Multimed Tools Appl (2006) 31: 309–325
DOI 10.1007/s11042-006-0042-2

Handling temporal heterogeneous data for content-based management of large video collections

Nicolas Moënne-Loccoz · Bruno Janvier · Stéphane Marchand-Maillet · Eric Bruno

Published online: 18 October 2006

Abstract Video document retrieval is now an active part of the domain of multimedia retrieval. However, unlike for other media, the management of a collection of video documents adds the problem of efficiently handling an overwhelming volume of temporal data. Challenges include balancing efficient content modeling and storage against fast access at various levels. In this paper, we detail the framework we have built to accommodate our developments in content-based multimedia retrieval. We show that not only our framework facilitates the development of processing and indexing algorithms but it also opens the way to several other possibilities such as rapid interface prototyping or retrieval algorithm benchmarking. Here, we discuss our developments in relation to wider contexts such as MPEG-7 and the TREC Video Track.

Keywords Video document · MPEG-7 · TREC video track · Content-based multimedia retrieval · ViCoDE

1 Motivations

Video data processing has for long been of high interest for the development of compression and efficient transmission algorithms. In parallel, the domain of content-based multimedia retrieval has developed, initially from text retrieval, then for images and now addressing video content retrieval. Whereas in text and image retrieval the volume of data and associated access techniques are well under control,

This work is funded by EU-FP6 IST-NoE SIMILAR (www.similar.cc) and the Swiss NCCR IM2 (Interactive Multimodal Information Management).

N. Moënne-Loccoz (✉) · B. Janvier · S. Marchand-Maillet · E. Bruno
Viper Group, Computer Vision and Multimedia Lab,
University of Geneva, Geneva, Switzerland
e-mail: Nicolas.Moenne-Loccoz@cui.unige.ch

this is largely not the case for video collection management. Not only video data volume may rapidly grow complex and huge but it also requires efficient access techniques associated to the temporal aspect of the data.

Efforts in video content modeling such as MPEG-7 [11] are providing a base for the solution to the problem of handling large amounts of multimedia data. While such a model is very well-suited to represent a single multimedia document, it cannot be used efficiently for accessing, querying and managing a large collection of such documents due to its inherent complexity. Unfortunately, most video retrieval systems presented in state of the art literature [1, 6] do not explicitly discuss the way they address such management issues.

In this paper, we detail the framework we have constructed for the management of video document collections in the context of our research in video content retrieval. Rather than presenting a temporal document model alone, our ultimate goal is to develop content characterization and indexing algorithms for the management of large video collections. When addressing such problems, one rapidly faces the need for a favorable context on which to base these developments and also that permits rapid and objective evaluation of research findings. From an extensible multimedia document model, we have built a database framework comprising all needed reference information to raw video documents. Efficient access to the original document is ensured by a generic accessor called OVAL that we have embedded within several prototyping platforms. This way, we are combining the benefits of a classical DBMS for rapid access to indexed description data with the efficient random access capabilities of our platform.

In Section 2, we are reviewing the model we propose for a multimedia document and associated description data. In Section 3, we detail how the data are produced, stored and efficiently accessed. Section 4 presents content-based video documents retrieval application relying on the proposed management framework. Throughout the paper, we briefly discuss the relation between our developments and common efforts with in particular the TRECVid [19] Retrieval Evaluation challenge.

2 Modeling temporal documents

The design of our framework is centered around the concept of temporal information. We consider that any part of our data store can be associated with a temporal stamp. The data itself may be located within either of the three layers depicted in Fig. 1. Namely, we follow a hierarchical scheme able to embed heterogeneous data such as an audio-visual (AV) stream (video) associated with meta-data and a set of key-frames (still pictures), themselves described by textual annotations. More formally, our scheme comprises:

- Document Information: global information about each document including meta-information and raw-data information. (Subsets of the *creation information*, *media information* and *usage information* of the MPEG-7 standard)
- Document structure: the temporal decomposition of video documents that comes from the temporal segments covered by the description data
- Document description: the set of description data that is either automatically extracted (*feature-based*) or entered manually by human operators (*semantic annotation*).

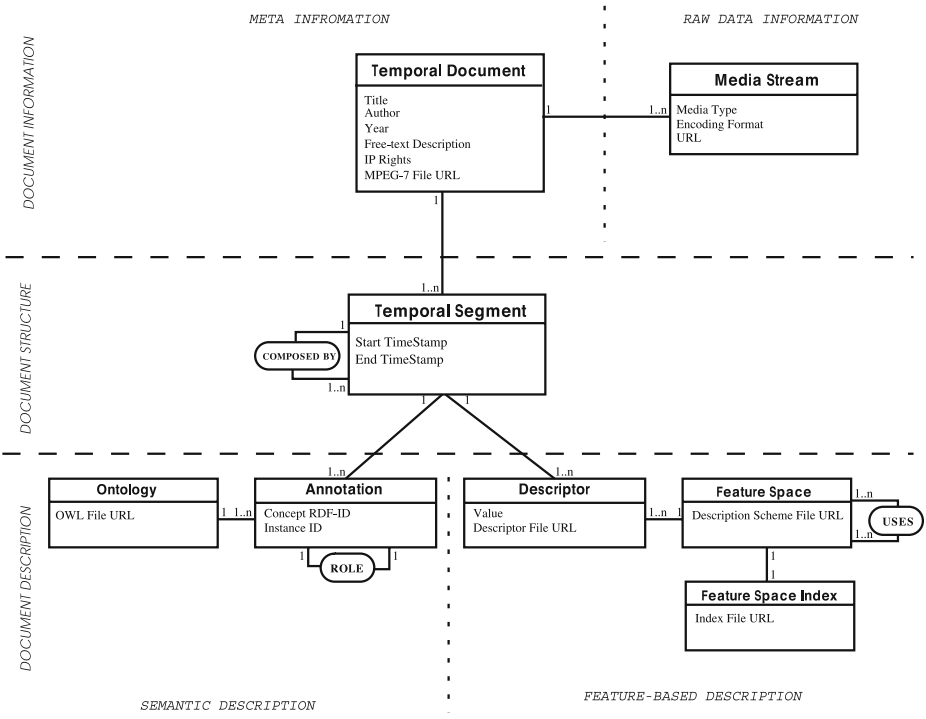

Fig. 1 Conceptual model of a video corpus representation

The key part of our model is the temporal decomposition of every document. We take the temporal dimension as a feature common to all modalities (visual, audio, textual) and exploit this property to create relations between pieces of information. By contrast, any other possible decomposition such as that proposed by the MPEG-7 standard would become an extra information attached to a particular information stream (e.g., the spatial decomposition of a key-frame).

The notion of a *temporal segment* is therefore the central building block for our model. It is initially defined as a continuous temporal interval over the multimedia stream S:

$$I_a^b(S) =]a, b[, \forall\, a, b \text{ s.t } 0 \leq \mathrm{a} \leq \mathrm{b} \leq \mathrm{T_S} \tag{1}$$

where T_S is the total length of the stream. A recursive definition gives a temporal segment as a composition of shorter temporal segments.

$$I_{(a_k)}^{(b_k)}(S) = \bigcup_k I_{a_k}^{b_k}, k = 1, \cdots, n \tag{2}$$

Any temporal pattern may therefore be defined within our scheme. Since no absolute temporal reference may be used, the definition makes sense only in association to a particular document (as identified by its *document information*). The converse is also true. To be valid, any piece of information should come with a temporal reference.

In particular, a complete document S is associated with $I_1^{T_S}(S)$ and any partition of S with a partition of that interval. Thus, our model readily copes with concurrent temporal segmentations of a given document.

2.1 Description spaces

Temporal segments organize the data along the temporal dimension. We define a further classification of the information contained in the *document description* layer (the temporal information) into main categories. We define the *asserted description* as the description that is given from an external knowledge source and the *deduced description* as being a description inferred or computed from the multimedia stream itself. Typically, the asserted description may be provided by a human operator annotating the document in question and therefore be located at a rather high semantic level. The deduced description is computed automatically and corresponds to the document features extracted from the data itself. This distinction places us in a favorable context for the development and test of multimedia information processing algorithms. For example, deduced description will form an automated characterization that the asserted description may help in evaluating (see Section 4 for an example).

In order to implement our data model, the distinction to consider is between *semantic description* and *feature-based description*, which corresponds to distinct and complementary storage modes.

2.1.1 *Semantic description*

Semantic description is integrated in the model through manual annotations. As free text annotation may provide a noisy description due to the lexical and cultural differences among annotators, the external knowledge is normalized by the use of an *ontology*. The semantic description therefore lists the set of instances of concepts (as defined by the ontology) that occur within a temporal segment. This scheme allows us to use generic multimedia annotation frameworks such as that given by the Semantic Web (see [9] for a more detailed proposition). As a complement, associations between instances may be created, according to their possible *roles*, as defined by the ontology. Note that our proposed model is directly able to represent different semantic descriptions, using various ontologies.

Clearly, tradeoffs are to be determined between the complexity of the ontology used and the level of description needed. An important factor to take into account is also the complexity of the annotation, strongly related to the size of the ontology at hand. In our research-oriented scheme however, the semantic description plays a crucial role. It provides a semantic organization of the content that may be used for high-level querying and browsing the collection, and for training or evaluation of classification or recognition algorithms.

2.1.2 *Feature-based description*

The main goal of our framework is to store, organize and create relations between automatically computed features. These are seen as a description deduced from a particular temporal segment. A feature-based description (or simply, a *descriptor*) of

a multimedia content is defined in relation to a *feature space*. In the general case, a descriptor attached to a temporal segment corresponds to a set of points or a trajectory within that feature space. Further, as some descriptors may be computed from other descriptors (e.g., shape descriptor computed from a spatial segmentation), feature spaces may be related through a *uses* relationship. Here again, our model closely matches the underlying architecture of the feature extraction procedures used.

For the sake of simplicity, simple descriptors are represented by their values. In the most complex case, we use external files storing these values. In order to access such descriptors, an index may be constructed for the corresponding feature space. A feature space index is a file storing the accessing methods along with the index data. For now, we have used complete distance matrices to index feature spaces, but for obvious computational reasons others indexing structures should be used. For example, tree-based index structures may be used, such as VP-Tree or M-Tree (see [5]).

Our framework therefore provides an efficient way to store the output of multimedia stream content analysis algorithms for evaluation or comparison purposes. The co-existence of both levels of description within an unified repository makes it easy to define evaluation or supervised training procedures. Further, as a complement to the semantic description, the feature-based representation of the temporal segments opens the way to constructing querying and browsing mechanisms.

3 Indexing temporal documents

3.1 Data generation

We have mapped our model onto a database schema. Our database currently handles more than 150 GB of video data coming from the two corpora gathered by the MPEG-7 and the TREC Video Retrieval Evaluation (2003 and 2004) communities. This heterogeneous set of videos contains many genres, including sport, sitcom series, variety program, TV news and documentaries. It illustrates typical TV broadcast by the variety of its content and is widely used as a benchmark for video analysis and indexing tasks.

Raw documents are processed in order to extract low-level information about their temporal structure (including shot detection), their activity content (camera displacement, regions of activity, event) and their global color and motion distribution. The speech transcripts extracted by Automatic Speech Recognition (ASR) at LIMSI laboratory [7] and all data made available on the TRECVid data (including annotations) is also stored in the database. These descriptors provide us various viewpoints on the raw documents according to their intrinsic audio-visual properties.

3.1.1 *Semantic annotations*

Documents semantic annotations (either manually generated or imported from TRECVid data corpus) relies on an ontology that is based on both the taxonomy presented in [16] and the lexicon of the TRECVid Collaborative Annotation Forum [10] (Fig. 2). This ontology is centered around the concept of a video shot. It

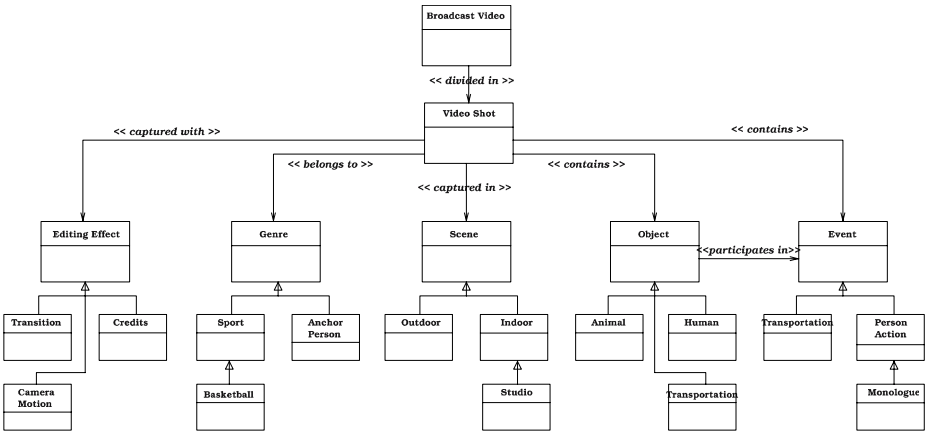

Fig. 2 Ontology for semantic annotation of video documents

is widely acknowledged that shots form essential semantic elements of a video stream. However, within our data model, shots are just a particular case of temporal segments.

Thus, other ontologies may be used, based for example on the concept of *scene* (set of *visually* correlated shots) or *story* (set of *semantically* correlated shots). This ontology creates annotations that provide us sufficient information for easy access to our database content and corresponds well to the documents features we wish to characterize automatically (e.g., events, scenes, objects, etc).

3.1.2 *Temporal partitioning*

Since the temporal structure of multimedia documents is central to our framework, the first step we take is to achieve temporal segmentation of multimedia streams. This approach is compatible with the fact of considering a video shot as a temporal unit for subsequent processing.

An automatic algorithm for video temporal segmentation based on the minimization of an information-based criterion has been developed [8]. It offers very good detection performance for abrupt as well as smooth transitions between shots. The algorithm proceeds according to the following steps.

The video content is first abstracted by a color dissimilarity profile using the classic color histogram and the Jeffrey divergence as similarity measure. The complexity of further processing is then reduced by robustly detecting non-ambiguous events such as hard transitions and sequences of still frames. An information-based segmentation is performed using a minimum message length (MML) criterion and a Dynamic Programming algorithm. This parameter-free algorithm uses information theoretic arguments to find the partitioning which agrees with the Occam's razor principle: the simplest model that explains data is the one to be preferred. The minimization process is fast by using the characteristics of video data like the presence of hard-cuts and redundancies to reduce the search for the solution. The computational complexity will depend on the video data but it is typically running in linear time.

Table 1 Performances of the shot boundaries detection

Performances	Our algorithm	Hardcut detection alone
Recall	92.4	67.6
Precision	80.2	78.8

At this stage, we obtain temporal segments whose definition is not guaranteed to match that of a shot. Since we see this level of decomposition as containing useful information, it is stored within our database and forms the most atomic temporal unit. However, to remain compatible with other studies, a final merging algorithm uses statistical hypothesis testing to group together segments that are unlikely to form different shots then stored as segment compositions.

As a basic example of evaluation facilitated by our framework, Table 1 presents the results of an experience using 70 videos of the TRECVid corpus and the evaluation framework of [17]. We used 35 h of news programs and the ground truth provided by the TRECVid community. The performances of the algorithm are comparable to the best results obtained by the participants of TRECVID 2003. The main advantage of our shot boundary detection algorithm is that we make a minimum number of assumptions about the definition of a video transition. The algorithm will detect any kind of special effect without any particular modeling. From these results, we have built confidence in our algorithm and used its results for the processing of streams where ground-truth was not available.

3.1.3 *Activity-based video decomposition*

Along with basic global descriptors of the temporal segments (color histograms, motion histograms, ASR descriptors), we extract from the visual streams, a more detailed description of the content. The aim [13] is to decompose a given video shot (as defined above) into several spaces characterizing meaningful parts of its content

- Capturing effects : trajectories of the affine parameters of the camera displacement
- Capturing environment : descriptors of the background
- Moving objects : salient regions of activity
- Events : trajectories of salient regions w.r.t background

Spatial salient points are extracted from each frame and matched between two successive frames. The global affine motion model (*Camera Displacement*) is estimated from the set of points trajectories. Salient regions of activity are extracted and tracked along the stream using the background model and the feature distribution of the points. As an example of extra information created from raw data and stored within our database in relation to the original data, Fig. 3 illustrates how a video shot may be represented by the plots of the affine parameters of the trajectories of the camera displacements, the mosaic of the scene (represented in the system by the MPEG-7 *Scalable Color Descriptors (SCD)* and *Non-Homogenous Texture* descriptors) and the set of salient regions of activity (represented in the system by the MPEG-7 *SCD* and *Motion Trajectories* descriptors).

The temporal granularity of such a description is not homogenous. Salient regions of activity could be defined on temporal segments that are subparts of a shot.

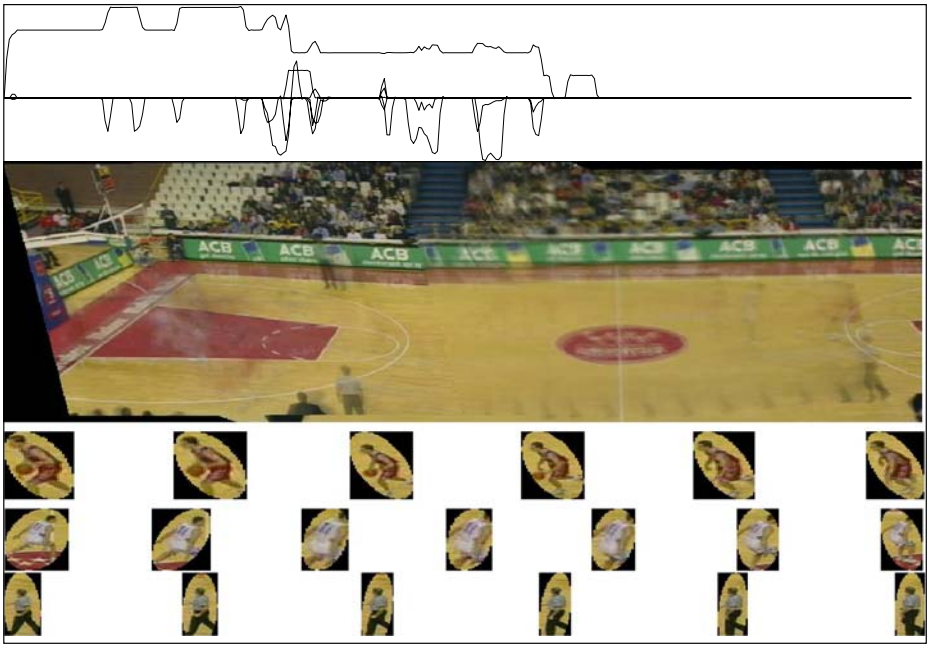

Fig. 3 Representation of the visual content of part of a basketball game using salient features analysis: trajectories of the six affine parameters of the camera displacement, mosaic of the scene, three samples of salient regions of activity

Here again, the temporal structure of our data model allows to describe documents at different temporal scales and to potentially combine descriptions of temporal segments into a coarser temporal granularity.

3.2 Data access

We now have a data repository that stores structured temporal audio-visual data enriched with low-level and semantic data. Basic access is given by the DBMS. The underlying model opens access to data using a document reference and a given temporal segment within it. From there, any information related to that temporal segment may be queried.

However, access to the raw data (audio-visual streams) of the temporal segments or to the complex spaces of their descriptions cannot be handled directly by the DBMS. Hence, an external index processor is proposed, that combines in a transparent way, different indexing methods for the different kind of data to be accessed.

3.2.1 *Raw data access*

In order to efficiently access the raw data of video documents that are usually stored in a compressed format, the index processor integrates a framework we have developed. OVAL (Object-based Video Access Library [12]) permits random access of data on AV streams. Typically, OVAL offers a common API on AV streams so

as to emancipate from the actual type of storage used for that particular stream (advantages of particular storage modes may however still be accessed, such as motion vector within an MPEG-2 stream). One advantage of OVAL over other data access libraries is that its abstraction enables generic VCR-like operations and also adds frame-precise random access facility to data streams. For example, using OVAL, a key-frame in a video stream is retrieved online by the sequence of `open`, `goto` and `extract` operations, thus avoiding physical duplication of data that may become obsolete. OVAL includes index pre-computation and buffering facilities so as to make the use of these operations as efficient as possible.

Using OVAL and coupled with our DBMS, the index processor forms a base for querying audio-visual data that makes transparent access at various levels and from different modes.

3.2.2 *Ontology-based annotation access*

Annotations may be accessed directly using the DBMS as *key* queries on the name of the concept queried. But, such queries may act only on leaf concepts of the ontology. In order to access annotations and still take into account the structure of the underlying ontology, i.e., the different relations between concepts of the ontology, the index processor integrates reasoning facilities provided by the SWKB engine developed in our lab [9]. SWKB permits structured knowledge inference, and hence is used to performed structured knowledge querying in an efficient way. Thus, the index processor is able to handle OWL description-logic predicate queries, providing the framework with powerful access to the annotations.

3.2.3 *Feature-space descriptions access*

Temporal segment descriptions associated to given feature spaces (e.g., color or motion) are, in general, given as vectors of high dimensions. Such vectors represent either a global descriptor of the temporal segment content (e.g., statistical moments) or a signature of the trajectory of sub-segments content descriptors (e.g., DWT signature, Piecewise linear representation). In order to perform searches in such high-dimensional spaces, different indexing structures may be used such as R-Tree [2], VA-File [21] or MVP-Tree [3]. Furthermore, temporal segment descriptions may be directly indexed as trajectories of vectors in which case other specific indexing structures have to be used (e.g., STR-Tree, TB-Tree [15], Multiversion-QuadTree [20]). The index processor proposed is able to integrate any kind of indexing structures, so that, depending on the feature-space and the queries performed, the access is processed in the most efficient way.

4 Retrieving temporal documents

Video retrieval systems aim at retrieving, from within a document collection, documents or parts of documents that correspond to a user query. The baseline of video retrieval systems is a simple video document browsing tool. A user may view all video documents, and finally decide which parts of which documents correspond to his/her needs. Clearly, such a system, while allowing complex queries is unusable in practice. A retrieval system should provide an automated answer to the user query within

a maximum time interval. Most state-of-the-art systems [1, 6] use content-based query-by-example (QBE), coupled with some textual retrieval capabilities in order to achieve this task. We show here how the above video management framework has been extended in the direction of creating a complete content-based video retrieval system using QBE and text retrieval.

4.1 Video document retrieval

We have developed a content-based retrieval system based on the relevance feedback paradigm. Users are able to formulate complex queries by iteratively providing positive and negative examples selected from within the documents retrieved by the system. The query is thus refined by user feedback until user satisfaction. The main challenge here is to develop an online and almost-real-time interactive scheme that is able to learn some semantic concepts in high dimensional feature spaces.

In order to avoid the use isolated descriptors that penalize response time, video segments are indexed by pairwise dissimilarities computed in a multimodal feature space. A dissimilarity space (see Pekalska et. al [14]) is then build, where data coordinates are their dissimilarities to training data. The main advantage of this technique is to reduce the dimensionality of the representation from the initial dimension of the feature space to the cardinality of the training set. A non-linear Fisher criterion is then optimized in this space of reduced dimension so as to obtain a new ranking function where positive elements tend to be placed on the top of the list while negatives are pushed to the end [22].

Our retrieval process is evaluated against the annotated TRECVid corpus (see Section 3 for details) on which random queries according to particular annotations are performed. Figure 4 displays the Precision-Recall graphs for the query "Hockey" (spanning around 0.5% of the 32,000 elements indexed in the database) and for various numbers of positive and negative examples. Dissimilarities are computed on color, motion and ASR histograms. The increasing performances when more and

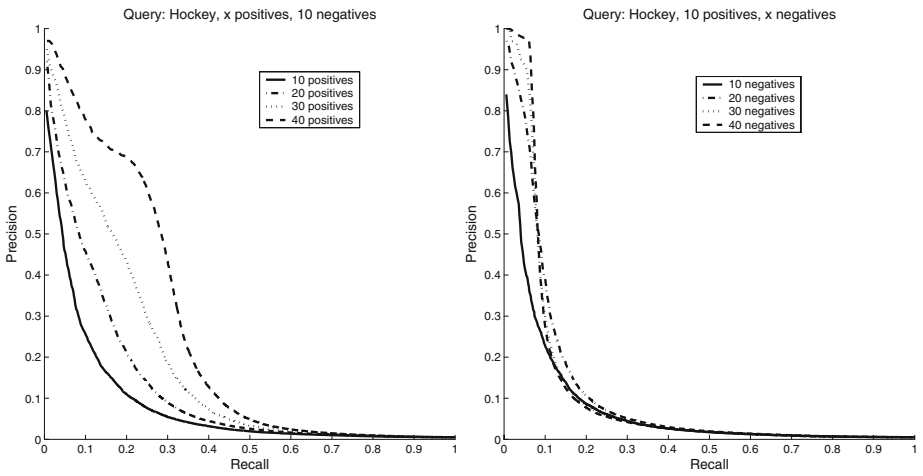

Fig. 4 Precision–recall graphs averaged on 100 instances of the query "Hockey" for an increasing number of positive and negative examples

more of examples are available show the ability of the system to learn semantic queries during relevance feedback loops.

Finally, we were interested in the computation time problem. Average response time for 20 negative and 5, 10, and 40 positive examples are, respectively, 1.4, 2 and 7.4 s while for 10 positive and 100 negative examples the time is 4.3 s. As the dimensionality of the representation space is equal to the number of positive examples, the response time increases according to their number. On the other hand, negative examples have less influence since they are just involved in the learning process.

4.2 ViCoDE video retrieval system

ViCoDE (Video Content Description and Exploration) is the video retrieval system putting together the data management framework described earlier (including the data corpora detailed in Section 3) and the above retrieval scheme. As such, it provides a user with efficient content-based QBE, enhanced with relevance-feedback interactions. It also provides textual retrieval for document meta-information and audio stream speech transcripts. In order to cope with the well-known "page zero" problem to which every QBE retrieval system is faced, instead of using inefficient query by sketch [4], *ViCoDE* provides several collection exploration methods that permit the user to find initial examples to seed the relevance-feedback loop.

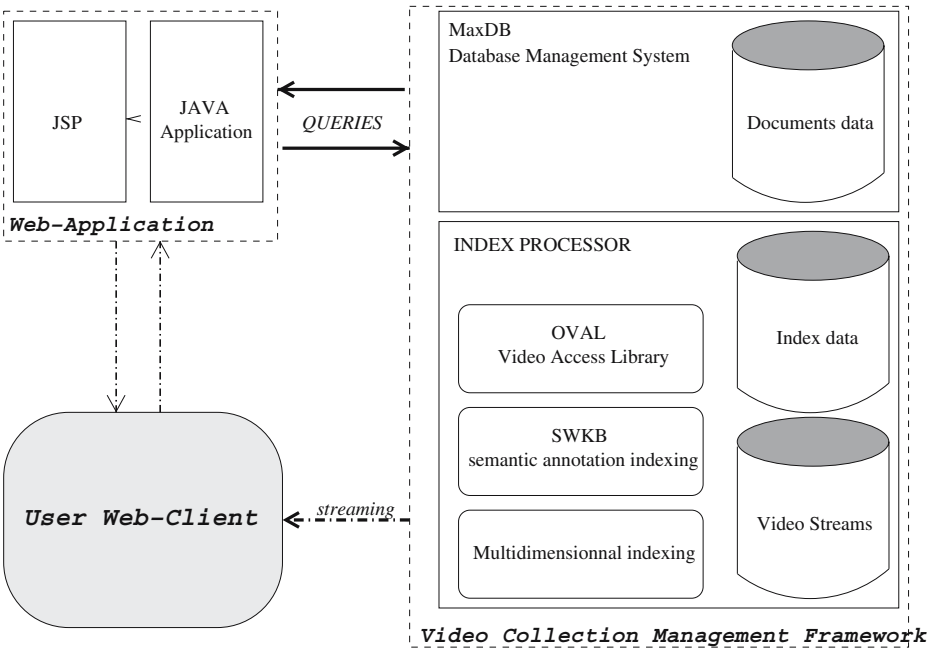

Fig. 5 *ViCoDE* system architecture

4.2.1 *ViCoDE architecture*

Practically, *ViCoDE* is a Java web-application extending our video collection management framework. Figure 5 shows the architecture of the system. User interactions are captured via the web-based interface. The *JAVA* web-application processes the queries and translates it as basic *SQL* statements (e.g., documents meta-information search, documents temporal structure browsing) or as index processor queries (e.g., keyframes generation, ontology based queries, multidimensional queries). Results are formatted as *Dynamic HTML*, via the JSP framework.

ViCoDE takes advantage of the efficient data access mechanism provided by our model, especially for the on-the-fly generation of the keyframes and video excerpts, the browsing through the document collection and the content-based retrieval algorithms.

4.2.2 *Video documents retrieval using ViCoDE*

As for most state-of-the-art video retrieval systems, *ViCoDE* is based on content-based queries by examples. It makes use of the low-level descriptions of queried video temporal segments to retrieve the most similar segments (in terms of the multimodal features available) within the collection. *ViCoDE* is based on the retrieval algorithm presented in the previous section which combines multimodal descriptions to characterize the user query, and exploits user interactions through a relevance-feedback loop.

ViCoDE also permits a user to query the collection for documents, based either on their meta-information (e.g., title, creation date, authoring), or on the audio stream speech transcripts.

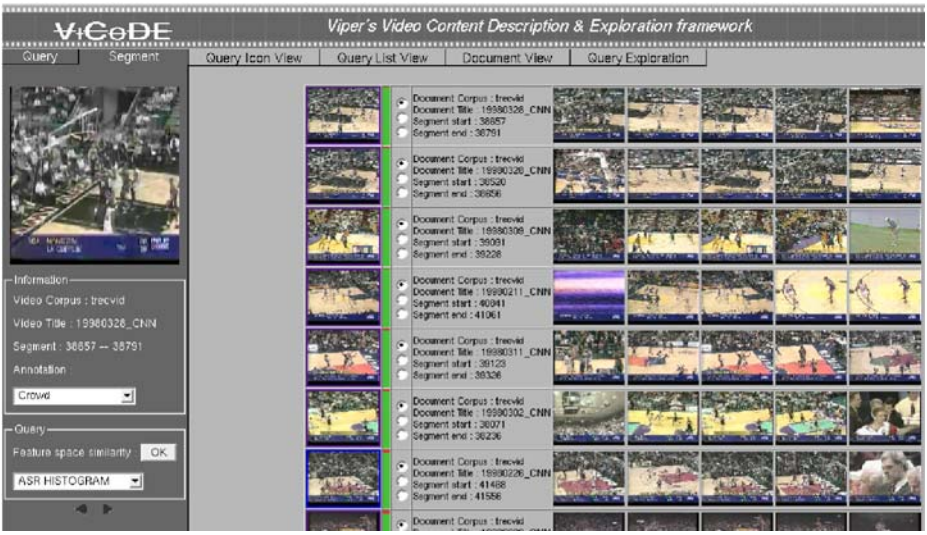

Fig. 6 *ViCoDE* displayed results for the query *Basketball*

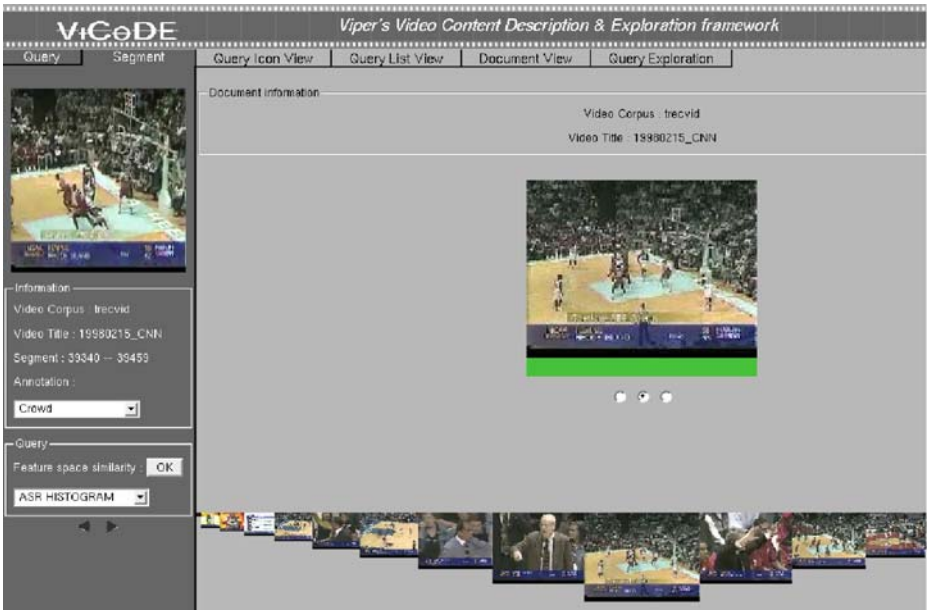

Fig. 7 *ViCoDE* document centered exploration interface

This retrieval method provides good performances (measured as P/R) for a user query, as it has been shown in the previous section. Figure 6 shows a typical result for the query *Basketball*.

Fig. 8 *ViCoDE* feature-space centered exploration interface

However, the user still should find initial positive samples for expressing his/her query. Textual queries may not be able to capture the user needs and, collection simple browsing is known to be inefficient. For that reason, *ViCoDE* provides two video document exploration methods, aiming at quickly browsing the collection to find examples for initiating the QBE.

First, the document-centric exploration (Fig. 7) permits a user to browse through a single document in an efficient way using some Fisheye-like viewing of the document temporal structure. As such, it captures the information provided by the document creator. It may also exploit the knowledge that a user have on a specific document.

The feature-space centric exploration permits a user to browse the collection from the point of view of a particular representation of the audio-visual content of segments. Given a feature-space (e.g., color, motion or events), segments are projected onto a 2D surface using a Sammon mapping [18] so that the user is able to figure out the region of the space where he may find positive samples for the query (Fig. 8).

By combining the different user interaction methods, and benefiting from the efficiency of our video collection management framework, *ViCoDE* provides efficient content-based video document retrieval in terms of precision/recall and moreover in terms of response time.

5 Conclusion

We are advocating for the use of an advanced data storage and retrieval framework for the development and evaluation of multimedia processing algorithms. We have based the development of our framework around the temporal properties of the data to be stored. Within our data model, raw data, annotations and extracted features coexist and may even overlap along the temporal dimension. Although not explicitly using any standard, we remain fully compatible with alternative description schemes such as MPEG-7 while not being constrained by their syntax or structure.

We have presented *ViCoDE*, a complete application based on our framework. We believe that the use of such a framework is unavoidable for the development of video indexing and retrieval applications. We further state that the very same framework may also serve for the evaluation. Duality between development and evaluation is made evident using an incremental annotation scheme whereby ground-truth is incrementally built for subsequent processing or objective systematic evaluation. Further developments will address the test and extension of our models to handle richer multimedia data.

References

1. Amir A, Berg M, Chang S-F, Hsu W, Iyengar G, Lin C-Y, Naphade M, Natsev A, Neti C, Nock H, Smith JR, Tseng B, Wu Y, Zhang D (2003) IBM research TRECVID-2003 video retrieval system. In: Proceedings of the TRECVID 2003 workshop, New York
2. Beckmann N, Kriegel H-P, Schneider R, Seeger B (1990) The R*-tree: an efficient and robust access method for points and rectangles. In: SIGMOD '90: Proceedings of the 1990 ACM SIGMOD international conference on management of data, Atlantic City, USA, pp 322–331

3. Bozkaya T, Ozsoyoglu M (1997) Distance-based indexing for high-dimensional metric spaces. In: Proceedings of the 1997 ACM SIGMOD international conference on management of data, Tucson, Arizona, pp 357–368
4. Chang S-F, Chen W, Meng HJ, Sundaram H, Zhong D (1997) VideoQ: an automated content based video search system using visual cues. In: Proceedings of the fifth ACM international conference on multimedia, Seattle, pp 313–324
5. Chávez E, Navarro G, Baeza-Yates R, Marroquin J (2001) Searching in metric spaces. ACM Comput Surv 33(3):273–321
6. Gaughan G, Smeaton AF, Gurrin C, Lee H, McDonald K (2003) Design, implementation and testing of an interactive video retrieval system. In: Proceedings of the 5th ACM SIGMM international workshop on multimedia information retrieval, Berkeley, California, pp 23–30
7. Gauvain J, Lamel L, Adda G (2002) The LIMSI broadcast news transcription system. Speech Commun 37(1-2):89–108
8. Janvier B, Bruno E, Marchand-Maillet S, Pun T (2003) Information-theoretic framework for the joint temporal partioning and representation of video data. In: Proceedings of the European conference on content-based multimedia indexing CBMI'03, Rennes, France
9. Jelmini C, Marchand-Maillet S (2004) OWL-based reasoning with retractable inference. In: Proceedings of the conference on coupling approaches, coupling media and coupling languages for information retrieval (RIAO'04), Avignon, France
10. Lin C-Y, Tseng BL, Smith JR (2003) Video collaborative annotation forum: establishing ground-truth labels on large multimedia datasets. In: Proceedings of the TRECVID 2003 workshop, Gaithersburg, Maryland
11. Manjunath B, Salembier P, Sikora T (eds) (2001) Introduction to MPEG-7: multimedia content description language. Wiley, New York
12. Moënne-Loccoz N (2004) OVAL: an Object-based Video Access Library to facilitate the developement of content-based video retrieval systems. Technical report, Viper group, University of Geneva, Switzerland
13. Moënne-Loccoz N, Bruno E, Marchand-Maillet S (2004) Video content representation as salient regions of activity. In: Proceedings of the international conference on image and video retrieval, CIVR'04, Dublin, Ireland
14. Pekalska E, Paclík P, Duin R (2001) A generalized kernel approach to dissimilarity-based classification. J Mach Learn Res 2:175–211
15. Pfoser D, Jensen CS, Theodoridis Y (2000) Novel approaches in query processing for moving object trajectories. In: VLDB '00: Proceedings of the 26th international conference on very large data bases, Egypt, pp 395–406
16. Roach M, Mason J, Xu L-Q, Stentiford F (2002) Recent trends in video analysis : a taxonomy of video classification problems. In: Proceedings of the international conference on internet and multimedia systems and applications, IASTED, Hawaii
17. Ruiloba R, Joly P, Marchand-Maillet S, Quenot G (1999) Towards a standard protocol for the evaluation of video-to-shots segmentation algorithms. In: International workshop in content-based multimedia indexing (CBMI), Toulouse, France
18. Sammon JW Jr (1969) A nonlinear mapping for data structure analysis. IEEE Trans Comput 18:401–409
19. Smeaton AF, Kraaij W, Over P (2003) TRECVID 2003 - an introduction. In: Proceedings of the TRECVID 2003 workshop, Gaithersburg, Maryland
20. Tzouramanis T, Vassilakopoulos M, Manolopoulos Y (2000) Multiversion linear quadtree for spatio-temporal data. In: Proceedings of the 4th East-European conference on advanced databases and information systems (ADBIS-DASFAA'2000), Prague, Czech Republik, pp 279–292
21. Weber R, Böhm K, Schek H-J (2000) Interactive-time similarity search for large image collections using parallel VA-files. In: ICDE, San Diego, California, p 197
22. Zhou X, Huang T (2004) Small sample learning during mutlimedia retrieval using biasmap. In: Proceedings of the IEEE conference on pattern recognition and computer vision, CVPR'01, vol. I, Hawaii, pp11–17

Nicolas Moënne-Loccoz received his M.S. degree from the University of Nice-Sophia Antipolis, France in 2001, and his Ph.D in computer science from the University of Geneva, Switzerland in 2005. He is working at the Computer Vision and Multimedia Laboratory, University of Geneva, Switzerland, as a postodoc research associate. His research interests focus on events-based description of video sequences for content-based indexing and retrieval.

Bruno Janvier received his M.S. degree from the Engineers School of Physics in Strasbourg, France in 2002, and is currently a Ph.D student at the Computer Vision and Multimedia Laboratory, University of Geneva, Switzerland. His research interests focus on video analysis and content-based video indexing and retrieval (CBVR).

Stéphane Marchand-Maillet received his PhD on theoretical image processing from Imperial College, London in 1997. He then joined the Institut Eurecom at Sophia-Antipolis (France) where he worked on automatic video indexing techniques based on human face localization and recognition. Since 1999, he is Assistant Professor in the Computer Vision and Multimedia Lab at the University of Geneva, where he is working on content-based multimedia retrieval as head of the *Viper* research group. He has authored several publications on image analysis and information retrieval, including a book on low-level image analysis. He currently leads the Benchathlon, a joint international effort for benchmarking content-based image retrieval systems.

Eric Bruno received his M.S. degree from the Engineers School of Physics in Strasbourg, France in 1995, and his Ph.D in signal processing from the Joseph Fourier University, Grenoble, France in 2001. Since 2002, he is working at the Computer Vision and Multimedia Laboratory, University of Geneva, Switzerland, as a research associate. His research interests focus on video analysis and content-based video indexing and retrieval (CBVR) which include motion estimation, region tracking, statistical learning of the video content and information retrieval.