

Supporting different search strategies in a video query interface

Mark D Dunlop (1) and Kieran McDonald (2)

1: Centre for Human Machine Interaction, Risø National Laboratory, Roskilde, Denmark
mailto: mark.dunlop@risoe.dk

2: Computer Applications, Dublin City University, Dublin, Ireland
mailto: kmcdon@compapp.dcu.ie

Abstract

This paper reports the design and development of the Diceman Query Application. This is the end-user query application for a video indexing and retrieval project based on the Diceman architecture for distributed internet content exchange using MPEG-7 and agent negotiation. The query application was developed to support different search strategies of users accessing large video archives that have been indexed with a complex indexing language. The paper describes the interface, its design, the strategies supported, and initial results from user tests of building complex queries using the query interface (including a discussion of end-users ability to formulate meaningful semantic queries using low-level indexing features). Finally the paper discusses the implications of the interface on the underlying search engine.

Introduction

With the advent of digital television and standards such as MPEG, large collections of digital video are becoming increasingly common. The MPEG7 emerging standard for multimedia content description will further accelerate the availability of video collections and enable large-scale search facilities for video, similar to web search engines for textual documents. While much research is ongoing into the automatic indexing of video streams, this paper focuses on design issues for the end user search interface. In particular it applies the search strategies defined by Pejtersen (1979), in the context of developing a library support environment for fiction access, to the development of a video query interface for professional users accessing collections indexed with complex indexing/description schemes. The paper also presents the results of an initial investigation into users' abilities to search for high-level concepts using low-level attributes (an approach that will be required given the financial constraints of high-level fine-detailed indexing).

The paper starts with a brief overview of related work in the domain of video retrieval, the paper then discusses the five search strategies of Pejtersen and their application to the development of a professional video retrieval query interface. Finally, the paper presents the initial findings on users' abilities to formulate queries using the interface and a brief discussion of the implications of the interface design on the underlying searching and matching approaches.

Video retrieval

Content based text retrieval (Frakes and Baeza-Yates 1992)(Sparck Jones and Willett 1997) is a long established research field which has developed complex and increasingly accurate techniques for matching natural language phrases to natural language documents. The last decade or so has seen an increase in attention on non-textual retrieval: with individual media work on image retrieval (e.g. Faloutsos et al 1994; Flickner et al 1995; Frankel et al. 1996) and speech retrieval (e.g. Sparck Jones et al. 1996; Wechsler, Munteanu & Schäuble 1998) and work on the combined medium of video retrieval.

For the purposes of retrieval, video material can either be considered as:

- a sequence of images or, more correctly as Liou et al. (1999) stress, as a series of shots;
- a soundtrack with associated video;

- a textually annotated temporal sequence;
- or as a combination of the above techniques.

Converting a raw sequence of images into a sequence of shots is no trivial task. Shot boundaries are rarely as sharp as cuts in early cinema footage, and considerable research effort is being expended trying to automatically segment video into shots (e.g. Zhang 1995, Liou et al. 1999). Additionally, the type of shot transitions, their frequency and speed are dependent on the material being viewed, for example documentaries have very different styles than quiz shows (O'Toole et al 1999). Once detected, shots can be used to create tables of content, aid extraction of key-frames and to help users browse and recognise scenes within videos (Lee, Smeaton and Furner 1999). In addition, research is continuing into the automatic indexing of other-camera level activities such as the type of shot transitions and camera motions (e.g. Akutsu and Tonomura 1994).

Speech recognition can be used, either with fixed time windows or associated with shots, to retrieve video based on the spoken soundtrack. Compared with image/shot-based work, this more closely resembles the traditional text retrieval work where retrieval would be based on a natural language phrase (in the case of speech retrieval, either spoken or typed) matching against (recognised) text. Indeed many techniques in video soundtrack matching are reminiscent of passage retrieval work in textual IR (Kaszkiel and Zobel 1997). Soundtrack-based retrieval, while clearly accessing only one aspect of a video is, however, more likely to support topicality-based searching, where a user searches for video on a specific topic. In addition the soundtrack and spoken words are often very memorable hence this approach is also likely to support known-clip retrieval. Both these types of search are likely to complement searching based on visual properties and video structure.

Alternative to image-based shot work and soundtrack-based retrieval some work has been carried out on using superimposed captions on, for example, news programs (e.g. Sato et al. 1999) and on using teletext/closed-caption subtitles (e.g. Brown et al. 1995). While somewhat restricting the material that can be automatically indexed, this is a powerful and fairly accurate approach for material that has been subtitled. In addition, with the increasing availability of movies with multi-lingual subtitles, this approach could provide some form of cross lingual retrieval.

The MPEG-7 initiative (Koenen et al 1998) has recently started the process of standardising the multimedia content description interface in the hope of bringing together many of the indexing directions discussed above within a single framework. MPEG-7 aims to specify a standard set of descriptors that can be used to describe various types of multimedia information (and how to define new descriptors and description schemes/structures). Given the success of the MPEG-1 and MPEG-2 standards for low-bandwidth broadcasting and digital television transmission respectively, the impact of MPEG-7 is likely to be considerable. Given the different approaches to video indexing and the requirements of different source domains, the task is, however, extremely complex.

This paper presents the design of an interface that supports users entering complex queries based on many different, typically visually-based, indexing descriptors. Based on professional users, the work was aimed at exploring design issues for creating searches on complex structures and indexing languages likely with MPEG-7 databases.

Diceman architecture

The Diceman project (Ward 1998) aims to provide an end-to-end solution for the indexing, storage, search and trading of digital AV content in order to support both content providers and content users. Heavily based on an agent architecture, Diceman aims to support multi-user searching of collections held by multiple content providers. Agents representing the content providers will attempt to interpret search requests intelligently, while agents representing content consumers will learn about user preferences and manage search results from multiple sources.

The Content Provider's Application (COPA) will integrate technologies for analysing and describing content before storing it in the DICEMAN database, a specialised database which is MPEG-7 aware.

The Diceman Query application was initially developed to act as a user front-end for searching a single COPA database and as a framework to test ideas regarding the usefulness of different query types. The Diceman COPA currently supports the indexing of video according to the description scheme and descriptors shown in figure 1. In line with much work on visual-based image retrieval, discussed earlier, the description language focuses mainly on camera motion, shot and colour information. However, for trial purposes character names were added to the description language.

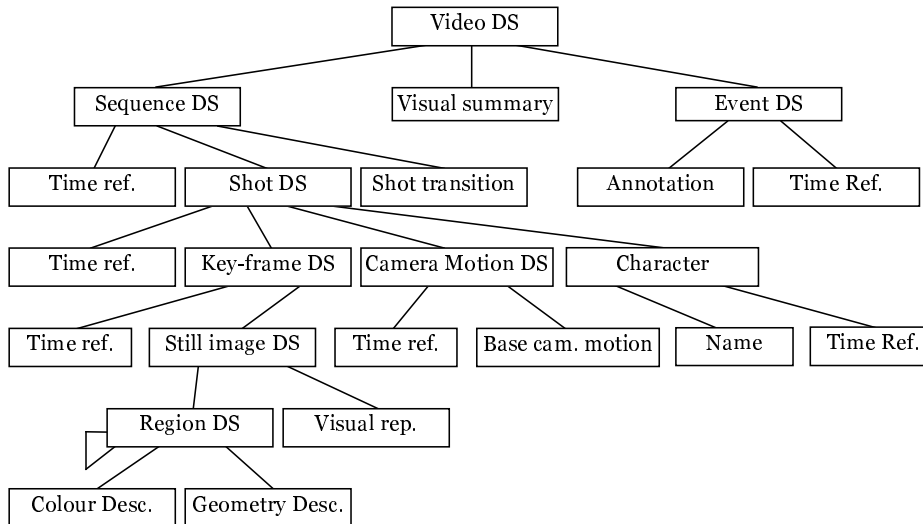


Figure 1: Description language for DICEMAN indexing

Query interface motivational scenarios

Early in the project, we met with an independent documentary filmmaker working in Ireland. Based on this interview, two scenarios were developed to direct the development work of the project. The first scenario is a classic “known item” search while the second is a more open search.

Scenario 1:

Sharon is looking through her rushes¹ for a clip when a woman at the Music Festival turns to look at the camera and pulls a face.

When starting to digitally edit a documentary, typically the video editor will digitise all their rushes at very low quality. As they proceed through the editing process the amount of material on disk that may be used will be reduced as more and more footage is eliminated. Periodically the material is recaptured at a higher quality to, again, fill the disk space. During this process the editor is constantly refining her idea of what material to include and comparing clips.

In this scenario the editor will know her material, having usually watched all the material before starting the editing process, and will be able to do fairly detailed descriptions of the material from memory. When wanting to select a specific clip, typically editors will know the shot/scene they are looking for and, unlike web searching, will know that the clip is present.

Scenario 2

Sharon is making a documentary about Galway where JFK made a famous speech in the main square. She wants a clip of President John F Kennedy talking to Taoiseach Jack Lynch.

¹ “a print of a motion-picture scene processed directly after the shooting for review by the director or producer”, *WWWebster Dictionary*.

Here the ideal clip would be cheap to licence for the purpose the documentary will be shown (e.g. local TV, national TV, world TV, cinema rights), with an easy to administer licensing agreement, of JFK and JL in Ayr Square in Galway (if not, somewhere not obviously not Ireland), of good quality production and available as a good quality recording with little delivery delay. Editors may compose the final segment from many different source segments depending on costs and feasibility of smoothly editing the source pieces together.

In this scenario, the user will not know if the shot exists, hence in common with the standard behaviour of stock image agencies, the query system should return a large selection of shots. This will not only help the user find the shot but is also needed to convince her of the quality and extensiveness of the search, i.e. increase her confidence in the results, and make it more likely that the retrieved set will contain items that cover different interpretations of the query.

Complex business models are in place for costing information and this is one of the main decision criteria for selecting shots to use. While outside the scope of the initial query interface development, managing costing information is one of the aims of the Diceman agent approach.

Diceman query application

In her work on access to fiction collections, Pejtersen (1979) identified five search strategies for accessing a large collection of books. In this project, we used three of these search strategies and discussion with users to drive the development of a query interface to the Diceman end-to-end agent-centred search and indexing system. The remaining two search strategies were not implemented but are discussed in light of video retrieval and the Diceman agent architecture.

Bibliographical Strategy

“The user is able to identify reading needs by author and title... This strategy appears to be a kind of decision table search using catalogs and card indexes.”²

For video retrieval, bibliographic information is composed of details such as the title, producer and year of release of the video production as a whole. Figure 2 shows the bibliographic details screen from the Diceman Query Application.

In line with most of the interface the user is given the ability to set importance sliders to highlight those elements of the search (s)he feels are more important or that (s)he is more confident with. In response to initial user trials, these sliders are initially set to minimal (zero) for non-entered data and jump to maximal (four) when a field is completed, the user can then reduce the weight of uncertain and less important elements. The implications of these sliders on the underlying search engine are discussed in more detail later.

² Quotes from Rasmussen, Pejtersen and Goodstein (1994).

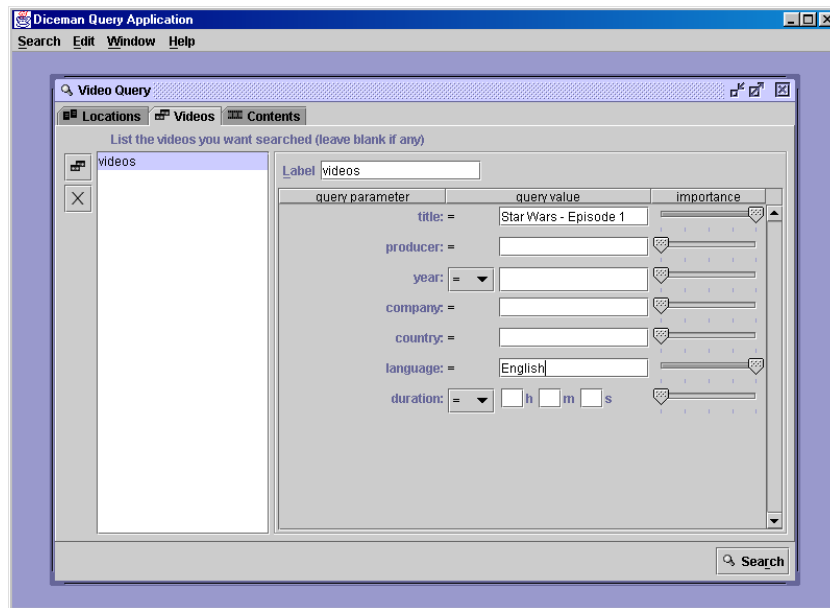


Figure 2: Bibliographic search screen

In terms of the scenarios, bibliographic search is likely to occur when searching for stock footage and is of limited research interest: if the user can specify these details then it is a simple database matching process to retrieve the desired video. Bibliographic searching could, however, also be used in conjunction with other search strategies to limit the space being searched (e.g. to limit the time period the material was shot in or to limit which rushes tapes to search).

Analytical Strategy

"The user explores reading needs systematically and compares them with the relevant aspects of the available books. This strategy is the rational, problem solving strategy."

The main form of search in Diceman is the support of complex, iterative query development. Figure 3 (main video query screen) shows the main search screen, the main section of which is composed of three columns:

- A visual representation of the user query (here showing a query for a shot containing the characters *Darth Maul* and *Yoda* but not *Anakin*). Items in this list can be edited, using the associated query form, to alter their properties (e.g. camera motion has a form where time, type, and speed of camera motion). Each of these properties has an associated editor based on their type (e.g. camera motion type can be selected from a pickable list, colours from a colour picker and numerical values from a text field with associated <,=,> operators). Query elements can also be optionally named for ease of recognition when refining queries or copying/pasting between queries.
- The importance of each element (here all equal and maximal). The sliders on the main query form allow for specification of the general importance of that element, whereas individual sliders within query element forms allow for more detailed specification of importance. Again, in line with the bibliographic details screen query element component sliders start at zero and jump to full once some specification is added.
- And a schematic time line (here showing no sequencing: elements can be dragged left/right to state elements occur before or after the other elements in a group - automatically creating sequence groupings as required, see figure 4). The timeline is intended to give a schematic representation of the query, somewhat mimicking the structure of matched clips. This is mainly

for feedback to the user that elements are composed and sequenced as (s)he intended but can also be used for limited query manipulation.

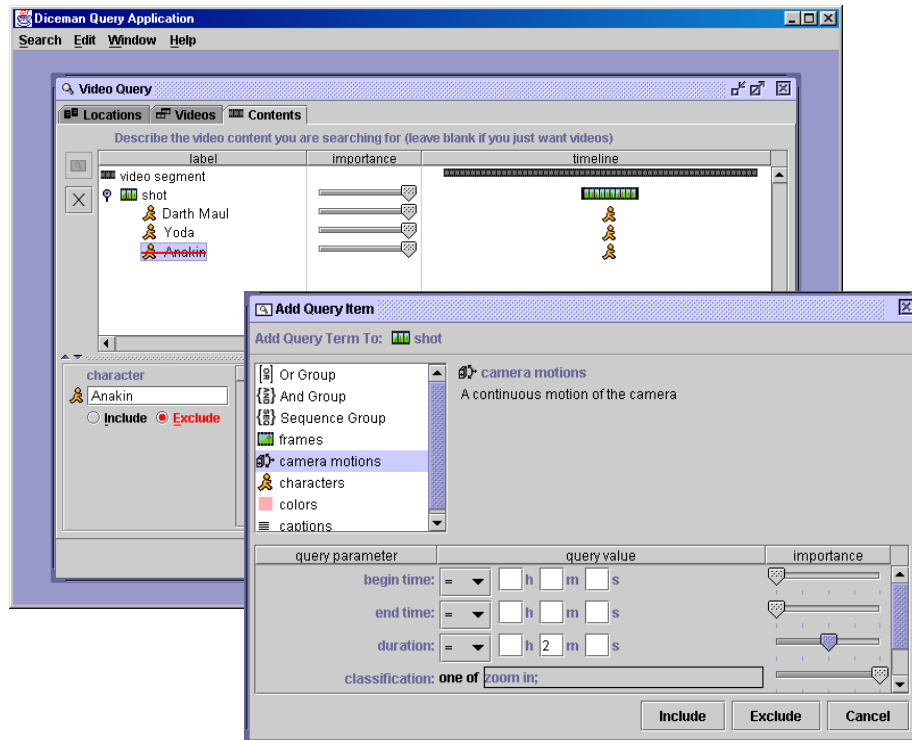


Figure 3: Main contents search and add query item screens

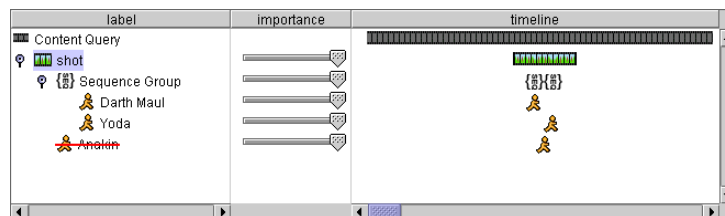


Figure 4: Revised query requiring Darth Maul to appear before Yoda

Superimposed on figure 3, the Add Query Item screen allows for specification of elements to be added to the query. When valid within the description language, all elements have a time stamp that can be from the start or before the end of the enclosing element (e.g. 30 seconds before the end of the shot) together with element specific attributes. Attributes of a query elements are ignored if left blank or given an importance of zero. By default, added items are ANDed together without sequencing information, but this can be altered and made explicit through use of the grouping elements AND, OR and SEQUENCE . These act as groups (similar to Windows Explorer folders) for other query elements and the grouping information is reflected both in the hierarchical view and the timeline view.

Once added, query items can be moved using copy-paste or drag-and-drop, deleted or have their attributes edited. As a result of initial user tests, it was decided that the interface should not enforce the strict hierarchical description language: the interface interprets user actions to create a valid query in terms of the description language. For example, the video indexing scheme may only allow shots to contain characters and scenes to contain shots, but from a query point of view the user should still be

able to search for a scene containing a character and not be artificially constrained to searching for shots that contain characters.

Following Pejtersen's observation on the systematic problem solving nature of searching, the interface supports query refinement by launching search results in a new window and allowing users to return to the query window to refine their query (e.g. importance sliders can be used to turn off element temporarily, new objects can be added, old items defined more precisely, or sequencing information added/changed). The aim of this approach is that users can start with a rough query to see how the system databases respond and then refine this query to as much detail as is required.

Similarity Strategy

"The user identifies their reading needs by mentioning a previously read book and asking for something similar."

To support similarity searching the Diceman system was altered to allow retrieved video clips to describe themselves. Upon request, the database server will return the main indexing elements for retrieved video clips, with *main* being defined by the individual database agents (possibly in negotiation with user agents). When a user selects "query like this", the retrieved element's description is fed into a new query window (the *liked clip* having been found using bibliographic or analytical searching or even as the result of a previous iteration of similarity searching). Here the user can simply carry out a new search based on the content of the selected item or edit the query before searching (by adding or deleting query elements, altering attribute specifications or importance sliders). Manipulation of query elements not only allows users to continue with the standard query-refinement process but also allows them to vary the weights of elements to stress those elements they are interested in (in essence specifying how they define similarity for this object and search).

Analytical and similarity searching are likely to be important for both known-clip and stock-clip searching. One would expect users to require more refinement when searching stock material as there will be more aspects to a query which have to be considered, but even over their own rushes users are unlikely to specify a query perfectly initially.

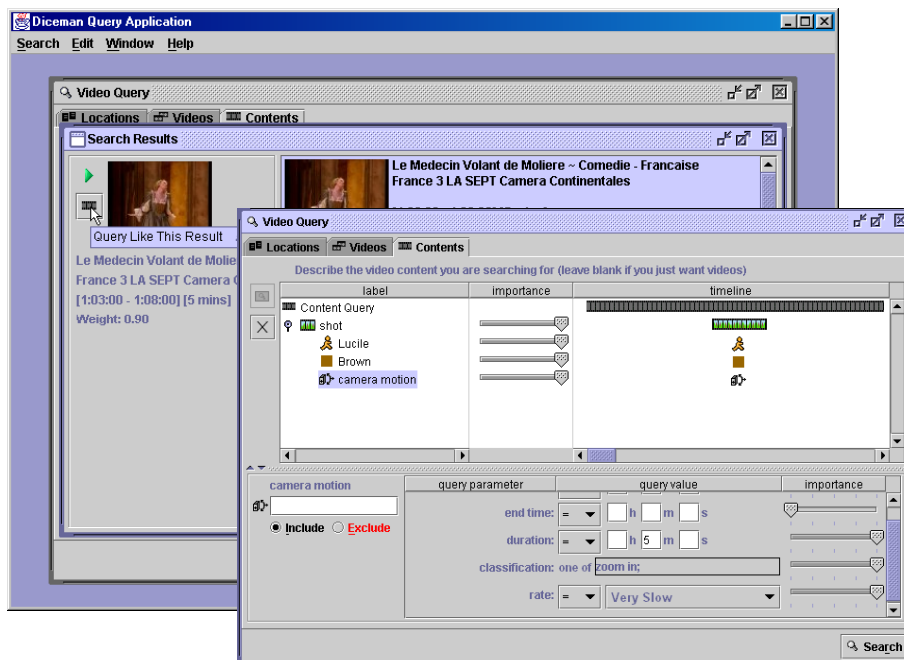


Figure 5: search by similarity

Empirical Strategy

“This strategy represents the use of shortcuts by the skilled librarian. It is based on the librarians’ prototypical classification of users and books. Titles to suggest are selected on the basis of the correlation experienced between user characteristics and typical reading habits. Thus, in addition to considering their expressed wishes, users are classified according to a number of information features (such as visual appearance, verbal style, dress, and age) and books are split up into simple genre classes.”

The empirical strategy is not currently supported by the Diceman system as it is more typical of casual use. However, this is a potential outcome of the agent model of Diceman: Agents could build up models of what individual known users will want and could support stereotypical initial set-ups for new users based on genre classes. Furthermore, user agents could interview new users, in a similar manner to librarians when faced with open information needs, or start by making basic assumptions based on available information (e.g. the user’s host domain, home page contents...). More investigation needs to be done into the potential of supporting the empirical strategy.

Browsing Strategy

“Finally, an information seeker in a library may have a need that is so ambiguous that a specification of a search template is avoided and, instead, the contents of a shelf or a database are scanned to find a match with the intuitively present need.”

Scanning the “shelves” of a large video archive on-line is likely to be very slow, but provider agents could be used to highlight, say, top-20 clips of all time or within predefined sections/genres (somewhat akin to front cover displayed books on library shelves). In addition, knowledge about users and their previous searches could be used to create user-biased hierarchies for browsing (c.f. the work of Harper, Mechkour and Muresan (1999) on clustering for web searching). Again this strategy is more typical of casual users so isn’t supported within the current Diceman interface, more research needs to be done to develop browsing support for large video archives as this is likely to be an important access methods for many users.

Using low-level attributes for high-level searches

At the start of the query interface development project, our intention was to develop a system that would support high level queries that directly support the strategies outlined above. With restrictions on the types of indexing supported, the final interface was, unfortunately, based mostly on camera and frame level information. This is likely to be a common problem with video databases - the cost of indexing semantic level attributes is likely to be prohibitive for most material (at least in terms of detailed, shot-by-shot, indexing). As part of our final evaluation we felt it worthwhile to conduct an initial test of whether users would be able to use these lower level attributes effectively.

In our final round of user testing, users were given a series of short queries to build using the query interface. These typically took the user around 45 minutes and were designed to exercise most areas of the application through targeted questions (e.g. “find shots which begin 15 minutes into a video and end with a fade”). After this set of tasks, users were asked to name a recent film they had seen and then to construct a query to search for their favourite scene within that film. Although the interface wasn’t capable of carrying out searches of, say, *Star Wars Episode I*³, all users felt confident that they had specified enough detail to search for the scene they had in mind. Typically user queries were a combination of character names (including negations to eliminate some scenes), shot structure (length of shot and rough position in the movie) and camera motions (such as long zooms or fast pans). However, some users also felt comfortable using very crude elements, such as dominant colour of a key-frame, to help reduce the search space (for example, desert scenes were given a sandy dominant colour to separate them from internal or forest scenes). While only a very initial study in

³ A commonly quoted movie.

user's ability to search using low level attributes, the results are encouraging, at least for known-item searching, especially if used within an iterative environment.

Implications on matching strategies

The design of an interface to a complex indexing language led to a long discussion of whether the system should support boolean or ranked querying. Boolean queries have in their favour strong user control over the elements - users can develop complex queries and can control precisely what elements are included and excluded from the returned sets. However, even computing science graduates are often not capable of composing all but the simplest boolean expressions correctly. As pointed out by Salton (1984), there are five major obstacles to successful use of boolean querying:

- “The formulation of good boolean queries is an art rather than a science; most untrained users are unable to generate effective query statements without assistance from trained searchers.
- The standard boolean retrieval methodology does not provide any direct control over the size of the output; some query statements may provide no output at all, whereas other statements provide an unmanageably large number of retrieved items.
- The boolean methodology does not provide a ranking of the retrieved items in any order of presumed usefulness, thus all retrieved items are presumed to be equally good, or equally poor, for the user.
- The boolean system does not provide for assignment of weights to the terms attached to documents or queries; thus each term is assumed to be as important as each other assigned term, the only distinction actually made is between terms that are assigned ... and terms that are not assigned... .
- The standard retrieval methodology may produce results which appear to be counter intuitive:
 - in response to an **or**-query ... a record or document with only one query term is assumed to be as important as a document containing all query terms;
 - in response to an **and**-query ... a document containing all but one of the query terms is considered as useless as a document with no query term at all.”

However, for a professional tool and to support detailed query refinement, we felt that more fine grain control was needed than typical of ranking based vector-space/probabilistic models. In line with Salton and the work of Fagin (1998) and Ortega et al (1998) we opted for a model based around the user composing *booleanesque* queries but the system treating these in a ranked fashion. Thus the user can make separations between *and* and *or*, can include *not* expressions but would not be faced with the strictness and unhelpfulness of traditional boolean models and can vary the weight of terms using importance sliders. Furthermore, the removal of strict boolean interpretations gives scope for the consumer agents and provider agents within the Diceman project to negotiate in order to provide the user with documents which are more likely to satisfy his/her needs.

Summary

This paper has presented an interface to a video search engine that is based around the Diceman MPEG-7 enabled architecture. The interface was developed to support three out of the five search strategies identified by Pejtersen, the other two strategies being discussed but not supported. We feel that the interface would allow users to search using different strategies and that an analysis in terms of these strategies was a valuable tool in the development of the system. The development of a search interface for a complex indexing language opens many questions concerning the model of indexing (boolean or ranked) and the ability for users and their agents to refine and easily control their queries and for suppliers' agents to best match users' needs. Further work needs to be done to address these issues for MPEG-7 both for expert and novice users.

Acknowledgements

The work reported here was carried out as part of the Diceman project on “Distributed Internet Content Exchange Using MPEG-7 and Agent Negotiation”. The DICEMAN⁴ project is part-funded by the European Commission under the ACTS RTD programme. The work was partly funded by The Danish National Research Foundation through its Centre for Human Machine Interaction, Teltec Ireland and Dublin City University.

The authors would like to thank Dublin City University’s Department of Computer Applications, Teltec Ireland and Risø National Laboratory for their support in this project - in particular Alan Smeaton, Liam Ward and Annelise Mark Pejtersen respectively. We would also like to thank our test users and Dermont Tynman for his time in discussing the project with us.

References

- A. Akutsu and Y. Tonomura, “Video Tomography: An Efficient Method for Camerawork Extraction and Motion Analysis”, *Proceedings of ACM Multimedia 1994*, pp 349-356, 1994.
- M.G. Brown, J.T. Foote, G.J.F. Jones, K. Sparck Jones and S.J. Young, “Automatic Content-Based Retrieval of Broadcast News”, *Proceedings of ACM Multimedia 95 - Electronic Proceedings (San Francisco, California)*, November 1995.
- R. Fagin, “Fuzzy Queries in Multimedia Database Systems”, *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, Seattle USA, pp. 1-10, ACM Press, June 1998.
- C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, “Efficient and Effective Querying by Image Content”, *Journal of Intelligent Information Systems*, **3**(3/4): 231-262, 1994.
- W.B. Frakes and R. Baeza-Yates (editors), *Information Retrieval: Data Structures and Algorithms*, Prentice Hall, 1992.
- M. Flickner, H.S. Sawhney, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, “Query by Image and Video Content: The QBIC System”, *IEEE Computer*, **28**(9): 23-32, 1995.
- C. Frankel, M. J. Swain, and V. Athitsos, *Webseer: An image search engine for the World Wide Web*, University of Chicago Technical Report 96-14, <http://infolab.ils.nwu.edu/webseer/>, August 1996.
- D. J. Harper, M. Mechkour and G. Muresan, “Document Clustering for Mediated Information Access”, *Proceedings of IRSG99: the 21st BCS-IRSG Annual Colloquium on IR Research*, Glasgow, Scotland, April 1999.
- M. Kaszkiel and J. Zobel, “Passage Retrieval Revisited”, *Proceedings of SIGIR 1997*, pp. 178-185, 1997.
- R. Koenen, S. Jeannin, F. Pereira, I. Sezan, A. Lindsay, F. Nack, S. Paek, and V.V. Vinod, “MPEG-7 context and objectives”, ISO report ISO/ IEC JTC1/ SC29/ WG11/ N2460, <http://drogo.cselt.it/mpeg/standards/mpeg-7/mpeg-7.htm>, Atlantic City, 1998.
- H. Lee, A. Smeaton, and J. Furner, “User Interface Issues for Browsing Digital Video”, *Proceedings of IRSG 99*, Glasgow, 1999.
- S. P. Liou, R. Hjelsvold, R. Depommier, and A. Hsu, “Efficient and Reliable Digital Media Archive for Content-based Retrieval”, *Multimedia Systems*, **7**(4): 256-268, 1999.
- C. O’Toole, A. Smeaton, N. Murphy, and S. Marlow, “Evaluation of automatic shot boundary detection on a large video test suite”, *Proceedings of The Challenge of Image Retrieval 1999*, Newcastle, 1999.
- M. Ortega, Y. Rui, K. Chakrabarti, K. Porkaew, S. Mehrotra, and T.S. Huang, “Supporting Ranked Boolean Similarity Queries in MARS”, *IEEE Transaction on Knowledge and Data Engineering*, **10**(6): 905-925, December 1998.
- A.M. Pejtersen, “Investigation of search strategies in fiction based on an analysis of 134 user-librarian conversations”, *Proceedings of IPFIS3*, Oslo, Norway, pp 107-132, 1979.

⁴ <http://www.teltec.dcu.ie/diceman/>

- J. Rasmussen, A.M. Pejtersen, and L. P. Goodstein, *Cognitive Systems Engineering*, Wiley International, 1994.
- G. Salton, *The Use of Extended Boolean Logic in Information Retrieval*, Cornell University Research Report TR84-588, January 1984.
- T. Sato, T. Kanade, E. K. Hughes, and M. A. Smith, "Video OCR: indexing digital news libraries by recognition of superimposed captions", *Multimedia Systems*, **7**: 385-395, 1999.
- K. Sparck Jones, G.J.F. Jones, J. T. Foote, and S.J. Young, "Experiments in spoken document retrieval", *Information Processing and Management*, **32** (4): 399-419, 1996.
- K. Sparck Jones and P. Willett (editors), *Readings in Information Retrieval*, Morgan Kaufmann, 1997.
- L. Ward, "As MPEG-4 goes to finishing school, MPEG-7 is learning its alphabet", *ACcenTS newsletter* (see also <http://www.teltec.dcu.ie/diceman/>), April 1998.
- M. Wechsler, E. Munteanu, and P. Schäuble, "New Techniques for Open-Vocabulary Spoken Document Retrieval", *Proceedings of SIGIR 1998*, pp 20-27, 1998.
- H.J. Zhang, C.Y. Low, S. W. Smoliar, and D. Zhong, "Video parsing, retrieval and browsing: an integrated and content-based solution", *Proceedings of ACM Multimedia 1995*, pp. 15-24, 1995.