CONTENT-BASED BROWSING IN LARGE NEWS VIDEO DATABASES

Mika Rautiainen, Timo Ojala, Tapio Seppänen
MediaTeam Oulu
P.O.BOX 4500, FIN-90014 University of Oulu
Finland
e-mail: mika.rautiainen@ee.oulu.fi, timo.ojala@ee.oulu.fi, tapio.seppanen@ee.oulu.fi

ABSTRACT
In this paper, we have evaluated the effectiveness of novel content-based browsing paradigm in video retrieval and compared it to the traditional model of content-based querying with relevance feedback. The presented model, cluster-temporal browsing, integrates feature clusters and chronological video structure dynamically in a single browsing view. A prototype of the browser has been evaluated in 70 hour news video collection by 17 test users in 24 predefined and highly semantic search topics. The contributions of this paper are: comparison of cluster-temporal browser against prevalent content-based query paradigm and evaluation of the effect of various browser parameters such as features and interface configurations in search performance. The main conclusions of the paper are that the cluster-temporal browsing surmounts traditional search paradigm in semantic search topics and that low-level visual features do not add to the search performance that is obtained using text search on speech transcripts.

KEY WORDS
content-based retrieval, browsing, indexing, video databases, interaction

1. Introduction

At the moment, two approaches are dominant in video search and browsing systems. The first is favoured by academic community dealing with video retrieval: content-based analysis is utilized to index and search video sequences and to create summarizations of long video sequences. In [1][2][3], some typical content-based systems are described. Such systems work well in constrained domains, but they have not proven very successful in associating features with the user's real information need because the semantic gap between mental and computational world is not yet surmounted by algorithms and systems. Second approach focuses on creating tools that make time-line based navigation in videos more efficient [4][5][7]. Some examples of such tools are fast forward, slide bars and hierarchical browsers [6]. These tools are based on user controlled interaction with the system and they are intuitive, unambiguous and easily adaptable by the users. However, they become time-consuming when size of the video database becomes significantly large. There is a need for another approach that combines the best properties of both. This paper is structured as following: Section 2 describes a dynamic cluster-temporal video browser application that employs content-based features and efficient navigation tools to obtain high search performance in large video databases. Section 3 describes interactive search experiments with the browser and Section 4 gives concluding remarks.

2. Cluster-Temporal Video Browser

2.1 Cluster-Temporal Browsing

The technique in realizing a content-based video browser is based on cluster-temporal browsing model [8]. It aims to reduce the effect of ambiguousness that is typically present in a traditional content-based example search by showing several similarity clusters concurrently.

The novelty of cluster-temporal browsing is in combining both inter-video similarities and local temporal relations of video shots in a single interface. During an interactive search the role of the system is to support the user, providing enough cues and dimensions to navigate through the vast search space towards the relevant objects. The role of this kind of system can be seen as a 'humble servant' as it tries not to constrain the navigation to a limited strategy, whereas typical content-based search systems throw users on mercy of search parameters and features. The name cluster-temporal browsing implies that the content-based feature clusters are not utilized alone, but together with the temporal context of the video.

Figure 1 shows the browsing interface. First row of the key frame images displays a bounded sequence of temporally adjacent shots in a video file that are presented chronologically as a time-line. At any time, user can scroll through the entire video file to get a fast overview of the entire shot sequence. The large panel below the video timeline gives user a view of similar shots from other videos in the database. The view is generated from the results of multiple content-based queries created from the example shots at the top row. The query results are organized into parallel columns to create a similarity matrix. The order of the shots is obtained from the content-based query results; columns are in top-down rank-order. Therefore the most similar shots are organized at the top of the similarity matrix. With the help of the similarity matrix,

a user can instantly see large numbers of additional shots that have content similar to the query video sequence at the top row. The similarity criterion is defined by the user-selected features and their combinations. For the experiments described in this report, visual and text features were made available in the browser.

Using the browser for navigation is straightforward. When an interesting shot is found from the similarity view, the current video sequence on top can be replaced with the source video of the shot of interest. After selecting it, the shot is positioned in the middle of the top row and its chronological neighbours are viewed next to it. System reorganizes the similarity view using the new set of shots as examples. At any time, user can update the similarity matrix by changing to other feature combinations. Each transition caused by browsing the timeline in the current video brings new shots to the top row. Because of the new examples appeared on the screen, the similarity view updates itself immediately. The requirements to update the similarity view are heavy, since the browsing speed should be close to real-time. To update a view, system must perform parallel query processing for several example-based queries. Multi-threaded index queries with efficient query cache provide reasonable access times even for the most complex feature configurations. Caching of results is essential when comparing the search performance of different feature configurations having varying computational complexity.

2.2 The Configurations of the Similarity View

There are two ways to organize result shots in a similarity view. The default layout puts the results of a single search into columns below the top row as shown in Figure 1. An alternative similarity view is illustrated in Figure 2. Here result shots are grouped based on their originating video file. The largest group of results originating from the same video file gets the highest rank. The ranked shot groups are displayed row-wise in the similarity view starting from left to right and from top to down. Each shot group shows a selection of key frames and speech transcripts. The grouping of results is aimed to help in understanding the contextual setting of the results and to give better structured information to the users.

Figure 1. Cluster-temporal browsing interface. Dotted rectangle indicates the similarity view, where content-based result clusters are organized column-wise.

Figure 2. Alternative similarity view: shots are grouped together if they originate from the same video file. The results are displayed using shot key frames. Speech transcripts are visible under the key frame groups.

a) **b)**

Figure 3. (a) Result container stores all the selected relevant shots and suggests more of similar shots based on the selected ones. (b) Fast action buttons are superimposed on each video item when mouse pointer is dragged over them.

2.3 Browser Tools and Navigation Aids

Based on the experiences from the previous experiments with cluster-temporal browser [8], additional features have been implemented to the browser application. First feature is browsing history for reversing navigational steps. It was the most desired add on according to test users' feedback after previous experiments. Fig. 1 shows the browsing history panel at the lower left corner of the interface. It collects the sequence of shots that have been selected for browsing. Second feature is a relevance feedback mechanism that is shown in Figure 3.a. When user finds a shot that is relevant to her search task, she appends it to the result container. System generates new content-based queries from the selected relevant shots and shows the results as a form of relevance feedback below

the relevant shots. Third feature is fast action buttons to speed up the activity selection for the shots of interest. As illustrated in Figure 3.b, the buttons are superimposed on the shot key frames when mouse pointer is dragged over. The buttons create following actions (from left to right): start shot playback, open originating video in browser, display additional information and append shot to the result container.

2.4 Retrieval System, Query Application and Search Parameters

The cluster-temporal browser is a part of a client-server retrieval system that also consists of a server (search engine) and a separate query application for parameterization of single run search. TRECVID reports [9][10] give an overview of the entire retrieval system and the results from the official experiments. A brief description about the search engine and query application follows.

Query application provides tools to define query attributes manually for the search engine. User can select any combination of example images or shots for visual search and select search parameters individually for each example. Textual query terms are created by typing words to a text box. After user has parameterized the search it is submitted to the server. Server distributes the query definitions to the respective search engines. When the result shots for the query have arrived, user can select any interesting shot from the result set as a start point for browsing with the browsing interface.

The search engine supports three different levels of search features: text, visual and concept search. The experiments in this paper use visual and text features and the concept features were excluded. The fusion of features is realized as a linear weighted combination of ranked similarities [11].

Visual search features are constructed from the two physical properties of a shot: (I) Color is the most widely used content feature in content-based retrieval research. Similarity by color gives initially very good perceptual correspondence between two color images that are small or short of details. After visual content is reviewed in detail, other properties for perceptual similarity emerge. (II) Structure of the edges in a visual imagery is a strong cue for many computer vision applications, such as classification of city and landscape images or segregating natural and non-natural objects. This can also provide invaluable in queries where statistical color information is insufficient in describing the main properties of an image. Following features have been used in our experiments: Temporal Color Correlogram (TCC) and Temporal Gradient Correlogram (TGC). These features are computed from a sequence of shot frames. More detailed description can be found from [11].

Text search is based on automatic speech recognition (ASR) and closed caption (CC) transcripts. For the test database used in these experiments, ASR text was produced at LIMSI [12] by converting spoken audio to text automatically. Stemming and stop word lists were used to preprocess the transcripts. To compute the matching score between database shots and a query term, prioritised ranking combined with weighed term frequency score was used. More details about text search are found in [10].

3. Retrieval Experiments with TRECVID 2004 News Video Database

The experiments with the cluster-temporal browser are based on the TRECVID 2004 benchmark [13]. National Institute of Standards and Technology (NIST) provided video test database and common segmentation for it (created by CLIPS-IMAG). The test database was about 70 hours of ABC and CNN news from the year 1998 consisting of 33367 shots. NIST also provided 24 semantic search topics that were used in the experiments. A topic contained one or more example clips of video or images and textual topic description to aid the search process.

This paper describes interactive search experiments that focus on testing the performance of cluster-temporal browser in highly semantic search tasks. Two experiments have been conducted. First focuses on testing different browser configurations in order to measure the significance of visual features in semantic search. Second test contrast traditional query and relevance feedback paradigm against cluster-temporal browsing to find out to what extent browser can improve the semantic search performance.

3.1 Experiments with Cluster-Temporal Browser Configurations

In this experiment, search engine was configured to use visual and text search with equal weights. The interactive experiment was carried out by a group of 12 test users, from which four users had prior experience on searching with the system. Novice test users were mainly information engineering undergraduate students having good skills in using computers, but had little experience in searching video databases. Experienced users had used cluster-temporal browser in previous experiments that were held a year before. They had about two hours of use experience before arriving to test. Only one of the test people was a native English speaker. All of the test users were used to searching the web. 12 users, 24 search topics and two variants of browser configurations were divided into following six search runs:

systemID: searcherID [starttopicID-endtopicID]

I1T: S1[125-130],S7[131-136],S2[137-142],S8[143-148]
I2VT: S2[125-130],S8[131-136],S1[137-142],S7[143-148]
I3T : S3[125-130],S5[131-136],S6[137-142],S4[143-148]
I4VT: S4[125-130],S6[131-136],S5[137-142],S3[143-148]
I5T: S10[125-130],S9[131-136],S11[137-142],S12[143-148]
I6VT: S9[125-130],S10[131-136],S12[137-142],S11[143-148]

System variant T disabled the visual search feature so that the browsing was entirely based on text search in speech transcripts. System variant VT combined visual search feature with text search. Each user did first six topics with one system configuration and then another six topics using another configuration. Half of the users used configuration T before VT, another half did the opposite. Shown configuration reduced the effect of learning and bias between the system variants. The effect of fatigue was alleviated with break and refreshments between system configuration change. The effect of learning within the topic groups (starttopicID-endtopicID) was not controlled, most of the users processed the topics in numerical order. All users were given half an hour introduction to the system, with a couple of example searches. Users were told to use 12 minutes for searching a topic, during which they selected shots that seemed to fit to the given topic description. The final result sets of 1000 shots for evaluation were created using selected results as examples to retrieve more shots for the result set. Total duration of the experiment was about three hours. Users also filled up questionnaires about their experiences. The test PCs were 0,8-2GHz PCs with Windows 2000/XP operating system installed.

Table 1. Search results for browser configurations

Search Run ID	MAP	# Relevant Returned
I1T (novice users)	0.210	726
I2VT (novice users)	0.179	678
I3T (experienced users)	0.212	767
I4VT (experienced users)	0.212	776
I5T (novice users)	0.212	723
I6VT (novice users)	0.201	721
Median (TRECVID 2004)	0.181	497
Max (TRECVID 2004)	0.337	980

Table 1 shows the mean average precision (MAP), which is a mean value of the average precisions from every search tasks [13], and total number of relevant shots returned. Also median and maximum performance over TRECVID 2004 interactive experiments are given to put the obtained results into perspective. The results indicate that there is no significant performance difference between the two browser configurations. However, the effect of experience is visible in the number of relevant shots returned: experienced users have returned on average 8% more relevant shots than novice users.

During the experiment, users had the possibility to switch between different similarity views (see Figures 1 and 2) and the use times for each were measured to estimate the user preference. The view that grouped the result shots in videos (Figure 2) was used only 436 minutes in total whereas the similarity view without grouping (Figure 1) was used 1069 minutes during the experiments. The elapsed times show that the user preference was towards ungrouped similarity view.

3.2 Experiments with Browser vs. Query with Relevance Feedback

The second experiment focused on the differences between cluster-temporal browser and traditional content-based search paradigm: query with relevance feedback. The experiment was carried out by a group of 5 test users. All users were novice with the search system but having good skills in using computers and only little experience in searching video databases. None of the test participants was a native English speaker. All of the participants were used to searching the web. 5 users, 24 search topics and two variants of browser configurations were divided into following two search runs:

systemID: searcherID [starttopicID-endtopicID]

I1Q: S1[125-130],S3[131-136],S2[137-142],S4[143-145],S5[146-148]
I2B: S2[125-130],S4[131-133],S5[134-136],S1[137-142],S3[143-148]

In system variant Q users were only allowed to use query application, where they had to manually select search parameters to generate results for each query. The users did not use cluster-temporal browser to navigate in the database. In addition to query application, users had access to relevance feedback that was built into result container as was described in Section 2.3. System variant B allowed users to utilize cluster-temporal browser during the search. The search time was limited to maximum of 12 minutes. During that time users were supposed to use the given system configuration the best way they could in order to find results for the given tasks.

The test conditions during the second experiment followed closely the first experiment. The second experiment was conducted after the official TRECVID 2004 experiments using ground truth information that was made available by NIST.

Table 2. Search results for the two system configurations

Search Run ID	MAP	# Relevant Returned
I1Q (novice users)	0.165	650
I2B (novice users)	0.202	681

Table 2 shows that the use of cluster-temporal browser in search results in 22% improvement in mean average precision over traditional query paradigm with relevance feedback mechanism. The difference is large enough to prove that cluster-temporal browsing is very capable of improving traditional content-based retrieval.

4. Conclusions

This paper described a dynamic interaction technique for content-based video retrieval. Cluster-temporal browser combines efficiently information from temporal video

structure and content-based feature clusters into a single view. It helps users to browse through large video collections and find shots that are relevant to their search task at hand. Extensive semantic search experiments have been conducted with a total of 17 test users in a large, 70 hour video collection. The experiments demonstrate that the cluster-temporal browsing improves search precision by 22% over traditional content-based query with relevance feedback paradigm. Using text and visual features combined in similarity view is equally efficient to using only text features. One cause for this is the given semantic search topics that require high level semantic meaning from the computed features. However, content-based visual similarity is based on low-level features that do not contribute to the search topics as much as speech transcripts. Users were given a possibility to change the organization of the result shots in the browser's similarity view. According to use statistics, users preferred ungrouped similarity view where the shots were organized column-wise.

In the future, higher level features will be employed to improve semantic search performance. High level features are lexical concepts that are detectable from video data using pattern recognition and classification techniques. Several concept detectors form a vocabulary that describes the content. By combining them with the speech transcript based features, a higher level similarity criterion can be used to supply more meaningful results through the browser.

5. Acknowledgements

We would like to thank the National Technology Agency of Finland (Tekes), Academy of Finland and Nokia Foundation for supporting this research.

References:

[1] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele & P. Yanker, Query by image and video content: The QBIC system, *IEEE Computer*, *vol. 38*, 1995, 23–31.

[2] A. K. Jain, A. Vailaya, & X. Wei, Query by video clip, *ACM Multimedia Syst.*, *vol. 7*, 1999, 369–384.

[3] A. Humrapur, A. Gupta, B. Horowitz, C. F. Shu, C. Fuller, J. Bach, M. Gorkani & R. Jain, Virage video engine, *SPIE Proc. Storage and Retrieval for Image and Video Databases V*, San Jose, CA, 1997, 188–197.

[4] J.-Y. Chen, C. Taskiran, A. Albiol, E. J. Delp & C. A. Bouman, ViBE: A compressed video database structured for active browsing and search, *Proc. SPIE: Multimedia Storage and Archiving Systems IV*, *vol. 3846*, Boston, MA, 1999, 148–164.

[5] J. R. Smith, VideoZoom spatial-temporal video browsing, *IEEE Trans. Multimedia*, *vol. 1*, 1999, 151–171.

[6] X. Zhu, J. Fan, A. K. Elmagarmid & W. G. Aref, Hierarchical video summarization for medical data, *Proc. SPIE: Storage and Retrieval for Media Databases*, *Vol. 4676,* San Jose, CA, 2002, 395-406.

[7] K. Wittenburg, J. Nicol, J. Paschetto & C. Martin, Browsing with dynamic key frame collages in Web-based entertainment video services, *Proc. IEEE International Conference on Multimedia Computing and Systems*, *vol. 2* 1999, 913 – 918.

[8] M. Rautiainen, T. Ojala, T. Seppänen, Cluster-temporal browsing of large news video databases, *Proc. IEEE International Conference on Multimedia and Expo*, *Vol. 2,* Taipei, Taiwan, 2004, 751 – 754.

[9] M. Rautiainen, J. Penttilä, P. Pietarila, K. Noponen, M. Hosio, T. Koskela, S.M. Mäkelä, J. Peltola, J. Liu, T. Ojala & T. Seppänen, TRECVID 2003 experiments at MediaTeam Oulu and VTT, *TRECVID Workshop at Text Retrieval Conference TREC-2003*, Gaithersburg, MD, 2003.

[10] M. Rautiainen, M. Hosio, I. Hanski, M. Varanka, J. Kortelainen, T. Ojala & T. Seppänen, TRECVID 2004 experiments at MediaTeam Oulu, *TRECVID Workshop at Text Retrieval Conference TREC-2004*, Gaithersburg, MD, 2004.

[11] M. Rautiainen, T. Ojala & T. Seppänen, Analysing the performance of visual, concept and text features in content-based video retrieval, *Proc. 6th ACM SIGMM International Workshop on Multimedia Information Retrieval,* New York, NY, 197-205.

[12] J.L Gauvain, L. Lamel & G. Adda, The LIMSI Broadcast News Transcription System, *Speech Communication, 37*(1-2), 2002, 89-108.

[13] TREC Video Retrieval Evaluation. http://www-nlpir.nist.gov/projects/trecvid/ (30.5.2005)