# Hybrid Query for Video Database System

**Lilly Suriani Affendey, Ali Mamat, Hamidah Ibrahim, Fatimah Ahmad**

*Faculty of Computer Science and Information Technology*
*Universiti Putra Malaysia, 43400 Serdang, Selangor*
*Tel : 03-89466549, Fax : 03-89466577*
*E-mail : suriani@fsktm.upm.edu.my, ali@fsktm.upm.edu.my, hamidah@fsktm.upm.edu.my, fatimah@fsktm.upm.edu.my*

**ABSTRACT**

*Many video retrieval systems support text and low-level feature-based querying, however only single homogeneous queries are allowed. In this paper we present a Video Database System (VDBS) based on our video data model that captures the hierarchical structure and contents of video to support hybrid query. The system supports query using similarity-based matching of low-level visual features as well as exact matching of textual attributes. The experiment shows that hybrid query gives more accurate results when compared to single query using text or image alone by eliminate results that are similar in colour but has different semantic, or vice versa.*

**Keywords**
*Video database, video query, similarity-based matching*

## 1.0 INTRODUCTION

Along with the growth of the information era and technology, abundant collections of multimedia data such as image, graphic, audio and video became available. Among these media, video contains the most features making it the most expressive and thus very complex to manage. Video provides a rich and lively resource for multimedia applications. Video resources can be divided into various categories such as instruction, entertainment, scientific recording, news, etc. However the availability of video resources does not necessarily imply accessibility and manipulability of video data. Large amount of video data and its audio-visual nature has made their manipulation very challenging. Problems are encountered with respect to the retrieval of the video content. Ordinary retrieval techniques are not suitable for practical usage within digital video libraries. The high volume of video data makes free browsing almost impossible. It is tedious and time consuming for a user to browse through a huge collection of video clips and find the desired part. However, the retrieval process can rely on textual annotation which is added as metadata through the semi-automatic cataloguing process. Nevertheless, this process requires a considerable amount of time to annotate. Furthermore, the text associated can sometime be vague and incomplete due to subjective human perception. These text descriptions should also describe the content sufficiently to help the users locate segments of interest or, at least, not exclude potential segments from the candidate list. Textual attributes of video segments are essential components of the video query, but not the only ones. Attributes such as the date of creation, title, sources are commonly available at the time of initial database population. Keywords and descriptions were added later using annotation tool. Query by text can be implemented using the database approach or the information retrieval techniques. Despite the limitations of the text annotation approach, it is still used to assign semantics. Semantics describe video concepts on a higher level, such as the description of objects, activities and events. And this is usually what an average user has in mind when formulating the query. Still, it is not reasonable to assume that all types of multimedia data can be described sufficiently with words alone. Thus, where images cannot be expressed textually it is useful to formulate query by example. Query by example is by far the most widely supported language in research prototypes for image databases. The two types of example used in querying video are image and video clip.

The rest of paper is organized as follows: The next section reviews related works. Section 3 explains the video database system and section 4 discusses the experiment and results. Finally section 5 gives the conclusion.

## 2.0 RELATED WORKS

Over the years research issues in video retrieval have been addressed from different perspectives and various experimental systems have been implemented. Content-based modeling can be categorized mainly into semantic or low-level visual features. Thus in the context of image and video data, content-based queries can be formulated using several techniques, which fall broadly into two categories: textual and visual. Textual annotation-based approach is for describing semantic or high-level concepts such as objects and events. While visual feature-based approach is used for indexing low-level visual features.

Video annotation is arbitrary video content description. Textual video annotation is usually in the form of keywords and free text, done manually and is very tedious. The differences may arise from their point of

views, their focuses, or their purposes of using the video. Other sources of video annotation come from closed caption, optical character recognition (OCR) or voice recognition (Ponceleon et al., 1998). Several attempts have been made towards extracting such information automatically to serve as video indexes. However, such automation cannot be fully achieved since they require further advancement in the image processing and computer vision techniques (Jiang & Elmagarmid, 1998). Although keywords and free text documents are often used to annotate video data, researchers also experiment other annotation forms. An iconic annotation of video content by arranging selected icons along Media Time Line was proposed (Davis, 1993). However, this approach is limited by its fix set of vocabulary and it does not exploit the textual information of video streams such as closed caption.

Another research used spatio-temporal logic to describe image and video content (Bimbo et al., 1995). In their prototype system, each stored image and video is processed and represented by spatio-temporal logic. The limitation of this approach was the image indexes were static and cannot be dynamically updated. Furthermore higher level concepts of spatio-temporal events were not addressed. This approach is not appropriate for heterogeneous user where the metadata need to be shared.

Oomoto (Oomoto & Tanaka, 1993) proposed a schema-less video object data model. Their focus was on the capabilities of object-oriented database features for supporting schema evolution and to provide a mechanism for sharing some descriptive data. A video frame sequence is modeled as an object with attributes, and attribute values to describe its contents. A semantically meaningful scene is a sequence of (not always continous) video frames. An interval is described by a pair of starting frame and ending frame, that denotes a continuous sequence of video frames. OVID is a prototype video object database system. The main components of OVID system are VideoChart, VideoSQL and Video Object Definition Tool. Each video object has an identifier, video frame numbers denoting the start and end frames, object annotations in the form of a set of attribute/value pairs and some methods such as play, inspect, disaggregate, merge and overlap. This system introduced interval inclusion inheritance. This mechanism, by which some attribute/value pairs of a video unit can be inherited by another video unit, enables sharing information among video units and can be used to speed up the process of annotation.

A frame-based video retrieval and sequencing system was designed and implemented by Chua and Ruan (Chua & Ruan, 1995). In order to retain the advantages of the structured approach while providing greater flexibility in modeling video data, a two-layered, concept-based retrieval model is employed in VRSS.

Hjelsvold (Hjelsvold et al., 1996) proposed a generic video data model. Their proposal combines ideas from the stratification and the segmentation approaches. The objective is to develop a framework where structuring, annotations, sharing and reuse of video data become possible. Their model is built upon an enhanced–ER model. A simple SQL-like video query language with temporal interval operators (eg. equals, before, etc.) is provided. This work concentrates on the structural part of a video in order to support video browsing. Thematic indexing is based on annotations, which give a textual description of the content of frame sequences. An experimental environment called VideoSTAR (Video STorage And Retrieval) was developed to highlight issues related to searching and browsing a shared video database. The VideoSTAR integrated video tool environment consists of a video player, a tool manager, and tools for searching, browsing, and registration of meta-data. A registration tool is used to register both content annotations and structure information. VideoSTAR contains a video query module that implements the algebra operations extended from the standard set operators. The operators take into account the temporal nature of the mapped video objects. The video query tool offers the query algebra directly to the user, thus it is unsuitable to users unfamiliar with the system. Furthermore VideoSTAR does not support query based on visual features and spatio-temporal relations between video objects.

The Logical Hypervideo Data Model (LHVDM) is based on a video abstraction hierarchy and semantic content descriptions (Jiang, & Elmagarmid, 1998). The video query language allows users to query and retrieve video based on content descriptions with spatial and temporal constraints. The multi-level data abstractions provide data independence, multi-user view sharing, and data reuse. However, video data access based on visual information is not supported.

With the emergence of vast amount of visual data the difficulties brought about by textual annotation approach becomes acute. To overcome them visual feature-based approach was proposed (Rui et al., 1999). Visual content can be modeled as a hierarchy of abstractions. At the first level are the raw pixels containing information about colour or brightness. Upon further processing yields features such as edges, corners, lines, curves, and colour regions. A higher abstraction could combine and interpret these features as objects and their attributes. At highest level are the human concepts of the objects and their relationships. Although low-level image features such as color, texture, shape and motion vectors of objects can be extracted and indexed, high level concepts and object attributes cannot be accurately extracted by automatic methods.

In the JACOB system (Ardizzone, & Cascia, 1997) a given video is automatically split into a sequence of shots, and a set of representing frames is extracted. The color, texture and motion features of these frames are extracted and stored as descriptors. User queries on these features are supported. However, queries on semantic features of the video data are not supported.

The VideoQ is a web-based system (Chang et al., 1997) that supports visual search of video database in terms of colour, texture shape, and object motion trails. Queries are input by sketching objects and specifying their visual features. A video object is a collection of regions that are grouped together under some criteria across several frames. A region is defined as a set of pixels in a frame, which are homogeneous in the features of interest to the user. For each region, the low-level features are extracted. These regions are further grouped into higher semantic classes known as video objects. Motion is the key attribute in VideoQ and the motion trajectory interface allows users to specify a motion trajectory for an object of interest. Users may also specify the duration of the object motion. Video queries are formulated by animated sketches. Users draw objects with a particular shape, paint color, add texture and specify motion to pose a query. Objects in the sketch are matched against those in the database and a ranked list of video shots complying with the requirements is returned. This system was developed to investigate the full potential of visual cues in content-based video search. Users can also search by keywords since video clips are catalogued into subject taxonomy, so users may navigate through the catalogue. However users can only perform homogeneous queries, in other words only a single type query can be performed at a time.

Ponceleon (Ponceleon, et. al., 1998) developed semi-automated techniques that combine manual input, and video and speech technology for automatic content characterization integrated into a single system called CueVideo. CueVideo integrates voice and manual annotation, attachment of related data, visual content search technologies (QBIC$^{tm}$), and novel multi view storyboard generation to provide a system where the user can incorporate the type of semantic information that automatic techniques would fail to obtain. An abstraction, which is representative of a record in the specific domain, was created. The primary abstraction they attempted to catalogue was technical talk. Each talk was represented as a combination of metadata in a relational database and the associated digital content on the file system. Other tables were created to store the logical groupings, annotations and video storyboard. However, the transcript generated by the speech recognition component has yet to be fully utilized for text indexing. The retrieval phase is envisioned as a web browser based interface through which end users may search, browse and view the content of a digital library. The users will be given several options in displaying the storyboard, and also an option to play selected segments of the video. Their work however, did not elaborate on the query formulation or the underlying query language.

Apart from the two main approaches described above, video content-based retrieval can also be based on other attributes of video data such as object's motion, cinematic effects like zooming, panning, etc., as well as spatio-temporal characteristics.

A prototype video database management system called BilVideo provides full support for spatio-temporal queries that contain any combination of spatial, temporal, object-appearance, external-predicate, trajectory-projection, and similarity-based object-trajectory conditions by a rule-based system built on a knowledge-base (Dondeler et al., 2003). BilVideo uses a rule-based approach to model spatio-temporal relations between objects. The query formulation is through a visual query interface. Users may query the system with sketches, and a visual query is formed by a collection of objects with some conditions, such as object trajectories with similarity measures, spatio-temporal orderings of objects, annotations, and events. Although they claimed that the system supports query on semantic, there were no description available. Furthermore, queries on video structure are not supported.

A video data model that extends the DISIMA image data model was proposed (Chen et al., 2003). Video components were added and links were set up between image and video data. The model expresses the semantics of video data content by means of salient objects and relationships among them. Connections between video data and DISIMA images are made through key frames, which are extracted from shot. Based on these connections, techniques used to query image data may be used to query video data. In addition, a set of new predicates has been defined to describe the spatio-temporal characteristics of salient objects in the video data. MOQL is used as a query language. This model however, lacks representation on video structure.

Many systems integrate querying at different levels. The most common approach is to integrate querying at the feature with querying at the semantic level. However, only homogeneous queries are allowed at all these levels. None of the systems or query languages supports hybrid queries at all these levels. In other words, none of the systems or query languages enables a user to specify textual, visual, motion, spatio-temporal, and audio attributes in the same query. To support hybrid querying at different levels the underlying data model must be able to deal with all different variations of video content. Furthermore the query language should support querying at different levels, namely, structural and content level. Another important characteristic of a video query language is visualization. In a multimedia environment where most information is inherently visual, it is more appropriate to query data using a visual query interface. The next section explains the basics on queries.

## 3.0 COMBINING EXACT AND SIMILARITY-BASED QUERIES

There are essential differences between querying multimedia data and conventional data from databases. The most important difference is that a response to a multimedia query typically provides not only a set of objects, but also the grades with which these objects qualify in terms of similarities. This is in contrast to a

traditional database where the answer to a query is simply a set of values.

An unweighted query Q consists of n atomic queries $q_1,\ldots,q_n$ which are regarded as predicates. A compound query (n > 1) combines atomic queries using the binary sentential connectives **and** ($\wedge$) and **or** ($\vee$). Beside these connectives the monadic connective, the negation operator **not** ($\neg$), is likewise important. A query can be defined as follows: Q := q | (Q [$\wedge|\vee$] Q)| $\neg$Q | (Q).

Atomic query for multimedia data is much harder to evaluate than an atomic query in a relational database. For example, when querying images it is reasonable and natural to ask for images that are somehow similar to some example image. In response to a query, a multimedia system might typically return a sorted list of items in the database that match the query best. Hence, there must be a notion of similarity, so that the most similar images are retrieved.

Compound queries are Boolean combinations of atomic queries. In compound query there could be a mismatch where the result of some queries is a sorted list, and for other queries, it is a set. These differences cause us to consider new mechanism to calculate the overall score. How do we combine such queries in Boolean combinations?

A solution for combining traditional database query and multimedia query was proposed in terms of "graded" or "fuzzy" sets (Fagin, 1996). A graded set is a set of pairs (x, g), where x is an object (such as a tuple), and g (the grade or score) is a real number in the interval [0,1]. We can think of a graded set as corresponding to a sorted list, where the objects are sorted by their grades. Therefore, a graded set is a generalization of both a set and a sorted list.

A number of different rules for evaluating Boolean combinations of atomic formulas in fuzzy logic are given as follows (Fagin, 1998). Consider the standard rules of fuzzy logic, where if x is an object and Q is a query, then let $\mu_Q(x)$ denote the score or grade of x under the query Q.

If we assume that $\mu_Q(x)$ is defined for each atomic query Q and each object x, then it is possible to extend to queries that are Boolean combination of atomic queries via the following rules.

**Conjuction rule:**

$\mu_{A\wedge B}(x) = \min \{\mu_A(x), \mu_B(x)\}$

**Disjuction rule:**

$\mu_{A\vee B}(x) = \max \{\mu_A(x), \mu_B(x)\}$

**Negation rule:**

$\mu_{\neg A}(x) = 1 - \mu_A(x)$

Details of formula for incorporating weights in scoring rules can be found in (Fagin, 1996). The next section discusses the video database system.

## 4.0 VIDEO DATABASE SYSTEM

Our Video Database System (VDBS) has three main modules namely the video shot detection, annotation and query interface, which are connected to an object-relational database management system. In VDBS a video clip is segmented into shots by the video shot detection module. A key frame is chosen to represent each shot. This still image will be stored in the database and its signature will be computed during the database population. Each video will be annotated in the video annotation module, which consist of three sub modules, namely the shot, scene and sequence annotation modules. Each shot will be textually annotated in the shot annotation module. Related shots can be grouped together to form a scene. Scenes are annotated in the scene annotation module. Groups of scenes can be composed to form a sequence. Sequences are annotated in the sequence annotation module. All text attributes are stored in the database. Each video shot is also represented by a key frame image. Low-level features of these images are extracted and stored in the object relational database. Through the query interfaces, users will be able to formulate queries on video structure, semantic as well as low-level features. The next section describes the query interface. The query interfaces provide three types of query formulation, namely query by text, query by image, and hybrid query formulation. These three types of query formulation are provided in order for us to make the comparison between single query modes against hybrid query formulation.

## 5.0 EXPERIMENTS AND RESULTS

Through the prototype VDBS, we compared the query results of single types query formulation against those obtained from hybrid query type formulation. VDBS is a video database system that supports query on semantic and low-level features of video. More importantly it supports hybrid query formulation, which is very useful since query by text or query by image alone is insufficient. VDBS caters for any video category where huge amount of video data needs to be searched. In this section, we present an example application for VDBS, which is searching news archive. In the broadcasting industry, searching for relevant video clips from news archives is a daily activity. Those involved in the preparation of video footage for news broadcasting needs to search through vast amount of archive materials in search of relevant video segments to republish. The current approach for accomplishing this task is through keyword search that was provided while cataloging the news. The video collections are not indexed by their content, thus searching for relevant video segments would require the person to sequentially view the whole video. Hence some other search mechanisms are needed to facilitate the search. The next section will describe our experiment using an example application to compare query formulation by text, image and a hybrid of text and image.

To illustrate the query formulation supported by VDBS, a news segment broadcasted by a local television channel was used as an example. Each segment can be annotated based on the visual as well as audio features. The news video is a 30 minutes clip and to generate the key frame images, the video was preprocessed using the Video Shot Detection module. By manually previewing the news video frame by frame, we can identify the exact number of shots which was then used to compare with the results from the VDBS Video Shot Detection module. A total of 352 shots represented by 352 key frames were obtained. Visual features are extracted from each key frame and they are stored in the database. Using the Video Annotation Module each of the 352 shots was annotated. Objects, activities, events as well as description of shots are stored in the database.

### 5.1 Categories of Query

The following sections show several examples of query formulation and the results. For each category of query, namely Query by Text, Query by Image and Hybrid Query, three different queries were formulated. In the experiments, key frame images representing sea, football and woman were used. Based on our experiments on query formulation using single types and hybrid query mechanism we plotted graphs to compare the number of images retrieved for various threshold values.

Figure 1 shows a comparison between query by image and hybrid query for 'sea'. The graph plots the threshold values against the number of images retrieved. The result shows the Hybrid Query type gives better results compared to Query by Images since the numbers of images retrieved were reduced. The result of the Query by Text was not shown in the graph as the threshold values are not relevant to this query type. Nevertheless, in the Query by Text there were 16 images retrieved. When compared to Query by Text, the Hybrid Query results was the same for threshold values 14 and above. However, by lowering the threshold value, the number of images retrieved was reduced.

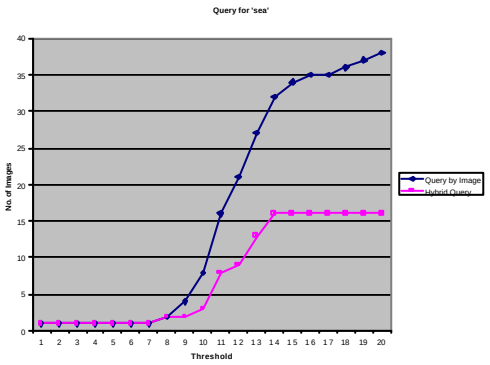


*Figure 1: Comparison between Query by Image and Hybrid Query for 'sea'*

Figure 2 shows a comparison between Query by Image and Hybrid Query for 'football'. The graph plots the threshold values against the number of images retrieved. The result shows the Hybrid Query type gives better results compared to Query by Image. In the Query by Text there were 33 images retrieved. When compared to query by Text, the Hybrid Query also shows better results where the number of images retrieved was less than 33. In fact by reducing the threshold values, the number of images retrieved was also reduced.

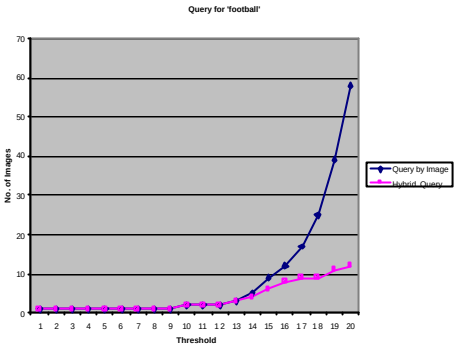


*Figure 2: Comparison between Query by Image and Hybrid Query for 'football'*

Figure 3 shows a comparison between Query by Image and Hybrid Query for 'woman'. The graph plots the threshold values against the number of images retrieved. Again, the result shows the Hybrid Query type gives better results compared to Query by Image. In the Query by Text there were 18 images retrieved. When compared to query by Text, the Hybrid Query also shows better results where the number of images retrieved was less than 33. In fact by reducing the threshold values, the number of images retrieved was also reduced.

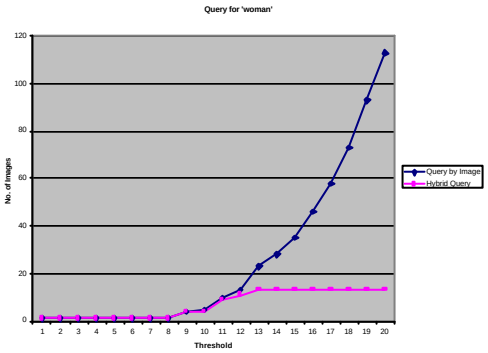


*Figure 3 : Comparison between Query by Image and Hybrid Query for 'woman'*

## 6.0 CONCLUSION

Query by text assumed that the video data has been sufficiently annotated, which is not reasonable for a huge amount of multimedia data. Nevertheless, query by text is more natural to end users. On the other hand, not every image can be described textually and hence query by example becomes useful. Thus, query by text complements query by example. Although many systems support these types of queries, none of them support hybrid query formulation. This is because query by text uses exact matching, while query by example uses similarity-based matching. These differences have

caused us to consider a combination of these two types of queries. Through the prototype VDBS, we were be able to compare the query results of single type query formulation against those obtained from hybrid query formulation. We also showed that through hybrid query formulations, we are able to get a better set of query results that are similar not only in terms of low-level features but semantic as well.

## REFERENCES


Ardizzone, E. and Cascia, M. Automatic Video Database Indexing and Retrieval. *Multimedia Tools and Applications* 1997; 4: 29-56.

Bimbo, A. D., Vicario, E. and Zingoni, D. Symbolic description and visual querying of image sequences using spatio-temporal logic. IEEE Transaction on Knowledge Data Engineering 1995;7(4): 609-622.

Chang, S. F., Chen, W., Meng, H. J., Sundaram, H. and Zhong, D. VideoQ – An Automatic Content-based Video Search System Using Visual Cues. In Proceedings of ACM Multimedia Conference 1997, Seattle, Washington.

Chen, L., Ozsu, T., and Oria, V. "Modeling Video Data For Content Based Queries: Extending the DISIMA Image Data Model," Proceedings of 9th International Conference on Multi-Media Modeling, Taiwan, pp. 169-189, January 2003.

Chua, T.S. and Ruan, L.Q. "A Video Retrieval and Sequencing System," *ACM Transactions on Information Systems*. Vol 13, No. 4, pp. 373-407, 1995.

Davis, M. Media Streams: An Iconic Visual Language For Video Annotation. In Proceeding of International Symposium on Visual Languages 1993; 196-202.

Donderler, M. E., Saykol, E., Arslan, U., Ulusoy, O., Gudukbay, U. 2003. BilVideo: Design and Implementation of a Video Database Management System. Kluwer Academic Publishers, Netherlands.

Fagin, R. "Combining Fuzzy Information from Multiple Systems," Proceedings of ACM Symposium on Principles of Database Systems (PODS'96), pp. 216-226, Montreal, Canada. 1996.

Fagin, R. "Fuzzy Queries in Multimedia Database Systems", Proceedings of ACM Symposium on Principles of Database Systems (PODS'96), pp. 216-226, Montreal, Canada.1996.

Hjelsvold, R., Midstraum, R. and Sandsta, O. "Searching and Browsing a Shared Video Database." *Multimedia Database Systems: Design and Implementation Strategies*, pp. 89-122. Kluwer Academic Publishers. 1996.

Jiang, H. and Elmagarmid, A. K. WVTDB – A Semantic Content-Based Video Database System on the World Wide Web. *IEEE Transactions on Knowledge and Data Engineering* 1998;10(6):947-966.

Oomoto, E. and Tanaka, K. OVID: Design and Implementation of a video object database system. IEEE Transaction on Knowledge and Data Engineering, 1993:5:629-643.

Ponceleon, D., Srinivasan, S., Amir, A., Petkovic, D. and Diklic, D. Key to Effective Video Retrieval: Effective Cataloging and Browsing. In Proceedings of the ACM Multimedia 1998; 99-107.

Rui, Y., Huang, T.S. and Chang, S. "Image Retrieval: Past, Present, and Future," *Journal of Visual Communication and Image Representation*, Vol 10, pp.1-23. 1999.