Using Description Logics for Indexing Audiovisual Documents

Jean Carrive[1], François Pachet[2], Rémi Ronfard[1]

[1]Institut National de l'Audiovisuel (INA)
[2]SONY CSL-Paris & Lip6 (Paris 6)

{jean, remi}@ina.fr, pachet@csl.sony.fr

Abstract

We address the problem of indexing broadcast audiovisual documents (such as films, news). Starting from a collection of so-called shots, we aim at building automatically high level descriptions of subsets of this collection, that can be used for annotating, indexing and accessing the document. We propose to represent documents and high level descriptions with the framework of description logics, enriched with temporal relations. We first define the problem as a classification problem. We then propose an algorithm to automatically classify sub-sequences of shots, based on a bottom-up construction of descriptions using the rule mechanism of the CLASSIC system.

Introduction

This study takes place in the field of audiovisual documents indexing. By audiovisual documents, we mean essentially video or film programs. Indexing is understood here in a very general sense, as the operation which allows whole or part of a document to be the result of a request. In practice, that goes from simple methods such as associating a few keywords with the whole document to much more sophisticated ones, such as describing deeply a document, for example with conceptual graphs [Simonnot 1996].

1.1 Temporal documents

A specific characteristic of all audiovisual documents is their temporal dimension. This temporal dimension has two sides: the multi-layered aspect of documents, and their structural aspect.

1.1.1 Documents are multi-layered

The various information concerning audiovisual documents may be organized in a multi-layered structure. Each layer contains temporal information concerning a particular aspect of the document. The most basic layer is the *shot layer,* which is basically the segmentation of the audiovisual data into a set of discrete temporal objects. Shots are usually considered as the smallest syntactic units of film language [Katz 1991]. Shots may be defined as what is filmed during one run of the camera, without edit. However, a most interesting information for indexing and understanding documents is the transition between shots. A *cut* means a brutal transition between two shots: the last image of the first shot is immediately followed by the first image of the second one. It can therefore be represented as a temporal objet with no duration (an event). Gradual transitions, such as a *fade* (*in* or *out*), *dissolve*, etc, are represented as standard temporal objects, intertwined between two shot objects.

Other typical layers are: the dialog layer for representing dialogs between characters. Yet another layer may be used for representing appearances of characters on screen, and so forth (see figure 1).

The information contained in each layer is typically derived from analysis algorithms. It is important to notice that some extraction algorithms may be executed *a priori*, such as the detection of shot transitions [Yeo 1995]. Other algorithms need contextual information, such as face detection. In the first case, some algorithms may be too costly to be executed on the whole document. This is the case for example for text extraction, where the document as to be firstly segmented in time and space.

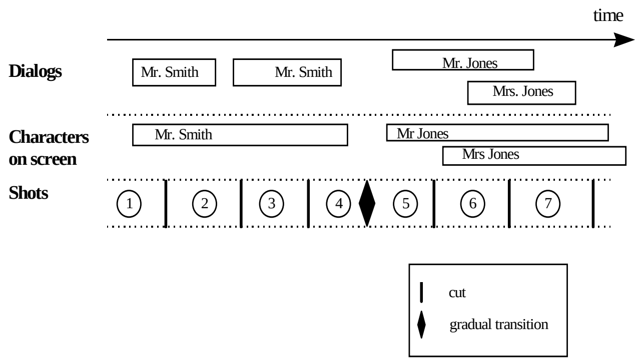

figure 1: example of multi-layered description

1.1.2 Structured documents

The second aspect of audiovisual documents is their hierarchical nature. In most cases, a document may be split into *successive* sequences which are in turn split into *shots* (see figure 2).

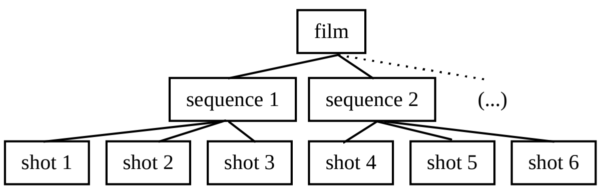

figure 2: hierarchical structure

Usually, TV researchers know more about documents, and can classify them into *document types*: for instance, the newscast of CNN at 8pm, specific sitcoms, variety shows, western movies, etc. Within one specific document type, documents share several characteristics, such as film sets, news readers, or the organization of shots or sequences over time. For example, the temporal structure of some particular news programs could be described *in general* as an alternation of *in sets* sequences and *report* sequences, where in sets sequences are composed for example of still shots (no camera motion) of the news reader (say Mr. Smith) separated by cuts, with the logo of the channel in the top right corner of the screen.

1.1.3 The taxonomy of film events

We claim that there exists a taxonomy for some elements of film, and that some (partial) formalization of this taxonomy may be given. A simplified taxonomy of traditional transition (punctuation) effects between two shots (from [Arijon 1976]) is shown figure 3.

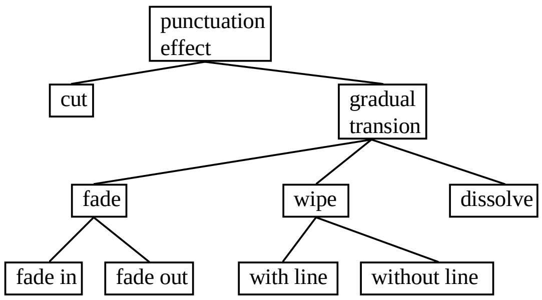

figure 3: taxonomy of punctuation effects

If we consider only subsets of the taxonomy where role fillers can be automatically extracted from the signal (as color histograms) it is important to notice that some parts of this taxonomy are entirely made up with *primitive* concepts, i.e. that no classification process can assign to an instance a position in the hierarchy. In the example above, an algorithm may detect a shot transition as a *gradual transition*, and then refine the description to *fade in*. However, concerning for example camera motions, an algorithm is unlikely to classify a camera motion as *pan*, and then refine the description to *pan right*.

1.1.4 Classification of temporal segments

The relative disposition of elements within a layer may convey some signification. For instance, in some contexts, a gradual transition between two shots may signify a transition between two sequences.

[Aigrain 1997] proposes general rules to group shots into sequences. One of these rules specifies that a gradual transition surrounded on each side by at least three cuts is likely to be a sequence transition. In the example of figure 1, according to this rule, there is a sequence limit between shots 4 and 5. This shot transition may be classified as « sequence limit ».

In order to describe meaningfully the document, elements of different layers are to be taken account. For example, in the example of the figure 1, a TV researcher might be interested by a shot where Mr. Smith is speaking during the whole shot, say, in sight of using this shot for a documentary about Mr. Smith. Shot number 3 meets these requirements. Thus, this shot may be classified under a concept « shot of Mr. Smith talking». It is important to note that these concepts may be considered as *specializations* of the basic concepts of the taxonomy of film events (section 1.1.3).

2. Using DL for analysis

Structural analysis is the process that yield the temporal structure of the document, from the initial audiovisual document and the various layers containing additional information on the document. This process is of course made easier when the document type is known *a priori* (which is most often the case), since the document type is associated with generic temporal structures, as seen in the preceding section. We claim that the structural analysis of an audiovisual document may be seen as a classification process. This involves 1) representing generic temporal structures for documents types (see section 1.1.2), and 2) devising an algorithm to aggregate primitive film events and classify them according to these generic temporal structures.

2.1 Description ogics and temporal classification

Description logics are knowledge representation languages; they allow to represent knowledge in a structured way by separating definitions of *concepts* (terminological representation system or *Tbox*) from description of *individuals* (assertional representation system or *Abox*). Concepts are sets of individuals and *roles* represent binary relations between individuals. Concepts and roles descriptions are organized in hierarchies with the *subsumption* relation [Nebel 1990].

The suitability of description logics for the representation of video was illustrated in [Ronfard 1997]. This paper explores the strengths and technical issues involved in using description logics for analyzing the video. Several description logics systems are available; the ideas proposed in this paper are implemented with the CLASSIC system [Borgida 1989].

Various works have been conducted to classify temporal structures, mainly in the field of plan recognition. [Devanbu 1996] and [Weida 1992] propose to extend the notion of subsumption to plans, while [Artale 1994] propose a formal language for reasoning about time and action. In section 2.3.4, we discuss how the problem of video indexing may be seen as a plan recognition problem.

2.2 Film events as concepts

It is natural to represent film events in our taxonomy as concepts in the sense of Description ogics. For instance, the concepts of a shot of the news reader in the example illustrated by the may represented by a CLASSIC concept as follows:

```
(define-concept 'READER-SHOT '(and
 SHOT
 (exactly 1 character-on-screen)
 (fills camera-motion still)
 (at-most 1 character-speaking)
 (all transition-to-next-shot CUT)
 (all transition-to-previous-shot CUT)
 ...))
```

As we can see, the CLASSIC concept represents only a part of the information : the temporal structure is not expressed. For instance, the concept definition above doesn't specify temporal relations between *character-on-*

screen and *character-speaking*. This is essentially due to the limitations of the description logics formalism. We propose to represent this structure using the rule-based inference mechanism of CLASSIC.

2.3 Grouping temporal units

In order to have some temporal segment classified, as a shot of the news reader, illustrated in section 1.1.4, one must first express this segment as a combination of some other segments. We express this combination as a grouping rule.

2.3.1 Structure expressed as grouping rules

The structure of the document is expressed as grouping rules which aggregate temporal forms of low level into temporal forms of higher levels. We have identified two main categories of grouping rules. In the first category, rules aggregate two instances of two distinct concepts into one instance of a concept of a higher level. In the second category, rules aggregate N instances of the same concept into one instance of a concept of a higher level.

In order to define these rules, we need to define the concept TEMPORAL, which represent temporal intervals. This concept is defined as follows :

```
(define-concept 'TEMPORAL '(and
  (exactly 1 begin)
  (all begin integer)
  (exactly 1 end)
  (all end integer)
  (< begin end)))
```

The general form of rules of the first category is:

$C1\ R\ C2 \rightarrow G$ *(1)*

Merge two segments

with:

C1, C2 temporal concepts (inheriting from TEMPORAL)

R : temporal relation

G : concept inheriting from TWO-TEMP-GRP, group of two temporal instances, defined by:

```
(define-concept 'TWO-TEMP-GRP '(and
 TEMPORAL
 (exactly 1 first-temporal)
 (all first-temporal TEMPORAL)
 (exactly 1 second-temporal)
 (all first-temporal TEMPORAL)))
```

The general form of rules of the second category is:

$G\ R\ C \rightarrow G'$ *(2)*

Merge segments into group

with:

C : temporal concept

R : temporal relation

G (and G') : concept inheriting from N-TEMP-GRP, group of several temporal instances, defined by:

```
(define-concept 'N-TEMP-GRP '(and
 TEMPORAL
 (at-least 2 element)
 (all element TEMPORAL)))
```

This last type of rules *(2)* try to aggregate an instance of the concept C into a pre-existing instance of G. This can be expressed by:

if *c* is an instance of C,
if *g* is an instance of G
if *c'* is an instance of C,
if *c'* is a role filler for the role `element` of *g*
if *c R c'*
if *c* is **not** a role filler for the role `element` of *g*
then *c* is added as a role filler for the role element of *g*

Some sub-categories have to be defined for each of the two main categories, in order to specify how to instantiate the resulting concept *G*. There are several ways to precise the role fillers of the resulting concept: that can be the common values of one very role of the premise concepts, the value of one particular role of one particular premise concept, the most specific generalization of values of one particular role, etc. Some particular role fillers are the values of the begin and end roles: these roles may be fills by calculating either the union of the temporal components ($\rightarrow_U$) or their intersection ($\rightarrow_I$).

2.3.2 The need for a temporal logic

The rules expressed above mention temporal constraints between temporal intervals: Mr. Smith talking *during* Mr. Smith on screen, for example. In order to represent these temporal constraints, we need a formalism to represent temporal relations. The choice of this formalism is important ; it must ensure a good compromise between expressiveness and tractability.

In our case, we propose to choose the temporal model presented by [11] – *Pointizable Interval Algebra* – which is based on the interval algebra of Allen [12]. In this model the consistency test is tractable, which is not the case in the full interval algebra of Allen. Disjunctions of Allen basic relations are here transformed into conjunctions of constraints on the bounds of these intervals. Only a subset of Allen interval algebra may be expressed in this way. For example, the temporal relation

A {*before* ∨ *meets* ∨ *overlaps*} B

is transformed to:

begin(A) < begin(B)
end(A) < end(B)

but the relation:

A {*before* ∨ *after*} B

has no equivalent.

2.3.3 Rule-based mechanism in Classic

We present here a first attempt to implement the grouping rules presented above using the rule-based mechanism offered by the CLASSIC system. We will first illustrate the grouping strategy given a simple example and then discuss the limitations of this implementation.

CLASSIC offers the possibility to associate a rule (roughly a Lisp expression) to a concept definition. Each time a new instance of this concept is instantiated, the rule is fired. In our case, we can associate a rule represented by: A R B → G to the concept A. When *a* instance of A is created, the behavior of the rule consists in searching all instances b_i of the concept B such that *bi* R *a*. *a* and b_i are then grouped together in a new instance of G.

For example, consider an emission where a *host* (for instance Mr. Johns in figure 1) interviews a *guest* (Mr. Smith). The interview sequence is made of shots of the host, shots of the guest and insert shots (for instance the hands of the guest, an element of the studio, etc.). Automatic tools gives a segmentation of the sequence into shots, and for each shots it gives the number of face regions (which in our case may be 0 or 1). Given a face region, automatic tools may say if corresponds to the host face, which is known by advance. Given those primitives, shots can be easily classified into tree disjoint classes: shots of the host (concept H), shots of the guest (concept G) and others shots (concept O).

Shot - reverse shots sequences are well-known cinema constructs where shots of two characters are shown alternatively. Making the assumption that in our case such sequences may reflect some interesting discussion between the host and the guest, we want to be able to extract shot - reverse shot sequences from the whole sequence.

In order to do that, given our rule formalism, we have first to group together the shots of the host which are directly followed by a shot of the guest. This is done by defining a rule that groups a shot of the host followed by a shot of the guest into an instance of H-G-SEQ:

$$H\ \{meets\}\ G \rightarrow_U H\text{-}G\text{-}SEQ$$

In a second step, we define a rule which groups together consecutive instances of H-G-SEQ into an instance of S-RS-SEQ, a concept which represents a shot - reverse shot sequence:

$$S\text{-}RS\text{-}SHOT\ \{meets\}\ H\text{-}G\text{-}SEQ \rightarrow_U S\text{-}RS\text{-}SEQ$$

The implementation of the grouping rules mechanism described above presents some obvious limitations. First, there may be an increasing cost of computational resources and, moreover, there is no *a priori* evidence that the system converges to a steady solution. However, this is a first attempt to implement our ideas and a second step will be to express them in a more formalized and more efficient way. On this subject, we are particularly interested in temporal extension of description logics concerning plan representation and plan recognition.

2.3.4 Video indexing as plan recognition ?

An important body of work has already been done concerning temporal extensions of description logics to represent and reason about plans. It is therefore important for us to determine to which extent the problem of video indexing may be expressed as a plan recognition problem.

The approaches which best suit our problem seem to be those adopted on the one hand by Weida and Litman in [Weida 1992] and on the other hand by Artale and Franconi in [Artale 1994]. The T-REX system [Weida 1992] integrates temporal constraint networks into a description logics formalism. A subsomption relation is defined for constraint networks, which allows to classify plans into a taxonomy. Plan recognition is done by a process which dynamically partitions the plan taxonomy into three modalities: *necessary*, *possible* and *impossible*. For example, we could represent a shot of the host followed by a shot of the guest (see section 2.3.3) with the following T-REX plan definition:

```
(defplan H-G-Seq
        ((step1 H)
         (step2 G))
        ((step1 (meets) step2)))
```

In an informal way, this expression refers to a plan composed of one instance of H (*step1*) and one instance of G (*step2*), *step1* and *step2* being constrained by the temporal relation *meets*.

[Artale 1994] propose a formal framework to represent temporal concepts (actions and plans) in a uniform way, which means that temporal operations are an integral part of the formalism, which was not the case in T-REX. The framework is provided with a well founded syntax, a formal semantics and a calculus. In this formalism, the concept of a shot of the guest followed by a shot of the host might be expressed by:

$$\begin{aligned}\mathrm{H\text{-}G\text{-}Seq} \doteq\ & \Diamond(x\ y)(x \mathrm{\ s\ } \#)(y \mathrm{\ f\ } \#)(y \mathrm{\ mi\ } x)\\ & ((*\mathrm{s1}:\mathrm{H})@x \wedge (*\mathrm{s2}:\mathrm{G})@y))\end{aligned}$$

In the expression above, the special variable # stands for the temporal interval at which the concept itself (in this case H-G-Seq) holds. The temporal existantial quantifier $\Diamond$ introduces temporal variables ; $*\mathrm{s1}$ and $*\mathrm{s2}$ are atomic parametric features: $*\mathrm{s1}$ is of type H and holds (represented by '@') during the x temporal interval. $(x \mathrm{\ s\ } \#)(y \mathrm{\ f\ } \#)(y \mathrm{\ mi\ } x)$ represent temporal constraints.

The temporal concept of two consecutive shots of different types is represented in both formalisms in a more natural way than our system currently can. However, some important concepts, for example a sequence of an undefined number of consecutive shots of the guest, can not be easily expressed in either formalisms. The CLASP system [Devanbu 1996], however more limited, would allow to express such "LOOP" constructs. We need to investigate how [Weida 1992] or [Artale 1994] can be extended in that direction in some more appropriate – and may be less expressive – formalism.

3. Conclusion

We have defined the problem of indexing audiovisual documents, and have shown that it involves classifying temporal structures using multi-layered information. The implementation with CLASSIC of the ideas expressed above was a first attempt to validate our approach on small examples. Some more formalized approaches are envisaged based on existing works in the literature.

Some open issues remain, concerning the specific problem of video indexing. First, a more convenient way to express a *sequence* as a succession of an undetermined number of temporal instances has to be found. Secondly, we need to investigate what minimal set of primitives should be provided by automatic extraction algorithms for the reasoning to be feasible.

The system is currently under development. Experimentation and evaluation is planned to take place at INA as part of the DiVAN[1] project in 1999.

Bibliography

[Aigrain 1997] Aigrain, P., Joly, P., Longueville, V. (1997), Medium Knowledge-Based Macro-Segmentation of Video into Sequences. *Intelligent Multimedia Information Retrieval*. A. P. M. Press.

[Arijon 1976] Arijon, D. (1976), *Grammar of film language*, Focal Press, London & Boston.

[Artale 1994] Artale, A., Franconi, E. (1994), *A Computational Account for Description Logic of Time and*

[1] . DiVAN is an Esprit project financed by the European Commission (Distributed Audiovisual Archives Network).

Action. Proc of the 4th International Conference on Principles in Knowledge Representation and Reasoning (KR94) pp: 3-14.

[Borgida 1989] Borgida, A., Brachman, R.J., McGuiness, D.L., Resnick, L.A. (1989), *CLASSIC: A Structural Data Model for Objects*. ACM SIGMOD Int. Conf. on Management of Data pp: 59-67.

[Devanbu 1996] Devanbu, P. T., Litman, D. (1996), "*Taxonomic Plan Reasoning.*" Artificial Intelligence **84** pp: 1-35.

[Katz 1991] Katz, S. D. (1991), *Film Directing Shot by Shot*, Michael Wiese Production.

[Nebel 1990] Nebel, B. (1990), "*Reasoning and Revision in Hybrid Representation Systems.*" LNAI **422** .

[Ronfard 1997] Ronfard, R. (1997), *Shot-level description and matching of video content*. SPIE 97, San Diego pp:

[Simonnot 1996] Simonnot, B. (1996), *Modélisation multi-agents d'un système de recherche d'information multimédia à forte composante vidéo*, Université Henri Poincaré - Nancy I pp**:** 259.

[Weida 1992] Weida, R., Litman, D. (1992), *Terminological Reasoning with Constraint Networks and an Application to Plan Recognition*. Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR'92), Cambridge, Massachussetts pp: 282-293.

[Yeo 1995] Yeo, B.-L., Liu, B. (1995), "*Rapid Scene Analysis on Compressed Video.*" IEEE Transactions on Circuits and Systems for Video Technology **5**(6) pp: 533-544.