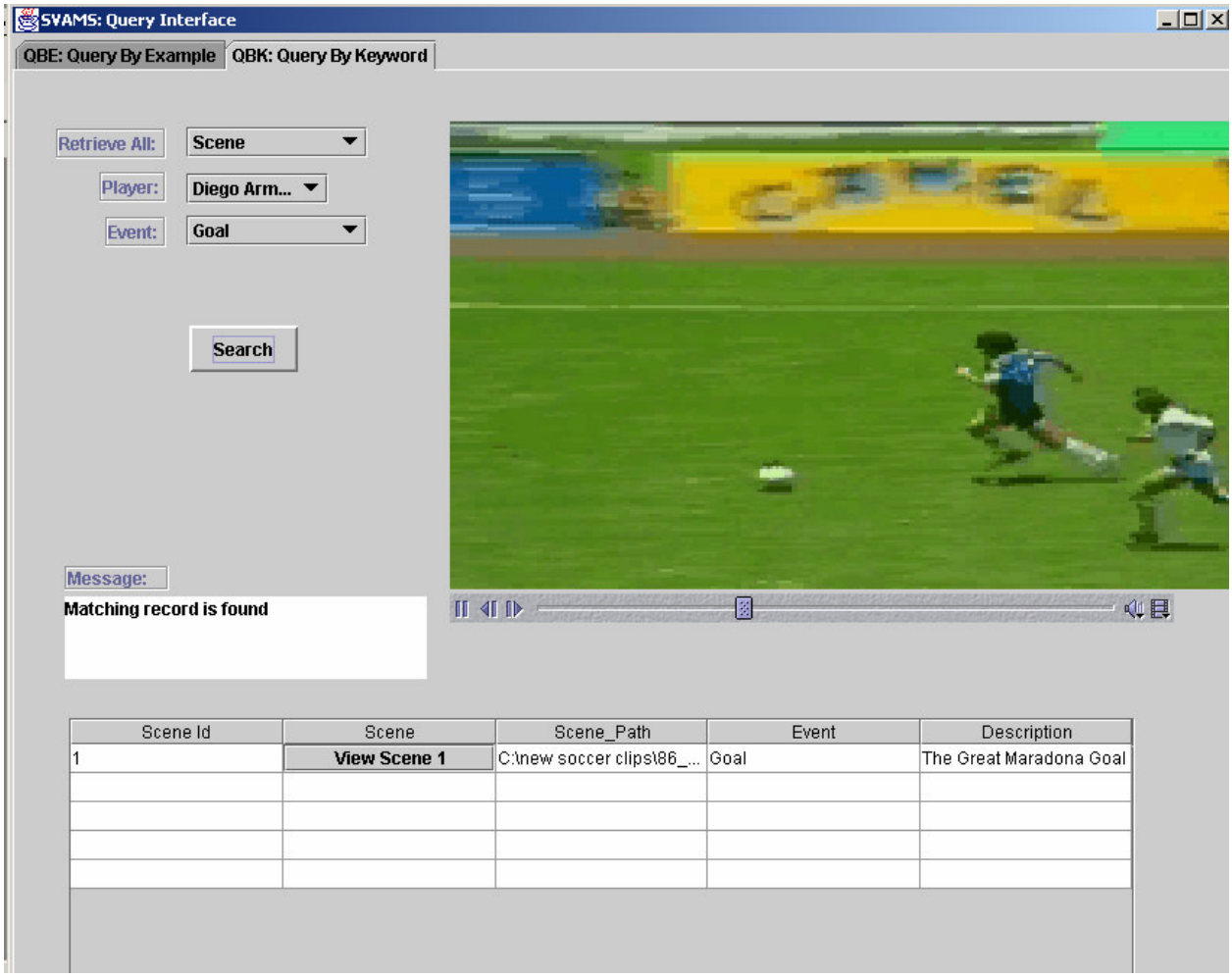

Figure 5.9 A Screenshot of the Query By Keyword Query Interface of SVAMS**.**

5.3 Summary

In this chapter, the specifics details of the prototype system SVAMS developed to demonstrate the practicality of the proposals made in this thesis has been discussed. We have discussed the platform under which the prototype system can be implemented. We have also dealt with the interMedia module integrated into Oracle 9i database that offers the storage and management of video data. A high-level architecture of the SVAMS in connection with the Oracle 9i database has been presented. Moreover, the query types that our prototype system allows have been explained. Finally, a description of the two main visual interfaces of the prototype system such as the Data Entry Panel and the query formulation panel has been presented.

CHAPTER 6

DISCUSSION

Since 1980's a lot of effort have been put into the development of efficient digital video data management techniques aimed at making video data as searchable as text data. Despite the tremendous amount of research efforts put into the area, existing video retrieval techniques have not reached to a satisfactory level of maturity, and the achievements obtained in the field have indicated a long way to go. This is, because video is a complex media and it is much more information-rich than text data, which in many cases needs superior data management techniques tailored to its behavior.

In this regard, as we have discussed in Chapter 4, the relational data model are not designed to handle the intricacy of video data, and the younger data model which is the Object-Oriented model has also its own flaws. This is where a new data model combining the best of both worlds came in, resulted in a paradigm shift, which is known as the OR-DBMS paradigm shift.

Video data management involves the fusion of many disciplines, amongst others, Computer Vision, which mainly focuses on the analysis of the perceptual aspect of a video or an image play the key role in availing techniques that can be employed for formulating queries based on perceptual features. Most research prototype and commercial retrieval systems support content-based retrieval using perceptual features. However, video retrieval based on perceptual features cannot discern the semantic meaning of the media in the manner a user wants. A retrieval system is only effective if it can return a query response of a user's subject of interest. At query time

good match in terms of visual features may not return the required query response. Hence, this approach alone cannot fully address the requirement of video data management.

Therefore, in this thesis a hybrid way of managing video content and its related data is proposed. In this context, in this thesis work three issues with regard to the provision of multi-criteria queries support are addressed. The first issue addressed is, the development of a generic video data model that can be used in unconstrained domain; in the proposed video data model in addition to the content of a video, metadata embedded in a video and mostly available in proprietary format is modeled, classified and consistently represented. The second issue addressed is, the design of a convenient video repositories models that can be used under OR-DBMSs environment. The third issue addressed is, the development of a video similarity-based retrieval technique that can conveniently be used under OR-DBMSs environment. In addition, a prototype system SVAMS is developed to demonstrate the practicality of our proposals.

6.1 The Video Data Model

Video is characterized by voluminous. To manage video data effectively and efficiently, it has to be segmented into meaningful and manageable units. To this effect, segmentation techniques are employed. After a unit of interest is determined, a video is segmented into its constituent elements, such as scenes or shots or Video Objects (VO), then, a data model is needed to describe or characterize and represent a video and its constituent units. Thus, there is a pressing need to have a video data model tailored to the behavior of vide data. The video data model we proposed distinguishes three parts: the frame-based scheme (or the frame-based representation scheme) which is intended for video data available in raw video streams, and object-based scheme (or object-based representation scheme) which is intended for video data available in object-based format (i.e. for video data available in MPEG-4 compressed bitstreams) and the

external description scheme which is intended to model, classify and represent metadata associated to a video and its constituent units.

The video data model we proposed revealed the key role that metadata can play in enhancing video retrieval systems by enabling the capturing of external descriptions (or high-level descriptions) of each video unit in a consistent and accurate manner using MPEG-7 (the standard content descriptor interface). Thus, the proposed video data model enables video retrieval systems to offer query support based on high-level descriptions of a video or its constituent units in addition to queries that can be formulated using low-level perceptual features.

6.2 The Video Data Repository

To manage the content of a video and its metadata under a DBMS environment, a suitable repository model is required. To this end, a video data repository model defined at the video and its constituent units is proposed. It is defined in such a way that the perceptual features (i.e. visual and temporal features) and high-level descriptions of a video are captured so as to provide multi-criteria query support. Thus, the chosen OR paradigm allows one to efficiently exploit the data management facilities of OR-DBMSs (i.e. support for both traditional alphanumeric and complex data types management).

6.3 Practical Demonstration

Using SVAMS, the prototype system, we demonstrated how the proposals we made in this thesis put into practice. The prototype system is a simple soccer video management system intended to manage soccer funs and hence only possesses basic functionalities that enable us to demonstrate the practicality of our proposals. However, it can be extended to incorporate additional tools that facilitate query formulation and query result displays.

CHAPTER 7

CONCLUSIONS AND FUTURE WORKS

Nowadays, digital data of audiovisual nature is becoming available to public access increasingly. In Chapter 1, we discussed how video is increasingly becoming a key element of multimedia computing. However, it's wide spread use is impeded due to a number of problems discussed in this thesis. The traditional data modeling techniques are not readily applicable for modeling video data. This is because video is content rich and complex. Thus, a data modeling technique that surpasses the power of traditional data modeling techniques, which is amenable to the behavior of video data, is needed.

We believe that effective video data management is the result of two complementary data management systems that is alphanumeric data management and complex data management. Developing a video retrieval system in a fresh start amounts to disregarding the maturity level that the conventional alphanumeric data managements systems have reached. Moreover, video data is multi-modal, besides the perceptual features it also consists of closed-caption text, which are amenable to traditional text and alphanumeric data management. Thus, content-based access to video can only be effective with the support of traditional text and alphanumeric data management techniques.

Therefore, in this regard we considered the existing methods in both domains (Computer Vision and DBMS) and proposed techniques for managing video data under OR-DBMS environment.

Major contributions of this thesis work are succinctly described below.

- A video data model, which distinguishes three parts, is introduced. The frame-based scheme (or representation scheme) is intended for video data available in raw video streams, the object-based scheme (or representation scheme) is intended for video data available in object-based format (i.e. for video data available in MPEG-4 compressed bitstreams), and the external description block is intended to model, classify and represent metadata associated to a video and its constituent units,
- Metadata as first class citizen of the video data model is introduced: metadata embedded in a video is modeled, classified and its role in enhancing retrieval efficiency is fully revealed. Moreover, through the proposed video data model, the benefits of the standard content descriptor interface MPEG-7 are directed to reach OR-DBMSs,
- We introduced a schematic video data repositories models defined at the video and its constituent elements, which can conveniently be used in the management of video collections under OR-DBMSs environment,
- A similarity-based video retrieval technique in the context of OR-DBMS is introduced,
- The important components of video data that need to be considered to fully represent and describe a video and its metadata are identified,
- A prototype system for a soccer application domain, which is called Soccer Video Archive Management System (SVAMS) that demonstrated the workability of the proposals made in this thesis, is developed.

Future work in this domain of research includes:

- based on the proposed video data model, the design of a video repository model for the object-based representation scheme of a video.
- video is multi-modal; in this regard, both representation schemes of the proposed video data model can be extended to mange audio data as well.
- the integration of similarity-based operators such as similarity-based join on both types of representation schemes.
- Key frames are not the unit of user's interest; the similarity-based retrieval technique proposed in the context of OR-DBMS can be extended to a course granularity such as a shot.

References

[1] J. R. Smith, S. Srinivasan, A. Amir, S. Basu, G. Iyengar, L. Ching-Yung, M. Naphade, D. Ponceleon, and B. L. Tseng, "Integrating Features, Models, and Semantics for TREC Video Retrieval", NIST TREC-10 Text Retrieval Conference, November 2001

[2] L. Chen II, M. Tamer Özsu, V. Oria: “Modeling Video Data for Content Based Queries: Extending the DISIMA Image Data Model”, MMM 2003: 169-189.

[3] S. Atnafu, L. Brunie, H. Kosch: “Similarity-Based Operators and Query Optimization for Multimedia Database Systems.”, IDEAS 2001: 346-355.

[4] F. I. Bashir, Ashfaq A. Khokhar, "Video Content Modeling: An Overview", Technical Report 09/2002.

[5] Y. Rui, Efficient Indexing, Browsing and Retrieval of Image/Video Content, University of Illinois at Urbana-Champaign, 1998. PhD Thesis.

[6] L. Chen and M. Tamer Özsu “Modeling of Video Objects In A Video Databases”, Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference.

[7] M. Tamer Özsu, “Principles of Distributed Database Systems”, Second Edition, Prentice-Hall, Inc., 1999.

[8] V. S. Subrahmanian, “Principles Of Multimedia Database Systems”, Morgan Kaufmann Publishers, Inc. 1998

[9] M. Petkovic, W. Jonker, “An Overview of Data Models and Query Languages for Content-based Video Retrieval”, International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet, l`Aquila, Italy, July 2000.

[10] M. Petković, W. Jonker.”A framework for Video Modeling”, Eighteenth IASTED International Conference on Applied Informatics, Innsbruck, February 2000.

[11] A. Hampapur, R. Jain, “Video Data Management Systems: Metadata and Architecture” in Multimedia Data Management, A. Sheth, W. Klas (ed), McGraw-Hill, 1998.

[12] T.G. Aguierre Smith, G. Davenport, “The Stratification System: A Design Environment for Random Access Video”, Proc 3rd Inter. Workshop on Network and operating system Support for Digital Audio and Video, La Jolla. C.A, 1992.

[13] F. I. Bashir, S. Khanvilkar, D. Schonfeld, A. Khokhar, "Multimedia Systems: Content-Based Indexing and Retrieval", E.E. Handbook, accepted for publication, 2003.

[14] N. Dimitrova, F. Golshani “Motion Recovery for video content classification”, ACM Transaction on Information Systems, 13:4, Oct. 1995, pp. 408-439.

[15] D. Schonfeld, D. Lelescu “VORTEX: Video retrieval and tracking from compressed multimedia databases-multiple object tracking from MPEG-2 bitstream, “(Invited Paper): Journal of Visual Communications and Image Representation, Special Issue on Multimedia Database Management, vol 11, pp. 154-182, 2000.

[16] M.R. Naphade, R. Mehrotra, a.M Fermant, J. Warnick, T.S Huang, A.M. Tekalp, “Probabilistic Multimedia Objects Multijects: A novel Approach to Indexing and Retrieval in Multimedia Systems”, Proc. I.E.E.E. International Conference on Image Processing, Volume 3, pages 536-540, Oct. 1998, Chicago, IL.

[17] D. A. Tran, K. A., Hua, Vu K. “Semantic Reasoning based Video Database Systems”, Proc. of the 11th Int’l Conf. On Database and Expert Systems Applcitions, pp. 41-50, September 4-8, 2000, London, England.

[18] A. Del Bimbo, “Visual Information Retrieval”, MORGAN KAUFMANN, 1999.

[19] R. Chbeir, S. Atnafu, L. Brunie, Image Data Model for Efficient Multi-Criteria Query in Medical Database; 14th International Conference on Scientific and Statistical Database Management (SSDBM 2002), 24th-26th July, pp.165- 175, Edinburgh, Scotland.

[20] A. Gupta, T. Weimouth and R. Jain., "Semantic queries with pictures: The VIMSYS model". Proc. of the 17th Int. Conf. On VLDB, September 1991, pp. 69-79.

[21] R. Jain and A. Hampapur., "Metadata in video databases, in Special Isuue on Metadta for digital Media, SIGMOD Record, Dec., 1994.

[22] G. Davenport, T.A. Smith and N. Pincever., "Cinematic primitives for multimeda IEEE Computer Graphics and Applications, July 1991:67-74

[23] A. Smith, T.G. and G. Davenport. 1992. "The Stratification System: A Design Environment for Random Access Video", Proc. 3rd Int. ACM Workshop on Networking and Operating System Support for Digital Audio and Video, La Jolla, CA, November 1192: pp. 250-261.

[24] H. Zhang. "Video Content Analyis and Retrieval". Handbook on Pattern Recognition and Computer Vision, World Scientific Publishing Company, 1997.

[25]. T. Lin, H. Zhang: "Automatic Video Scene Extraction by Shot Grouping". ICPR 2000: 4039-4042

[26].M. Cooper, and Jonathan Foote. "Conference on Image Processing", In Proceedings of the International, Thessaloniki, Greece. October 7-10,

[27] M. E. Donderler, O. Ulusoy, U. Gudukbay, "A Rule-based Approach to Represent Spatio-temporal Relations in Video Data", International Conference on Advances in Information Systems (ADVIS'2000), Lecture Notes in Computer Science (Springer-Verlag), vol.1909, October 2000.

[28] M. Petković “Content-based video retrieval”, Ph.D. Workshop, in conjunction with VII. Conference on Extending Database Technology

[29] V. Oria, M.T. Özsu, L. Liu, X. Li, J.Z. Li, Y. Niu, and P. Iglinski, "Modeling Images for Content-Based Queries: The DISIMA Approach", Second International Conference on Visual Information Systems, San Diego, CA, December 1997, pages 339-346.

[30] P. Lyman, H.R. Varian. How Much Information?, A research report at the School of Information Management and Systems at the University of California at Berkeley, Regents of the University of California, October 2000. Available on: http://www.sims.berkeley.edu/how-much-info/, (Consulted on April, 2003).

[31] Road Ward. Oracle interMedia User’s Guide and Reference, Release 9.0.1, Oracle Corporation, Part No. A88786-01, 2001.

[32] R. Konen . An excerpt from the new standard, courtesy of INTERNATIONAL ORGANIZATION FOR STANDARDIZATION(ISO), WG11(MPEG), Theory in Focus, CCTV theory explained,.

[33] R. Weiss, A. Duda, D.K. Gifford, “Content-based Access to Algebraic Video”, Proc. of Int. Conf. On Multimedia Computing and Systems, IEEE Press, 1994. pp. 140-151.

[34] MPEG-4 video verification model version-11, ISO/IEC JTC1/SC29/WG11, N2171, Tokyo, March 1998.

[35] V. Kashyap and A. Sheth. Semantics Heterogeneity in Global Information Syatems: the Role of Metadata, Context and Ontologies. In M. Papazoglou and G. Schlageter (eds.). Cooperative Information Systems: Current Trends and Directions. Springer-Verlag, 1997, pp. 139-178.

[36]J.K Wu, A. Desai Narasimhalu**, B.M. Mehtre, C.P. Lam**, Y.J. Gao. CORE:: a content-based retrieval engine for multimedia information systems, Institute of Systems Science, National University of Singapore, Heng MuiMui Keng Terrace, Kent Ridge, Singapore-0511, Multimedia Systems-Verlag 1995, pp. 25-41.

[37] MPEG-1: "Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbps", ISO/IEC 1117-2:Video, (November 1991).

[38] MPEG-2: "Generic coding of moving pictures and associated audio information", ISO/IEC 13818-2 Video, Draft International Standard, (November 1994).

[39] R. Koenen. MPEG-4 Overview - (V.21 – Jeju Version), ISO/IEC JTC1/SC29/WG11 N4668, MPEG Requirements Group, March 2002. Available at: http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm

[40] J.M. Martinez. MPEG-7 Overview (version9.0). ISO/IEC JTC1/SC29/WG11N5525, MPEG Requirements Group, Pattaya, March 2003. Available at: http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm

[41] J. Bormans, K. Hill. MPEG-21 Overview v.5 **.** ISO/IEC JTC1/SC29/WG11N5525, Shanghai, October 2002. Available at: http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm

[42] S. Boll, W. Klas, P. Amit Sheth: "Overview on Using Metadata to manage Multimedia Data. Multimedia Data Management", 1998: 1-24

[43] D. A Tran., K. A. Hua, Vu K. "Video Graph: A Graphical Object-based Model for Representing and Querying Video Data", In the proc. of ACM Int'l Conference on Conceptual Modeling (ER 2000), October 9-12, Salt Lake city**,** USA**.**

[44] P. Aigrain, H. Zahng, D. Petkovic, “Content-based Representation and Retrieval of Visual Media: A State-of-the-Art Review”, Multimedia Tools and Applications, Kluwer Academic Publishers, 3(3), 1996, pp. 124-132.

[45] H. Wactiar. Extracting and Visualizing Knowledge from Video and Film Archives, Volume 8, Issue 6 (http://www.jucs.org), July 2002, Springer Co. Pub. Also appeared on the proceedings of I-KNOW’02 Conference, Graz, Austria, July 11-12, 2002. Available at: http://www.know-center.at/en/conference/I-know02/program.htm (Consulted on Nov. 24, 2002).

[46] M. Castellanos & M. Garcia-Solaco. ACM SIGMOD Record vol 20, #4, pp. 44-48, special refereed issue: A. Sheth (guest ed.): Semantic Issues in Multidatabase Systems, Dec. 1991.

[47] Naphade M.R. Mehrotra R., Fermant a.M, Warnick J., Huang T.S, Tekalp A.M., “ A High Performance Shot Boundary Detection Algorithm using multiple cues”, Proc. I.E.E.E International Conference on Image Processing, Volume 3, pages 536-540, Oct 1998, Chicago, II.

[48] M.J. Swain and DH. Ballard. “Color Indexing”, International Journal of Computer Vision, 7(1):11-32, 1991.

[49] M. E. Donderler, O. Ulusoy, U. Gudukbay, “A Rule-based Approach to Represent Spatio-temporal Relations in Video Data”, International Conference on Advances in Information Systems (ADVIS'2000), Lecture Notes in Computer Science (Springer-Verlag), vol.1909, October 2000.

[50] T. Lin, HongJiang Zhang, “Automatic Video Scene Extraction by Shot Grouping”, ICPR 2000: 4039-4042.

[51] M. Cooper, and Jonathan Foote, “Scene Boundary Detection Via Video Self-Similarity Analysis”, In Proceedings of the International Conference on Image Processing, Thessaloniki, Greece. October 7-10, 2001.

[52] H. Zhang, C. Y. Low, S. W. Smoiar, and D. Zhong, “Video parsing, retrieval and browsing: An integrated and content-based solution”, In Proc. ACM Conference on Multimedia, 1995.

[53] Y. Zhuang, Y. Rui, T.S. Huang, and S. Mehrotra, “Adaptive key frame extraction using unsupervised clustering”, In Proc. IEEE International Conference on Image Processing, 1998.

[54] P. O. Gresle and T. S. Hung, “Gisting of Video Documents: Key frames selection algorithm using relative activity measure”, In Proceedings of the 2nd International Conference On Visual Information Systems, 1997.

[55] W. Wolf, “Key frame selection by motion analysis”, In Proc. IEEE International Conferencing, Acoust, Speech, and Signal Processing, 1996.

[56] B. Erol and F. Kossentini, “Automatic Key Video Object Plane Selection Using the Shape Information in the MPEG-4 Compressed Domain”, IEEE Transactions on Multimedia, vol. 2, no 2, pp.129-138, June 2000.

[57] B. Erol and F. Kossentini, “Retrieval of video objects using compressed domain shape features”, Proceedings of IEEE ICECS Conference, December 2000.

[58] Oh J., Chowdary T., “An efficient technique for measuring of various motions in video sequences”, Proceedings of 2002 International Conference on Imaging Science, Systems and Technology (CISST’02), Las Vegas, NV, June 2002.

[59] H. Zhang, S. W. Somiar, and J. J. Wu, “Content-based video browsing tools”, In Proceedings of ISETC/SPIE Conference on Multimedia Computing and Networking, 1995.

[60] D. Zhlong, H. Zhang, and S.F Chang, “Clustering methods for video browsing and annotation”, Technical Report, Columbia University, 1997.

[61] S. Berchtold, C. Boehm, B. Braunmueller, D. A. Keim and H. P. Kriegel. A Cost Model for Nearest Neighbor Search in High-Dimensional Data Space, ACM PODS, Arizona, 1997, p.78-86.

[62] T. Seidl, H.P Kriegel. Optimal Multi-Step k-Nearest Search, SIGMOD, WA, USA, 1998, p. 154-165.

[63] G.R. Hjaltason and H. Samet. Ranking in Spatial Databases, Springer-Veriag, Berlin, 1995, Proceedings of the 4th International Symposium on Large Spatial Databases – SSD’95, p. 83-95.

[64] S. Berchtold, C. Boehm, D. A. Keim and H. P. Kriegel. A Cost Model For Nearest Neighbor Search in High-Dimensional Data Space, Proceedings of the ACM PODS Conference, Tucson, Arizona, May 1997, p. 78-86

[65] Tony Lin, Chong-Wah Ngo, Hong-Jiang Zhang, and Qing-Yun Shi, “Integrating color and spatial features for content-based video retrieval”, IEEE Int. Conf. Image Processing (ICIP), invited paper, oral presentation, Greece, Oct. 7-10, 2001. (EI)

[66] Waleed E. Farag, Hussein Abdel-Wahab. “Adaptive Key Frames Selection Algorithms for Summarizing Video Data”, JCIS 2002: 1017-1020

[67] Tianming Liu; Hong-Jiang Zhang; Feihu Qi . “A novel video key-frame-extraction algorithm based on perceived motion energy model”, Circuits and Systems for Video

Technology, IEEE Transactions on Volume: 13, Issue: 10, Year: Oct. 2003. p. 1006-1013

[68] Nelson M. Mattos. “SQL99, SQL/MM, and SQLJ: An Overview of SQL Standards”, IBM Database Common Technology.

[69] S-F. Chang, A. Puri, T. Sikora, and H. Zhang. Special issue on MPEG-7 (12 papers). IEEE Transactions on Circuits and Systems for Video Technology, 11(6), June 2001.

[70] Agathoniki Trigoni, G. M. Bierman. “Inferring the Principal Type and the Schema Requirements of an OQL Query”, BNCOD 2001, p.185-201.

[71] Oomoto, E. and K. Tanaka. “OVID: Design and implementation of a video object database system”, IEEE Trans. On Knowledge and Data Engineering, 1993, 5(4):629-643.

[72] Ardizzone, E. and M. La Cascia. “Video indexing using optical flow field”, Proc. ICIP’96, International Conference on Image Analysis, Vol. 3, pp. 831-834.

[73] Ardizzone, E. and M. La Cascia. “Automatic video database indexing and retrieval”, Multimedia Tools and Applciations, 1997, 4(1):29-56.

[74] Chang. S.F., W. Chen, H.J. Meng, H. Sundaram and D. Zhong. “A fully automated content based video search engine supporting spatio-temporal queries”, in Special Issue on Image/Video Processing for Interactive Multimedia, IEEE Trans. On Circuit and Systems for Video Technology, 1998.