A Survey of Semantic Content-Based Multimedia Models

Harry W. Agius

Centre for Multimedia, School of Computing, Information Systems and Mathematics, South Bank University, UK

The increasing use of multimedia within information systems has led to the development of models and techniques that seek to capture information regarding the semantic content of video and audio. This paper surveys this emerging multimedia research area and discusses how successful it has been.

Keywords: Semantic Content-Based Multimedia Models, Video, Audio, Syntax, Semantics, Multimedia Information Systems

1. Introduction

Multimedia information systems (MMISs) handle retrieval and processing mechanisms for static media, such as text and graphics, as well as for dynamic, time-variant media, such as video and audio (Burrill et al., 1994; Angelides and Dustdar, 1997). This is achieved by representing all information uniformly, as a bit stream (Agius and Angelides, 1997a). This is an issue of *syntax*, because the emphasis is placed on the organisation and representation of information in the MMIS, whether this be the bit stream (e.g. text represented through ASCII codes, video represented through formats such as MPEG, and audio represented through Wave and other formats) or objects presented on-screen.

However, it has come increasingly to be realised that these issues do not address how to use video or audio effectively within an MMIS. Without knowledge of its content a bit stream remains a bit stream that cannot be interpreted. To use and interact with it, the bit stream must be converted into a form that can be understood. This is a *semantic* issue because the emphasis is on the meaning depicted within videos and audios.

This paper surveys the emerging area of semantic content-based multimedia modelling, where the emphasis is placed on what is taking place within the media, i.e. the meaning of the content, as opposed to the format of how this content is stored, which is an issue of syntax. Section 2 elaborates on syntax and semantics for video and audio. Section 3 discusses semantic content-based multimedia modelling techniques within four groups: (1) those modelling 'physical' (i.e. syntactic) content information as colour, texture, and camera motion; (2) those concerned with representing the spatial and temporal location of content objects; (3) stratification-based techniques; and (4) formal techniques. Section 4 closes the paper by discussing the success of existing techniques.

2. Multimedia Syntax and Semantics

The distinction between multimedia syntax and semantics separates pixel and semantic representations in still and full-motion video, and signal and semantic representations in audio. Pixel representations are concerned with the storage of arrays of values, in which each value represents the data associated with a pixel in the image. For a bitmap this value is a binary digit; for a colour image, the value may be a collection of numbers or an index indicating the intensities of various key colours, e.g. red, green and blue (Steinmetz and Nahrstedt, 1995). Pixel representations for video applications increasingly take advantage of motion-compensated transform coding methods, as in MPEG (Le Gall, 1991; Meyer-Boudnik and Effelsberg, 1995) and H.261 (Liou, 1991). These image representations are described in terms of video frames divided into arbitrary square blocks and, as such, are mathematically intensive.

Intelligent image understanding techniques have sought to move toward more semantic representations of images by attempting to recognise objects within images. However, they have only partly attempted to bridge the gap. Image understanding is necessarily process-oriented, focusing on three broad stages (Chang and Hsu, 1992): (1) image analysis and pattern recognition; (2) image structuring and understanding; and (3) spatial reasoning and image information retrieval.

At present, however, image understanding researchers do not completely agree on a common representation for important tasks, e.g. the appropriate decomposition of an object into parts that enable efficient recognition is still a subject of basic research (Mundy, 1995). Also, the techniques are not rich enough to capture the information necessary for comprehensive processing and are therefore inadequate for domain- and task-independent image understanding (Gudivada and Raghavan, 1995). Moreover, there is a predominance within image understanding of merely identifying objects and not necessarily any further information about the objects. For example, we may be concerned with what a particular motor vehicle is doing within an image: Is it parked? Is it racing? Has it crashed? Which way is it facing?

Full-motion video offers further complications. Understanding that relies on the contents of the video frames is a very difficult problem. Current successful efforts at visual querying of image databases fail to capture and exploit the massive information contained in video. Video is temporal, spatial, and often unstructured; the combined video and audio signals convey an abundance of information (Kanade, 1996). While Swanberg et al. (1992) argue that, in many cases, video information is structured in the sense that there exists both a strong spatial order within individual frames and a strong temporal order among different frames pertaining to the same scene, a broader perspective would reveal this to be only true to a limited extent, e.g. in scenes from a news programme. Furthermore, it is the temporal nature of video that brings to the fore issues concerning what particular objects are doing within the video. For example, suppose we want to determine all those frames in which a specified object performs a particular act, such as video frames in which a white horse is galloping. Whereas recognising the white horse is relatively easy, selecting frames in which the horse is galloping (and not jumping or cantering) is extremely difficult.

Audio is often put to a variety of uses, including speech, music and sound effects. Audio signal representations are always concerned with the storage of digital samples. These are discrete numbers representing the amplitude of the analogue sound waveform at regular time intervals. The greater the number of bits used to approximate the height of the waveform, the closer the resultant waveform - reconstructed from the stream of discrete numbers - will be to the original analogue waveform. For example, if eight bits are used in sampling the amplitude, then the amplitude may take on one of 256 possible values at each interval. With fewer bits, however, less possible values are available, and so the shape of the digitally reconstructed waveform will become less discernible, resulting in lower quality sound (Steinmetz and Nahrstedt, 1995).

The use of digitised music has led to more symbolic forms of music representation, the most popular being the MIDI (Music Instrument Digital Interface) data format included in the standard. More bespoke approaches to music include encoding the sheet music into a digital representation (Rader, 1996). However, Kanade (1996) explains that audio is also intrinsically linked to video. The audio signal includes language information in the form of narration and dialogue that, when transcribed, provide direct indices to the video content. Natural language analysis of the transcript, together with production notes and other text information about the video, can determine the narrative's subject area and theme. This understanding can be used to generate summaries of each video segment for icon labelling, browsing, and indexing. The audio signal conveys other information, including pauses, silence, music, and laughter. These bits of information can supplement the other structured descriptors, e.g. pauses might be useful in identifying natural start and stop positions for video segmentation.

Intelligent speech analysis and speech generation techniques have both sought to move toward more semantic representations of speech. The former by attempting to recognise who is speaking, what is being said (i.e. what words), or how something is being said within digital audio (e.g. angrily), thus moving from signals to semantics; the latter by attempting to transform text into speech, thus moving from semantics to signals. However, they have only partially bridged the gap, for similar reasons as intelligent image understanding techniques.

Speech analysis, like image analysis, is necessarily process-oriented, focusing on three broad stages (Steinmetz and Nahrstedt, 1995): (1) acoustic and phonetic analysis; (2) syntactical analysis (speech recognition); and (3) semantic analysis (speech understanding). In contrast, speech generation uses one or more of the following techniques (Steinmetz and Nahrstedt, 1995): pre-recorded speech samples; time-dependent speech concatenation; or frequency-dependent sound concatenation. With the latter two the process focuses on translating text into a sound script which is then translated into a speech signal. The existing body of algorithms and data structures for speech analysis and generation are not extensive and, again, tend to be domain-specific and task-dependent. They also are predominantly speaker-dependent.

The following section reviews semantic content-based multimedia modelling techniques which have sought to move towards more fuller representations of video and audio content.

3. Existing Research on Semantic Content-Based Multimedia Modelling

Efforts to represent semantic multimedia content have centred around the development of models that may be seen to fit within one of four groups, according to the technique they employ: (1) those modelling 'physical' (i.e. syntactic) content information, such as colour, texture, and camera motion; (2) those concerned with representing the spatial and temporal location of content objects; (3) stratification-based techniques; and (4) formal techniques. This section surveys semantic content-based multimedia models according to these four groups.

3.1. Physical Models

These models are primarily concerned with 'physical' content information, which is typically syntactic in nature, e.g. colour, texture, and camera motion.

At the NTT Human Interface Laboratories, Japan, Tonomura et al. (1994) developed methods for video parsing where each shot (a logical video segment) is then further analysed to obtain features of the video content, called *video indexes*. The indexes are organised into two kinds of structures: the *link structure* describes the link relations between shots, and the *content structure* stores information about the scene and objects as obtained by shot analysis. Camera work information suggests the scene's spatial situation, while representative colour information provides some information about the objects. Techniques are discussed to automatically extract this information.

The data model used in IBM's Query by Image Content (QBIC) system (Barber et al., 1995; Flickner et al., 1995) stores still images or video scenes that contain objects (subsets of an image), and video shots that consist of sets of contiguous frames and contain motion objects. This data model is used for both database population (where images and videos are processed to extract and store features describing their content) and querying (where the user composes a query graphically). The content used in both cases includes the colour, texture, shape, sketch, and location of image objects and regions. For video, content includes object and camera motion.

3.2. Techniques for Locating Content Objects

Models within this category are focused on identifying the spatial and temporal location of content objects, often for enabling user interaction with the video and audio.

Visual Repair (Goodman, 1993) is a prototype explanation generation component for an intelligent multimedia training system in the domain of Apple Macintosh IIcx repair. Video is used in the student's repair plans, to illustrate to the student what he has advised should be done to fix the fault, and when giving help at the student's

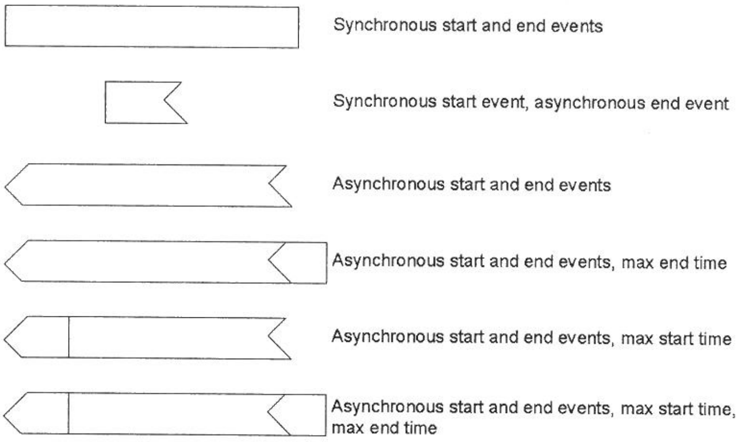
Synchronous start and end events
Synchronous start event, asynchronous end event
Asynchronous start and end events
Asynchronous start and end events, max end time
Asynchronous start and end events, max start time
Asynchronous start and end events, max start time, max end time

Fig. 1. The six types of units in the timeline-tree model. Source: Hirzalla et al. (1995).

request. Relevant parts of the video are graphically highlighted as it is played. The beginning frame of each video has information associated with it about the content of that video frame (the name of important elements and the size and location of each element). That information can be used to automatically generate graphics that are superimposed on the video in order to point out the recipient objects of important actions during the execution of the presentation plan.

Burrill et al. (1994) propose the use of Sensitive Regions (or 'hot-spots'), which use pre-editing to define regions of interest within video frames. The regions are identified through the use of polyhedral 3D volumes, on the representational axes 'width', 'height', and 'time'. In specific implementations, the authors suggest that the model can be extended to attach application-dependent semantics to the objects delineated within these regions, but they do not discuss this any further. In its simplest form, the approach can be used as a trigger mechanism which enables the user to click within the hot-spot, e.g. actors, stage 'props' and scenery, to identify the object or invoke some hyperlink to another part of the underlying hyperbase. The authors explain that the concept of Sensitive Regions could also be used for non-visual objects such as background music and film mood.

Hirzalla et al. (1995) use an enhanced timeline (a timeline is a graph representing the flow of media over time) with six basic units as a theoretical model for interactive multimedia (Figure 1). The symbol, Choice$_i$ (C_i) is also used where each user choice results in a different timeline, thus i refers to timeline$_i$, where $i \geq 0$. There are many different timelines, and so timeline$_i$ is a timeline that branches from timeline$_j$, where $j < i$. C_i distinguishes between temporal equalities and inequalities with other asynchronous events. Events that share temporal equalities (that is, that do not admit terms such as 'at least' or 'at most') carry the same symbol; otherwise the symbols differ. A data structure, containing three fields, that determines the user action that initiates the object is associated with C_i: **user_action** describes what input should be expected from the user, such as 'keypress-y' or 'left-mouse'; **region** establishes which region of the screen (if applicable) is a part of the action, e.g. 'rectangle(100, 100, 150, 180)'; and **destination_scenario_pointer** names a pointer to some other part of the scenario, or even a different scenario.

Figure 2a shows an example car demonstration scenario, where a user is presented with a graphic of a car. Embedded onto the presentation screen (i.e. modelled in the scenario as a combination of a user action (mouse click) and a region on the screen) are three hot-spots: the hood, the door, and the background. Each choice triggers either text explaining the features of the car's engine, a video-audio clip showing and explaining the interior of the car, or the disappearance of the car, respectively. If the user does not respond within a certain time frame, the car image disappears. If the user chooses the hood and gets the text object, he might then choose to listen to the engine. The

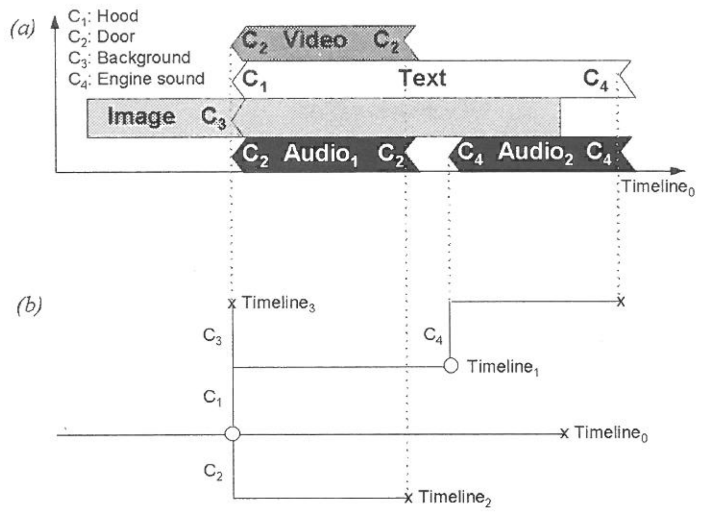

Fig. 2. The timeline-tree model represents interactive scenarios like this car demonstration using: (a) an expanded timeline; and (b) a tree-like structure that traces all possible timelines. Source: Hirzalla et al. (1995).

presentation ends after the audio plays. Since the end events of 'Text' and 'Audio$_2$' objects have temporal equality, both are labelled with C_4. Figure 2b shows the tree corresponding to the interactive scenario in Figure 2a. The small circles represent branches where user actions may change the course of the scenario (i.e. the times that asynchronous events corresponding to the symbols at the circle become activated - they are deactivated only when the presentation flow branches to another timeline). If the user makes no choices, the current timeline simply plays itself out (timeline$_0$); otherwise, users traverse the timeline tree, viewing custom presentations (timeline$_1$ through timeline$_4$) determined by their choices. At most one choice, C_i, can be selected at a time. Consequently, the presentation flow will branch to timeline$_i$. Each 'x' represents possible scenario ending points.

The IntelligentPad architecture (Tanaka, 1996) is based on pads, each of which consists of a display object, which defines both its view on the display screen and its reaction to user events, and a model object, which defines its internal state and behaviour. Pads may be used to represent container objects (container media that carry content information), media objects (container objects with their content objects), and reference frames (which indirectly specify the corresponding sub-portion of content, with time segments working as temporal reference frames and rectangular areas working as spatial reference frames). For the access of non-articulated (that is, non-machine recognisable) content objects, i.e. those in images, movies and sounds, in a media object, the media object can be provided with a special slot named 'reference_frame' that receives the location and size of a reference frame and returns the corresponding portion of its content information. Spatial reference frames can be represented as transparent pads that minimally cover the target content objects.

3.3. Stratification-Based Techniques

These models assign strata to contiguous segments of video and audio which provide descriptions of the content of the segment. The detail and makeup of such descriptions vary considerably between the models.

Parkes (1988; 1990) proposes a model for handling descriptive data for video information that is used in the CLORIS intelligent multimedia tutoring system. The model has two basic concepts: *events* and *settings*. An event is a hierarchical description of a video scene based on PART-OF relationships. For instance, suppose a video scene A shows how to use a micrometer. The event 'Using the micro meter' is assigned to A. This is the root of the description. It consists of four sub-events: 'Remove micro from case', 'Clean micro', 'Measure metal', and 'Record measure'. Each event corresponds to some portion of video A. The event 'Clean micro' itself consists of four further sub-events,

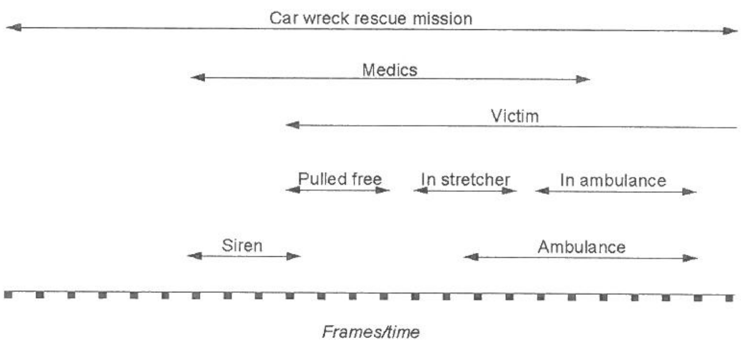

Fig. 3. Example of strata in the Stratification System.

'Hold micro', 'Lift cloth', 'Wipe rod', and 'Replace cloth'. These may each consist of further sub-events. A setting corresponds to different representations of the same object in the real world. For instance, the binary relations 'zoom in' and 'zoom out' are defined between these settings.

EVA (Mackay, 1989; Mackay and Davenport, 1989) is a video annotator system, written in Athena Muse and developed at MIT. It provides software researchers with the facilities to create labels and annotation symbols prior to a session and then permits live annotation of video during an experiment. Although EVA is a useful tool for analysing (particularly live) video data, the capability to share descriptive information among annotated video scenes is relatively weak. It is not fully addressed what operations are needed to compose/decompose the annotated video scenes.

The Stratification System (Aguierre Smith and Davenport, 1992) is a video annotation system that uses the concept of stratification to assign descriptions to video footage, where each stratum refers to a sequence of video frames. The strata may overlap or totally encompass each other. Figure 3 shows an example of video footage annotated by strata. Strata are stored in files accessible by a simple keyword search. A user can find a sequence of interest, but cannot easily determine the context in which it appears because of the absence of relationships between the strata.

Oomoto and Tanaka (1993) propose the video object data model as a new modelling construct for video database management. They consider that any portion of a video frame sequence is an independent entity, and so make it possible to define a *video object*, which corresponds to a certain set of video frame sequences. It has its own attribute-value pairs to represent the content (meanings) of the corresponding video scene. Figure 4 shows an example video object database. The main features of the video object data model are as follows. There is no assumption of a specific database scheme such as classes and a class hierarchy, so users can define any attribute's structure for each video object. Interval inclusion inheritance is used, whereby some descriptive data of video objects can be inherited by other video objects. For example, in Figure 4, object 3 (O_3) has attribute 'Location' and its value 'America'; thus O_4 to O_7 also have this attribute-value by the interval inclusion relationship. Finally, video objects are composed based on an IS-A hierarchy. The authors define several operations, *interval projection, merge* and *overlap*, for video objects that compose new video objects. These operations also derive, based on the IS-A hierarchy, the attribute-values of the synthesised video object from the original video objects.

Media Streams (Davis, 1993) is an iconic visual language that enables users to create multilayered, iconic annotations of video content. Icons denoting objects and actions are organised into cascading hierarchies of increasing levels of specificity. Additionally, icons are organised across multiple axes of descriptions such as objects, characters, relative positions, time, or transitions. The icons are used to annotate video streams represented in a timeline. Currently, around 2,200 iconic primitives can be

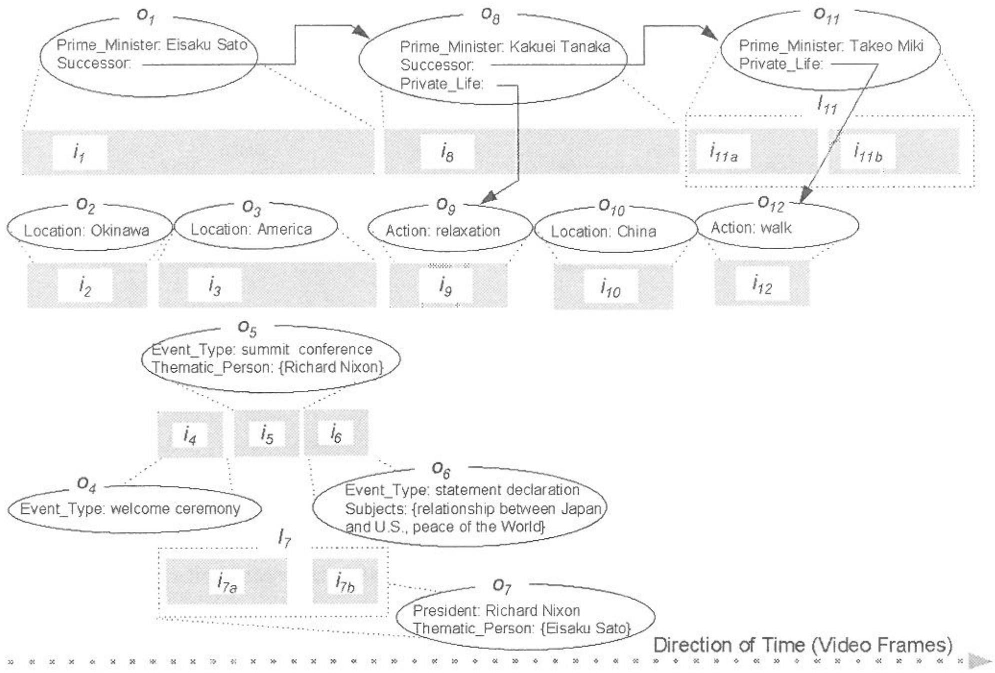
o_1
Prime_Minister: Eisaku Sato
Successor:
o_8
Prime_Minister: Kakuei Tanaka
Successor:
Private_Life:
o_{11}
Prime_Minister: Takeo Miki
Private_Life:
I_{11}
i_1
i_8
i_{11a}
i_{11b}
o_2
Location: Okinawa
o_3
Location: America
o_9
Action: relaxation
o_{10}
Location: China
o_{12}
Action: walk
i_2
i_3
i_9
i_{10}
i_{12}
o_5
Event_Type: summit conference
Thematic_Person: {Richard Nixon}
i_4
i_5
i_6
o_4
Event_Type: welcome ceremony
o_6
Event_Type: statement declaration
Subjects: {relationship between Japan and U.S., peace of the World}
I_7
i_{7a}
i_{7b}
o_7
President: Richard Nixon
Thematic_Person: {Eisaku Sato}
Direction of Time (Video Frames)

Fig. 4. Example video object database using the video object data model. Source: Adapted from Oomoto and Tanaka (1993).

browsed. However, this user-friendly visual approach to annotation is limited by a fixed vocabulary. Also, it does not exploit textual data such as closed-captioned text.

Little et al. (1993; 1995) propose a system that supports content-based retrieval of video footage. They define a specific data scheme composed of Movie, Scene, and Actor relations with a fixed set of attributes. The system requires manual feature extraction, then fits these features into the data scheme. It permits queries on the attributes of movie, scene, and actor. Having selected a movie or a scene, a user can scan from scene to scene. To achieve this, the model uses an object composition Petri net (OCPN) to represent the interconnections of the various scenes, based on the earlier work of Little and Ghafoor (Little and Ghafoor, 1990; Little and Ghafoor, 1991; Little and Ghafoor, 1993; Little, 1994). An OCPN uses the structure of a Petri net to maintain synchronisation between the various elements (in this case, scenes) in a multimedia presentation (in this case, a movie). Unfortunately, the data model and the virtual video browser are limited because descriptions cannot be assigned to overlapping or nested video sequences as in the Stratification System. Moreover, the system is focused on retrieving previously stored information and is not suitable for users who need to create, edit, and annotate a customised view of the video footage.

The algebraic video data model (Weiss et al., 1995) consists of hierarchical compositions of video expressions with high-level semantic descriptions, constructed using video algebra operations. Video algebra is used as a means of combining and expressing temporal relations, defining the output characteristics of video expressions, and associating descriptive information with these expressions. Interaction with algebraic video is accomplished through four activities: Edit and Compose, Play and Browse, Navigate, and Query. The operations that support playback, navigation, and content-based queries are grouped together as interface operations. The fundamental entity of the model is a *presentation*, a multi-window spatial, temporal, and content combination of video segments. Presentations are described by video expressions. The most primitive video expression creates a single-window presentation from a raw video segment. These segments are specified using the name of the raw video and a range within it (Figure 5).

Adah et al. (1996) present a content-based model for video data that has been implemented within a prototype system, AVIS. The model represents three main types of entities within the video.

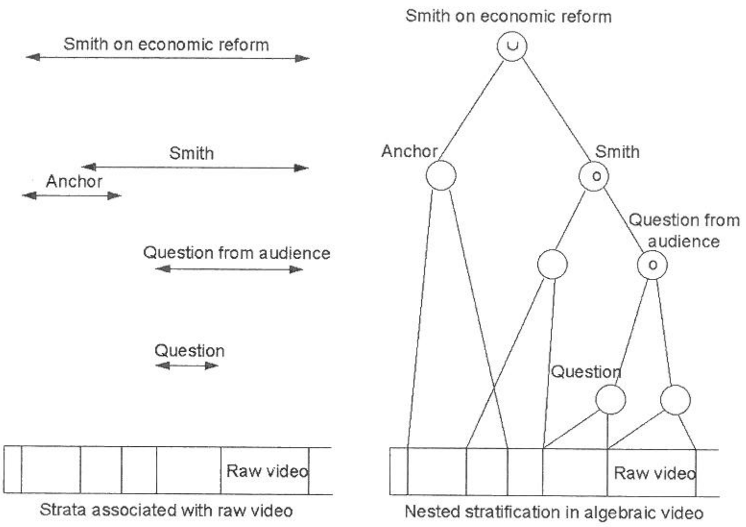

Fig. 5. Nested stratification in the video algebra data model. Source: Weiss et al. (1995).

Video objects are present in video frames and include characters and objects that are present in a movie. 'Invisible' objects may also be modelled. Therefore, some object X may be present inside a cupboard (which is visible) even though X cannot be physically seen. **Activity types** describe the (generic) subject of a given video frame sequence, such as 'murder' or 'giving a party'. Multiple activities may occur simultaneously. **Events** are instantiations of an activity type which make the activity more specific. Activity types are therefore general groups containing many events. Two further sub-entities that are used to construct events help distinguish events from activity types: (1) **Roles** are descriptions of certain aspects of an activity, and they may involve objects (e.g. 'victim' and 'murderer' are roles in the activity 'murder') and descriptions (e.g. 'murder motive' and 'murder weapon'); (2) **Teams** are sets of roles (objects/descriptions) that jointly describe an event; that is, they are instantiations of the roles in an activity type, e.g. for the event 'murder', the team involved might consist of Tom in the role 'victim', and Dick and Harry both in the role 'murderer', a gun may play the role of the 'murder weapon', while 'mugging' is the role 'mur der motive'. These entities are represented using association maps and a specially adapted form of segment trees, which the authors refer to as frame segment trees.

Informedia (Christel et al., 1995; Kanade, 1996; Wactlar et al., 1996) is a digital video library system that uses integrated image, speech, and language understanding for the creation and exploration of the library. Informedia's off-line creation facilities work as follows. Using speech recognition techniques, Informedia converts each videotape's sound track to a textual transcript. A language understanding system analyses and organises the transcript, then stores it in a full-text information retrieval system. Image understanding techniques segment video sequences, detect and identify objects (human faces and text), obtain a visual characterisation of the scene, identify the representative images for the skim video (comprising the significant words and images of the original video), and match images by incorporating language and speech information. Thus, for a particular video clip, Informedia stores information about the following: when scenes change, the different forms of camera motion within the clip (e.g. pan, static, zoom), the location of identified faces, the location of identified text, the word relevance, and the audio level. These are used later for interactive retrieval by a user of the indexed video library.

Jabber (Kazman et al., 1996) uses content-based indexing of an audio stream to access the parallel streams produced by video conferences. It performs speech recognition on the audio stream, then groups the recognised words into semantically-linked trees. Jabber uses four forms of indexing (which may be combined): **indexing by intended content**, where meetings (i.e. the data streams) are indexed by users, in real time, according to an explicit agenda that accompanies the meeting; **indexing by actual**

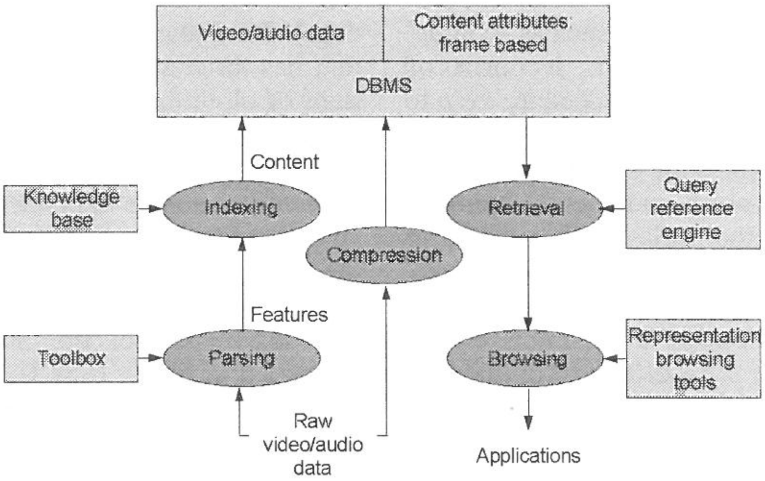

Fig. 6. The video management architecture of the Video Classification project. Source: Smoliar and Zhang (1994).

content where meetings are indexed by what was said or done via a speech recognition system which creates text-based records from the stored audio track from which clusters of related words (which in turn relate to topics) are identified and used as indexes back into the original streams; **indexing by temporal structure**, where meetings are indexed by their structure in terms of human interactions over time; **indexing by application record**, where a log of a computer application's activi ty can be kept and used as an index back into the audio/video streams, e.g. object creations, deletions, modifications, changing focus, grouping, and undoing.

3.4. Formal Techniques

Models within this category use formal techniques, usually based on mathematics, in order to specify the content information.

The Video Classification project at the Institute of Systems Science, National University of Singapore (Smoliar and Zhang, 1994), has developed an architecture that characterises the tasks of managing video content (Figure 6). The parsing and indexing aspects of the architecture deal with semantic content. In parsing, the video source material is segmented into individual camera shots, which then serve as basic units for indexing; then, different camera techniques are identified, e.g. panning and tilting, zooming; finally, content models are applied to the identification of context-dependent semantic primitives. In indexing, the video clips are tagged according to the semantic primitives of the images and then inserted into the database. The various subject matter categories of the material being indexed are represented in a hierarchy as a tree, where each node is a knowledge representation frame, permitting specialisation and generalisation among the categories. The Video Classification project is also working on audio and preliminary algorithms have begun to be developed that detect content changes in an audio signal. Plans are to develop models of audio events, similar to the models used in image-based content parsing, e.g. in a sports video, very loud shouting followed by a long whistle might indicate that someone has scored a goal, in which case the system should recognise an 'event'.

At Hiroshima University, Japan, Yoshitaka et al. (1994) developed an object-oriented technique for the composition of domain knowledge in a multimedia database system. The domain knowledge describes how the system views the target multimedia data for content-based retrieval. Domain knowledge, *Dk*, is a way for a class to present knowledge representing a certain concept held by objects in the class. It is defined as a triple:

$$Dk(C) =< Fi[fi, extp], Op[op[fi, mf]], \\ Cm[cd, fi, v] >$$

C denotes a concept representing a pseudo object which derives from the objects in a class by providing the domain knowledge. A pair of brackets represents a set. Fi represents the features constituting a concept C, such as 'colour' and 'length' for the concept 'hair'. It consists of a feature item name fi and a procedure $extp$ to extract the information from objects in the associated class. Op defines the semantics of operators appearing in a query and how the operator is evaluated during the retrieval. The semantic behaviour of an operator may change depending on the class of objects to be evaluated. For example, the behaviour of an operator '$=$' for objects in an integer class differs from that in a colour class. A member of Op consists of an operator op and a set of descriptions of semantic behaviours corresponding to the operator, given by the combination of a specific item fi and a function mf for evaluating the fitness between the extracted value of feature item fi and a data value v. The higher the value, the more the object satisfies the query condition. Cm converts a condition value cd specified in a query into a certain data value (or a certain range of values) v whose data type is the same as that of the data values of fi. Therefore, both fi and v are the same type and are processed through mf. v can be a certain function $f(cd)$ that returns a certain data value (or a certain range of values).

Brink et al. (1995) propose the media abstraction, expressed as a 7-tuple:

$$M = (ST, fe, \lambda, R, F, Var_1, Var_2)$$

Var_1 is a set of objects called state variables, ranging over states. Var_2 is a set of objects called feature variables, ranging over features. ST is a set of objects called states. All files containing a photograph will be separate states in the media abstraction. fe is a set of object features. These may include persons of interest (e.g. Tony Blair, Gordon Brown) and inanimate features (e.g. Houses of Parliament, 10 Downing Street). λ is a map from ST to functions from fe to $[0, 1]$. It specifies the confidence of a particular feature occurring in a given image. For instance, $(\lambda(s_2))$ (Tony Blair) $= 0.7$ indicates that the certainty of Tony Blair occurring in state s_2, which may be a picture, is 70 percent. R is a set of fuzzy interstate relations (of possibly different arities) on the set ST; and F is a set of fuzzy feature-state relations. Each relation in F is a map from either $fe^i \times ST$ to $[0, 1]$ (when relationships between features are independent of state) or $fe^i \times ST$ to $[0, 1]$, where $i \leq 1$ (when relationships between features are state dependent). For instance, a relation called *is_wearing* that has three arguments (a person's name, an item of clothing, and a colour) would change from state to state – the same person may be dressed differently in two different pictures. It is therefore a state-dependent relation. Hence, an extra, fourth argument, must be added to it: the state name. A sample tuple for this relation, ('Tony Blair', 'tie', 'red', file5) : 0.99, says there is a 99 percent certainty that in the picture contained in file 5, Tony Blair is wearing a red tie. For state-independent relations, there is no need to add an extra state-name argument. The media abstraction may also be used to model domains involving audio input in a similar manner. The authors do not, however, explain how video content may be modelled.

4. Concluding Discussion

This paper began by distinguishing between multimedia syntax, where emphasis is placed on the organisation and representation of the bit stream or on-screen objects, and multimedia semantics, where the emphasis is on the meaning of the depicted content. The paper then reviewed existing semantic content-based multimedia modelling techniques within four groups: (1) 'physical' models; (2) techniques for locating content objects; (3) stratification-based techniques; and (4) formal techniques. The complex nature of video and audio has made focusing on content-based multimedia semantics a much more difficult problem than has been the case with text (Aigrain et al., 1996; Agius and Angelides, 1997b; Lee et al., 1997). Consequently, existing models have tended to specialise on the representation of one or two specific content-based semantics, such as the representation of objects, which has limited their *general* application within MMISs. However, as the review has highlighted, existing models have also suffered from a number of more specific shortcomings.

The explicit way in which sequences of video and audio are split and grouped together within existing models has been basic and predominantly video-oriented. Audio is frequently underspecified compared to video, or is disregarded altogether. Video and audio are either treated in unison as one inseparable unit or audio is left unspecified. For example, QBIC, Visual Repair, CLORIS, the Stratification System, the video object data model, and the model underlying the Video Classification project all fail to cater for audio. In contrast, Jabber is completely audio-oriented, with no facilities for the handling of video. Even in models where facilities for both video and audio are provided, the functionality for audio has been inferior to that provided for video.

Although most semantic content-based models represent on-screen objects, very few of the models are concerned with the *location* of these objects, e.g. through the use of on-screen coordinates. Frequently, content-based multimedia models have been satisfied with merely representing the *presence* of a content object in a particular frame or set of contiguous frames. Exceptions have included those approaches utilising 'hot-spots', such as Sensitive Regions, the timeline-tree model, and the IntelligentPad architecture. Moreover, very few of the models have addressed the issue of determining the *relative* location of objects.

The modelling of incidents occurring within the media stream has also been poorly addressed, but has received some attention in the stratification-based approaches and the formal techniques. However, the semantic information has been of a very unstructured form. For example, the algebraic video data model relies on attached strings of text, as does the Stratification System. Other models which take a more structured approach, such as the video object data model, still essentially put text strings into arbitrary attribute-value pairs. Semi-structured information makes processing on this information, e.g. in terms of identifying and comparing terms, more difficult than if the information were fully structured. The temporal relationships between the incidents (e.g. W occurs during X, Y occurs before Z) has also been inadequately addressed with many models providing no capability for this, e.g. the models underlying QBIC and Visual Repair, the Sensitive Regions model, the timeline-tree model, IntelligentPad, EVA, the Stratification System, Jabber, the model underlying the Video Classification project, and the media abstraction. One exception is the model used in the Virtual Video Browser which is based on OCPNs.

Finally, only a handful of models provide the ability for the user to directly interact with the video and audio.

Problems and imperfections are to be expected in any field which is young and newly emerging. We have already seen some progress within the area of semantic content-based multimedia modelling, evidenced by existing techniques, within a very short space of time. We can therefore expect that, as the area of semantic content-based multimedia modelling evolves further, the above shortcomings will be ironed out, the techniques will become comprehensive, and the models thus more suitable for general application.

5. References

ADAH, S., CANDAN, K. S., CHEN, S.–S., EROL, K., SUBRAHMANIAN, V. S., The Advanced Video Information System: data structures and query processing, *Multimedia Systems*, **4** (1996), No. 4, 172–186.

AGIUS, H. W., ANGELIDES, M. C., Desktop video conferencing in the organisation, *Information & Management*, **31** (1997a), No. 6, 291–302.

AGIUS, H. W., ANGELIDES, M. C., Integrating logical video and audio segments with content-related information in instructional multimedia systems, *Information and Software Technology*, **39** (1997b), No. 10, 679–694.

AGUIERRE SMITH, T. G., DAVENPORT, G., The Stratification System: a design environment for random access video, In *Proceedings of the Third International Workshop on Network and Operating Systems Support for Digital Audio and Video*, (1992), pp. 250–261.

AIGRAIN, P., ZHANG, H. J., PETKOVIC, D., Content-based representation and retrieval of visual media: a state-of-the-art review, *Multimedia Tools and Applications*, **3** (1996), No. 3, 179–202.

ANGELIDES, M. C., DUSTDAR, S., *Multimedia Information Systems*, Kluwer Academic Publishers, Boston, MA, 1997.

BARBER, R., EQUITZ, W., FALOUTSOS, C., FLICKNER, M., NIBLACK, W., PETKOVIC, D., YANKER, P., Query by content for large on-line image collections, In *A Guided Tour of Multimedia Systems and Applications*, B. Furht, M. Milenkovic, Eds., (1995) pp. 357–378, IEEE Computer Society Press, Los Alamitos, CA.

BRINK, A., MARCUS, S., SUBRAHMANIAN, V. S., Heterogeneous multimedia reasoning, *Computer*, **28** (1995), No. 9, 33–39.

BURRILL, V., KIRSTE, T., WEISS, J., Time-varying Sensitive Regions in dynamic multimedia objects: a pragmatic approach to content-based retrieval from video, *Information and Software Technology*, **36** (1994), No. 4, 213–223.

CHANG, S.–K., HSU, A., Image information systems: where do we go from here? *IEEE Transactions on Knowledge and Data Engineering*, **4** (1992), No. 5, 431–442.

CHRISTEL, M., KANADE, T., MAULDIN, M., REDDY, R., SIRBU, M., STEVENS, S., WACTLAR, H., Informedia Digital Video Library, *Communications of the ACM*, **38** (1995), No. 4, 57–58.

DAVIS, M., Media Streams: an iconic visual language for video annotation, In *Proceedings of the IEEE Symposium on Visual Languages*, (1993) Bergen, pp. 196–202.

FLICKNER, M., SAWHNEY, H., NIBLACK, W., ASHLEY, J., HUANG, Q., DOM, B., GORKANI, M., HAFNER, J., LEE, D., PETKOVIC, D., STEELE, D., YANKER, P., Query by image and video content: the QBIC system, *Computer*, **28** (1995), No. 9, 23–32.

GOODMAN, B. A., Multimedia explanations for intelligent training systems, In *Intelligent Multimedia Interfaces*, M. T. Maybury, Ed. (1993) pp. 148–171, The AAAI Press / The MIT Press, Menlo Park, CA.

GUDIVADA, V. N., RAGHAVAN, V. V., Guest editors' introduction: Content-based image retrieval systems. *Computer*, **28** (1995), No, 9, 18–22.

HIRZALLA, N., FALCHUK, B., KARMOUCH, A., A temporal model for interactive multimedia scenarios, *IEEE MultiMedia*, **2** (1995), No, 3, 24–31.

KANADE, T., Immersion into visual media: new applications of image understanding, *IEEE Expert*, **11** (1996), No. 1, 73–80.

KAZMAN, R., AL–HALIMI, R., HUNT, W., MANTEI, M., Four paradigms for indexing video conferences, *IEEE MultiMedia*, **3** (1996), No, 1, 63–73.

LE GALL, D., MPEG: a video compression standard for multimedia applications, *Communications of the ACM*, **34** (1991), No. 4, 46–58, (Reprinted in *A Guided Tour of Multimedia Systems and Applications*, B. Furht, M. Milenkovic, Eds., (1995) pp. 95–107, IEEE Computer Society Press, Los Alamitos, CA)

LEE, J. C.–M., LI, Q., XIONG, W., VIMS: a video information management system, *Multimedia Tools and Applications*, **4** (1997), No. 1, 7–28.

LIOU, M., Overview of the $p \times 64$ kbit/s video coding standard, *Communications of the ACM*, **34** (1991), No. 4, 59–63.

LITTLE, T. D. C., Time-based media representation and delivery, In *Multimedia Systems*, J. F. K. Buford, Ed. (1994) pp. 175–200, ACM Press / Addison–Wesley, New York, NY.

LITTLE, T. D. C., AHANGER, G., CHEN, H.–J., FOLZ, R. J., GIBBON, J. F., KRISHNAMURTHY, A., LUMBA, P., RAMANATHAN, M., VENKATESH, D., Selection and dissemination of digital video via the Virtual Video Browser, *Multimedia Tools and Applications*, **1** (1995), No, 2, 149–172.

LITTLE, T. D. C., AHANGER, G., FOLZ, R. J., GIBBON, J. F., REEVE, F. W., SHELLENG, D. H., VENKATESH, D., A digital video-on-demand service supporting content-based queries, In *Proceedings of the First ACM International Conference on Multimedia*, (1993) Anaheim, CA, pp. 427–436.

LITTLE, T. D. C., GHAFOOR, A., Synchronization and storage models for multimedia objects, *IEEE Journal on Selected Areas in Communications*, **8** (1990), No. 3, 413–427.

LITTLE, T. D. C., GHAFOOR, A., Spatio-temporal composition of distributed multimedia objects for value-added networks, *Computer*, **24** (1991), No. 10, 42–50.

LITTLE, T. D. C., GHAFOOR, A., Interval-based conceptual models for time-dependent multimedia data, *IEEE Transactions on Knowledge and Data Engineering*, **5** (1993), No. 4, 551–563, (Reprinted in *A Guided Tour of Multimedia Systems and Applications*, B. Furht, M. Milenkovic, Eds., (1995) pp. 257–269, IEEE Computer Society Press, Los Alamitos, CA)

MACKAY, W. E., EVA: an experimental video annotator for symbolic analysis of video data, *SIGCHI Bulletin*, **21** (1989), No. 1, 68–71.

MACKAY, W. E., DAVENPORT, G., Virtual video editing in interactive multimedia applications, *Communications of the ACM*, **32** (1989), No. 7, 802–810.

MEYER–BOUDNIK, T., EFFELSBERG, W., MHEG explained, *IEEE MultiMedia*, **2** (1995), No. 1, 26–38.

MUNDY, J. L., The Image Understanding Environment program, *IEEE Expert*, **10** (1995), No. 6, 64–73.

OOMOTO, E., TANAKA, K., OVID: design and implementation of a video-object database system, *IEEE Transactions on Knowledge and Data Engineering*, **5** (1993), No. 4, 629–643, (Reprinted in *A Guided Tour of Multimedia Systems and Applications*, B. Furht, M. Milenkovic, Eds., (1995) pp. 323–337, IEEE Computer Society Press, Los Alamitos, CA)

PARKES, A. P., CLORIS: a prototype video-based intelligent computer-assisted instruction system, *In Proceedings of RIAO '88*, (1988), pp. 24–50.

PARKES, A. P., SELF, J. A., Towards 'interactive video': a video-based intelligent tutoring environment, In *Intelligent Tutoring Systems: At the Crossroad of Artificial Intelligence and Education*, C. Frasson, G. Gauthier, Eds., (1990) pp. 56–82, Ablex Publishing Corporation, Norwood, NJ.

RADER, G. M., Creating printed music automatically. *Computer*, **29** (1996), No. 6, 61–68.

SMOLIAR, S. W., ZHANG, H. J., Content-based video indexing and retrieval, *IEEE MultiMedia*, **1** (1994), No. 2, 62–72.

STEINMETZ, R., NAHRSTEDT, K., *Multimedia: Computing, Communications and Applications*, Prentice Hall PTR, Upper Saddle River, NJ, 1995.

SWANBERG, D., SHU, C.–F., JAIN, R., Architecture of a multimedia information system for content-based retrieval, In *Proceedings of the Third International Workshop on Network and Operating Systems Support for Digital Audio and Video*, (1992), pp. 387–392.

TANAKA, Y., IntelligentPad as meme media and its application to multimedia databases, *Information and Software Technology*, **38** (1996), No. 3, 201–211.

TONOMURA, Y., AKUTSU, A., TANIGUCHI, Y., SUZUKI, G., Structured video computing, *IEEE MultiMedia*, **1** (1994), No. 3, 34–43.

WACTLAR, H. D., KANADE, T., SMITH, M. A., STEVENS, S. M., Intelligent access to digital video: Informedia project, *Computer*, **29** (1996), No. 5, 46–52.

WEISS, R., DUDA, A., GIFFORD, D. K., Composition and search with a video algebra, *IEEE MultiMedia*, **2** (1995), No. 1, 12–25.

YOSHITAKA, A., KISHIDA, S., HIRAKAWA, M., ICHIKAWA, T., Knowledge-assisted content-based retrieval for multimedia databases, *IEEE MultiMedia*, **1** (1994), No. 4, 12–21.

Received: December, 1997
Accepted: March, 1998

Contact address:
Harry W. Agius
Centre for Multimedia
School of Computing, Information Systems and Mathematics
South Bank University
Southwark Campus
103 Borough Road
London SE1 0AA
UK
phone: +44 (0)171-815 7663
fax: +44 (0)171-815 7499
e-mail: harryagius@acm.org
home page: http://www.sbu.ac.uk/~agiushw/

HARRY WAYNE AGIUS is a Senior Lecturer in Computer Science and a Fellow of the Centre for Multimedia at South Bank University, where he is undertaking research into semantic content-based multimedia modelling, instructional multimedia information systems, and multimedia information superhighways. He holds a BSc in Computing and Information Systems (1994), an MSc in Analysis, Design and Management of Information Systems (1995), and a PhD in Information Systems (1997), all from The London School of Economics and Political Science (University of London). He is a member of the ACM and the IEEE.