Modeling of Video Spatial Relationships in an Object Database Management System*

John Z. Li, M. Tamer Özsu, and Duane Szafron
Department of Computing Science,
University of Alberta
Edmonton, Canada T6G 2H1
{zhong,ozsu,duane}@cs.ualberta.ca

Abstract

A key aspect in video modeling is spatial relationships. In this paper we propose a spatial representation for specifying the spatial semantics of video data. Based on such a representation, a set of spatial relationships for salient objects is defined to support qualitative and quantitative spatial properties. The model captures both topological and directional spatial relationships. We present a novel way of incorporating this model into a video model, and integrating the abstract video model into an object database management system which has rich multimedia temporal operations. The integrated model is further enhanced by a spatial inference engine. The powerful expressiveness of our video model is validated by some query examples.

1 Introduction

Management of multimedia data poses special requirements for database management systems. Many applications depend on spatial relationships among multimedia data. There is significant research on spatial relationships in image databases and geographic information systems (GIS) [1, 5, 6, 13, 16, 17, 19, 21], but very little research has been done on spatial modeling in the context of video data. Video related work mostly concentrates on temporal relationships [7, 10, 12, 14, 20]. We argue that a video spatial model is an essential part of an abstract multimedia information system model which can be used as the basis for declarative queries.

Information about the spatial semantics of a video must be structured so that indexes can be built to efficiently retrieve data from a video database. A *video* consists of a number of *clips*. A *clip* is a consecutive sequence of *frames*, which are the smallest units of video data. *Spatial data* pertains to spatial-oriented objects in a database, including points, polygons, surfaces, and volumes. Spatial relations have been classified [18] into several types, including *topological relations* that describe neighborhood and incidence (e.g., overlap, disjoint); *directional relations* that describe order in space (e.g., south, northwest); and *distance relations* that describe space range between objects (e.g., far, near). We focus on the first two types, i.e., topological and directional relations.

One of the most important issues in modeling video spatial relationships is how to handle user queries. The special requirements of multimedia query languages in supporting spatial relationships have been investigated within the context of specific applications such as image database systems and geographic information systems [19]. From a user's point of view, the following requirements are necessary to support spatial queries in a multimedia information system:

- Support should be provided for object domains which consist of *complex* spatial objects in addition to simple points and alphanumeric domains.
- Support should exist for *direct spatial searches*, which locate the spatial objects in a given area of images. This can resolve queries of the form "*Find all the faces in a given area within an image or a video frame*".
- It should be possible to perform *hybrid spatial searches*, which locate objects based on some attributes and some associations between attributes and the spatial objects. This can resolve queries of the form "*Display the person's name, age, and an image in which he/she is riding on a horse*".
- Support should exist for *complex spatial searches*, which locate spatial objects across the database by using set-theoretic operations over spatial attributes. This can resolve queries of the form "*Find all the roads which pass through city X*".

*The research is supported by a grant from the Canadian Institute for Telecommunications Research (CITR) under the Network of Centres of Excellence (NCE) program of the Government of Canada.

- Support should be provided to perform *direct spatial computations*, which compute specialized simple and aggregate functions from the frames.

- Finally, support should exist for *spatio-temporal queries* which involve not only spatial relations, but temporal relations as well.

We use the Common Video Object Tree model (CVOT) [10] to build an abstract model. This abstract CVOT model is integrated into a temporal object model to provide concrete object database management system (ODBMS) support for video data. The system that we use in this work is TIGUKAT[1] [15], which is an experimental system under development at the University of Alberta. The major contributions of this paper are: the introduction of a unified representation of spatial objects, comprehensive support for user spatial queries, and support for user spatio-temporal queries. Our work is further enhanced by a rich set of spatial inference rules.

The rest of the paper is organized as follows. Section 2 reviews the related work in object spatial representations. Section 3 introduces our representation of object spatial properties and relationships. Section 4 describes a new video model and a novel integration of the new model into an OBMS. Section 5 shows the expressiveness of our spatial representation by discussing some query examples. Section 6 presents our concluding remarks.

2 Related Work

Egenhofer [6] has specified eight fundamental topological relations that can hold between two planar regions. These relations are computed using four intersections over the concepts of *boundary* and *interior* of pointsets between two regions embedded in a two-dimensional space. These four intersections result in eight topological relations. A spatial SQL [5], based on this topological representation, supports direct spatial search, hybrid spatial search, complex spatial search, and direct spatial computation.

Papadias et al. [16, 17] assume a construction process that detects a set of special points in an image, called *representative points*. Every spatial relation in the modeling space can be defined using only these points. Two kinds of representative points are considered: *directional* and *topological*. In the case of using two representative points the directional relations between objects can be defined as intervals which may facilitate the retrieval of spatial objects from a database using an R-tree based indexing mechanism [17].

[1]TIGUKAT (tee-goo-kat) is a term in the language of Canadian Inuit people meaning "objects." The Canadian Inuits (Eskimos) are native to Canada with an ancestry originating in the Arctic regions.

Nabil et al. [13] propose a two dimensional projection interval relationship (2D-PIR) to represent spatial relationships based on Allen's interval algebra and Egenhofer's 4-intersection formalism, which enable a graph representation for pictures based on 2D-PIR to be constructed. In order to overcome some problems of using the minimum bounding rectangle (MBR) with boundaries parallel to horizontal and vertical axes in the 2D-PIR representation, two alternative solutions are proposed: slope projection and the introduction of topological relations. However, neither of these two solutions is complete in the sense that there still exist cases that cannot be handled by the 2D-PIR representation.

The Video Semantic Directed Graph (VSDG) model is a graph-based conceptual video model [4]. One feature of the VSDG model is an unbiased representation of the information that provides a reference framework for constructing a semantically heterogeneous user's view of the video data. This model also suggests using Allen's temporal interval algebra to model spatial relations among objects. However, their definitions of such spatial relations are both incomplete and unsound.

Abdelmoty et al. [1] extend the 4-intersection formalism [6] for topological relations to represent *orientational* relations. The orientational relations require a reference object called an *origin* to establish a spatial relation. The directional relations between two objects are defined using the intersections of these four semi-infinite areas. Hernández [9] defines the composition of topological and directional relations, with the result being pairs of topological/directional relations. Composition is accomplished using *relative topological orientation nodes* as a store for intermediate results. This work is extended in [3] to handle composition of distance and directional relations.

3 Spatial Properties of Salient Objects

A *salient object* is an interesting physical object in a video frame. Each video frame usually has many salient objects, e.g. persons, cars, etc. We use the term objects to refer to salient objects whenever this will not cause confusion.

3.1 Spatial Representations

It is a common strategy in spatial access methods to store object approximations and use these approximations to index the data space in order to efficiently retrieve the potential objects that satisfy the result of a query [17]. Depending on the application domain, there are several options in choosing object approximations. MBR has been used extensively to approximate objects because they need only two points for their representation. While MBR demonstrates some disadvantages when approximating non-convex or diagonal objects, they are the most commonly used approximations in spatial applications. Hence, we use MBR to represent objects in our system.

Definition 1 The *bounding box* of a salient object A_i is defined by its MBR (A_{ix}, A_{iy}) and a depth A_{iz} where $A_{ix} = [x_{s_i}, x_{f_i}], A_{iy} = [y_{s_i}, y_{f_i}], A_{iz} = [z_{s_i}, z_{f_i}]$. x_{s_i} and x_{f_i} are A_i's projection on the X axis with $x_{s_i} \leq x_{f_i}$ and similarly for y_{s_i}, y_{f_i}, z_{s_i}, and z_{f_i}. The *spatial property* of a salient object A_i is defined by a quadruple $(A_{ix}, A_{iy}, A_{iz}, C_i)$ where C_i is the centroid of A_i. The centroid is represented by a three dimensional point (x_i, y_i, z_i). This can be naturally extended by considering a time dimension: $(A^t_{ix}, A^t_{iy}, A^t_{iz}, C^t_i)$ to capture the spatial property of a salient object A_i at time t.

The spatial property of an object is described by its bounding volume and centroid. Suppose the spatial property of A_i is $(A^{t_1}_{ix}, A^{t_1}_{iy}, A^{t_1}_{iz}, C^{t_1}_i)$ at time t_1 and is $(A^{t_2}_{ix}, A^{t_2}_{iy}, A^{t_2}_{iz}, C^{t_2}_i)$ at time t_2. The displacement of A_i over time interval $I = [t_s, t_f]$ is $DISP(A_i, I) \equiv \sqrt{(x_i^{t_s} - x_i^{t_f})^2 + (y_i^{t_s} - y_i^{t_f})^2 + (z_i^{t_s} - z_i^{t_f})^2}$ which is the movement of the centroid of A_i. Also the Euclidean distance between two objects A_i and A_j at time t_k is $DIST(A_i, A_j, t_k) \equiv \sqrt{(x_i^{t_k} - x_j^{t_k})^2 + (y_i^{t_k} - y_j^{t_k})^2 + (z_i^{t_k} - z_j^{t_k})^2}$ which is also characterized by the centroid of A_i and A_j. Our goal is to support both quantitative and qualitative spatial retrieval.

Spatial qualitative relations between objects are very important in multimedia object databases because they implicitly support *fuzzy queries* which are captured by similarity matching or qualitative reasoning. Allen [2] gives a temporal interval algebra (Table 1) for representing and reasoning about temporal relations between events represented as intervals. The elements of the algebra are sets of seven basic relations that can hold between two intervals and their inverse relations.

Relation	Symbol	Inverse	Meaning
B before C	`b`	`bi`	BBB CCC
B meets C	`m`	`mi`	BBBCCC
B overlaps C	`o`	`oi`	BBB CCC
B during C	`d`	`di`	BBB CCCCC
B starts C	`s`	`si`	BBB CCCCC
B finishes C	`f`	`fi`	BBB CCCCC
B equal C	`e`	`e`	BBB CCC

Table 1. 13 Temporal Interval Relations

The temporal interval algebra essentially consists of the topological relations in one dimensional space, enhanced by the distinction of the order of the space. We consider 12 directional relations in our model and classify them into the following three categories: *strict directional relations* (north, south, west, and east), *mixed directional relations* (northeast, southeast, northwest, and southwest), and *positional relations* (above, below, left, and right). The definitions of these relations in terms of Allen's temporal algebra are given in Table 2. The symbols $\wedge$ and $\vee$ are the standard logical *AND* and *OR* operators, respectively. A short notation $\{\}$ is used to distribute the $\vee$ operator over interval relations. For example $A_{ix}\,\{\mathtt{b}, \mathtt{m}, \mathtt{o}\}\,A_{jx}$ is equivalent to $A_{ix}\,\mathtt{b}\,A_{jx} \vee A_{ix}\,\mathtt{m}\,A_{jx} \vee A_{ix}\,\mathtt{o}\,A_{jx}$.

Among Egenhofer's eight topological relations there are two inverse relations: *covers* vs *covered_by* and *inside* vs *contains*. Hence, only six topological relations are defined here, as shown in the last part of Table 2. Note that the definitions of directional and topological relations are based on two dimensional (2D) space since video frames are usually mapped into 2D images. In 3D space, the depth of an object has to be considered and the extension is straightforward. Figure 1 shows all the cases of A_i *northwest* of A_j ($A_i\,\mathtt{NW}\,A_j$).

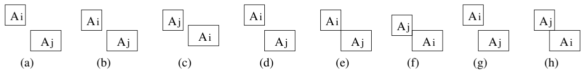

Figure 1. All the Cases of $A_i\,\mathtt{NW}\,A_j$

Figure 2 shows all the topological relations. While any two spatial objects always have a topological relation, they may not have any directional relation. For instance, consider objects A_i and A_j in the case of $A_i\,\mathtt{OL}\,A_j$ in Figure 2. A_i and A_j have no directional relation. This coincides with our intuition about spatial objects.

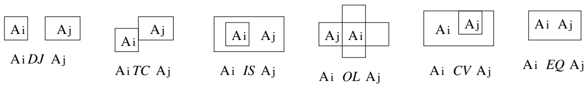

Figure 2. Definitions of Topological Relations

In our definition, if two objects overlap, they do not have any directional relation. This is certainly an arguable definition. In Figure 3 it is natural to say A_i *overlaps* A_j in (a) and A_i *west* of A_j in (c). However, it may not be reasonable to say they are still true in cases (b) and (d). The problem comes from the representation of the temporal interval algebra which does not distinguish the degree of the overlap regions.

3.2 Reasoning about Spatial Relations

Logic-based representations, such as rules, are used in qualitative spatial reasoning since they provide a natural and

Relation	Meaning	Definition
$A_i\ \texttt{ST}\ A_j$	South	$A_{ix}\ \{\texttt{d}, \texttt{di}, \texttt{s}, \texttt{si}, \texttt{f}, \texttt{fi}, \texttt{e}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{b}, \texttt{m}\}\ A_{jy}$
$A_i\ \texttt{NT}\ A_j$	North	$A_{ix}\ \{\texttt{d}, \texttt{di}, \texttt{s}, \texttt{si}, \texttt{f}, \texttt{fi}, \texttt{e}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{bi}, \texttt{mi}\}\ A_{jy}$
$A_i\ \texttt{WT}\ A_j$	West	$A_{ix}\ \{\texttt{b}, \texttt{m}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{d}, \texttt{di}, \texttt{s}, \texttt{si}, \texttt{f}, \texttt{fi}, \texttt{e}\}\ A_{jy}$
$A_i\ \texttt{ET}\ A_j$	East	$A_{ix}\ \{\texttt{bi}, \texttt{mi}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{d}, \texttt{di}, \texttt{s}, \texttt{si}, \texttt{f}, \texttt{fi}, \texttt{e}\}\ A_{jy}$
$A_i\ \texttt{NW}\ A_j$	Northwest	$(A_{ix}\ \{\texttt{b}, \texttt{m}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{bi}, \texttt{mi}, \texttt{oi}\}\ A_{jy}) \vee (A_{ix}\ \{\texttt{o}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{bi}, \texttt{mi}\}\ A_{jy})$
$A_i\ \texttt{NE}\ A_j$	Northeast	$(A_{ix}\ \{\texttt{bi}, \texttt{mi}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{bi}, \texttt{mi}, \texttt{oi}\}\ A_{jy}) \vee (A_{ix}\ \{\texttt{oi}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{bi}, \texttt{mi}\}\ A_{jy})$
$A_i\ \texttt{SW}\ A_j$	Southwest	$(A_{ix}\ \{\texttt{b}, \texttt{m}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{b}, \texttt{m}, \texttt{o}\}\ A_{jy}) \vee (A_{ix}\ \{\texttt{o}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{b}, \texttt{m}\}\ A_{jy})$
$A_i\ \texttt{SE}\ A_j$	Southeast	$(A_{ix}\ \{\texttt{b}, \texttt{m}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{b}, \texttt{m}, \texttt{o}\}\ A_{jy}) \vee (A_{ix}\ \{\texttt{oi}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{b}, \texttt{m}\}\ A_{jy})$
$A_i\ \texttt{LT}\ A_j$	Left	$A_{ix}\ \{\texttt{b}, \texttt{m}\}\ A_{jx}$
$A_i\ \texttt{RT}\ A_j$	Right	$A_{ix}\ \{\texttt{bi}, \texttt{mi}\}\ A_{jx}$
$A_i\ \texttt{BL}\ A_j$	Below	$A_{iy}\ \{\texttt{b}, \texttt{m}\}\ A_{jy}$
$A_i\ \texttt{AB}\ A_j$	Above	$A_{iy}\ \{\texttt{bi}, \texttt{mi}\}\ A_{jy}$
$A_i\ \texttt{EQ}\ A_j$	Equal	$A_{ix}\ \{\texttt{e}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{e}\}\ A_{jy}$
$A_i\ \texttt{IS}\ A_j$	Inside	$A_{ix}\ \{\texttt{d}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{d}\}\ A_{jy}$
$A_i\ \texttt{CV}\ A_j$	Cover	$(A_{ix}\ \{\texttt{di}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{fi}, \texttt{si}, \texttt{e}\}\ A_{jy}) \vee (A_{ix}\ \{\texttt{e}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{di}, \texttt{fi}, \texttt{si}\}\ A_{jy}) \vee$ $(A_{ix}\ \{\texttt{fi}, \texttt{si}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{di}, \texttt{fi}, \texttt{si}, \texttt{e}\}\ A_{jy})$
$A_i\ \texttt{OL}\ A_j$	Overlap	$A_{ix}\ \{\texttt{d}, \texttt{di}, \texttt{s}, \texttt{si}, \texttt{f}, \texttt{fi}, \texttt{o}, \texttt{oi}, \texttt{e}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{d}, \texttt{di}, \texttt{s}, \texttt{si}, \texttt{f}, \texttt{fi}, \texttt{o}, \texttt{oi}, \texttt{e}\}\ A_{jy}$
$A_i\ \texttt{TC}\ A_j$	Touch	$(A_{ix}\ \{\texttt{m}, \texttt{mi}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{d}, \texttt{di}, \texttt{s}, \texttt{si}, \texttt{f}, \texttt{fi}, \texttt{o}, \texttt{oi}, \texttt{m}, \texttt{mi}, \texttt{e}\}\ A_{jy}) \vee$ $(A_{ix}\ \{\texttt{d}, \texttt{di}, \texttt{s}, \texttt{si}, \texttt{f}, \texttt{fi}, \texttt{o}, \texttt{oi}, \texttt{m}, \texttt{mi}, \texttt{e}\}\ A_{jx} \wedge A_{iy}\ \{\texttt{m}, \texttt{mi}\}\ A_{jy})$
$A_i\ \texttt{DJ}\ A_j$	Disjoint	$A_{ix}\ \{\texttt{b}, \texttt{bi}\}\ A_{jx} \vee A_{iy}\ \{\texttt{b}, \texttt{bi}\}\ A_{jy}$

Table 2. Directional and Topological Relation Definitions

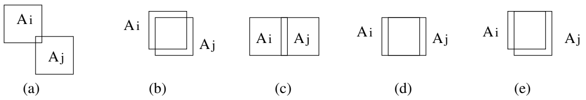

Figure 3. Some Non-directional Spatial Cases

flexible way to represent spatial knowledge [16]. Such representations usually have well defined semantics and simple inference rules that can be integrated into any deductive system. For example, if there are A_1 $north$ of A_2, and A_2 $overlap$ A_3, and A_3 $north$ of A_4, then we deduce A_1 $above$ A_4, which can be expressed as a rule

$$A_1\ \texttt{NT}\ A_2 \wedge A_2\ \texttt{OL}\ A_3 \wedge A_3\ \texttt{NT}\ A_4 \Rightarrow A_1\ \texttt{AB}\ A_4.$$

A spatial inference rule can support spatial analysis without transforming any spatial knowledge into the domain of underlying coordinates and point-region representations. We have constructed a comprehensive set of spatial inference rules [11] and have proven the correctness of those rules. A broad range of qualitative spatial queries are supported as both topological and directional relations are considered. Since all the rules are propositional Horn clauses, they can be easily integrated into any multimedia object database by using either a simple inference engine or a lookup table.

4 Video Modeling

Video modeling is the process of translating raw video data into an efficient internal representation which helps to capture video semantics. The procedural process of extracting video semantics from a video is called *video segmentation*. In this section we briefly introduce the Common Video Object Tree (CVOT) model (a video model) and its integration into a temporal OBMS.

4.1 The Common Video Object Tree Model

There are several different ways to segment a video into clips, two of which are *fixed time intervals* and *shots*. A *fixed time interval* segmentation approach divides a video into equal length clips using a predefined time interval (e.g. 2 seconds) while a *shot* is a set of continuous frames captured by a single camera action. Two common problems with existing models are restrictive video segmentation and poor user query support. The CVOT model [10] is primarily designed to deal with these two problems. One unique feature of the CVOT model is that a clip overlap is allowed. This can provide considerable benefit in modeling *events* which are discussed in Section 4.3. Generally, a smooth transition of one event to another event, *event fading*, requires having some scene or activity overlap between the end of the previous event and the start of the next event. Such a transition phase is usually reflected in a few frames as shown in Figure 4.

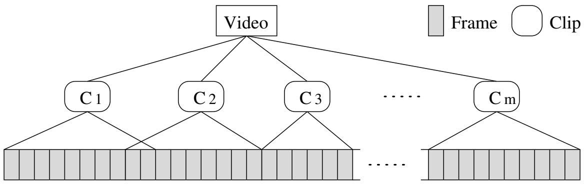

Figure 4. Stream-based Video

The main purpose of the CVOT model is to find all the

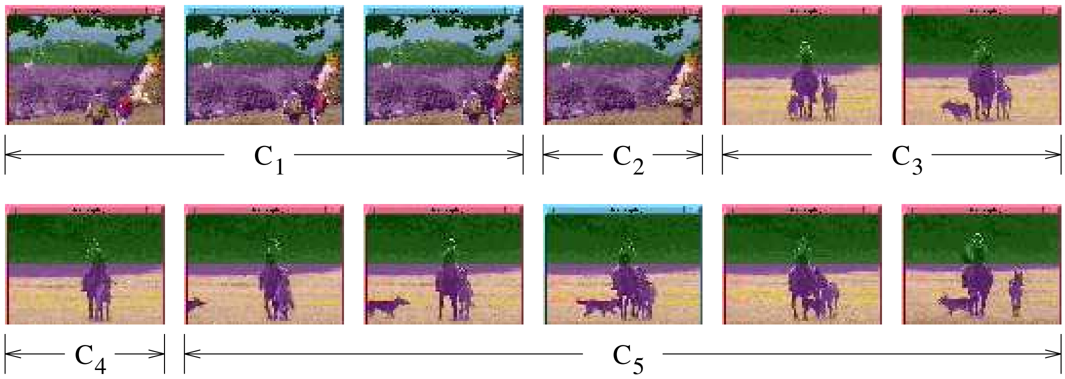

Figure 5. Salient Objects and Clips

common objects among clips and to group clips according to these objects. A tree structure is used to represent such a clip group. The *time interval* of a clip is defined according to the clip's starting frame and ending frame.

Example 1 Figure 5 shows a video in which John and Mary walk toward their house. Later, Mary rides a horse on a ranch with her colt and dog. Let us assume that the salient objects are $SO = \{\mathsf{john}, \mathsf{mary}, \mathsf{house}, \mathsf{tree}, \mathsf{horse}, \mathsf{colt}, \mathsf{dog}\}$. If the video is segmented as in Figure 5, then we have five clips $C = \{C_1, C_2, C_3, C_4, C_5\}$ with john, mary, house, and tree in C_1, john, house, and tree in C_2, mary, horse, colt, and dog in C_3, mary, horse, and colt in C_4, and mary, horse, colt, and dog in C_5. Figure 6 shows a CVOT instance for Figure 5. The CVOT model directly supports queries of the type "*Find all the clips in which a salient object appears*" and "*How long does a particular salient object occur in a video*".

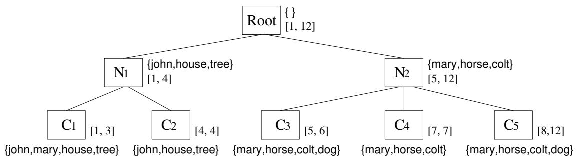

Figure 6. A CVOT Built from Figure 3

4.2 The OBMS Support

CVOT is an abstract model. To have proper database management support for continuous media, this model needs to be integrated into a data model. We work within the framework of a uniform, behavioral object model such as the one supported by the TIGUKAT system [15]. The important characteristics of the model, from the perspective of this paper, are its *behaviorality* and and its *uniformity*. The model is *behavioral* in the sense that all access and manipulation of objects is based on the application of behaviors to objects. The model is *uniform* in that every component of information, including its semantics, is modeled as a *first-class object* with well-defined behavior.

The primitive objects of the model include: *atomic entities* (reals, integers, strings, etc.); *types* for defining common features of objects; *behaviors* for specifying the semantics of operations that may be performed on objects; *functions* for specifying implementations of behaviors over types; *classes* for automatic classification of objects based on type; and *collections* for supporting general heterogeneous groupings of objects. In this paper, a reference prefixed by "`T_`" refers to a type, "**C_**" to a class, "*B_*" to a behavior, and "`T_X`< `T_Y` >" to the type `T_X` parameterized by the type `T_Y`. For example, `T_person` refers to a type, **C_person** to its class, *B_age* to one of its behaviors and `T_coll`< `T_person` > (`T_coll` stands for `T_collection`) to the type of collections of persons. A reference such as David, without a prefix, denotes some other application specific reference. Consequently, the model separates the definition of object characteristics (a *type*) from the mechanism for maintaining instances of a particular type (a *class*).

Temporality has been added to this model [8] as type and behavior extensions of the type system discussed above. Figure 7 gives part of the time type hierarchy that includes the temporal ontology and temporal history features of the temporal model. Unary operators which return the lower bound, upper bound, and length of the time interval are defined. The model supports a rich set of ordering operations among intervals, e.g., *before*, *overlaps*, *during*, etc. (see Table 1) as well as set-theoretic operations, viz. *union*, *intersection* and *difference*. A time duration can be added or subtracted from a time interval to return another time interval. A time interval can be expanded or shrunk by a specified time duration.

One requirement of a temporal model is an ability to adequately represent and manage histories of objects and

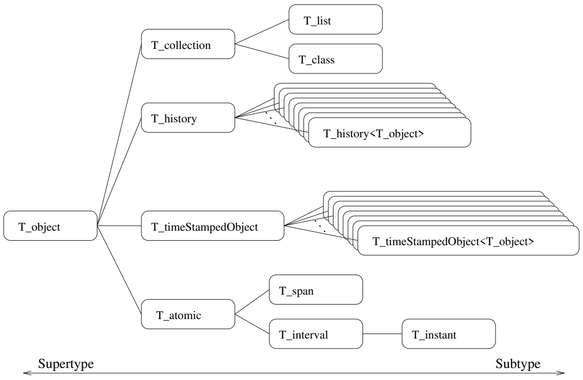

Figure 7. The Basic Time Type Hierarchy

real-world events. Our model represents the temporal histories of objects whose type, is T_X as objects of the T_history<T_X> type as shown in Figure 7. A temporal history consists of objects and their associated timestamps (time intervals or time instants). A *timestamped object* (of type T_timestampedObject<T_X>) knows its timestamp and its associated object (value) at the timestamp. A temporal history is made up of such objects. Table 3 gives the behaviors defined on histories and timestamped objects. Behavior *B_history* defined on T_history<T_X> returns the set (collection) of all timestamped objects that comprise the history. Another behavior defined on history objects, *B_insert*, timestamps and inserts an object into the history. The *B_validObj* behavior allows the user to get the objects in the history that were valid at (during) the given time.

Each timestamped object is an instance of the T_tsObj<T_X> type where T_tsObj stands for T_timeStampedObject. This type represents objects and their corresponding timestamps. Behaviors *B_value* and *B_timeStamp*, defined on T_tsObj, return the value and the timestamp of a timestamped object, respectively.

4.3 System Integration

Integrated multimedia systems can result in a uniform object model, simplified system support and, possibly, better performance. Figure 8 shows our video type system. The types that are in a grey shade are directly related to the CVOT model and they will be discussed in detail in the following subsections.

4.3.1 Integrated System Model

We start by defining the T_video type to model videos. An instance of T_video has all the semantics of a video and is modeled as a history of clips. We model a clip set by defining the behavior *B_clips* in T_video. *B_clips* returns a history object of type T_history< **T_clip** >, whose elements are timestamped objects of type T_clip (**T_tsObj** < **T_clip** >).

The behavior *B_cvotTree* on T_video returns an instance of a CVOT for a video. A common question to myVideo would be its length (duration). This is modeled by the *B_length* behavior. Video information should also include metadata, such as the publishers, producers, publishing date, etc. A video can also be played by using *B_play*[2].

Each clip has a set of consecutive frames, which is modeled by T_history<T_frame>. All the salient objects within a clip are grouped by the behavior *B_slObjects* which returns an instance of T_coll< **T_history** < **T_slObjects** >>. Similarly, all the events within a clip are grouped by the behavior *B_events*, which returns an instance of T_coll< **T_history** < **T_event** >>.

The basic building unit of a clip is the frame which is modeled by T_frame in Table 3. A frame knows its location within a clip and such a location is modeled by a time instant (*B_location*), which can be a relative frame number. We model frames within a clip as a history which is identical to how we model clips within a video. Different formats of a frame are defined by the behavior *B_format* of T_frame. *B_format* on type T_frameFormat, an enumerated type, defines the format of a frame. The content of a frame, *B_content*, is an image which defines many image properties such as width, height and color.

4.3.2 Modeling Video Features

The semantics or contents of a video are usually expressed by its *features* which include video attributes and the relationships between these attributes. Typical video features are salient objects and *events*. An *event* is a kind of activity which may involve many different salient objects over a time period, like holding a party and riding a horse etc.

Since objects can appear multiple times in a clip or a video, we model the history of an object as a timestamped object of type T_history< **T_slObject** >. The behavior *B_slObjects* of T_clip returns all the objects within a clip. Using histories to model objects and events results in powerful queries, as will be shown in the next subsection. Furthermore, it enables us to uniformly capture the temporal semantics of video data because a video is modeled as a history of clips and a clip is modeled as a history of frames. *B_activity* on T_event in Table 3 identifies the type of events and *B_roles* identifies all the objects involved in an event. *B_inClips* indicates all the clips in which this event occurs. It is certainly reasonable to include other information, such as the location and the real-world time of

[2] A full set of behaviors can, of course, be defined on T_video to enable typical actions, such as pause, fast forward, and rewind. We do not elaborate on these any further in this paper.

T_history<T_X>	*B_history*:	T_coll<T_tsObj<T_X≫
	B_insert:	T_X,T_interval → T_boolean
	B_validObj:	T_interval → T_coll<T_tsObj<T_X≫
T_tsObj<T_X>	*B_value*:	T_X
	B_timeStamp:	T_interval
T_video	*B_clips*:	T_history<T_clip>
	B_cvotTree:	T_tree
	B_search:	T_slObject, T_tree → T_tree
	B_length:	T_span
	B_publisher:	T_coll<T_company>
	B_producer:	T_coll<T_person>
	B_date:	T_instant
	B_play:	T_boolean
T_clip	*B_frames*:	T_history< T_frame >
	B_slObjects:	T_coll<T_history<T_slObject≫
	B_events:	T_coll<T_history<T_event≫
T_frame	*B_location*:	T_instant
	B_format:	T_videoFormat
	B_content:	T_image
T_event	*B_activity*:	T_eventType
	B_roles:	T_coll<T_slObject>
	B_inClips:	T_video → T_history< T_clip >
T_slObject	*B_inClips*:	T_video → T_history< T_clip >
	B_category:	T_slObjectCategory
	B_status:	T_status
T_spObject	*B_xinterval*:	T_interval
	B_yinterval:	T_interval
	B_zinterval:	T_interval
	B_centroid:	T_point
	B_area:	T_real
	B_disp:	T_interval, T_interval → T_real
	B_distance	T_spObject, T_interval → T_real
	B_south:	T_spObject → T_boolean
	B_north:	T_spObject → T_boolean
	B_west:	T_spObject → T_boolean
	B_east:	T_spObject → T_boolean
	B_northwest:	T_spObject → T_boolean
	B_northeast:	T_spObject → T_boolean
	B_southwest:	T_spObject → T_boolean
	B_southeast:	T_spObject → T_boolean
	B_left:	T_spObject → T_boolean
	B_right:	T_spObject → T_boolean
	B_below:	T_spObject → T_boolean
	B_above:	T_spObject → T_boolean
	B_equal:	T_spObject → T_boolean
	B_inside:	T_spObject → T_boolean
	B_overlap:	T_spObject → T_boolean
	B_cover:	T_spObject → T_boolean
	B_touch:	T_spObject → T_boolean
	B_disjoint:	T_spObject → T_boolean

Table 3. Primitive Behavior Signatures

an event, into type `T_event`, but they are not important to our discussion.

Any object occupying some space is an instance of `T_spObject`. In type `T_slObject`, a subtype of `T_spObject`, the behavior *B_inClips* returns all the clips in which the object appears. *B_category* describes the category of objects, such as static objects (e.g. mountains, houses, trees) and mobile objects (e.g., cars, horses, boats). *B_status* may be used to define some other attributes of objects, such as *rigidness*. The rest of the behaviors are related to the directional and topological relations and are self-explanatory. Table 3 also shows the behavior signatures of spatial objects.

5 Query Examples

In this subsection we present some examples to show the expressiveness of our model from the spatial properties point of view. We first introduce an object calculus [15]. The alphabet of the calculus consists of object constants (a, b, c, d), object variables (o, p, q, u, v, x, y, z), dyadic predicates $(=, \in, \notin)$, an n-ary predicate (C, P, Q), a function symbol (β) called a *behavior specification* (Bspec), and logical connectives $(\exists, \forall, \wedge, \vee, \neg)$. A *term* is a constant, a variable or a Bspec. An *atomic formula* (*atom*) has an equivalent Bspec representation. From atoms, *well-formed formulas* (WFFs) are built to construct the declarative calculus expressions of the language. WFFs are defined recur-

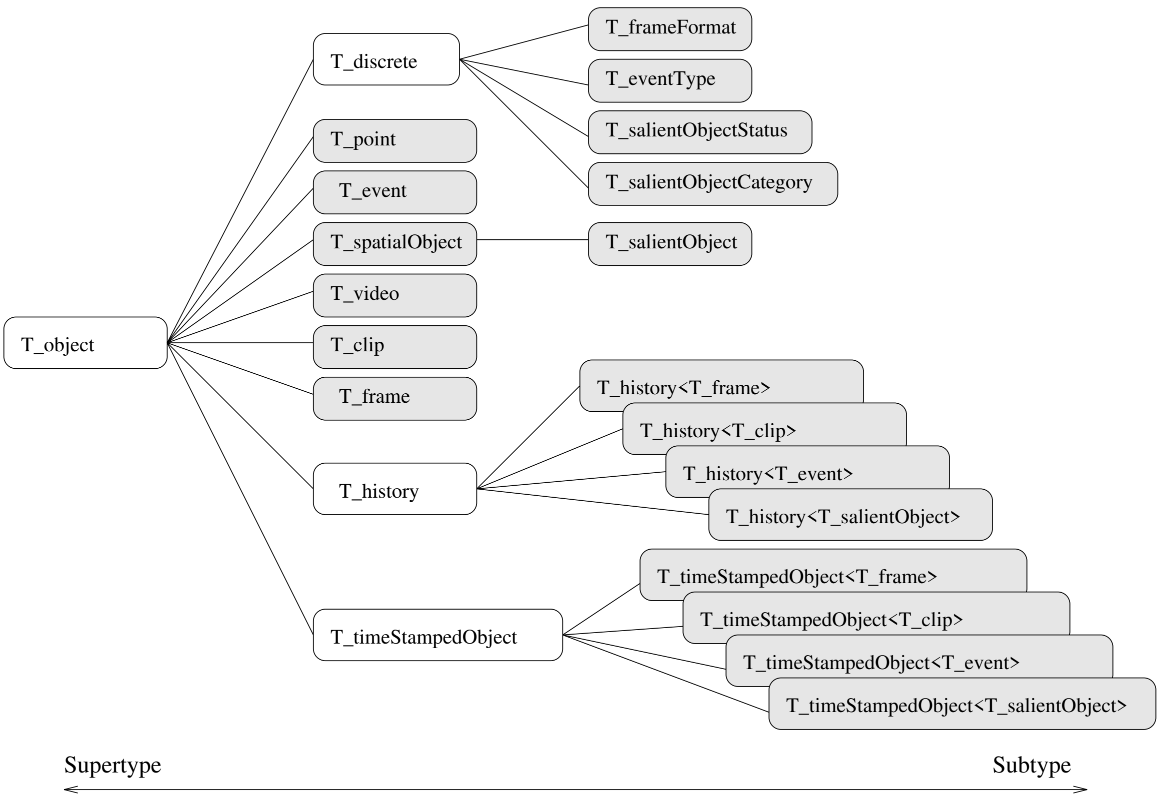

Figure 8. The Video Type System

sively from atoms in the usual way using the connectives $\wedge, \vee, \neg$ and the quantifiers $\exists$ and $\forall$. A query is an object calculus expression of the form $\{t_1, \ldots, t_n | \phi(o_1, \ldots, o_n)\}$ where $t_1, \ldots, t_n$ are the terms over the multiple variables and $o_1, \ldots, o_n$. ϕ is a WFF.

We assume that all the queries are posted to a particular video instance myVideo and salient objects and events are timestamped objects as discussed in Section 4. We also assume that all clips are timestamped clips and $\mathtt{c} \in \mathtt{myVideo}.B\_clips.B\_history$ where c is an arbitrary clip. For simplicity, if a clip, salient object, or event belongs to a timestamped object class **C_tsObj**, we omit it in the query calculus expressions.

Query 1 Is the salient object a in clip c?

$\{\mathsf{q} \mid \mathsf{q} = \mathsf{a}.B\_timeStamp.B\_during(\mathsf{c}.B\_timeStamp)\}$.

The query checks whether the time interval of object a is a subinterval of clip c. For convenience, predicate $IN(\mathsf{o},\mathsf{c})$ is used to denote that object o is in clip c.

Query 2 Find all the objects in a given area a at time t.

$$\{z \mid \exists \mathsf{c}(\mathbf{C\_interval}(t) \wedge \mathbf{C\_slObject}(\mathsf{a}) \wedge \mathbf{C\_history}(\mathsf{x}) \wedge \mathbf{C\_collection}(\mathsf{y}) \wedge \mathsf{x} \in \mathsf{c}.B\_value.B\_slObjects \wedge \mathsf{y} \in \mathsf{x}.B\_history \wedge \mathsf{t}.B\_during(\mathsf{y}.B\_timeStamp) \wedge \mathsf{z} = \mathsf{y}.B\_value \wedge \mathsf{z}.B\_inside(\mathsf{a}))\}$$

where c is an instance of a timestamped clip. Suppose we can find a clip (c) in which some object (y) appears at time t (t.B_during(y.$B\_timeStamp$)), then this object (y) is selected to check if it is inside area a.

Query 3 Find all the objects that are very close to object a.

$$\{z \mid \exists \mathsf{y}(\mathbf{C\_history}(\mathsf{x}) \wedge \mathbf{C\_real}(\mathsf{h}) \wedge IN(\mathsf{a},\mathsf{c}) \wedge \forall \mathsf{x} (\mathsf{x} \in \mathsf{c}.B\_slObjects \wedge \mathsf{y} \in \mathsf{x}.B\_history \wedge \mathsf{a}.B\_timeStamp.B\_during(\mathsf{y}.B\_timeStamp) \wedge \mathsf{y}.B\_value.B\_distance(\mathsf{a}.B\_value).B\_lessthan(h) \wedge \mathsf{z} = \mathsf{y}.B\_value))\}$$

where a is an instance of $\mathtt{T\_tsObj} < \mathtt{T\_spObject} >$ and h is a predefined threshold value for measuring *very close*. In this query formula we locate the clip c in which a appears and go through all the salient objects in c. If any object shows up at a's time (a.$B\_timeStamp$.B_during(y.$B\_timeStamp$)), then the distance between this object and a is computed and its value is compared with a predefined threshold h.

Query 4 Find a video clip in which a dog approaches Mary from the left.

$$\{\mathsf{c} \mid \exists \mathsf{x} \exists \mathsf{x}_2 \exists \mathsf{x}_3 \exists \mathsf{y} \exists \mathsf{y}_2 \exists \mathsf{y}_3 (\mathbf{C\_history}(\mathsf{x}) \wedge \mathbf{C\_history}(\mathsf{y}) \wedge \mathbf{C\_real}(\mathsf{h}_1) \wedge \mathbf{C\_real}(\mathsf{h}_2) \wedge \mathsf{x},\mathsf{y} \in \mathsf{c}.B\_value.B\_slObjects \wedge \mathsf{x}_2,\mathsf{x}_3 \in \mathsf{x}.B\_history \wedge \mathsf{y}_2,\mathsf{y}_3 \in \mathsf{y}.B\_history \wedge \mathsf{x}_2.B\_value = \mathsf{dog} \wedge \mathsf{y}_2.B\_value = \mathsf{mary} \wedge \mathsf{x}_2.B\_timeStamp.B\_equal(\mathsf{y}_2.B\_timeStamp)$$

$\wedge x_2.B\_value.B\_left(y_2.B\_value) \wedge x_3.B\_value = a \wedge$
$y_3.B\_value = b \wedge x_3.B\_timeStamp.B\_equal$
$(y_3.B\_timeStamp) x_3.B\_value.B\_left(y_3.B\_value) \wedge$
$x_3.B\_timeStamp.B\_after(x_2.B\_timeStamp) \wedge$
$x_2.B\_value.B\_disp(x_2.B\_timeStamp,$
$x_3.B\_timeStamp).B\_greaterThan(h_1) \wedge y_2.B\_value.$
$B\_disp(x_2.B\_timeStamp, x_3.B\_timeStamp).$
$B\_lessThan(h_2))\}$

where dog and mary are two instances of `T_slObject`. Suppose clip c is what we are looking for and two salient objects, denoted by x_2 and x_3, are introduced to represent dog and to reflect different time stamps. The same strategy is used for the object mary. We compute the dog's displacement over the time period and enforce this displacement to be greater than a predefined value h_1 to insure that enough movement is achieved. The displacement of mary is also computed and is required to be less than a predefined value h_2. This particular requirement of mary is to guarantee that it is the dog approaching Mary from the left, instead of Mary approaching the dog from the right.

6 Conclusions

Spatial relationships play a very important role in multimedia information systems. In this paper we explore the spatial properties of salient objects in a video object database. The major contribution of this work is that the proposed spatial model supports a comprehensive set of queries. Both the qualitative and quantitative spatial properties of objects are considered. We show that the integrated CVOT model supports the above requirements. The support for object spatial relationships is further strengthened by incorporating a rich set of spatial inference rules. A uniform approach to modeling video objects using histories is also discussed and the expressiveness of the CVOT model is demonstrated by means of example queries within the context of the TIGUKAT system. We intend to build a video query language based on the CVOT model. The spatial, temporal, and spatio-temporal queries can be translated into the query calculus and then the query algebra. It is then possible to optimize these queries using object query optimization techniques.

References

[1] A. I. Abdelmoty and B. A. El-Geresy. An intersection-based formalism for representing orientation relations in a geographic database. In *Proc. of ACM Conf. on Advances in GIS Theory*, Gaithersburg, MD, 1994.

[2] J. F. Allen. Maintaining knowledge about temporal intervals. *Commun. of ACM*, 26(11):832—843, 1983.

[3] E. Clementini, J. Sharma, and M. J. Egenhofer. Modelling topological spatial relations: Strategies for query processing. *Computers and Graphics*, 18(6):815—822, 1994.

[4] Y. F. Day, S. Dagtas, M. Iino, A. Khokhar, and A. Ghafoor. Object-oriented conceptual modeling of video data. In *Proc. of Int'l Conf. on Data Eng.*, Taiwan, 1995.

[5] M. Egenhofer. Spatial SQL: A query and presentation language. *IEEE Trans. on Knowledge and Data Eng.*, 6(1):86—95, Jan. 1994.

[6] M. Egenhofer and R. Franzosa. Point-set topological spatial relations. *Int'l J. of Geographical Information Systems*, 5(2):161—174, 1991.

[7] S. Gibbs, C. Breiteneder, and D. Tscichritzis. Data modeling of time-based media. In *Proc. of ACM SIGMOD*, pages 91—102, Minneapolis, May 1994.

[8] I. A. Goralwalla, Y. Leontiev, M. T. Özsu, and D. Szafron. Modeling time: Back to basics. TR-96-03, Dept. of Comp. Sci., Univ. of Alberta, Feb. 1996.

[9] D. Hernández. *Qualitative Representation of Spatial Knowledge*. Springer-Verlag, New York, 1994.

[10] J. Z. Li, I. Goralwalla, M. T. Özsu, and D. Szafron. Video modeling and its integration in a temporal object model. TR-96-02, Dept. of Comp. Sci., Univ. of Alberta, Jan. 1996.

[11] J. Z. Li, M. T. Özsu, and D. Szafron. Spatial reasoning rules in multimedia management systems. TR-96-05, Dept. of Comp. Sci., Univ. of Alberta, Mar. 1996.

[12] T. C. C. Little and A. Ghafoor. Interval-based conceptual models for time-dependent multimedia data. *IEEE Trans. on Knowledge and Data Eng.*, 5(4):551—563, August 1993.

[13] M. Nabil, J. Shepherd, and H. H. Ngu. 2D projection interval relationships: A symbolic representation of spatial relationships. In *Proc. of 4th Int'l Symp. on Large Spatial Databases*, pages 292—309, Portland, Aug. 1995.

[14] E. Oomoto and K. Tanaka. OVID: Design and implementation of a video-object database system. *IEEE Trans. on Knowledge and Data Eng.*, 5(4):629—643, Aug. 1993.

[15] M. T. Özsu, R. J. Peters, D. Szafron, B. Irani, A. Lipka, and A. Munoz. TIGUKAT: A uniform behavioral objectbase management system. *The VLDB J.*, 4:100—147, 1995.

[16] D. Papadias and T. Sellis. The qualitative representation of spatial knowledge in two-dimensional space. *The VLDB J.*, 4:100—138, 1994.

[17] D. Papadias, Y. Theodoridis, T. Sellis, and M. J. Egenhofer. Topological relations in the world of minimum bounding rectangles: A study with R-trees. In *Proc. of ACM SIGMOD*, pages 92—103, San Jose, CA, May 1995.

[18] D. Pullar and M. Egenhofer. Toward formal definitions of topological relations among spatial objects. In *Proc. of the 3rd Int'l Symposium on Spatial Data Handling*, pages 165—176, Sydney, Australia, 1988.

[19] N. Roussopoulos, C. Faloutsos, and T. Sellis. Spatial data models and query processing. *IEEE Trans. on Software Engineering*, 14(5):639—650, May 1988.

[20] G. A. Schloss and M. J. Wynblatt. Building temporal structures in a layered multimedia data model. In *Proc. of ACM Multimedia'94*, pages 271—278, San Francisco, CA, 1994.

[21] P. Sistla, C. Yu, and R. Haddack. Reasoning about spatial relationships in picture retrieval systems. In *Proc. of the 20th Int'l Conf. on VLDB*, pages 570—581, 1994.