Proceedings of the IASTED International Conference on
Internet and Multimedia System and Applications
October 18-21, 1999 Nassau, Bahamas

Issues in Designing Contemporary Video Database Systems

Oge Marques and Borko Furht

Department of Computer Science and Engineering
Florida Atlantic University
777 Glades Road, Boca Raton, FL 33431-0991
{omarques, borko}@cse.fau.edu

Abstract ¾ **Video databases became an active field of research during the last years. Many universities and research groups are building prototypes and the first commercial products are becoming available. This paper surveys some of the most important open issues that should be taken into account when designing a contemporary video database system.**

Index Terms ¾ **Video databases, Multimedia database systems, Video indexing and retrieval, Digital libraries.**

1. Introduction

The field of distributed multimedia systems has experienced increasing research and development efforts during the last decade. One of the main goals envisioned by multimedia researchers is the creation of huge digital libraries accessible to users worldwide. These large and complex multimedia databases must store all types of multimedia data, e.g. text, images, animations, graphs, drawings, audio, and video clips. Video information plays a central role in such systems and therefore the design and implementation of video database systems has become a major topic of interest in the last five years or so.

The amount of video information stored in archives worldwide is huge. Conservative estimates state that there are more than 6 million hours of video already stored and this number grows at a rate of about 10 percent a year [1]. Significant efforts have been spent in recent years to make the process of video archiving and retrieval faster, safer, more reliable and accessible to users anywhere in the world. Progress in video digitization and compression, together with advances in storage media, have made the task of storing and retrieving raw video data much easier. Evolution of computer networks and the growth and popularity of the Internet have made it possible to access these data from remote locations.

However, raw video data alone has limited usefulness, since it takes far too long to search for the desired piece of information within a videotape repository or a digital video archive. Attempts to improve the efficiency of the search process by adding extra data (henceforth called *metadata*) to the video contents do little more than transferring the burden of performing inefficient, tedious and time-consuming tasks to the cataloguing stage. There must be better ways to automatically store, catalog, and retrieve video information with greater understanding of its contents. And this is the challenge behind the design of contemporary video database systems.

In this paper we examine the state of the art, ongoing research, and open issues in designing video database systems.

2. Video database systems in a nutshell

The primary goal of a Video Database System (VDBS) is to provide an environment both convenient and efficient for retrieving and storing database information [2]. More specifically, the main purpose of video database systems is to provide a pseudo-random access to the sequential video data, in other words, to overcome the sequential and time-consuming process of viewing video [2] [3]. This goal is normally achieved by dividing a video recording into meaningful segments, indexing those segments, and representing the indexes in a way that allows easy browsing and retrieval. Therefore, a VDBS is basically a database of indexes (pointers) to a video recording [3].

In addition to its primary objective, the development of a VDBS is also driven by other arguments, common to general database systems, such as the ability of sharing data, enforcing standards, implementing security measures, providing data independence, reducing redundancy, avoiding inconsistencies, maintaining integrity, balancing conflicting requirements, and making information available on demand [1].

Figure 1 presents a simplified block diagram of a typical VDBS. Its main blocks are:

- **User interface**: friendly, visually rich interface that allows the user to interactively query the database, browse the results, and view the selected video clips.

- **Query / search engine**: responsible for searching the database according to the parameters provided by the user.
- **Digital video archive**: repository of digitized, compressed video data.
- **Visual summaries**: representation of video contents in a concise, typically hierarchical, way.
- **Indexes**: pointers to video segments or story units.
- **Digitization and compression**: hardware and software necessary to convert the video information into digital compressed format.
- **Cataloguing**: process of extracting meaningful story units from the raw video data and building the corresponding indexes.

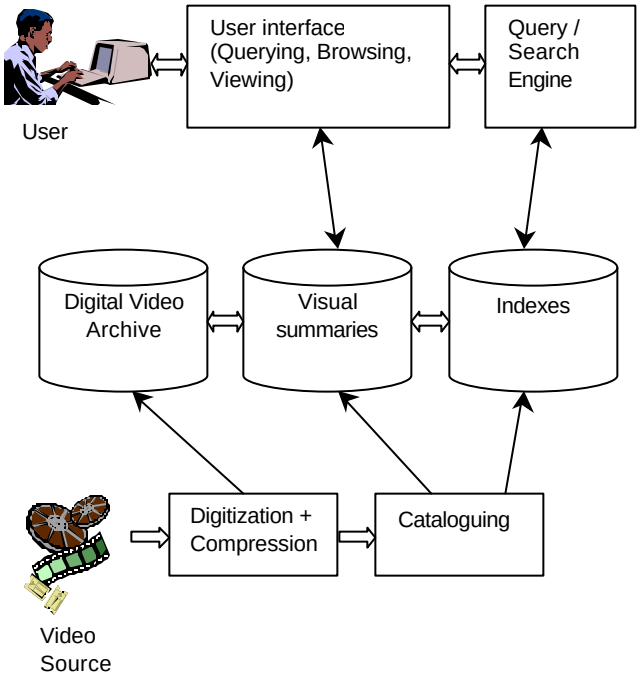

Figure 1. Block diagram of a VDBS (based on [2]).

3. Building a Video Database System

Preliminaries

Building a video database requires far more than digitizing and compressing video data. The main task is to automatically extract indexes from the video stream by dividing the original recording into segments and adding textual and/or symbolic information (metadata) to the indexes. Next, these indexes must be represented in visual summaries in a way that allows easy browsing and retrieval. Since video databases tend to be huge, many database design issues come into picture, such as the choice of effective database models and the yet open problem of efficient (incremental) indexing of very large databases. Moreover, extraction of low/intermediate features that will eventually be used for query-by-content operations also takes place at this stage. The issues of video segmentation, metadata representation, feature extraction, and video abstraction are explained below. The concept of visual summaries and possible ways of implementing them is discussed in Section 5. For more details on the design and implementation of multimedia database systems, see [4], [5], and [6].

Video segmentation

Video segmentation, also called *video parsing*, consists in partitioning video sequences (also called *segments* or *stories*) into scenes which can be further subdivided into individual shots, as illustrated in Figure 2. A *shot* can be defined as "a continuous action on screen resulting from what appears to be a single run of the camera" [7]. A *scene* is a sequence of shots that focus on the same point of interest, while a *sequence* can be defined as "a series of related shots and scenes that form a single, coherent unit of dramatic action" [7].

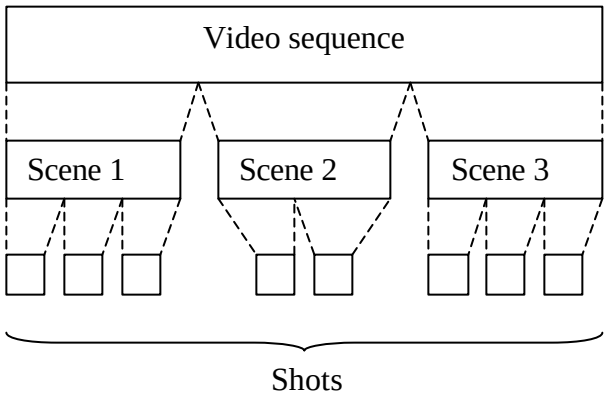

Figure 2. Video parsing (segmentation) consists of partitioning a video sequence into video scenes and further into video shots.

Current shot detection algorithms perform fairly well and are able to detect shot boundaries even in the presence of gradual transition effects such as fade, wipe, and dissolve. Scene detection, however, is still an open issue for which two major approaches have been tried: (i) the use of filming rules (e.g. transition effects, shot repetition, presence of music in the soundtrack) to allow shot grouping and event detection; and (ii) the use of *a priori* rules for segmenting programs (e.g. TV news), whose structure is fairly rigid and predictable. Experts agree that "general scene detection requires content analysis at a higher level and it should not be expected that this task will be fully automated in the near future based only on the visual content analysis using current image processing and computer vision techniques" [8].

Metadata

Until recently, video data used to be stored in analog format and the associated information (metadata) slowly moved from textual catalog cards to computerized databases. There are three categories of metadata [5]: (a) *content-dependent metadata* (e.g. the facial features of a news anchorperson, the derivation of camera movements); (b) *content-descriptive metadata* (e.g. the impression of anger or happiness based on facial expression); and (c) *content-independent metadata* (e.g. name of the director of a movie). Only content-dependent metadata can be automatically extracted from the video contents. Contemporary video database systems assume that video data and metadata are stored in digital format, which gives rise to several questions, such as:

- Which metadata is needed?

The answer to this question depends heavily on the scope of the application and the amount of detail one wants to keep track of. A TV news video clip might need, at the very minimum, day, time, duration, TV station, and 3 to 10 keywords that best describe the specific clip. A Hollywood movie, on the other hand, could be described simply by its title and year (which should be enough to distinguish it from its remakes and sequels), or could add as much metadata as to even include the name of the newly hired intern of the costume crew.

- How much metadata is needed?

Defining the amount of metadata to be stored directly impacts the cataloguing and query stages. The more metadata, the longer it will take to catalog it, either manually or automatically. On the other hand, the richer the queries by keywords can be.

- How could we automatize the process of metadata extraction from the video contents?

This is one of the key questions researchers are trying to answer these days. The goal is to maximize the amount of content-dependent metadata automatically extracted from the video contents and to reduce the amount and need for human intervention in the process to a minimum.

As a final note, it is important to mention that standardization of metadata descriptors and integration to the video data into a single model is under study. The MPEG-7 standard [9] committee is working on the specification of a standard set of descriptors as well as Description Schemes (DSs) for the structure of descriptors and their relationships. This combination of descriptors and description schemes will be associated with the content itself to provide a uniform method for labeling visual information at the semantic level and allow fast and efficient searching for multimedia contents. MPEG-7 will also standardize a language to specify description schemes and the schemes for encoding the descriptions of multimedia content. It is important to note that, however useful they are, neither automatic nor semi-automatic feature extraction algorithms will be inside the scope of the standard. The search engines will not be specified within the scope of MPEG-7, either. Those aspects are illustrated in Figure 3.

Figure 3. Scope of MPEG-7 (from [9]).

Feature extraction

Another important action that takes place in the cataloguing phase is feature extraction. Most of these low-level features were originally used in image processing and computer vision, and later adapted to video. Typical examples of such features are texture and color. Once extracted, those features are stored in a feature vector, which might later be compared against a user-defined feature vector during the query-by-content stage.

Some of the important open questions at this stage are:

- Which features should be extracted?

Color, shape and texture have been used with limited success for image databases that support query-by-content. However, these features are frame-dependent and do not reflect the dynamics of video data. Furthermore, newer, and better features – possibly combined and weighted in a customized manner as to adapt themselves to users' preferences – are needed for video databases.

- How well do these features represent the semantic contents of the video data?

Video information is a semantically rich and complex concept [1]. Video retrieval will be effective only if there are ways of mapping semantic (high-level) features onto low-level features. Bridging this semantic gap is still a research challenge. Attempts to solve the semantic gap problem include the identification of semantic primitives (*objects*, *role*, *actions* and *events*) as abstractions of visual signs, the association of semantics with visual signs, and, more recently, the use of semiotic methodologies, in an attempt to investigate the *sense* conveyed by low-level features such as camera breaks, colors, and editing effects [10].

Video abstraction

Video abstraction is the process of extracting the main visual information of a video sequence and presenting it in a way that should be much shorter than the original video. Three possible approaches to video abstraction are: (i) key-frame extraction, where each video shot is represented by one of its frames; (ii) video icons, whose main variants are 3-D icon and video mosaic; and (iii) video summaries, in which additional sources of information, such as audio and text are combined to create a highlight of a long video sequence.

The automatic construction of an effective video summary for general video materials remains an open research topic [8].

4. Querying the video database: the designer's perspective

There are two major categories of queries: textual and visual. Query-by-text rely on keywords and show limited usefulness in video databases, since it may be difficult to know under which keywords a given video material has been indexed. Querying the video database based on visual content is normally performed by extracting the visual attributes provided by the user (either directly or through an example), building a feature vector and calculating metric distances between this vector and every feature vector previously stored. While mathematically elegant, this classic method bears no resemblance with psychological similarity models which arguably better model the way human beings perceive similarity [10].

Very often, query results may not be exactly the ones the user had in mind when she formulated the query. To overcome this problem, the system should allow interactive refinement of the query. At a more advanced stage, the system should also learn from these interactions and improve its performance in subsequent searches, in a process known as relevance feedback. That is the focus of several ongoing research efforts in video querying.

5. Querying the video database: the user's perspective

The user interface is a crucial component of a VDBS. Ideally such interface should be simple, easy, friendly, functional, and customizable. It should provide integrated browsing, viewing, searching, and querying capabilities in a clear and intuitive way. This integration is extremely important, since it is very likely that the user will not always stick to the best match found by the query engine. More often than not the user will want to check the first few best matches, browse through them, preview their contents, refine her query, and eventually retrieve the desired video segment.

Searching the VDBS contents should be made possible in several different ways, either alone or combined. For instance a user should be able to perform a pictorial query, e.g. querying by similarity (using another video clip or a static image as a reference) and a query by keyword simultaneously.

Query options should be made as simple, intuitive and close to human perception of similarity as possible. Users are more likely to prefer a system that offers the "Show me more video clips that look similar to this (image or video)" option, rather than a sophisticated interactive tool to edit that video shot key-frame's histogram and perform a new search. While the latter approach might be useful for experienced technical users with image processing knowledge, it does not apply to the average user and therefore has limited usefulness. An ideal VDBS query system would hide the technical complexity of the query process from the end user, who might wonder: "how do they manage to retrieve exactly the video I want?"

6. Video databases on the Internet

Making a VDBS accessible through the Internet, particularly on the Web, extends its usefulness to users anywhere in the world at the expense of new design constraints which are addressed below [11]:

- Visual information on the Web is highly distributed, minimally indexed, and schema-less.
- The query and retrieval stages have no control over the cataloguing process and must rely on possible metadata stored in HTML tags associated with the images and video clips.
- In order to keep the query response time below a tolerable limit (typically two seconds), the number of visual features used for comparison and matching has to be kept low.
- The user interface should work with reduced-size images and videos until the final stage, when the user issues an explicit request.
- The use of content-based query methods may be deferred until a stage where the scope of the search has been reduced to a specific semantic category, selected by the user.

7. Conclusion

Research on video databases is very active. Some of the important aspects currently being investigated include:

- Higher degree of automation of the cataloguing process [12].

- Standardization of metadata descriptors.
- Better, simpler, and more functional user interfaces with integrated browsing, navigating, viewing, searching, and querying capabilities.
- Use of machine learning techniques to improve performance based on users' feedback.
- Automatic extraction of semantic information from video data.
- Stronger integration among different abstraction levels, from the feature level up to the semantic level.
- Use of multimedia features (audio, video, and textual information) for video classification.

The ultimate goal is to enable users to retrieve the desired video clip among massive amounts of video data in a fast, efficient, semantically meaningful, friendly, and location-independent manner.

References

[1] R. Hjelsvold, "VideoSTAR – A database for video information sharing", Dr. Ing. Thesis, Norwegian Institute of Technology, November 1995.

[2] B.-L. Yeo and M. M. Yeung, "Retrieving and Visualizing Video", *Communications of the ACM*, Vol. 40, No. 12, December 1997.

[3] R. Bryll, "A Practical Video Database System", Master Thesis, University of Illinois at Chicago, 1998.

[4] S. Marcus and V.S. Subrahmanian, "Multimedia Database Systems", http://www.cs.umd.edu/projects/hermes/publications/postscripts/mm1.ps.

[5] B. Prabhakaran, *Multimedia Database Management Systems*. Boston: Kluwer Academic Publishers, 1997.

[6] K. C. Nwosu, B. Thuraisingham, and P. B. Berra, *Multimedia Database Systems: Design and Implementation Strategies*. Boston: Kluwer Academic Publishers, 1996.

[7] I. Konigsberg, *The Complete Film Dictionary – 2nd ed*. New York: Penguin, 1997.

[8] H.-J. Zhang, "Content-based video browsing and retrieval", in *Handbook of Internet and Multimedia Systems and Applications*, B. Furht (ed.). Boca Raton: CRC Press, 1999.

[9] http://drogo.cselt.stet.it/mpeg/standards/mpeg-7/mpeg-7.htm

[10] A. Del Bimbo, "A Perspective View on Visual Information Retrieval Systems", Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries, Santa Barbara, California, 1998.

[11] S.-F. Chang, J. R. Smith, M. Beigi, and A. Benitez, "Visual Information Retrieval from Large Distributed Online Repositories". *Communications of the ACM*, Vol. 40, No. 12, December 1997.

[12] D. Petkovic, "Challenges and Opportunities in Search and Retrieval for Media Databases", Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries, Santa Barbara, California, 1998.