A Generic Data Model for Video Content Based Retrieval

Cyril Decleir[1] Mohand-Saïd Hacid[2] Jacques Kouloumdjian[1]

(1) LISI-INSA, 20 av. A. Einstein
F-69621 Villeurbanne-France
{cdecleir,koulou}@lisi.insa-lyon.fr
Tel.: (+33) 04 72 43 85 88, Fax.: (+33) 04 72 43 80 99

(2) LuFg TCS, RWTH Aachen, Ahornstraße 55, 52074 Aachen, Germany
hacid@cantor.informatik.rwth-aachen.de
Tel.: (+49) 241 80 21131, Fax.: (+49) 241 88 88360

This work is partially supported by the French Department of Telecommunications (CNET)
Research action no. 96 ME-17 - Content Video Indexing and Querying - "Sésame Project"

Keywords: video indexing, video databases, query languages.

Abstract

This paper presents a generic data model and a rule-based query language for content based video access. The model allows user-defined attributes as well as explicit relations between object of interest. A declarative, rule-based query language is used to infer relationships about information represented in the model.

1 Video Indexing

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 because automated, high-level interpretation is nowadays impossible. While storing raw video streams in analogic or digital formats is possible, to archive efficiently video documents in order to enable efficient retrieval means being able to store and manage *video content* (e.g., raw features, annotations). Recently, this field has received a particular attention. Two approaches have been considered: (1) systems that use only automated feature extraction, like *Jacob* [4], and (2) more elaborated systems for concrete applications [3] [1] [5] that integrate user specifications in their video data model.

In this paper, we develop a generic data model to represent user-annotations as well as computer extracted features, and a rule-based query language. The model is based on the notion of objects of interest that can be annotated using attributes. Objects can be linked together by means of explicit relation names. We first present the model, then the corresponding query language.

2 Generic Video Data Model

We introduce in this section the concepts of the model.

First solutions to video indexing consisted in a simple summary of the video document content. In general, the temporal dimension of that media was ignored. To get a better grab on video content, the time dimension has to be taken into account. Historically, a first solution was brought by *segmentation*, where the time-line is partitioned and each element is individually annotated. Then *stratification* was introduced, which allowed a finer indexing. To take into account the temporal aspect of video, both methods use a technique that consists in linking one or several annotations to a temporal interval of the video document. In our model, we use an extension of stratification, in the form of *generalized interval*. A generalized interval is an union of single, non-overlapping time intervals. Doing this way, the same information that occurs in separate parts of a video can be handled in an uniform way.

While temporal management is a key point in the video indexing field, annotation structuring is another concern. Basic solutions, such as free text, are still widely used, but their exploitation is heavily supported. Several solutions have been developed to enhance the handling of annotations, such as the one which consists in creating a specific video description language [2], or the one which consists in annotation typing (*location*, *people* and *event*) [3]. In our model, we extend this last approach by annotating by means of user-defined attributes. Therefore, categories can be implemented upon user's need and video document specificities.
Finally, we introduce two useful entities to support annotation: *relations* and *objects*. Up to this point, we were describing some video content feature individually. But how interactions between features can be captured? For instance, if two persons, say A and B are shaking hands, how can we represent this fact efficiently? For this purpose we introduced *objects* and *relations*. Objects are entities used to label relevant features extracted from video content. In the previous example, we should introduce two objects: A and B. Then, to represent the fact that A and B are shaking hands, we simply introduce a relation *ShakingHands(A,B)*.
At this point, we can give a mathematical structure of the model. A video document is seen as a 7-tuple:

$$V = (\mathcal{I}, \mathcal{O}, \mho, \mathcal{R}, \Phi, \lambda_1, \lambda_2)$$

- $\mathcal{I}$: a set of generalized intervals. A video sequence is split into a set of arbitrary generalized intervals. Note that generalized intervals overlapping is possible. Each generalized interval is seen as an abstract object.
- $\mathcal{O}$: a set of entities (objects) describing information within a video sequence. According to application needs, only information which is of interest for a given exploitation is considered. These entities are determined by analyzing each generalized interval.
- $\mho$: a set of atomic values (e.g., integers, strings).
- $\mathcal{R}$: a set of relations. These relations link objects within generalized intervals.
- Φ : a set of constraints describing time intervals associated with generalized intervals.
- $\lambda_1 : \mathcal{I} \longrightarrow 2^{\mathcal{O}}$ maps each generalized interval to a subset of $\mathcal{O}$.
- $\lambda_2 : \mathcal{I} \longrightarrow \Phi$ maps each generalized interval to a set of interval constraints.

The *object* is the central entity of our model. Relationships among objects are represented by means of attributes and explicit relation names.

3 Rule-Based Query Language

We briefly introduce here, through examples, a rule-based query language that can be used to infer relationships about information represented in the model. The language is built using elements of the previously defined sets. The temporal aspect is captured through the use of conjunction of constraints which are written $t\theta c$ or $c\theta t$ where t is a temporal variable, c is a constant and $\theta \in \{<, \leq, =\}$.

Here are some examples of queries:
The query *"list the objects appearing in the domain of a given generalized interval g"* can be expressed as:

$$GeneralizedInterval(g), Object(O), O \in oset(g).$$

oset is a predefined function. It returns the set of objects associated with the generalized interval g.

The query *"list all sequences where the object o appears"* can be expressed as:

$$GeneralizedInterval(G), Object(o), o \in oset(G).$$

The query *"does the object o appear in the domain of a given temporal frame* $[a, b]$*"* can be expressed as (provided that t is a time variable):

$$\begin{aligned} &GeneralizedInterval(G), Object(o), o \in oset(G),\\ &\quad Interval(G, I), \{I \Rightarrow (a \leq t \wedge t \leq b)\}. \end{aligned}$$

Rules can also be used to infer new relationships: if we want to know all pairs of sequences with their common objects, we define the following rule:

$$\begin{aligned} &same - object - in(G_1, G_2, O) : -\\ &\quad GeneralizedInterval(G_1), GeneralizedInterval(G_2),\\ &\qquad Object(O), O \in oset(G_1), O \in oset(G_2). \end{aligned}$$

4 Conclusion

We have presented a video data model and a rule-based query language. Since the model is flexible, it can fit many applications needs. We are currently investigating a formal definition of the query language, together with the problem of sequence presentation.

References

[1] S. Adali, K.S. Candan, S. Chen, K. Erol, and V.S. Subrahmanian. Advanced video information system : Data structures and query processing. *Multimedia Systems*, 1996.

[2] M. Davis. Media streams: An iconic visual language for video annotation. In Ephraim P. Glinert and Kai A. Olsen, editors, *Proceedings of the 1993 IEEE Symposium on Visual Languages*, pages 196–203, Bergen, Norway, August 1993. IEEE Computer Society Press.

[3] R. Hjesvold and R. Midtstraum. Modelling and querying video data. In *20th VLDB Conference*, 1994.

[4] M. La Cascia and E. Ardizzone. Jacob : Just a content-based query system for video databases. In *ICASSP-96, Atlanta*, 1996.

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