Object-Oriented Conceptual Modeling of Video Data

Young Francis Day, Serhan Dağtaş, Mitsutoshi Iino
Ashfaq Khokhar, and Arif Ghafoor
Distributed Multimedia Systems Laboratory
School of Electrical Engineering
Purdue University, West Lafayette, IN 47907

Abstract

In this paper, we propose a graphical data model for specifying spatio-temporal semantics of video data. The proposed model segments a video clip into subsegments consisting of objects. Each object is detected and recognized, and the relevant information of each object is recorded. The motions of objects are modeled through their relative spatial relationships as time evolves. Based on the semantics provided by this model, a user can create his/her own object-oriented view of the video database. Using the propositional logic, we describe a methodology for specifying conceptual queries involving spatio-temporal semantics and expressing views for retrieving various video clips. Alternatively, a user can sketch the query, by examplifying the concept. The proposed methodology can be used to specify spatio-temporal concepts at various levels of information granularity. [1]

1 Introduction

The key characteristic of video data is the spatial/temporal semantics associated with it, making video data quite different from other type of data such as text, voice and images. A user of video database can generate queries containing both temporal and spatial concepts. However, considerable semantic heterogeneity may exist among users of such data due to difference in their pre-conceived interpretation or intended use of the information given in a video clip. Semantic heterogeneity has been a difficult problem for conventional database [3], and even today this problem is not clearly understood. Consequently, providing a comprehensive interpretation of video data is a much more complex problem.

[1] This research was supported in part by the National Science Foundation under grant number 9418 767-EEC

Most of the existing video database systems either employ image processing techniques for indexing of video data or use traditional database approaches based on keywords or annotated textual descriptors.

For indexing, keywords and textual descriptions for video data have also been suggested in [6], based on a generalization hierarchy in object-oriented realm. Video segments can be joined or concatenated based on the semantics of this hierarchy. However, this approach is very tedious since the perception of video contents is done manually by users, not through an automatic image processing mechanism. A video system that automatically parses video data into scenes using a color histogram comparison routine have been proposed in [5]. To locate frames containing desired objects, a method based on comparing the color histogram maps of objects is used. In [4], a hierarchical video stream model is proposed, that uses a template or histogram matching technique to identify scene change in a video segment. A video segment is thus divided into several subsegments. In each subsegment, a frame-based model is used to index the beginning frame of a subsegment from which objects are identified. In this system a video stream is parsed, and the information is stored in the database. However, this system has many limitations. First, the textual descriptions are manually associated with video segments. Second, the system provides parsing only for a specific type of video.

In order to address the issues related to user-independent view and semantic heterogeneity, we propose a semantically unbiased abstraction for video data using a directed graph model. The model allows to represent spatio-temporal aspects of information associated with objects (persons, buildings, vehicles, etc.) present in video data. However, such an automated video data-base system requires an effective and robust recognition of objects present in the video database. Due to the diverse nature of the video data, we can use various techniques currently available ac-

1063-6382/95 $4.00 © 1995 IEEE

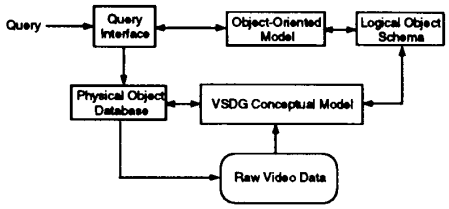

Figure 1: System abstraction

cording to the requirements of different situations that may occur in the input. For each input video clip, using a database of known objects, we first identify the corresponding objects, their sizes and locations, their relative positions and movements, and then encode this information in the proposed graphical model. The encoded video data may be semantically highly rich. Therefore a unified framework is needed for the users to express and for the system to process semantically heterogeneous queries on the unbiased encoded data. For this purpose, we propose an object-oriented approach that provides an elegant paradigm for representing user's view of the data. It is a hierarchical scheme that provides the necessary framework for a user to compose the views of the data with the maximum flexibility and at the same time allows processing of heterogeneous queries by evaluating the proposed graphical abstraction. For this purpose we also define an interface between these modeling paradigms. Another reason for this modeling approach is to provide an efficient indexing mechanism for on-line query processing without performing computations on the raw video data since such computation can be quite extensive. The proposed VSDG can be generated off-line and subsequently can be used to process user's queries on-line. The architecture of the proposed system is shown in Figure 1.

The organization of this paper is as follows. Section 2 describes the assumptions, techniques and the proposed graphical model. Two examples of video clips are used to illustrate the model. In Section 3, an object-oriented approach is presented for users to specify the perceived view of video data. The use of predicate logic for specifying spatio-temporal concepts and query types are also presented there. Section 4 describes the computer vision/image processing requirements of the proposed model. The paper is concluded in Section 5.

2 Graph-Based Conceptual Modeling

Generally, most of the wordly knowledge can be expressed by describing the interplay among physical objects in the course of time and their relationship in space. Physical objects may include persons, buildings, vehicles, etc. A video database, is a typical replica of this wordly environment. In conceptual modeling of video data, it is therefore important that we identify physical objects and their relationship in time and space. Subsequently, we can represent these relations in a suitable structure that is useful for users to manipulate. Temporal relations among objects has been previously modeled by using Petri-Net [1], temporal-interval [2], languages, etc. For spatial relations, most of the techniques are based on projecting objects on a two or three dimensional coordinate system. Very little attempt has been made to formally express spatio-temporal interactions of objects in a single framework. In this section, we propose a graphical model to capture both spatial and temporal semantics of video data. The proposed model is referred as Video Semantic Directed Graph (VSDG) model. The most important features of the VSDG model is an unbiased representation of the information while providing a reference framework for constructing semantically heterogeneous user's view of the video data. Using this model along with the object-oriented hierarchy (discussed in Section 3) the system shown in Figure 1 can be used for on-line query processing, with performing any computation on the actual raw video data. In the following sections, modeling of space and time information associated with objects is described in detail.

2.1 Spatio-Temporal Modeling over a Sequence of Frames (a Clip)

The spatial attribute, of a salient physical object present in the frames can be extracted in form of bounding volume, Z, that describes the spatial projection of an object in three dimensions. It is a function of bounding rectangular (Y), centroid, and depth information related to the object. The bounding rectangular is computed with reference to a coordinate system with an origin at the lower left corner of each frame. Both Z, and Y are expressed as :

$$Bounding\ Rectangular\ (Y) = (width, height, x, y)$$

$$Bounding\ Volume\ (Z) =$$

$$(Bounding\ Rectangular, centroid, depth)$$

Temporal modeling of a video clip is crucial for users to construct views or to describe episode/events in the clip. Episodes can be expressed by collectively

interpreting the behavior of physical objects. The behavior can be described by observing the total duration an object appears in a given video clip and its relative movement over all the frames. For example, occurrence of a slam-dunk in a sport video clip can be an episode in a users specified query. The processing of this query requires evaluation of both spatial and temporal aspects of various objects.

Temporal information of objects can be captured by specifying the changes in the spatial parameters associated with the bounding volume (Z) of objects over the sequence of frames. At the finest level, these changes can be recorded at each frame. Although, this fine-grained temporal specification may be desirable for frame based indexing of video data, it may not be required in most of the applications. The overhead associated with such detailed specification may be formidable. Alternatively, a coarse-grained temporal specification can be maintained by only analyzing frames at δ distance apart [7]. This skip distance (δ) may depend upon the complexity of episodes. δ is an interger with frame as its unit. There is an obvious tradeoff between the amount of storage needed for temporal specification and the detail of information maintained by the VSDG model.

2.2 The Proposed Model

Formally, both the spatial and temporal specifications of a clip can be represented as a directed graph, as shown in Figure 2, that consists of n video segments, labeled V_1, V_2, ..., V_n. In this model, time spans of physical objects are represented as circular nodes in the graph. Salient objects within a segment are grouped between two rectangular nodes, whereas such a node marks the appearance of a new physical object. Within a segment there can be an arbitrary number of objects. A video clip may consist of several such segments. An object may appear in any number of segments. In order to capture temporal semantics of object via VSDG, a motion vector can be associated with each object, represented by a circular node. The formal definition of VSDG is following.

A VSDG is a directed graph with a set of circular nodes (P), a set of rectangular nodes (T), and a set of arcs (A) connecting circular nodes to rectangular nodes. A circular node has an attribute describing the duration for which an object (person, vehicle, etc.) appears in a video segment. A rectangular node corresponds to an event in a video clip whenever a new physical object appears. In other words, a rectangular node marks the start of a new segment, that differs from its predecessor segment in terms of appearance of

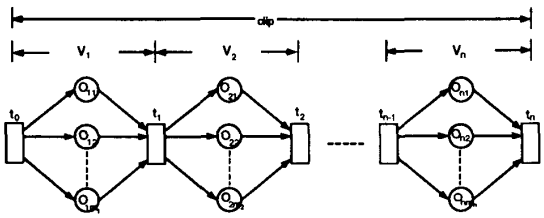

Figure 2: VSDG representation of a clip

a new physical object. Each circular node has exactly one incoming arc and one outgoing arc. The duration of a video segment V_i is max(duration(O_{i1}), ..., duration(O_{im_i})), where m_i is the number of salient objects O_{ij} ($1 \leq j \leq m_i$) appearing in segment V_i. Each circular node has the following attributes:

- $D{:}P \rightarrow I$,

is a mapping from the set of circular nodes to the set of durations in terms of frames.

- $W{:}P \rightarrow (Z_1, \ldots, Z_{\lfloor \frac{l}{\delta} \rfloor})$,

is a mapping from the set of circular nodes to the set of motion vetors. The element Z_i ($\forall i$ ($1 \leq i \leq \lfloor \frac{l}{\delta} \rfloor$)) of this vector is the *Bounding Volume* at *i-th* sampled frame. In other words, Z_i = (*Bounding Box*$_i$, *depth*$_i$, *centroid*$_i$), and *Bounding Box*$_i$ = (*width*$_i$, *height*$_i$, x_i, y_i), where l represents the number of frames associated with object O_i in a given video segment and δ is the time granularity for tracking motion of every object in a video segment.

From the motion vectors, we can derive the relative spatial relationship between any two objects for any sampled frame. The relative movements between two objects can be described as a set of mutual spatial relationships, as in [8]. For any sampled frame, the relative position between objects O_i and O_j can be captured by the spatial relationship between their projections on each coordinate axis, x, y, and z. In other words, for O_i, $R_{ij} = (SR_{ij_x}, SR_{ij_y}, SR_{ij_z})$ represents the spatial relationship of projections of O_j with respect to O_i on each axis. If there are k concurrent objects, between two rectangular nodes, then k - 1 such vectors are generated for each O_i. Therefore, in a given segment of a VSDG, if an object O_j appears concurrently with O_i for l frames, then O_i has a vector $M_{ij} = (R_{ij_1}, \ldots, R_{ij_{\lfloor \frac{l}{\delta} \rfloor}})$. Similarly, O_j has a vector $M_{ji} = (R_{ji_1}, \ldots, R_{ji_{\lfloor \frac{l}{\delta} \rfloor}})$. To determine the speed of an object, we take the difference $Z_j - Z_0$ and divide it by $\lfloor \frac{l}{\delta} \rfloor \delta$. For O_i, vectors R and M are derived information and are additional attributes associated with circular node representing O_i.

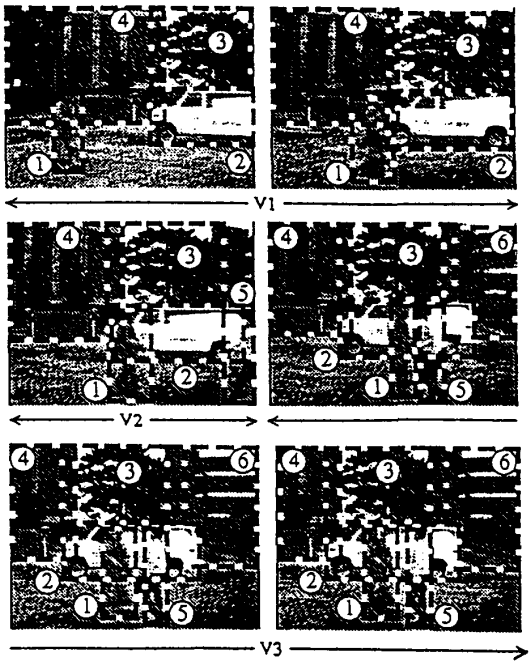

Figure 3: Snapshots of two example video clips

2.3 An Example of VSDG-Based Modeling

In this section, we use a video clip shown in Figure 3 to illustrate the proposed model. In the example video clip (Figure 3(a)), a car (object 2) and a person (object 1) appear first, then the camera moves toward the right and two persons (object 1 and object 5) are walking toward each other and shake hands. Assuming that proper object recognition methods are used to identify these objects, we can appropriately define the bounding volumes information for the objects. The complete VSDG model, for the example video clip is given in Figure 4, which describes the information about various objects and their temporal behaviors. The VSDG in Figure 4, has four rectangular nodes which correspond to three different scene changes. The first rectangular node (t_0) marks the start of video clip, t_1 indicates the appearance of objects O_5, t_2 indicates the appearance of object O_6, and t_3 indicates the end of the video clip. There are a total of six objects, O_1, O_2, O_3, O_4, O_5, O_6, and some objects appear in multiple scenes. For example, O_1, O_2, O_3, and O_4 appear in video segments V_1 and V_2.

3 A VSDG-Based Object-Oriented Model

As mentioned earlier, the objective of the system, shown in Figure 1, is to process video queries on-line. We, therefore, need to represent the rich semantics of video data using a suitable data model. In this section, we propose an object-oriented model which can be interfaced with the VSDG model. The objective of proposing an object-oriented paradigm is two folds. First, this can be used by the users to define their conceptual queries involving motion in a systematic way. Second, it can allow processing of user's conceptual queries by evaluating VSDG. Therefore, the system processes users' queries with the assistance of the object-oriented views. In other words, an object-oriented view serves as a "knowledge base" for the system. In Section 3.1, we describe how a user can construct views about the video data consisting of various objects.

3.1 An Object-Oriented Based User's Defined View

As we have mentioned video data is represented by three entities, spatial, temporal and physical objects. For user to query video data, we propose an object-oriented approach which provides an elegant paradigm for user's view representation. It is a hierarchical schema that allows a user to compose their views about the video data. The objective is to offer the maximum flexibility to a user to represent their semantics and at the same time allow processing of heterogeneous query by executing the proposed VSDG model. Generally, in any worldly knowledge three entities of interest are: spatial, temporal, and physical. The spatial entity called Conceptual Spatial Object (CSO) is the spatial concept associated with an object which can be extracted at the frame level. For example, *Presiding* a meeting attaches a meaning to some spatial area. For this concept, a person in a frame may be identified such that he/she is either standing or sitting on a chair in the center of a meeting room. Another example of a spatial concept is 'sitting'. A person may be sitting on a chair or some physical object. In this case, we have a conceptual spatial object 'sitting' with attributes 'a physical object which can sit' and 'a physical object being sit on', and they are related by the 'sitting' relationship. Conceptual Temporal Object (CTO) defines the concept that extends over a sequence of frames. It may involve several CSOs or CTOs (may be combination of both). There can be several temporal relationships

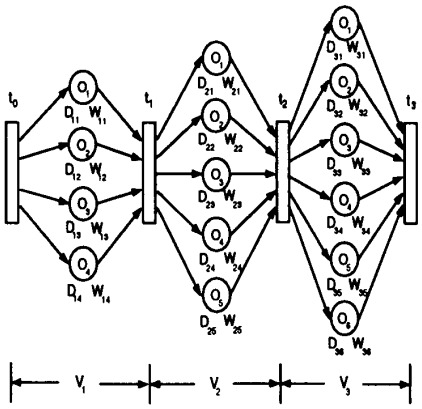

Figure 4: VSDG representation of the example clip

among physical objects (described below). Examples are 'slamdunk', 'walking', etc. A formal definition of CTO is given in Section 3.2. Lastly, Physical Objects (PO) are the physical objects described above which correspond to places in VSDG. Some examples are persons, tree, houses, etc.

Both CSO and CTO can be called logical objects. In general, any concept or semantic view of a user can be expressed using a set of rules from predicate logic that operate upon CSOs, CTOs, and POs. The foundation of these rules is given in Section 3.2. The objects and rules together can describe complex behavior involving multiple objects.

For video data, a user can use combination of various abstractions to construct his/her view of the video data. The important feature of this hierarchy, and in general for any object-oriented abstractions, is that each terminal node is either a CTO, a CSO, or a PO. Any complex video query is expressed as a function of these terminal nodes and processing of such query requires execution of some CTO and CSO over the specified PO's. As an example, consider a sports video database which can be used by multiple users with widely different interests. Figure 5 describes an object hierarchy of view/knowledge which a user would like to construct.

A fan may view the video data as the collection of players, event, and teams. Furthermore, in his/her view, there are three types of players, forward, guard, and center. There are two types of event, individual_event and team_event. Teams consist of those from NBA and NCAA. A sports fan can generate a query such as 'Give the video clips where

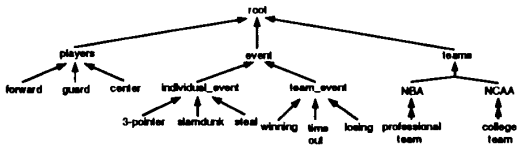

Figure 5: Fan's view

Michael Jordan (i.e., a PO) has a slam-dunk (CSO + CTO)'. The system first finds the image of Michael Jordan from the *picture* attribute of *player* class, and selects a set of video clips, then the *slamdunk* CTO is involved with its inputs supplied by the places in VSDG of selected video clips. On the other hand, a coach would like to identify those video segments where *pass* (CTO) occurs around the right sideline of the front court. The system invokes methods associated with a CTO called *pass* and with a CSO called *right sideline* and uses them to evaluate VSDG.

The definitions of some of the classes used in this example are given in Table 2. The entries are self-explanatory. The methods are coded based on predicate logic and describe the spatio-temporal processing related to the event of that object. In class NBA, "SetOf" is used to specify the association abstraction. The method "slamdunk" will be defined later.

3.2 Predicate Logic for Spatio-Temporal Semantics

In this section we describe a methodology to express concepts to express queries for video data using CSO and CTO and rules based on predicate logic.

3.2.1 Spatial Predicate

Here we define two sets of spatial predicates. The first set specifies predicates involving absolute measures in video data, while the second set represents spatial predicates based on relative positions among objects in video data. These sets are defined on two arbitrary physical objects "a" and "b". Similar relations are described in [11, 9, 10].

Set I :

$D(t, a, b)$: Distance between a and b at time t (may not need t)

$\equiv \sqrt{(x_{at} - x_{bt})^2 + (y_{at} - y_{bt})^2 + (z_{at} - z_{bt})^2}$

$DP(t_1, t_2, a)$: Displacement of a between t_1 and t_2

$\equiv \sqrt{(x_{at1} - x_{at2})^2 + (y_{at1} - y_{at2})^2 + (z_{at1} - z_{at2})^2}$

$RC(t, a, b)$: The relative coordinate of b to a at time t

$\equiv (x_{bt} - x_{at}, y_{bt} - y_{at}, z_{bt} - z_{at})$

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.