Modeling and Querying Video Data: A Hybrid Approach*

Cyril Decleir
LISI-INSA Lyon, Bâtiment 501
F-69621 Villeurbanne Cedex
cdecleir@lisi.insa-lyon.fr

Mohand-Saïd Hacid
LuFG Theoretical Computer Science
RWTH Aachen,
Ahornstraße 55, 52074 Aachen, Germany
hacid@cantor.informatik.rwth-aachen.de

Jacques Kouloumdjian
LISI-INSA Lyon, Bâtiment 501
F-69621 Villeurbanne Cedex
koulou@lisi.insa-lyon.fr

Abstract

This paper develops a video data model and a rule-based query language for video retrieval. A video sequence is split into a set of fragments. Each fragment can be analyzed to extract the information of interest that can be put into a database. This database can then be searched to find information of interest. Two types of information are considered: (1) the entities (objects) of interest in a video sequence, (2) generalized strata (a set of fragments), which contain these entities. To represent these information, our data model allows facts as well as objects and constraints. We present a declarative, rule-based, constraint query language that can be used to infer relationships about information represented in the model. The language has a clear declarative and operational semantics.

1. Introduction

Recent progress in compression technology has made it possible for computer to store efficiently pictures, audio and even video. Nevertheless, if such media are widely used in today's communication, efficient computer exploitation is still lacking. Many databases should be created to face the increasing development of advanced applications, such as video on demand, video/visual/multimedia databases, monitoring, virtual reality. that need to integrate these media. As a consequence, there is a crucial need for techniques allowing an easy development of such applications. In this paper, we focus on video exploitation, which is one of today's most complex and required media, whose computer handling possibilities have been growing quickly since the availability of efficient compression techniques such as MPEG or QuickTime. Video analysis and content retrieval based on semantics require multi-disciplinary research effort in areas such as computer vision, image processing, data compression, databases, information systems (see [12, 5]). Facilities should be available for users to view video material in a non-sequential manner and to build new sequences from others. To facilitate retrieval, all useful semantic objects and their features appearing in the video must be indexed. The use of keywords or free text [6, 14] to describe video content is not sufficient [4]. Additional techniques are needed. As stated in [4], the issues that need to be addressed are: (1) the representation of video information in a form that facilitates retrieval and interaction, (2) the organization of this information for efficient manipulation, and (3) the user-friendly presentation of the retrieved video sequences. The time-dependent nature of video is of considerable importance in developing adequate data models and query languages.

From database point of view, video data presents an interesting challenge in the development of data models and query languages. For example, the data model should be expressive enough to capture several characteristics inherent to video data, such as movements, shapes, variations, etc.

Despite the consensus of the central role video databases will play in the future, there is little research work on finding a common terminology and semantic foundations for representing and querying video information. This paper is a contribution in this direction. The framework presented here integrates formalisms developed in video, constraint,

*This work is partially supported by France Télecom (through CNET/CCETT) research contract no. 96 ME 17.

and object databases. The paper builds on the works of [1, 9, 7, 10, 11] to propose a data model for video databases and a declarative, rule-based, constraint query language, that has a clear declarative and operational semantics. We make the following contributions:

1. We develop a simple video data model on the basis of relation, object and constraint paradigms. Users can freely describe and retrieve video sequence according to their viewpoints.

2. We propose a declarative, rule-based, constraint query language that can be used to infer relationships from information represented in the model, and to intentionally specify relationships among objects. It allows a high level specification of video data manipulations.

The model and the query language use the point-based approach to represent periods of time associated with video sequences. First-order queries can then be conveniently asked in a much more declarative and natural way [13].

For lack of space, this extended abstract suppresses many of the details and all proofs.

Paper outline: This paper is organized as follows. Section 2 presents some useful definitions. Section 3 formally introduces the video data model. Section 4 describes the underlying query language. Section 5 draws conclusions.

2. Basic Definitions

This section provides the preliminary concepts that will be used to design the video data model and the underlying rule-based, constraint query language.

Definition 1 (Dense Linear Order Inequality Constraints) *Dense order inequality constraints are all formulas of the form* $x\theta y$ *and* $x\theta c$*, where* x*,* y *are variables,* c *is a constant, and* θ *is one of* $=, <, \leq$ *(or their negation* $\neq, \geq, >$*). We assume that these constants are interpreted over a countably infinite set* $\mathcal{D}$ *with a binary relation which is a dense order. Constants,* $=$*,* $\leq$*, and* $<$ *are interpreted respectively as elements, equality, the dense order, and the irreflexive dense order of* $\mathcal{D}$*.*

Complex constraints are built from primitive (atomic) constraints by using logical connectives. We use the special symbol, $\Rightarrow$, to denote the entailment between constraints, that is, if c_1 and c_2 are two constraints, we write $c_1 \Rightarrow c_2$ for c_1 entails c_2. $c_1 \Rightarrow c_2$ if and only if $c_1 \wedge \neg c_2$ is unsatisfiable.

Definition 2 (Set-Order Constraints) *Let* $\mathcal{D}$ *be a domain. A set-order constraint is one of the following types:*

$$c \in \widetilde{X}, \widetilde{X} \subseteq s, s \subseteq \widetilde{X}, \widetilde{X} \subseteq \widetilde{Y}$$

where c *is a constant of type* $\mathcal{D}$*,* s *is a set of constants of type* $\mathcal{D}$*, and* $\widetilde{X}$*,* $\widetilde{Y}$ *denote set variables that range over finite sets of elements of type* $\mathcal{D}$*.*

Our set-order constraints are a restricted form of set constraints [3], involving $\in$, $\subseteq$, and $\supseteq$, but no set functions such as $\cup$ and $\cap$.

Definition 3 (Time Intervals) *An interval is considered as an ordered pair of real numbers* (x_1, x_2)*,* $x_1 \leq x_2$*. This definition refers to the predicate* $\leq$*. If* t *is a time variable, then an interval* (x_1, x_2) *can be represented by the conjunction of the two primitive dense linear order inequality constraints* $x_1 \leq t$ *and* $t \leq x_2$*.*

Definition 4 (Generalized Strata) *A generalized strata is a set of pairwise non overlapping intervals. Formally, a generalized strata can be represented as a disjunction of time intervals.*

3. Video Data Model

- *Objects and object identity :* Objects are entities of interest in a video sequence. In our model, we refer to objects via their logical object identities, which are nothing but syntactic terms in the query language. We have essentially two types of objects: (1) generalized strata objects, which are abstract objects resulting from splitting a given video sequence into a set of smaller sequences; (2) semantic objects which are entities of interest in a given video sequence.

- *Attributes :* Objects are described via attributes. If an attribute is defined for a given object, then it also has a value for that object.

- *Relations :* It has been argued many times that objects do not always model real world in the most natural way, and there are situations when the use of relations combined with objects leads to a more natural representation. Although relations can be encoded as objects, this is not the most natural way of handling relations and so we prefer to have relations as first-class language constructs.

We assume the existence of the following countably infinite and pairwise disjoint sets of atomic elements:

- relation names $\mathcal{R} = \{R_1, R_2, \ldots\}$;
- attributes $\mathcal{A} = \{A_1, A_2, \ldots\}$;
- (atomic) constants $\mathcal{D} = \{d_1, d_2, \ldots\}$;
- object identities or oid's $\mathcal{O} = \{id_1, id_2, \ldots\}$.

Furthermore, in order to be able to associate a time interval to a generalized strata object, we allow a restricted form of dense linear order inequality constraints to be values of attributes. We define the set $\tilde{C}$ whose elements are:

- Primitive (atomic) constraints of the form $t\theta c$ where t is a variable, c is a constant, and θ is one of $<, =, >$;
- conjunctions, and disjunctions of primitive constraints.

Definition 5 (Value) *The set of values is the smallest set containing $\mathcal{D} \cup \mathcal{O} \cup \tilde{C}$ and such that, if $v_1, \ldots, v_n$ $(n \geq 1)$ are values, then so are $\{v_1, \ldots, v_n\}$ and $[A1 : v_1, \ldots, A_n : v_n]$.*

Definition 6 (Video Object) *A video object (denoted* v-object*) consists of a pair (oid, v) where:*

- *oid is an object identifier which is an element of $\mathcal{O}$;*
- *v is an m-tuple $[A_1 : v_1, \ldots, A_m : v_m]$, where A_i ($i \in [1, m]$) are distinct attribute names in $\mathcal{A}$ and v_i ($i \in [1, m]$) are values.*

If $o = (oid, v)$ with $v = [A_1 : v_1, \ldots, A_n : v_n]$, then $attr(o)$ denotes the set of all attributes in v (i.e. $\{A_1, \ldots, A_n\}$), and $value(o)$ denotes the value v, that is, $v = value(o)$. A value v_i is denoted by $o.A_i$.

3.1. Example

Let us see how the example given in [1] can be modeled in our framework. First let us recall the example: It concerns the movie: *The Rope* by Alfred Hitchcock. In the movie, two friends, Philip and Brandon decide to commit the perfect crime. They want to prove they are of the privileged group of people who are allowed to kill just for sake of killing and not receiving any punishment for it. Hence, they kill their friend David and hide him inside a chest in the living room. To sign their masterpiece, they give a party where they invite friends of David (David's girlfriend Janet, Janet's old boyfriend Kenneth), and his parents (David's father Mr. Kentley, David's aunt Mrs. Atwater). These individuals would talk about David, not suspecting that David's body is in the same room they are standing. In addition to these people, they invite, as a challenge, their old mentor Rupert Cadell who is known to be very intelligent and suspicious. Rupert will prove worthy of his reputation and he will immediately understand the extraordinary circumstances. As the movie progresses, Rupert will keep asking questions and gather clues to find out what is wrong.

Let us consider, for example, two generalized strata:

1. The first (gi_1) corresponds to the period of time in the sequence where the crime is committed. This interval contains four objects of interest: Philip, Brandon, David, and the Chest. The three objects Philip, Brandon, and David have an attribute, called role. Role-filler for Philip and Brandon is "murderer", and role-filler for David is "victim".
2. The second (gi_2) corresponds to the period of time in the sequence where the party is given. This interval contains objects Philip, Brandon, David, Janet, Kenneth, Kentley, Atwater, Rupert Cadell, and Chest.

Attributes are associated with these two generalized strata. The *subject* attribute gives a brief description about the main activity occurring in the generalized strata. The attribute *entities* is intended to give semantic objects of interest in the generalized strata. Another common attribute is *duration*, which define the strata's temporal boundaries using a constraint. Additional attributes can attached to generalized strata.

The following is a simple database extract indexing, in part, by content the two generalized strata gi_1 and gi_2.

$gi_1 = (id_1, [entities : \{o_1, o_2, o_3, o_4\}, duration : (t > a_1 \wedge t < b_1), subject : \text{"murder"}, victim : o_1, murderer : \{o_2, o_3\}])$
$gi_2 = (id_2, [entities : \{o_1, o_2, o_3, o_4, o_5, o_6, o_7, o_8, o_9\}, duration : (t > a_2 \wedge t < b_2), subject : \text{"Giving a party"}, host : \{o_2, o_3\}, guest : \{o_5, o_6, o_7, o_8, o_9\}])$

$o_1 = (id_3, [name : \text{"David"}, role : \text{"Victim"}])$
$o_2 = (id_4, [name : \text{"Philip"}, realname : \text{"Farley Granger"}, role : \text{"Murderer"}])$
$o_3 = (id_5, [name : \text{"Brandon"}, realname : \text{"John Dall"}, role : \text{"Murderer"}])$
$o_4 = (id_6, [])$
$o_5 = (id_7, [name : \text{"Janet"}, realname : \text{"Joan Chandler"}])$
$o_6 = (id_8, [name : \text{"Kenneth"}, realname : \text{"Douglas Dick"}])$
$o_7 = (id_9, [name : \text{"Mr.Kentley"}, realname : \text{"Cedric Hardwicke"}])$
$o_8 = (id_{10}, [name : \text{"Mrs.Atwater"}, realname : \text{"Constance Collier"}])$
$o_9 = (id_{11}, [name : \text{"Rupert Cadell"}, realname : \text{"James Stewart"}])$

$in(o_1, o_4, gi_1)$
$in(o_1, o_4, gi_2)$
...

The first statement says that the generalized strata gi_1 has a duration given by the interval $[a_1, b_1]$. The entities of interest in this fragment of sequence are o_1, o_2, o_3, o_4. It also says that this fragment of a sequence deals with the murder (the value of the attribute *subject*), where the object o_1 (David) is the victim, the objects o_2 (Philip) and o_3 (Brandon) are the murderers.
The last two statements are facts that define a relationship between the objects o_1 (David) and o_4 (Chest) within the generalized strata gi_1 and gi_2.
Note that in the first two statements, t is a temporal variable, and a_1, a_2, b_1 and b_2 are positive integers. A generalized strata does not necessarily correspond to a single continuous fragment. This is because a meaningful scene does not always correspond to a single continuous sequence of fragments. In this case, the value describing the period of time associated with a generalized strata will be a disjunction of atomic constraints.

4. Rule-Based, Constraint Query Language

In this section, we present the declarative, rule-based query language that can be used to reason with facts and objects in our video data model. The language consists of two constraint languages on top of which relations can be defined by means of definite clauses.

This language has a model-theoretic and fix-point semantics.

4.1. Syntax

We fix a countable set $\mathcal{D}$ of constants (i.e., atomic values), called the domain, and a set $\mathcal{O}$ of object identifiers (oid's), which are disjoint.

Let $V_{\mathcal{D}}$ and $V_{\mathcal{O}}$ be disjoint countable sets of variables, used to denote constants from the domain $\mathcal{D}$ and object identifiers, respectively. The elements of $V_{\mathcal{D}}$ are called value-variables, and those in $V_{\mathcal{O}}$, oid-variables.

The terms of the language are:

- *value-terms*, which are of two forms: (i) the constants in $\mathcal{D}$, and (ii) the variables in $V_{\mathcal{D}}$.
- *oid-terms*, which are of two forms: (i) the oid's in $\mathcal{O}$, and (ii) the variables in $V_{\mathcal{O}}$.

Definition 7 (Predicate Symbol) *We define the following predicate symbols:*

- *each* $\mathsf{P} \in \mathcal{R}$ *with arity* n *is associated with a predicate symbol* P *of arity* n*,*
- *a special unary predicate symbol* AnyStrata*.*
- *a special unary predicate symbol* AnyObject*.*
- *we assume the presence of the built-in predicate "=".*

Definition 8 (Atom) *If* P *is an* n*-ary predicate symbol and* $t_1, \ldots, t_n$ *are terms, then* $\mathsf{P}(t_1, \ldots, t_n)$ *is an atom.*

Definition 9 (Rule) *A* rule *in our language has the form:*

$$r : H \leftarrow L_1, \ldots, L_n, c_1, \ldots, c_m$$

where H *is an atom,* $n, m \geq 0$*,* $L_1, \ldots, L_n$ *are (positive) literals, and* $c_1, \ldots, c_m$ *are constraints.*

Optionally, a rule can be named as above, using the prefix "r : ", where r is a constant symbol. We refer to H as the head of the rule and refer to $L_1, \ldots, L_n, c_1, \ldots, c_m$ as the body of the rule.

Definition 10 (Range-restricted Rule) *A rule* r *is said to be range-restricted if every variable in the rule occurs in a body literal. Thus, every variable occurring in the head occurs in a body literal.*

Definition 11 (Program) *A program is a collection of range-restricted rules.*

Definition 12 (Query) *A query is of the form* Q :?$q(\bar{s})$*, where* q *is referred to as the query predicate, and* $\bar{s}$ *is a tuple of constants and variables.*

Our language has a declarative model-theoretic and a fix-point semantics.

4.2. Examples

Let us give some simple examples of queries. In the following, uppercase letters stand for variables and lowercase letters stand for constants.

The query "list the objects appearing in a given sequence (generalized strata) g" can be expressed by the following rule:

$$q(O) \leftarrow AnyStrata(g), AnyObject(O), O \in g.entities$$

In this example, g is a constant and O is the output variable. Here, we suppose that for a given generalized strata, the set-valued attribute "entities" gives the set of semantic objects of interest in that generalized strata. This query involves an atomic (primitive) constraint. To compute the answer set to the query, we need to check the satisfiability of the constraint $O \in g.entities$ after O being instantiated.

The query "does the object o appear in the domain of a given temporal frame $[a, b]$" can be expressed as:

$$q(o) \leftarrow AnyStrata(G), AnyObject(o), o \in G.entities, \\ G.duration \Rightarrow (t > a \wedge t < b)$$

Where t is a temporal variable. This query involves one primitive constraint $o \in G.entities$, and a complex arithmetic constraint $G.duration \Rightarrow (t > a \wedge t < b)$. To compute the answer set to the query, we need to check satisfiability of these two constraints.

The query "list all generalized strata where the objects o_1 and o_2 appear together" can be expressed as:

$$q(G) \leftarrow AnyStrata(G), AnyObject(o_1), AnyObject(o_2), \\ \{o_1, o_2\} \subseteq G.entities$$

The query "list all pairs of objects, together with their corresponding generalized strata, such that the two objects are in the relation "Rel" within the generalized strata", can be expressed as:

$$q(O_1, O_2, G) \leftarrow AnyStrata(G), AnyObject(O_1), \\ AnyObject(O_2), O_1 \in G.entities,$$

$$O_2 \in G.entities, Rel(O_1, O_2, G)$$

The query ”find the generalized strata containing an object O whose value for the attribute A is val” can be expressed as:

$$q(G) \leftarrow AnyStrata(G), AnyObject(O), \\ O \in G.entities, O.A = val$$

5. Conclusion and Future Work

There is a growing interest in video databases. We believe that formal developments will help understanding related modeling and querying problems. This will lead to the design of intelligent systems for managing and exploiting video information.

In this paper, we have addressed the problem of developing a video data model and a formal, rule-based, constraint query language that allow the definition and the retrieval of video data by content. The primary motivation of this work was that objects and time intervals are relevant in video modeling and the absence of suitable supports for these structures in traditional data models and query languages represent a serious obstacle. Objects allow to maintain an object-centered view inherent to video content. Attributes and relations allow to capture relationships between entities. This multi-paradigm approach simplifies the indexing of video sequences. We have developed a declarative, rule-based, constraint query language to reason about the video content. It provides a much more declarative and natural way to express queries.

There are many interesting directions to pursue. An important direction of active research is to significantly extend the query language to allow dynamic creation of generalized strata. This is very useful for virtual editing [8] which is recognized to be important in some video applications.

Another important direction is to study the problem of sequence presentation. Most existing research systems use template-based approach [2] to provide the automatic sequencing capability. In this approach, a set of sequencing templates is predefined to confine the user's exploration to a certain sequencing order. The problem is that this approach is domain-dependent and relies on the availability of a suitable template for a particular query.

We are investigating these two important research directions.

References

[1] S. Adali, K. S. Candan, S.-S. Chen, K. Erol, and V. S. Subrahmanian. Advanced video information system: Data structures and query processing. *ACM-Springer Multimedia System*.

[2] T. G. Aguierre-Smith and N. C. Pincever. Parsing movies in context. In *Proceedings of USENIX (Summer 1991), USENIX Association, Berkeley, California*, pages 157–168, 1991.

[3] A. Aiken and E. L. Wimmers. Solving systems of set constraints (extended abstract). In I. C. S. Press, editor, *Proceedings of the 7th Annual IEEE Symposium on Logic in Computer Science, Santa Cruz, California*, pages 329–340, 1992.

[4] T.-S. Chua and L.-Q. Ruan. A video retrieval and sequencing system. *ACM TOIS*, 13(4):373–407, Oct. 1995.

[5] N. Dimitrova. The myth of semantic video retrieval. *ACM Computing Surveys*, 27(4):584–586, Dec. 1995.

[6] B. J. Haan, P. Kahn, V. A. Riley, J. H. Coombs, and N. K. Meyrowitz. Iris hypermedia services. *Communications of the ACM*, 35(1):35–51, Jan. 1991.

[7] R. Hjelsvold and R. Midtstraum. Modelling and querying video data. In *Proceedings of VLDB'94, Santiago, Chile*, 1994.

[8] W. E. Mackay and G. Davenport. Virtual video editing in interactive multimedia applications. *Communications of the ACM*, 32(7):802–810, July 1989.

[9] S. Marcus and V. S. Subrahmanian. Foundations of multimedia database systems. *Journal of the ACM*, 43(3):474–523, 1996.

[10] E. Oomoto and K. Tanaka. Ovid: Design and implementation of a video-object database system. *IEEE TKDE*, 5(4):629–643, Aug. 1993.

[11] D. Srivastava, R. Ramakrishnan, and P. Z. Revesz. Constraint objects. In *Proceeding of PPCP'94*, number 874 in LNCS, pages 218–228. Springer Verlag, 1994.

[12] A. TOIS. Special issue in video information systems. *ACM Transactions on Information Systems (TOIS)*, 13(4), Oct. 1995.

[13] D. Toman. Point vs. interval-based query languages for temporal databases. In *Proceedings of PODS'96, Montréal, Canada*, pages 58–67, June 1996.

[14] Y. Tonomura. Video handling based on structured information for hypermedia systems. In *Proceedings of the International Conference on Multimedia Information Systems'91, Singapore*, pages 333–344. McGraw Hill, New York, 1991.