7 A MULTI-MODEL FRAMEWORK FOR VIDEO INFORMATION SYSTEMS

Uma Srinivasan, Craig Lindley, and Bill Simpson-Young

CSIRO Mathematical and Information Sciences
North Ryde, NSW, Australia

Abstract: In order to develop Video Information Systems (VIS) such as digital video libraries, video-on-demand systems and video synthesis applications, we need to understand the semantics of video data, and have appropriate schemes to store, retrieve and present this data. In this paper, we have presented an integrated multi-model framework for designing VIS application that accommodates semantic representation and supports a variety of forms of content-based retrieval. The framework includes a functional component to represent video and audio analysis functions, a hypermedia component for video delivery and presentation and a data management component to manage multi-modal queries for continuous media. A metamodel is described for representing video semantic at several levelss. Finally we have described a case study - the FRAMES project - which utilises the multimodel framework to develop specific VIS applications.

7.1 INTRODUCTION

Video information systems (VIS) such as digital video libraries, video-on-demand systems, and video synthesis applications introduce new challenges in the management of large collections of digital video and audio data with associated texts, images, and other objects. In order to manage digital videos, we need to understand the semantics of video data, and have appropriate schemes to store, retrieve and present it. Researchers have studied video data from different perspectives. The pattern recognition community has largely concentrated on image data in video and has come up with algorithms that can detect patterns in the data at the visual level (Aigrain *et al*, 1996). The database community has focussed on logical structures that facilitate indexing of video sequences for retrieval purposes. In order to manage large

digital collections, we need a video[1] management strategy that can exploit the research outcomes of both of these groups, and at the same time manage the complex semantics of continuous visual media where interpretations are subjective and domain dependent. A conceptual model for such applications must address a number of issues.

It should be capable of representing the high level semantics of a visually rich medium such as a video.

(i) It should support operators and constructors that allow manipulation of a continuous (temporal) medium such as video.

(ii) It should be able to represent low level visual features such as shot boundaries, camera operations, and object tracking, which can be extracted from video data streams.

(iii) It should be capable of modelling audio features such as distinct sound patterns present in the audio stream of a digital video.

(iv) It should be supported by a storage model that supports storage and delivery of video sequences, based on their features.

(v) It should include presentation functions that can provide navigation and browsing facilities to view videos.

(vi) It should provide video composition functions to generate virtual videos.

In section 7.2 of this paper we explore the semantics of video data and present a metamodel with 8 semantic levels, in section 7.3 we identify the requirements of a typical VIS application, and in section 7.4 we present an integrated modelling framework for designing VIS applications. The primary purpose of this framework is to present a unified approach for video management, which includes functions for creating, storing, managing and presenting video applications. In section 7.5, we present a case study - the FRAMES project - to illustrate the application of the proposed multi-model framework in developing an experimental environment to provide access to video material.

Much of the work on developing conceptual models for multimedia data has resulted in models that address specific issues of multimedia data such as representation of image features (Gupta 1997, Flickner, et al. 1995), synchronisation and presentation (Adjeroh and Lee 1996), video navigation and browsing (Simpson-Young and Yap 1996, Arman et al 1994) and models for video management, (Zhang, Low and Smoliar 1995, Hjelsvold, Midstraum and Sandsta 1996). While each of these functions is essential, focussing on only one model is not sufficient to address all aspects of delivering an interactive video application. This paper takes a more holistic approach to modelling and presents a multi-model approach for developing VIS applications. While many of the individual modelling levels have been demonstrated by other research groups, FRAMES is unique in that it

[1] When using to the term *video*, we include audio and video components unless the context dictates otherwise.

integrates these into a comprehensive overall framework. The FRAMES project has developed most of the elements of this framework, together with appropriate processing modules and interfaces. Ongoing research is addressing the specific content of the different models.

7.2 SEMANTICS OF VIDEO INFORMATION SYSTEMS

The conceptual model of any information system has to represent objects and relationships well understood by users in a given application domain. For VIS applications, in addition to application domain objects, a conceptual model has to include video objects, meanings associated with those video objects, and other associated objects and attributes that are derived from video data. Film semiotics, pioneered by the film theorist Christian Metz (1974), has identified five levels of cinematic codification that cover visual features, objects, actions and events depicted in images together with other aspects of the meaning of the images. These levels, all of which can be represented within a VIS metamodel, are:

1. the *perceptual level*: the level at which visual phenomena become perceptually meaningful; the level at which distinctions are perceived by a viewer within the perceptual object. This level includes perceptible visual characteristics, such as colours and textures. This level is the subject of a large amount of current research on video content-based retrieval (see Aigrain *et al*, 1996).
2. the *cinematic level*: the specifics of formal film and video techniques incorporated in the production of expressive artefacts ("a film", or "a video"). This level includes camera operations (pan, tilt, zoom), lighting schemes, and optical effects. Automated detection of cinematic features is another area of vigorous current research activity (see Aigrain *et al*, 1996).
3. the *diegetic level*: at this level the basic perceptual features of an image are organised into the four-dimensional spatio-temporal world posited by a video image or sequence of video images, including the spatiotemporal descriptions of agents, objects, actions, and events that take place within that world. An example of an informal description at this level may be "Delores Death enters the kitchen, takes a gun from the cutlery drawer and puts it into her handbag". This is the "highest" level of video semantics that most research to date has attempted to address, other than by associating video material with unconstrained text (allowing video to be searched indirectly via text retrieval methods, eg. Srinivasan *et al*, 1997).
4. the *connotative level*: metaphorical, analogical, and associative meaning that the denoted (ie. diegetic) objects and events of a video may have. The connotative level captures the codes that define the culture of a social group and are considered "natural" within the group. Examples of

connotative meanings are the emotions connoted by actions or the expressions on the faces of characters, such as "Delores Death is angry and vengeful", or "Watch out, someone's going to get a bullet!".

5. the *subtextual level*: more specialised meanings of symbols and signifiers. Examples might include feminist analyses of the power relationships between characters, or a Jungian analysis of particular characters as representing specific cultural archetypes. For example, "Delores Death violates stereotypical images of the passivity and compliance of women", or "Delores Death is the Murderous Monster Mother".

Lindley and Srinivasan (1998) have demonstrated empirically that these descriptions are meaningful in capturing distinctions in the way images are viewed and interpreted by a non-specialist audience, and between this audience and the analytical terms used by film-makers and film critics. Modelling "the meaning" of a video, shot, or sequence requires the description of the video object at any or all of the levels described above. The different levels interact, so that, for example, particular cinematic devices can be used to create different connotations or subtextual meanings while dealing with similar diegetic material.

Despite their importance, the connotative and subtextual levels of video semantics have generally been ignored in attempts to represent video semantics in computing science research to date, despite being a major concern for film-makers and critics[2]. Perhaps one reason for this is that these are not levels of annotation that can be expected to be automated in the short term (in the longer term, memory- and case-based approaches may make some degree of automation feasible). The requirement for the provision of these descriptive annotations changes the production model for interactive video systems (compared with the production model for tradition linear video productions). Authorship is no longer limited to the assembly of the video data, but extends to the generation of descriptive annotations that facilitate and constrain the ways in which the video data can be interactively and dynamically reused. Authorship also extends to high level virtual video prescriptions upon which virtual video generation is based (see below).

All of the above five levels concern the description of the visual and auditory content of video data. We can also identify a level of descriptive information that describes a video production or component without describing its visual and auditory content:

6. the *bibliographic* level is concerned with information about the video such as production details (production crew and principle creative authors, year, etc.), inventory information such as original film gauge or type of tape, standard and format, etc.

---

[2] This may be due to video semantics work being closely aligned with automated video analysis work with its origins in robotics and industrial/manufacturing applications.

Finally, in addition to these six levels, for digital video systems we can identify two additional levels concerned with the semantics of digital video data representation and presentation:

7. the *system* level, concerned with operating systems and networked data file storage and manipulation. This is the level at which the video data is simply "a file" that can be moved, copied, etc.
8. the *video data* level, concerned with the interpretation of the contents of a video data file as a time-ordered series of frames, each substructured as an array of colour values corresponding to a grid of pixels. This level also includes various video compression and encoding models.

An orthogonal way of modelling a video sequence[3] is to distinguish a set of its attributes representing those patterns (features) in the data that can be automatically detected from the bit streams from those attributes that must be manually defined. We shall call the former attributes the *data-driven attributes* of a video sequence. Patterns or features that can be detected so far mostly belong to the perceptual level and cinematic level of codification. By this definition, we recognise that the data-driven attributes that can be determined will change (in particular, increase in number and type) with evolving feature detection technology. The second set of attributes of a video sequence in this orthogonal scheme contains authored models/descriptions of a video. These attributes are manually annotated descriptions of video content, including descriptions of automatically detectable features. In this paper we shall refer to such attributes as *authored -description attributes.*

Some important aspects of video semantics cut across multiple content levels. For example, issues concerned with temporal sequencing of video include elements at the video data level (frames), the cinematic level (shots), and the diegetic level (a scene as a sequence of shots implied by location and time of day). Hence the time order of a video sequence can be subdivided at various levels of granularity. Figure 7.1 shows the different abstraction levels of this subdivision. These levels of subdividing a sequence represent different sizes of video object on which search and retrieval can be conducted; ideally all of these levels should be accommodated in the conceptual schema of a VIS application.

[3] Video sequence here is a generic term that represents any level of video abstraction such as clip, scene and shot.

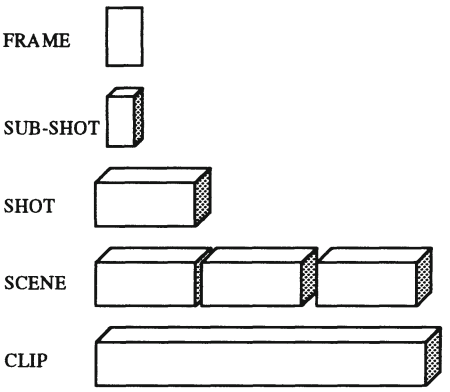

Figure 7.1: Levels of abstraction of a video sequence

Different representations may be used to model these descriptions, with each representation supporting different forms of query or computational operation upon the description. Alternate representations may include textual descriptions or structured database models, such as relational or object models. Text descriptions support search by ranked text retrieval routines using free text descriptions as queries. Relational models support search by SQL queries. Complex object models can depict object hierarchies, allowing search on (or via) subtype/supertype relationships, and can include object encapsulation to represent different abstraction levels of video sequences. The attributes of complex video objects represented in the model could be attributes that are generated automatically by video analysis tools.

7.3 FUNCTIONAL REQUIREMENTS OF A TYPICAL VIS

A typical VIS requires some component sub-systems to manage and present video data. A few key modules are described here. User requirements will have to determine which of these modules are essential for a given application.

7.3.1 Video Analysis

In order to manage digital video sequences, we need video analysis tools to partition the video into the various levels of temporal grouping described above. A shot is usually indexed by a key frame, ie., a still image that represents the shot. Key frames can be further indexed by content-based features such as colour, texture and shape, using image analysis software such as (Gupta 1997). Additionally, shots can also be characterised by camera work such as pan and zoom operations (Aigrain, *et al*, 1996, Gu, *et al*, 1997).

7.3.2 Audio Analysis

The audio signals available in digital videos can provide valuable information when analysing video programs. Although speech recognition has proved to be a difficult problem, other aspects of audio analysis such as aligning speech with textual transcripts (Robert-Ribes and Mukhtar 1997) has been found to be very useful. In some cases audio signals can provide valuable information about non-audio events; for example in the sports domain, loud sound bursts such as a crowd cheer indicate the highlights of a sporting event. Other tools for audio content processing and analysis are reported in (Pfieffer *et al*, 1996).

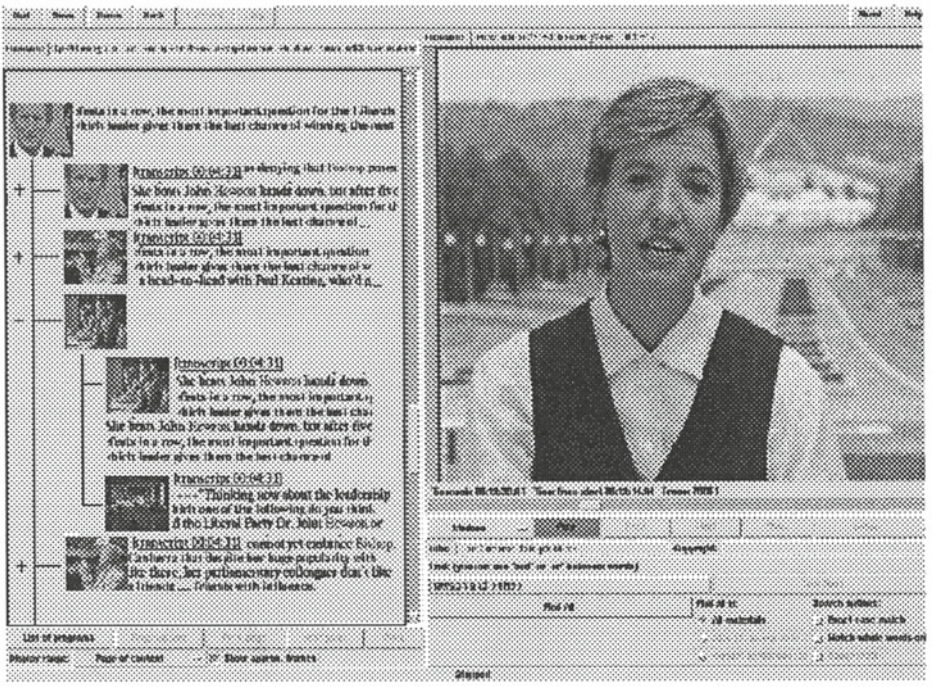

Figure 7.2: Hierarchical browsing using multiple representations (images © Australian Broadcasting Corporation)

7.3.3 Interaction and Presentation

As video is a visually rich medium, users often need to browse through a video before saving a required video sequence. Navigation and browsing tools, therefore, form an important component of a VIS application. Effective browsing through video material requires a combination of actual video material and other available alternate representations of the material such as shotlists containing descriptions of the sequence, transcripts of audio content, and sets of key frames (Yap, Simpson-Young and Srinivasan, 1996). Figure

7.2 shows an example of such an interface using the FRANK client (Simpson-Young and Yap 1996).

7.3.4 Video Composition

Although conceptual models augmented with video analysis tools can be used directly for search and retrieval of video content, applications such as automatic video generation and virtual video synthesis require specific video composition rules as criteria for generating virtual videos from an archive of video clips. Virtual videos can be specified by an author in the form of virtual video prescriptions (Lindley and Vercoustre, 1998). A virtual video prescription is a high level description of the structure of a (possibly interactive) video program. Interpretation of a prescription involves the interpretation engine reading queries embedded in the prescription and sending them to appropriate query processors that will return references to video components that satisfy those queries. The references are then sent by the interpreter to a video server for execution, thereby generating the dynamic virtual video presentation for the user of the system. Although a virtual video prescription contains embedded database queries expressed in terms of both data-driven attributes and authored-description attributes of the video, the prescription itself is also a data structure that can be searched and queried upon. The overall VIS schema may therefore accommodate the representation of virtual video prescriptions, either as generic (searchable) text files, or structured according to specific document type definitions (DTDs).

7.3.5 Data Management

As video information is a combination of visual, audio and text material, a VIS application should be capable of handling multi-modal queries, where each mode or media traditionally has a different indexing mechanism. For example, a sports channel may wish to retrieve a video sequence that shows *'a close-up of Davidson's play accompanied by a loud cheer'*. In this query the *close-up* is a framing characteristic (cinematic level) defined on a diegetic object (character), the *loud cheer* is an audio signal and the description about *Davidson's play* could be textual information[4]. That is, a single query may involve accessing data represented by multiple data types. The data management function has to offer support for managing video objects whose attributes could be audio/video feature extraction functions, text documents and key frame images.

Traditional database management systems that store multimedia data as binary large objects (BLOBS) are no longer adequate to handle functional

[4] A close-up could be represented as a manual annotation or automatically detected, either on demand or during a prior video analysis operation.

attributes that may be generated in real-time. A database management system to store video objects requires the ability to extend traditional data types with suitable operators for manipulating video sequences, constructors for video composition and delivery mechanisms for presentation. The requirements of a multimedia database management system have been discussed in a number of recent papers (Nwosu *et al* Eds 1996). The Object-relational model is well suited to be the logical model for VIS applications as it supports derived data types and abstract data types to represent video, audio and image data (Subrahmanyam 1997).

The requirements specified thus far need a modelling approach that can model complex video objects, generate functional attributes, support specification of video composition rules within a virtual video prescription and provide presentation capabilities that can support navigation and browsing.

7.4 A FRAMEWORK FOR DESIGNING VIDEO APPLICATIONS

In order to develop VIS applications, we need a modelling framework that is rich enough to support the video semantics described in section 7.2, and robust enough to meet the demands of evolving technology in the areas of video content analysis and automatic video generation, as described in the previous section.

Video analysis requires extensive computation of digital data. While the objects involved in the computation could be represented within an object-oriented model (or any semantic model), the emphasis in this case lies in describing the computations involved. This calls for a thrust towards a functional component, as the functional modelling approach is well suited for non-interactive programs, where the main purpose is to compute a function. By contrast, databases often have a trivial functional model, since their purpose is to store and organise data, not to transform them.

In the case of the video presentation, the emphasis is on accessing information in a structured way, which is a typical hypermedia application requirement. Design of hypermedia applications involves capturing and organizing the structure of complex domain and making it clear and accessible to users (Tomas *et al* 1995). The hypermedia model provides a control structure that supports navigation through data based on its content. While this is ideal where nodes consist of persistent data such as text, finding a general way to indicate dynamic media such as a portion of video or audio is a difficult problem. This requires augmenting a hypermedia model with a storage model that supports multimedia data types and complex temporal relationships among data items that support high level presentation semantics. (Hardman, *et al* 1994).

A VIS application model therefore has to draw from both (functional and hypermedia) modelling approaches in order to meet the VIS requirements.

These models need to be supported by a storage model that supports operations on complex data types and temporal relationships.

Given these requirements, now we present an integrated multimodel framework to support the complex environment of a VIS application.

7.4.1 The Integrated Multi-model Framework

Figure 7.3 shows the proposed integrated framework for developing VIS applications. The framework illustrates the different modelling components that are appropriate for delivering the functional modules of a VIS application. The modelling components are described in greater detail in the rest of this section.

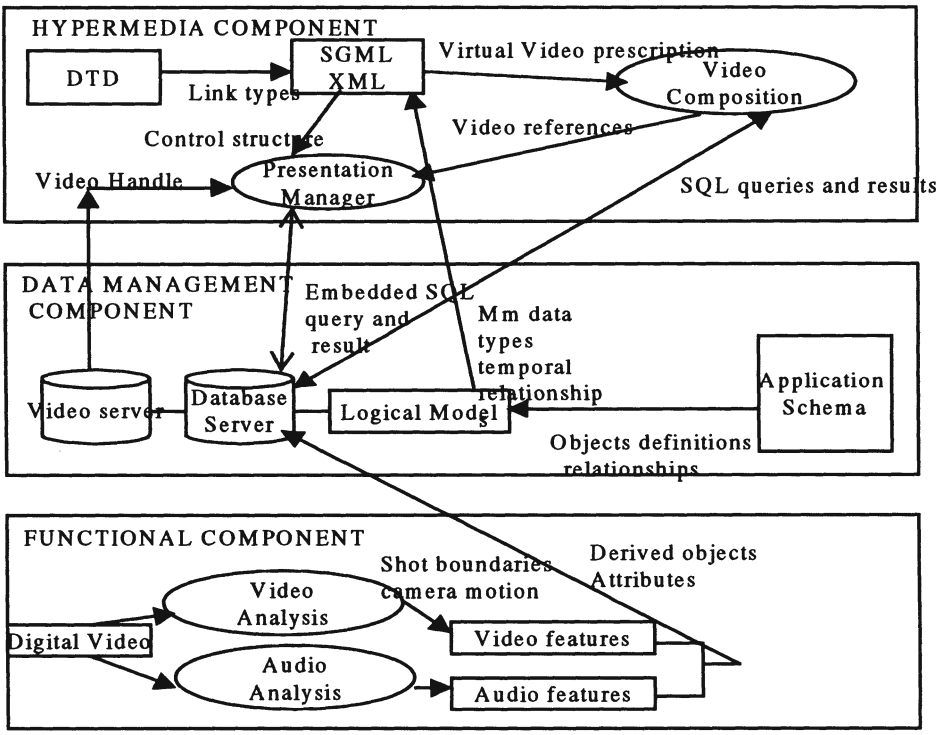

Figure 7.3: An Integrated Framework for VIS applications

In figure 7.3, the functional component of the application deals with the modules that are computation-intensive, such as video analysis and audio analysis. The features automatically generated by the video analysis and the audio analysis components are stored as derived objects and attributes in the database. The hypermedia component deals with the presentation manager that uses an SGML or XML document to support video navigation and browsing. (In a web-based environment environment, the presentation manager could have components on the server side eg, CGI script, and/or on the client side eg, a Java applet or a Java script code.) A video composition

process within the hypermedia component reads queries from a SGML/XML document and uses them to query the contents of the database. The complex data types used to represent video data are available to the presentation manager from the logical database model. The logical model is created from the VIS application schema. The application schema represents video semantics at the different levels discussed in section 7.2.

7.4.1.1 Functional Component. Digital video data forms the input to the video analysis module. The video analysis module identifies features determined by patterns in the video data. The type of features could be camera shots, camera type such as pan, zoom, object identification etc. Many algorithms are available for shot detection, ie, parsing videos into distinct shots (Arman, Hsu and Chiu 1993, Feng, Meng, Juan and Chang 1995, Zhang, Low and Smoliar 1995, Gu, Tsui and Keightley 1996). Such algorithms that deal with the non-trivial data structures are best specified as part of the functional component in the integrated framework. Similarly the audio processing module also forms part of the functional component.

Video parsing and audio analysis are usually performed before the video is made available for browsing, navigation and querying. Both video analysis and audio analysis functions produce derived objects and attributes that are stored in the database.

While a camera shot could form a fundamental browsing unit, a group of shots may also be aggregated into a meaningful unit or 'scene' for querying video content. What constitutes a scene has to be determined by the user-requirements of the video application.

7.4.1.2 Hypermedia Component. The hypermedia design approach is well suited for presentation of videos as it offers control structures well suited for navigation and browsing. The presentation manager has to support navigation through a video at several levels of abstraction, and also allow navigation across multiple videos in applications such as digital video libraries and virtual video generation. Management of these functions is facilitated by the use of structured documents.

A structured document basically separates the presentation style from logical structure and content. A DTD or Document Type Definition provides a framework for the elements that constitute a structured document and also specifies the rules and relationships between the elements that constitute a document. Using a standard such as SGML or XML it is possible to define a structured document that has elements to describe the semantic structure of a video. In FRANK (Simpson-Young and Yap 1996) this is achieved by using the Text Encoding Initiative (TEI) DTD which supports a semantically rich document structure.

The presentation manager uses the control structures and the links in a structured document to facilitate authoring as well as querying contents of the Video database. Using SGML-compliant tags it is possible to embed SQL queries within a document. The results of the query are displayed through the presentation manager.

In applications that require video composition, the video composition module utilises a standard such as SGML or XML to specify virtual video prescriptions which are read by the video composition component of the application. The video composition component directs queries represented in a prescription to appropriate query processors and sends the video components returned from the queries to the presentation manager. This generates a virtual video program from various levels of sequences, possibly within separate video data files.

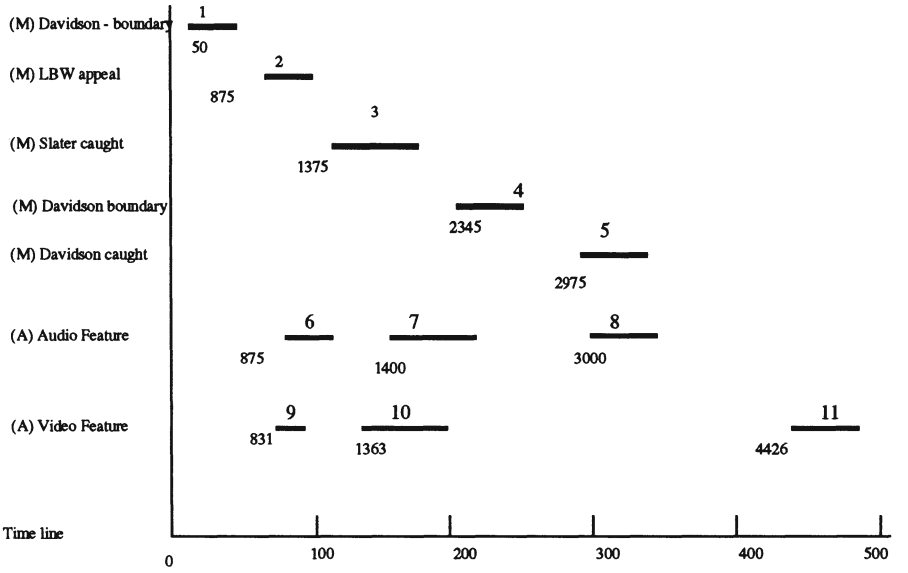

Figure 7.4: A Sports Sequence

7.4.1.3 Data Management Component. The object definitions and relationships are represented in the VIS application schema, which in turn is mapped onto the logical model of the database server. The logical model defines the multimedia data types and temporal relationships needed to represent video sequences. For VIS applications, the database server is usually connected to a media server to manage the high volume video data and deliver continuous video in a synchronised manner to the presentation manager.

Video data has temporal requirements that have implications on storage, manipulation and presentation. In the storage model, the logical representation of a video sequence is a time-ordered sequence of frames and operations on video sequences are temporal operations on time intervals. The following example illustrates the need for a temporal mapping of video objects.

A video sequence may spread over an interval of 5000 frames i.e., 3 mins and 20 secs. (A video in PAL format delivers 25 frames per second.) The objects and/or events of significance may occur at different intervals in the

sequence. Temporal attributes of a video sequence such as startOffset and endOffset map the data storage pattern onto a temporal presentation order. Figure 7.4 shows some significant events in a sports sequence. Events marked (M) are manually annotated events and events marked (A) are automatically detected events. In this example, the audio feature is a loud burst of sound, and the video feature is an automatically generated camera zoom-in operation. The automatically detected camera shot boundaries (not shown in the diagram) are present at different points within this interval.

A query to retrieve Davidson's play accompanied by a close-up camera operation would require operations on time intervals 1, 4, 5, 9 and 10. In order to satisfy a comprehensive range of temporal ordering requirements in queries, we need a number of set operations on time intervals in order to reconstruct total time sequence that encapsulates all and only those subsequences that satisfy a query. In this case the subsequence order is (1⊕9⊕10⊕4⊕5). The ⊕operation here is defined as follows: the ⊕ operator takes two intervals and returns a new interval such that the start-time of the new interval is the lesser of the start-times of the two intervals and the end-time of the new interval is the greater of the two end-times. This definition ensures that any overlapping intervals or intersections (as in 1 and 9) are not duplicated and gaps between two intervals (as in 9 and 10) are included in the returned sequence. The ⊕ operator defined here is only one example of a temporal operator. Allen (1983) has shown that there are thirteen disjunct relationships that can exist between two temporal intervals. In the context of videos, a comprehensive set of temporal operators have to be defined to handle a variety of queries incorporating specific temporal constraints.

The video application should have the capability to manage such temporal data and its associated relationships. Spatial relationships can also be represented in detail, supporting dedicated spatial query forms. However, the FRAMES project has not yet addressed spatial data in this level of detail.

Table 1 presents a summary of the essential components and the factors that need to be considered while developing a conceptual model for a VIS application. Column 1 shows the required components of a VIS application. Column 2 shows the services provided by these components. Column 3 shows the data structures and functions involved in delivering the type of services shown in column 2. Coulmn 4 shows the appropriate modelling methodology or formalism suited to deliver the component shown in col. 1.

The next section describes the FRAMES architecture that utilises the multimodel framework presented in this section.

7.5 THE FRAMES ARCHITECTURE

The functional architecture being used in the FRAMES[5] project utilises this multimodel framework to develop video content models for VIS applications.

Table 7.1. Summary of Data structures and Modelling Formalism

Requirement	Services	Data Structures and Functions	Modelling Formalism
Video analysis	Camera-shot boundary detection Categorisation of camera operations Video indexing by low level features	Derived objects and attributes, Complex computation functions to generate video objects and attributes	Functional Model
Audio Analysis	Detection of interpretive classes characterised by distinct sound patterns	Complex computation functions to generate audio objects and attributes	Functional Model
Video composition	Virtual video generation	Video composition functions, SGML/XML DTDS for presentation	Object relational model Hypermedia model - DTD
Video Interaction and presentation	Presentation generation Web-friendly navigation and browsing facility	SGML/XML DTDs for user interaction Document interpretation and management functions	Hypermedia Model -DTD
Data management	Database functions,	Object relational tables, complex objects, abstract data types, temporal mapping of video sequences Object-relational operations	Storage model to support mm data types

A core component of the architecture is a video semantics metamodel, which is a model of the different ways in which video semantics can be expressed. The metamodel represents video semantics based on film semiotics described in section 7.2. The primary hypothesis is that modelling the meaning of a video sequence can involve description of the video object

[5] The FRAMES project is being carried out within the Cooperative Research Centre for Research Data Networks established under the Australian Government's Cooperative Research Centre (CRC) Program.

at one or more of these levels. The metamodel is used as the canonical basis for the definition of specific content models for a given VIS application. Since all specific video content models are expressed in terms of objects, attributes and relationships that are predefined in the metamodel, and all queries on video data are also expressed in terms of the same metamodel, the system guarantees a commonality of language during the search process.

7.5.1 Video Semantics Metamodel

5.1.1 The Cinematic Level of Video Semantics. The cinematic level of video semantics is concerned with the specifics of how formal film and video techniques are incorporated in the production of expressive artefacts ("a film", or "a video") in such a way as to achieve artistic, formal, and expressive integrity. The process is complex, partially codified within various stylistic conventions, and tightly linked to other levels of meaning. Common examples of cinematic techniques (extracted from Arnheim, 1971) include shot characteristics, (eg. effect of angles, size, and relative placement on how one object is interpreted in relation to others in the diegetic space, etc.), mobile camera, shot speed (frame rate), optical effects, and principles of montage (eg. long strips for quiet rhythm, short strips for quick rhythm, climactic scenes, tumultuous action).

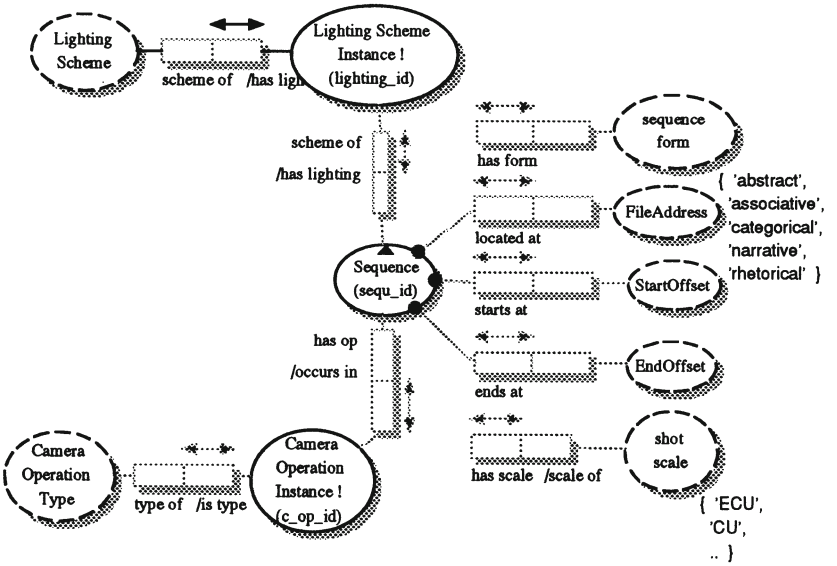

Figure 7.5: Schema for the Cinematic Level of Video Semantics

The initial FRAMES demonstrator includes very simple cinematic annotations, shown on the conceptual model of Figure 7.5. The model has been developed using the Object Role Modelling Formalisam (Halpin 1995). The model includes the file address and start and end offsets of a modelled video segment. A shot scale label can be used for sequences that correspond

with single shots. The sequence is also classified as having one of five primary sequence forms ie, abstract, associative, categorical, narrative, and rhetorical (described in Bordwell and Thompson, 1997). The FRAMES project has developed extensions to this model that include camera operations, lighting schemes, and other cinematic characteristics.

7.5.1.2 The Diegetic Level of Video Semantics. *Diegesis* designates the sum of a film's denotation: the narration, the fictional space and time dimensions implied in and by the narrative, the characters, locations, events, and other narrative elements considered in their denoted aspect (Metz, 1974). Based upon this definition, we define the diegetic meaning of video data as the four-dimensional spatio-temporal world that it posits, including agents, objects, actions, and events that take place within that world.

Davis (1994) considers a number of specific ontological issues in the representation of video semantics that address the diegesis of video data as a spatio-temporal world. The sequencing of shots allows the construction of many types of space (including real, artificial, and impossible spaces). For real locations it is possible to distinguish the actual location of the recording, the spatial location inferred by a viewer of an isolated shot, and the spatial location inferred when a shot is viewed in the context of a shot sequence. The virtual spaces created by videos require the use of relative three-dimensional spatial position descriptions. Video requires techniques for representing and visualising the complex structure of the *actions* of characters, objects, and cameras. For representing the action of bodies in space, the representation needs to support the hierarchical decomposition of units, spatially and temporally. Conventionalised body motions (walking, sitting, eating, talking, etc.) compactly represent motions that may involve multiple abstract body motions (represented according to articulations and rotations of joints). Much of the challenge of representing action is in knowing what levels of granularity are useful. *Time* (analogously to space) requires the representation of actual time and both possible and impossible visually-inferred time.

The initial FRAMES conceptual model includes five primary diegetic entity types, as shown on Figure 7.6: characters, objects, locations, speech acts, and actions. Diegetic modelling is an area of considerable ambiguity, since it can extend to arbitrary degrees of complexity in modelling the structure of any of these basic types and their interrelationships. A major modelling consideration is whether to include various details within the structure of the conceptual model, or to include them within more generic and unconstrained descriptive fields. For example, if a character performs an action upon another primary entity type, could this be modelled as an instance of a structural relationship with the other entity type, or could alternatively be incorporated into the "action description" associated with an action. Including the substructure of actions within the conceptual model supports more direct forms of processing upon action descriptions. If the detail is held within generic description fields, access to that detail requires

more complex and time-consuming processing of the internal content of descriptions. Processing the text of descriptions is made more difficult if the format of the text is not constrained; if it is constrained, then it is simpler to break this format information out into the conceptual model.

Decisions regarding the complexity of the conceptual model involve a trade-off between increasing the complexity of creating models on the one hand, and being able to process the models with more discrimination on the other. It is a major aim of the FRAMES project to gain an understanding of how much complexity is required in order to synthesise coherent video sequences of different types, and to develop tools and techniques for creating models having appropriate detail as efficiently as possible.

7.5.1.3 The Connotative and Subtextual Levels of Video Semantics. The connotative level of video semantics is the level of metaphorical, analogical, and associative meaning that the denoted (ie. represented diegetic) objects and events of a video may have. The connotative level captures the cultural codes that define the culture of a social group and are considered "natural" within the group.

The subtextual level represents a range of possible 'readings' or interpretations of video content, and hence is an important level of video semantics. The level of *subtext* corresponds to the level of hidden and suppressed meanings of symbols and signifiers, preceding and extending the immediacy of intuitive consciousness. The subtextual level is more specifically concerned with the levels of meaning that may *not* be immediately apparent to a reader (ie. viewer).

For both the connotative level and the subtextual levels, a definitive representation of "the meaning" of video content is in principle impossible. The most that can be expected is the development and representation of a body of evolving interpretations and their interrelationships.

An ideologically neutral content-based search and retrieval system must not restrict the range of possible interpretations of images. If such a system uses content representations, it must represent and support different views of content.

The FRAMES conceptual model accommodates connotations and subtext associated directly with video sequences, or indirectly with sequences via characters, locations, and objects.

7.5.1.4 Interactions Between Semantic Levels. Interaction between interpretation paradigms at the different levels of meaning are highly complex, which makes a universal film syntax impossible. The current version of the FRAMES schema includes very simple interconnections between levels (eg. Figure 7.7 shows interconnections between the cinematic and diegetic levels. Ongoing research will address the development of more systematic knowledge and information models to represent semantic interactions within specific film styles and genres. Interdependencies between levels create the need to protect schema and data integrity. For example, if a diegetic entity is deleted, and it has connotations associated

with it, there should be a systematic way of managing the referential dependency (it cannot be assumed that all deletions should cascade to dependent entities). Tools for managing data integrity are an area for ongoing research. The elements of FRAMES architecture are shown on Figure 7.8.

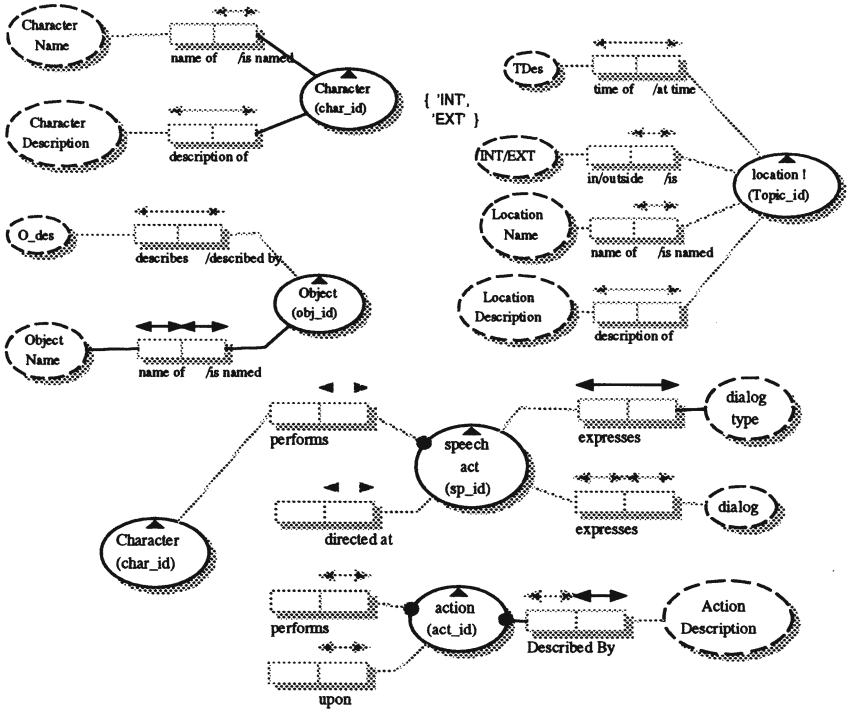

Figure 7.6: Schemas for the Diegetic Level of Video Semantics.

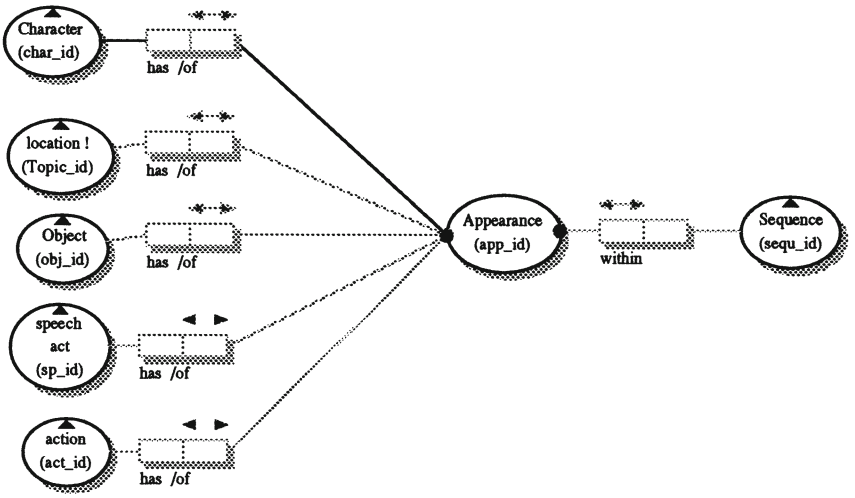

Figure 7.7: Schema Associating Diegetic Level Objects with the Cinematic Level of Video Semantics

7.5.2 Components of the FRAMES Architecture

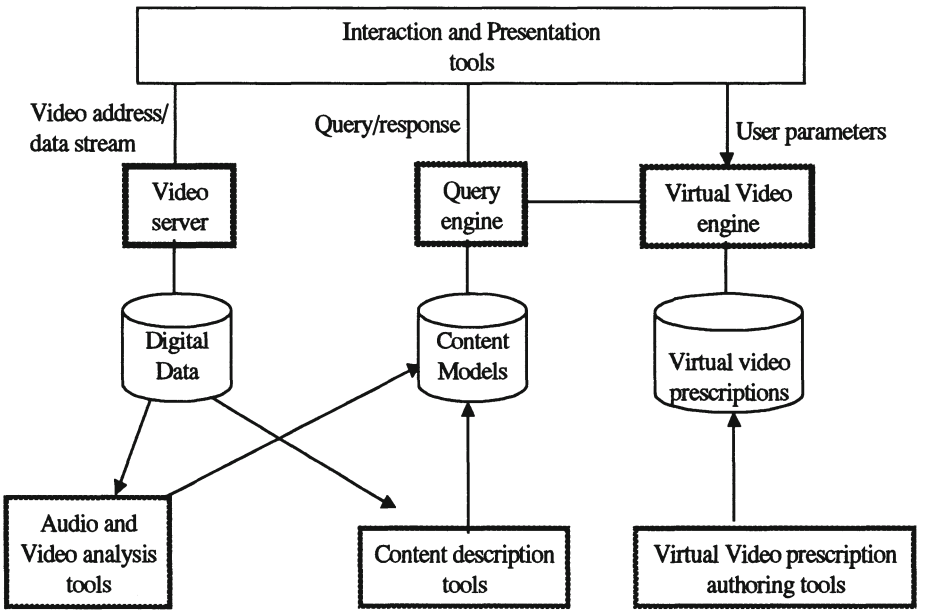

Figure 7.8: The FRAMES Architecture

7.5.2.1 Content description tools. The content description tool provides a way of developing a conceptual model for a specific VIS application. This model (also referred to as the content model) is developed using the semantics represented in the metamodel. The content model provides an integrated environment (section 7.4) and describes the video sequences, associated objects and operations on the objects as per the requirements of a given VIS application. Applying constraints on the conceptual model generates the logical model for the DBMS. The data model allows access and retrieval of video data by supporting appropriate indexing mechanism required during query processing. Queries can be specified directly by users using a *query authoring tool* (which is part of the interaction and presentation tools of figure 7.8) that provides structured interfaces for expressing queries in terms of components specified by the metamodel. Well-formulated queries (ie. those expressed in terms of the metamodel and according to the syntax of the query language) are dispatched to a *query processor*. The query processor performs matching of queries against content models in order to find references to video objects in the video database that satisfy each query. Since queries can be expressed at various levels of abstraction, it is possible for a sequence of video components , rather than a primitive video data object, to satisfy a query. An answer to a query may then be a list of

references to primitive video objects, or a list including sequences of such objects.

7.5.2.2 Virtual Video Prescription Tools. Virtual video composition is carried out by a virtual video engine. The virtual video engine is used when the video data is to be delivered to users in the form of highly structured and coherent video productions. In this case, a *virtual video prescription authoring tool* is used to specify the high level structure of the virtual video in the form of a *virtual video prescription.* A virtual video prescription includes embedded queries that are executed by a *virtual video prescription interpreter* in order to synthesise a complete *virtual video* having content tuned to specific user requirements at the time of interpretation. Virtual videos are specified and presented using the hypermedia modelling approach.

7.5.2.3 Automated Video/Audio analysis tools. The *automated video analysis tools* include algorithms for characterising video data in terms of visual features (eg. using colour histograms and texture measures), in terms of automatically detectable camera operations (eg. pan, tilt, and zoom), and basic object and shape detection. Automated cut detection is incorporated for parsing video data streams that extend beyond a single cut (ie. sequences or complete productions). The audio analysis tools include algorithms for detecting distinct loud sounds and characterisation of audio events such as music, voice, etc.

7.5.2.4 Presentation Tools. The presentation tools include the user interaction component, which includes query, navigation and browsing. Experience with the FRANK navigation and browsing tool (Simpson-Young and Yap, 1996) suggests that a web-based browsing tool provides an appropriate interface for searching, navigating and browsing through video material.

7.6 CONCLUSION AND ONGOING WORK

In this paper, we have presented a multi-model framework for designing VIS applications. The framework includes a functional component to represent video and audio analysis functions and a hypermedia component for video delivery and presentation. Modelling video data involves understanding the semantics of visual information. Towards this end, we have described the various levels at which video data can be interpreted. We have then presented a metamodel for modelling video content. Finally we have described a case study - the FRAMES project - which utilises the multimodel framework to develop specific VIS applications.

We plan to extend our research in two areas; query languages for continuous visual media, and virtual video generation. We will continue to develop tools and infrastructure to allow specialists to articulate interpretive models at different levels and then use these models in support of video search, retrieval, browsing and synthesis. On the application side, work will

involve developing systems for specific domains such as the sports domain, which will include multi-modal queries to query sports highlights.

Acknowledgments

The authors wish to acknowledge that this work was carried out within the Cooperative Research Centre for Research Data Networks established under the Australian Government's Cooperative Research Centre (CRC) Program and acknowledge the support of the Advanced Computational Systems CRC under which the work described in this paper is administered.

References

ADJEROH AND LEE (1996): Synchronisation and User Interaction in Distributed Multimedia Presentation Systems, *Multimedia Database Systems, Kluwer Academic publishers, 1996* .

AIGRAIN, P. ZHANG, H. and PETKOVIC, D. (1996): Content-Based Representation and Retrieval of Visual Media, *Multimedia Tools and Applications,* 3, 179-202.

ALLEN, J.F. (1983): Maintaining knowledge about temporal intervals, *Communications of the ACM, vol.26 No.11*, 882-843.

ARMAN, DEPOMMIER, HSU, CHIU (1994)*: Content-based Browsing of Video Sequences, Proceeedings of ACM international Conference on Multimedia '94, California, 1994.*

ARMAN, F., HSU, A., and CHIU, M.Y. (1993): Image processing on compressed data for large video databases, *Proc. ACM Multimedia 93, Anaheim, California, August 1993*, pp 267-272.

ARNHEIM, R., (1971): *Film as Art*, University of California Press.

BORDWELL, D. and THOMPSON,K. (1997): *Film Art: An Introduction, 5th edn.*, McGraw-Hill.

DAVIS, M., Knowledge Representation for Video", *Proceedings of the 12th National Conference on Artificila Intelligence*, AAAI, MIT Press, pp. 120-127, 1994.

FENG, J., LO, K-T. and MEHRPOUR, H. (1996): Scene change detection algorithm for MPEG video sequence, *IEEE International Conference on Image Processing (ICIP96)*, September 1996.

FLICKNER, SAWHNEY, NIBLACK, ASHLEY, HUANG, DOM, GORKHANI, HAFNER, LEE, PETKOVIC, STEELE AND YANKER (1995): Query by image and video content: The QBIC system, *IEEE Computer*, 28(9), pp 23-32.

GORKY (1994): Multimedia Information Systems, *IEEE Multimedia* , Spring 1994.

GRIFFIEON, J., YAVATKAR, R. and ADAMS, R. (1996): Automatic and Dynamic Identification of Metadata in Multimedia.

GU, L., TSUI K. AND KEIGHTLEY D. (1997): Dissolve detection in MPEG compressed video, *Proceedings of IEEE International Conference on Intelligent Processing Systems, October 1997.*

GU, L., TSUI, K. and KEIGHTLEY D. (1996): Camera shot boundary detection in MPEG compressed video, *Technical Report, CSIRO Mathematical and Information Sciences, December 1996.*

GUPTA, A. (1997): Visual Information Retrieval: A Virage Perspective, *Visual Information Retrieval White Paper, http://www.virage.com/wpaper.*

HALPIN (1995): Conceptual schema and Relational database Design, 2nd edition, Prentice Hall, Sydney, Australia.

HARDMAN, L., BULTERMAN.D.C.A, VAN ROSSUM, G., The Amsterdam Hypermedia Model, Communications of the ACM, Vol.37, No.2, February, 1994.

HJELSVOLD,MIDSTRAUM AND SANDSTA (1996*): Searching and Browsing a Shared Video Database, Multimedia Database Systems*, Kluwer Academic publishers, 1996 .

ISAKOWITZ, T, STOHR, E, and BALASUBRAMANIAN , P., (1995): *RMM: A Methodology for Hypermedia Design*, Communications of the ACM vol. 38, No. 8, August 1995.

KIM M., CHOI J. G. AND LEE M. H. (1998): Localising Moving Objects in Image Sequences Using a Statistical Hypothesis Test, *Proceedings of the International Conference on Computational Intelligence and Multimedia Applications, Churchill, Victoria, 9-11 Feb., 836-841.*

LINDLEY C. A. & VERCOUSTRE A. M. (1998): *Intelligent Video Synthesis Using Virtual Video Prescriptions*, International Conference on Computational Intelligence and Multimedia Applications, Churchill, Victoria, 9-11 Feb.

LINDLEY C. A. 1997 A Multiple-Interpretation Framework for Modeling Video Semantics, *ER-97 Workshop on Conceptual Modeling in Multimedia Information Systems.*

LINDLEY C. A. AND SRINIVASAN U. 1998 "Query Semantics for Content-Based Retrieval of Video Data: An Empirical Investigation", Storage and Retrieval Issues in Image- and Multimedia Databases, August 24-28, in conjunction with 9th International Conference DEXA98 Vienna, Austria.

METZ C. 1974 *Film Language: A Semiotics of the Cinema*, trans. by M. Taylor, The University of Chicago Press.

NWOSU, K. THURAISINGHAM, B. and BRUCE BERRA, P. (1996): *Multimedia Database Systems,* Kluwer Academic Publishers, 1996.

PFEIFFER, S. FISCHER, S. and EFFELSBERG ,W. (1996): Automatic Audio Content Analysis, *Proceedings of ACM Multimedia, Boston, 1996.*

ROBERT-RIBES, J. and MUKHTAR, R.G., (1997): Automatic Generation of Hyperlinks between Audio and Transcript, *Fifth European Conferenceon Speech Communication and Technology,* September 1997.

SIMPSON-YOUNG, W. and YAP, K. (1996): FRANK: Trialing a system for remote navigation of film archives, *SPIE International Symposium on Voice and Video Communications, Boston, November 1996.*

SRINIVASAN, GU, TSUI AND BILL SIMPSON-YOUNG (1997): A Data Model to support Content-based Search on Digital Video Libraries, *Australian Computer Journal, Vol.29, No 4, November 1997.*

SUBRAHMANYAM V.S., *Principles of Multimedia Database Systems,* Morgan and Kaufmann, 1998.

YAP, K., SIMPSON-YOUNG, W, and SRINIVASAN, U. (1996): Enhancing Video Navigation with existing alternate Representations, *First International Conference on Image Databases and Multimedia Search, Amsterdam, August 1996.*

ZHANG, H., LOW, C.Y. and SMOLIAR S.W. (1995): Video parsing and browsing using compressed data, *Multimedia Tools and Applications*, 1(1), pp 89-111.