Interactive Search in Large Video Collections

Andreas Girgensohn, John Adcock, Matthew Cooper, and Lynn Wilcox
FX Palo Alto Laboratory
3400 Hillview Avenue, Bldg. 4
Palo Alto, CA 94304, USA
{andreasg, adcock, cooper, wilcox}@fxpal.com

Abstract

We present a search interface for large video collections with time-aligned text transcripts. The system is designed for users such as intelligence analysts that need to quickly find video clips relevant to a topic expressed in text and images. A key component of the system is a powerful and flexible user interface that incorporates dynamic visualizations of the underlying multimedia objects. The interface displays search results in ranked sets of story keyframe collages, and lets users explore the shots in a story. By adapting the keyframe collages based on query relevance and indicating which portions of the video have already been explored, we enable users to quickly find relevant sections. We tested our system as part of the NIST TRECVID interactive search evaluation, and found that our user interface enabled users to find more relevant results within the allotted time than those of many systems employing more sophisticated analysis techniques.

Categories & Subject Descriptors: H.5.1 [**Information interfaces and presentation**]: Multimedia information systems – *video*.

General Terms: Algorithms; Design; Human Factors.

Keywords: Video search, keyframe collages, text analysis.

INTRODUCTION

While searching text documents is a well-studied process, it is less clear how to best support search in video collections. Typically text documents can be treated as units for the purpose of retrieval. However, treating whole videos as units will often not lead to satisfactory results. This is true in the case of news videos where a 30-minute news program is broken up into stories of one or two minutes in length.

Our approach to this problem is to support users in rapidly searching through such video collections. Our target users are analysts who want to combine information from several sources or video producers who want to locate video for reuse. While the latter will frequently use libraries with extensive meta-data to support retrieval, our goal is to support the search in video collections where such meta-data is not available. We assume that time-aligned text, such as transcripts, automatically recognized speech, or closed captions, is available.

To validate our approach, we participated in this year's interactive search component of a video retrieval evaluation

CHI 2005, April 2–7, 2005, Portland, Oregon, USA.
ACM 1-59593-002-7/05/0004.

Figure 1: The interactive search interface. (A) Story keyframe summaries in the search results (B) Search text and image entry (C) TRECVID topic display (D) Media player and keyframe zoom (E) Story timeline (F) Shot keyframes (G) Relevant shot list

called TRECVID sponsored by the National Institute of Standards and Technology (NIST) [9]. In the interactive search, participants have access to four months worth of broadcast news video from the U.S. ABC and CNN networks (about 60 hours). Participants are asked to answer questions such as "find shots of Bill Clinton speaking with at least part of a US flag visible behind him." Some of the TRECVID participants use very elaborate video analysis techniques to support the search [4]. For example, one very successful system allows the user to search for visual features such as animals, buildings, or people [2].

Our system design philosophy is to automate parts of the system but to let the users directly perform tasks that they can do better. For this application, the system and the users collaborate to improve the information retrieval precision and recall. Precision is the fraction of relevant retrieved documents among all documents retrieved and recall is the fraction of relevant retrieved documents among all possible relevant documents. Our system works to maximize the information retrieval recall without compromising precision. The users are mostly responsible for the precision by browsing through the visually presented candidates and by selecting the truly relevant ones. The system performs a second automation step after the interactive session to supplement the user-selected shots with additional search results that are deemed similar.

Figure 2: Story keyframe montage example. The keyframe montage at the left is constructed from the 9 shot keyframes of the story at the right selected and cropped based on their relevance to the query "Sam Donaldson".

Our system pre-processes the text transcript to segment each half hour video into smaller semantically-related units (stories) that are of a length better suited for standard text retrieval techniques and visual presentation and summarization (see Figure 1). We also provide limited support for visual search to deal with situations where the visual information is more important than the associated text (e.g., to find sunsets).

In the next section, we describe the search user interface. We then present the components of the back-end system. Finally, we present the results of the TRECVID evaluation and conclude with a discussion of implications.

USER INTERFACE

Searches in video collections can return a large number of relevant results. It is important to present those results in a form that enables users to quickly decide which of the results best satisfy the user's original information need. Our system displays search results in a form that makes it easy for users to determine which results are truly relevant. The system makes use of a prior segmentation of each video into shots and stories. Video shots are uninterrupted sequences of strong visual coherence, generally taken by a single camera without turning it off. Stories are semantically related groups of shots, where the semantic information comes from a time-aligned text transcript.

Our interactive search system, pictured in Figure 1, has several novel features. First, our system provides several relevance-based visualizations such as listing search results in relevance order with image sizes also determined by relevance, a timeline with color-coding, and colored bars for keyframes. Second, we visualize stories as keyframe collages where keyframes are selected and sized by their relevance to a query so that the same story would be shown differently for different queries. Third, within a topic session, we mark visited stories so that the user can avoid needless revisiting of stories. We present the three types of UI elements that we developed to surface the novel features.

Shot and Story Visualization

Results for a query are returned as a set of stories, sorted by query relevance. Each story is represented as a collage of keyframes from the video shots contained in the story. The selected keyframes and their sizes depend on the relevance of the corresponding video shots to the query. The most-relevant shots are selected and their keyframes are combined to form a story keyframe-collage. The size allotted to each portion in this 4-image montage is determined by the shot's score relative to the query. Figure 2 shows an example of this where the query was "Sam Donaldson" and the shots most relevant to the query are allocated more room in the story thumbnail, in this case the 2 shots of the 9 total shots in the story that depict Sam Donaldson. Rather than scaling down the keyframes, they are cropped to preserve details in reduced size representations. The keyframes for the shots comprising a story can be seen expanded in a separate pane by selecting the keyframe-collage for a story (see Figure 1F).

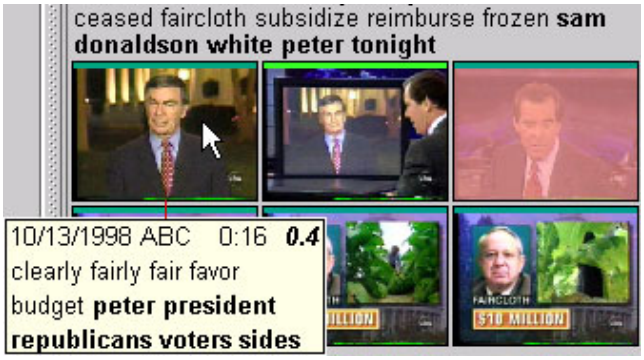

Figure 3: Tool tip showing distinguishing keywords and bold query keywords.

Tool Tips

It is useful to provide feedback to the user to indicate why a particular document was deemed relevant to the query and how the document is different from other documents. A tool tip for story and video shot keyframes provides that information to the user in the form of keywords distinctive for the story and keywords related to the query (see Figure 3). The terms with the highest tf-idf scores (term frequency * inverse document frequency [8]) distinguish the shot or story best from all other shots or stories in the collection. This favors the terms that frequently appear in the shot but appear only in a few other shots. The terms in bold are most related to the query. With a standard tf-idf retrieval approach, those would be the query terms appearing in the story. When latent semantic analysis (LSA) search is used, a relevant document may not contain any of the words used in the query. In this case we use the latent semantic space to identify terms in the document that are most similar to the query.

Overlays

Semi-transparent overlays are used to provide three cues. A gray overlay on a story icon indicates that it has been previously visited (see Figure 1A and E). A red overlay on a shot icon indicates that it has been explicitly excluded from the relevant shot set (see Figure 1F). A green overlay on a shot icon indicates that it has been included in the results set (see Figure 1F). For color blind users, we also have the option to replace the translucent overlays with different patterns superimposed on the keyframes.

Horizontal colored bars are used along the top of stories and shots to indicate the degree of query-relevance, varying from black to bright green. The same color scheme is used in the timeline depicted in Figure 1E.

BACK END

We pre-process videos to segment them into stories with a text-based latent semantic analysis (LSA) of the transcripts. For searching, we give users the choice among literal keyword text search, LSA-based text search, and visual similarity search. At the completion of a topic, the system automatically finds additional relevant video shots.

Data Pre-processing

As the lowest-level unit, we use video shots that are provided as a reference by TRECVID. We perform an automatic pre-processing step to identify topic or story units from the automatically recognized speech. We build a latent semantic space (LSS) [1] treating the stopped and stemmed [7] text tokens for each video shot in the testing corpus as a separate document. We then project the text for each shot into this shot-based LSS and compute a similarity matrix for the shots in a video using cosine similarity. A checkerboard kernel is passed over the similarity matrix and points of highest novelty are chosen as story boundaries, as in [3]. After determining story boundaries, we create a new LSS from the stories to be used for retrieval. We also use Lucene [6] to create literal text search indices for both stories and video shots. Finally, we compute color correlograms for the keyframes of video shots to be able to support visual search.

Search Engine

Queries are specified by a combination of text and images. The searcher can opt to perform a text-only or image-only search by leaving the image or text query area empty. For the text portion of the query, the searcher can choose between a literal keyword text search (weighted by tf-idf) and a LSA-based text search. When determining text-query relevance for shots, the shots inherit part of the retrieval score of their parent stories to properly handle terms that co-occur in the same story but in different shots.

An image similarity matching capability is provided based on color correlograms [5]. Correlogram provide signatures for color groupings in images and tend to produce better image search results than color histograms or similar measures. To generate an image-similarity relevance score at the story level, the maximum score from the component shots is propagated to the story.

Post Query Processing

After the searcher finishes searching for video shots relevant to a topic, the system attempts to find additional relevant shots. We use two strategies to select additional shots. First, we address the fact that shots are sometimes segmented at the wrong place by adding all shots bracketing the shots selected by the user. Second, we issue additional queries to find shots similar to the ones the user selected. We use three variants for the second strategy. The first variant *(WEIGHTED)* uses the weighted average of the scores of all queries issued by the user to compute a new score for every video shot in the collection. The second variant *(LSA1)* combines the text from all user-selected shots to form a single LSA query and we add the best results from that query. The third variant *(LSA2)* uses the text from every user-selected shot to form a separate LSA query and combines the query results as in the WEIGHTED method.

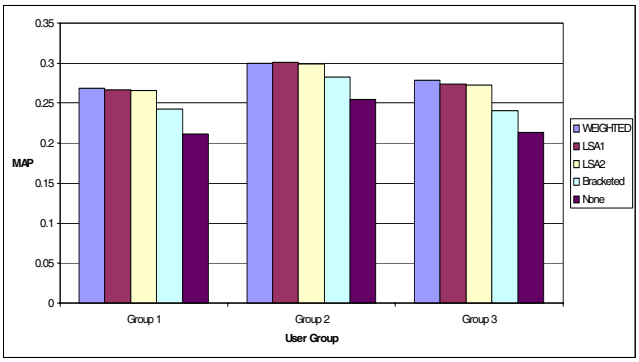

Figure 4: Overall mean average precision (MAP) performance by user group and post-processing system type employed. The "None" column is the MAP performance of the user selected shots without any automatic augmentation

TESTS AND RESULTS

The TRECVID evaluation consists of 24 topics (one of which had no relevant shots in the test set and was discounted). 15 minutes are allowed for answering each topic. Because answering all topics would take 6 hours, most TRECVID participants assign subsets of topics to individual searchers. We employed 6 searchers (5 male; 1 female) to each answer 12 topics. All searchers have experience with video processing but most of them had not used the user interface before their 30-minute training session with a different news video collection. None of the searchers had seen the test collection or the topics before the search session. We grouped the topics into quarters and assigned them to the searchers in a standard latin square arrangement such that every searcher had a different combination of quarters. We then grouped searchers who had answered complementary sets of topics to create 3 groups of 2 searchers.

Measuring task completion times does not provide a good indication of performance because the searchers used the maximum allotted time of 15 minutes for almost all topics. TRECVID evaluates search results by computing the average precision (AP) for each topic. This is the average of the precision values obtained after each relevant shot is retrieved. Relevant shots that are not retrieved are assumed to have a precision of 0. The mean over all topics (mean average precision; MAP) is used to compare results.

Figure 4 shows the MAP results for the three groups of searchers. In addition to the results for the user-selected shots, the figure also shows the results to the three post-processing strategies described in the previous section. The post-processing strategies have similar performance (*WEIGHTED* is best and *LSA2* worst) and increase the MAP by 0.054 on average. While there are large differences in performance among the groups of searchers, those differences are fairly small compared to the differences among the results submitted by other systems.

Figure 5 shows our median and maximum AP by topic along with the overall maximum and median scores taken from all groups. With a few notable exceptions, our median tracks a path between the overall median and maximum, and our maximum achieves maximum or near-maximum overall

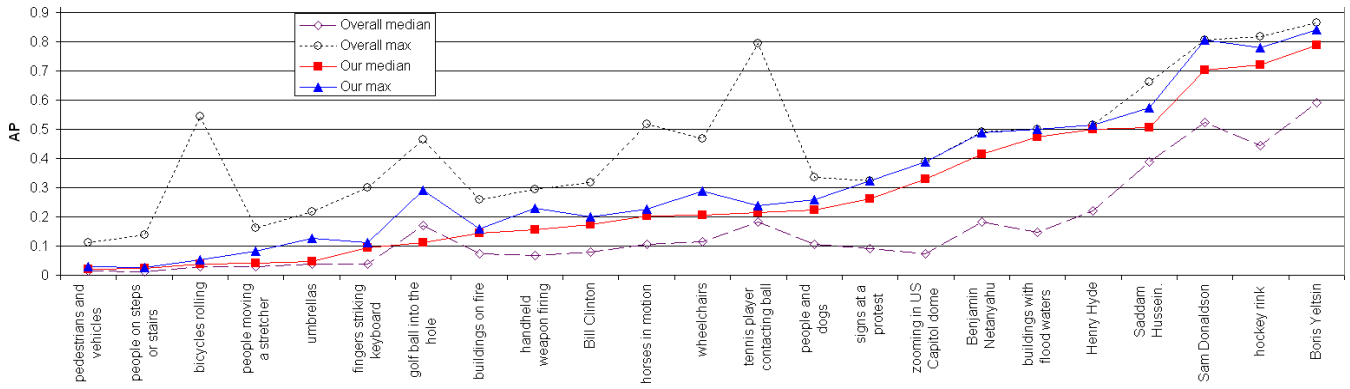

Figure 5: Average precision (AP) by topic with median and maximum. Sorted by our median performance.

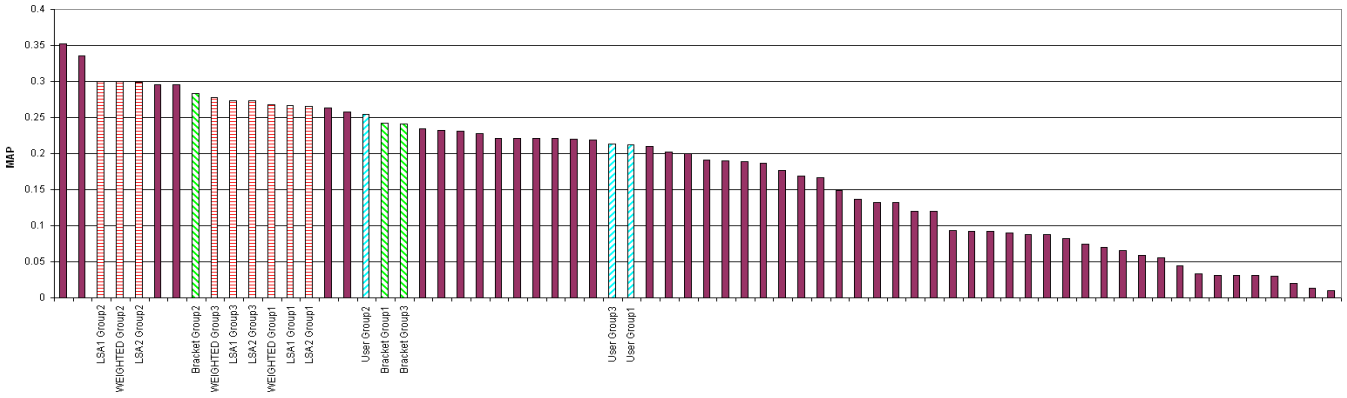

Figure 6: Interactive search mean average precision (MAP) scores for the entire set of submissions. Our scores with and without automatic post-processing are shown as striped bars. Other groups' submissions as solid bars.

scores for several topics. Figure 6 shows the MAP performance of our system with different post-processing strategies compared to all TRECVID submissions.

Our best submission placed 3rd overall and only 4 submissions from 3 groups performed better than our worst performing submission [9]. Those 3 groups (University of Amsterdam/MediaMill, CMU, and IBM) have very mature image retrieval efforts and employ very sophisticated semantic image processing and feature detection. For example, the top-scoring MediaMill system uses a semantic lexicon with 32 concepts such as aircraft, bicycle, or Bill Clinton. This allows them to do well in TRECVID 2004 topics such as *"find shots of one or more bicycles rolling along."*

CONCLUSIONS

We presented our approach to supporting users in searching video collections. Our system directed the users' attention to promising results and let them judge the relevance. We provided an efficient user interface that enables users to quickly browse retrieved video shots. We used text-based story segmentation and adapted the visualization of stories to the relevance with respect to the current query. Even without the use of elaborate media analysis techniques, this approach was competitive with groups employing very mature image understanding systems.

REFERENCES

1. M.W. Berry, S.T. Dumais, G.W. O'Brien, Using Linear Algebra for Intelligent Information Retrieval, *SIAM Review*, 37(4), p. 573-595, 1995.
2. M. Christel and N. Moraveji. Finding the Right Shots: Assessing Usability and Performance of a Digital Video Library Interface. *Proc. ACM Multimedia*, pp. 732-739, 2004.
3. M. Cooper and J. Foote. Scene Boundary Detection Via Video Self-Similarity Analysis. *Proc. IEEE Intl. Conf. on Image Processing*, pp. 378-381, 2001.
4. A.G. Hauptmann and M.G. Christel. Successful Approaches in the TREC Video Retrieval Evaluations. *Proc. ACM Multimedia*, pp. 668-675, 2004.
5. J. Huang, S.R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih. Image Indexing Using Color Correlograms. *Proc. IEEE Comp. Soc. Conf. Comp. Vis. and Patt. Rec.*, pp. 762-768, 1997.
6. Jakarta Lucene. http://jakarta.apache.org/lucene/
7. M.F. Porter. An Algorithm for Suffix Stripping. *Program*, 14(3), pp. 130-137, 1980.
8. G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5), pp. 513-523, 1988.
9. TRECVID. http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html