ACCESSING VIDEO ARCHIVES USING INTERACTIVE SEARCH

M. Worring[1]*, G.P. Nguyen*[1]*, L. Hollink*[2]*, J.C. van Gemert*[1]*, D.C. Koelma*[1]

[1]Mediamill/University of Amsterdam
worring@science.uva.nl, http://www.science.uva.nl/˜worring
[2]Department of Business Informatics, Free University Amsterdam

ABSTRACT

In this presentation we present a system for interactive search in video archives. In our view interactive search is a four-step process composed of indexing, filtering, browsing, and ranking. We have experimentally verified, using 22 groups of two participants each, how users apply these steps in the interactive search and how well they perform.

1. INTRODUCTION

Video archives are typically queried for shots of a specific person or object involved in some specified event like the famous encounter of Lewinsky and Clinton, or for a generic setting, person, or object e.g. shots of people skating on a lake. These queries are at the conceptual level and cannot be specified easily using low-level visual information or keywords. Hence, the primary entry point to a video archive should be a collection of high-level concepts.

Due to the semantic gap [4], the collection of high-level concepts for which indices can be successfully derived is limited. The high-level *indexing* yields a broad categorization of the information space only. At query time these categorizations can be used for *filtering* the dataset to obtain a limited set of candidate relevant shots. Within one of the induced subclasses of the information space one can only use low-level indices. End-users of a search system, however, cannot be expected to define their query need in terms of such low-level indices. Therefore, image based systems often rely on query-by-example [4] to specify the query need and then use *ranking* based on visual similarity to find the result. This is effective if one has examples to start with, often this is not the case. Therefore, to specify the query, users should select appropriate examples from the dataset itself by *browsing* through the dataset.

On the basis of the above observations, we decompose the interactive search task into four steps. The first step, the indexing, is only performed once, the other three namely filtering, browsing, and ranking can be used in an iterative fashion in the interactive search task. The whole process is illustrated in figure 1. In the following sections we describe the different processing steps and report on an experiment conducted within the context of the TRECVID benchmark [3] to verify the quality of the proposed system.

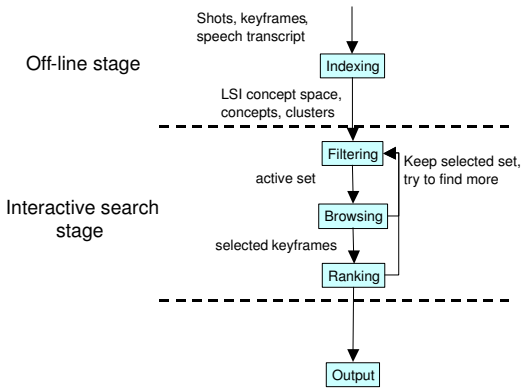

Fig. 1. *Overview of the different processing steps and their relations.*

2. THE PROCESSING STEPS

A video document is a combination of a synchronized auditory and visual stream where the visual stream is decomposed into shots. In our system shots are the basis for retrieval. The speech is transcribed automatically [1] and aligned with the shots so every shot is associated with a specific part of the transcription. Finally, every shot is represented by one or more keyframes. The shots and keyframes are all provided by TRECVID. We now describe in more detail the four processing steps that are performed in searching for specific shots.

2.1. Indexing

The aim in the indexing step is to provide users with a set of high-level entry points into the dataset.

We use a set of 17 specific concept detectors developed by CMU [2] for the TRECVID, ranging from generic ones like female speech to specific ones like car/truck/bus. These detectors have been developed in different ways. Some are

truly multimodal e.g. the NewsSubjectMonologue detector, others are using only one of the modalities. The quality of the result also varies for the different concepts. These concepts form one high-level entry point.

We augment the high-level concepts by deriving textual concepts from the speech recognition result using Latent Semantic Indexing (LSI) . To that end, we construct a vector space by taking all words found. We then perform stopword removal using the SMART's english stoplist. This results in a 18.117 dimensional vector space. The vector space is then reduced to 400 dimensions using Principal Component Analysis. Thus, we effectively decompose the information space into a small set of broad concepts, where the selection of one word from the concept reveals the complete set of associated words also.

For all keyframes in the dataset we also perform low-level indexing by computing the global *Lab* color histograms using 32 bins for each channel. To structure these low-level visual descriptions of the dataset, the whole dataset is clustered using k-means clustering with random initialization. The k in the algorithm is set to 143 as this is the number of images our display will show to the user (11 rows of 13 keyframes).

2.2. Filtering

The indexing step is performed in the off-line stage. The first step in the interactive search stage is filtering of the dataset to retrieve an active set of limited size containing shots that conform to a set of user selected high-level concepts.

To select the active set, the user takes a combination of the following three query specifications:

- *High-level concept:* which can be used as a positive filter (concept should be present) or as a negative filter (should not be present).
- *Textual Concept:* specified by a user defined word and automatically related to all associated words found in the LSI space.
- *Keyword:* which must match a word in the text associated with a shot exactly.

The first two query classes yield a ranking. So we restrict the active set to contain 2000 shots maximum, leading to approximately 4000 keyframes. Users can combine the query mechanism using an *and* function (but this usually leads to very small sets) or the ranked result is an alternation between the results obtained for the selected query specification mechanisms.

2.3. Browsing

At this step in the process we assume that the user is going to select examples from within the active set. As the filtering is already based on high-level concepts the browsing step relies on low-level descriptions. In particular, search is based on the *Lab* histograms of the keyframes of the shots, where similarity of two keyframes is defined by the Euclidean distance of the two histograms. Users should select examples based on these histograms and distances.

Browsing requires a visualization mechanism that on the one hand provides an *overview* of the dataset, while showing sufficient *detail* on the other. Furthermore, the visualization should give the user an insight in the *structure* of the dataset.

To give the user an overview of the data the user can decide to look at the clustered data, rather than the whole dataset. In this visualization mode, the center of each cluster for which some element is present in the active set is presented on the screen. Showing the structure of the information space is complicated as we have to make a 2-dimensional display. The keyframes, however, are embedded in the high-dimensional space induced by the 96 dimensions of the *Lab* histogram. It is the structure in this high-dimensional space that we want to visualize.

A well known technique for this purpose is multi dimensional scaling (MDS). This is a visualization techniques which displays points embedded in a high dimensional space onto the screen in such a way that the distances between points are preserved as good as possible.

MDS fails when the data is clustered in non-elliptical shapes e.g. when the cluster is in the form of a spiral. To solve this problem one can use the ISOMAP algorithm [5]. This graph based technique first constructs the nearest neighbor graph in the high-dimensional space. Distance between two points is then redefined as the distance of the shortest path between the points in the graph. This distance matrix is than the input to the MDS algorithm as described above [5]. We use the above method for visualizing the set of keyframes, where a point corresponds to the center point of the keyframe.

In addition to the overview and structure based overview, users can see details by inspecting specific keyframes, read the associated text, and see the keyframes of the shot and/or surrounding shots[1]. On the basis of this inspection the user performs a *selection operation* to get a set of one or more example images.

2.4. Ranking

When the user has selected a set of suitable images, the user can perform a ranking through query by example us-

[1] Our system actually can also play the shot as a clip, but this was not used in the experiments as with limited search time too much time would be spend on viewing the video clips.

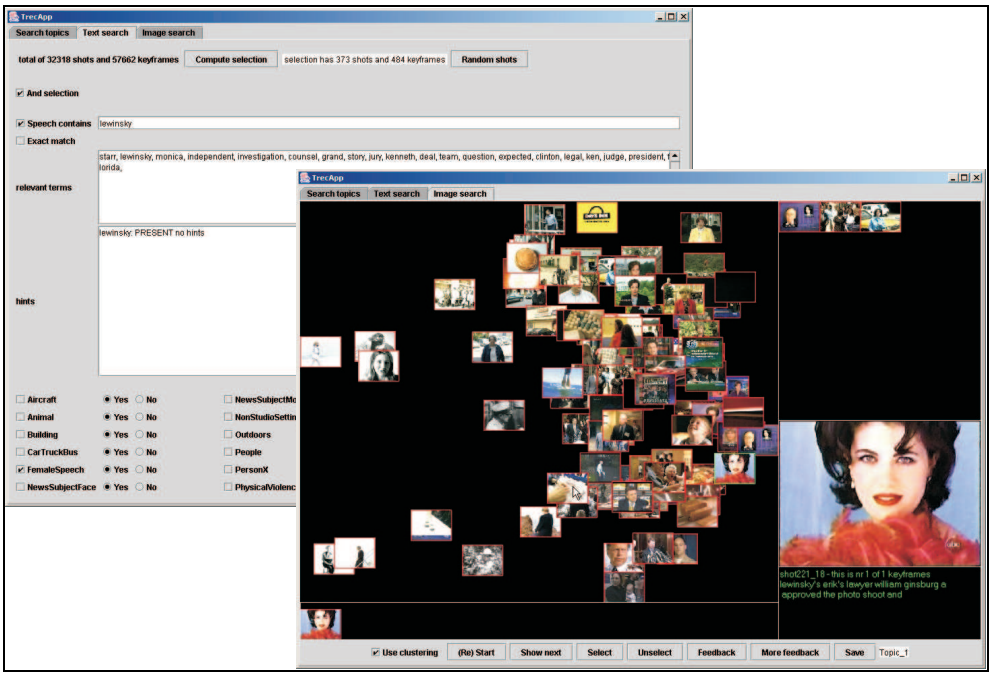

Fig. 2. *Screendump of the GUIs used for filtering (topleft) and browsing/ranking.*

ing again the *Lab* histograms with Euclidean distance. In the result the closest matches within the filtered set of 2000 shots are computed, where the system alternates between the different examples selected. An illustration of the user interface used is presented in figure 2.

3. EXPERIMENTAL SETUP

3.1. Search protocol

For the TRECVID 24 topics have to be found in a dataset consisting of 60 hours of video from ABC, CNN, and C-SPAN. For the experiments total search is limited to 15 minutes per topic to assure that e.g. sequential scanning of all candidates is not an option. The following is the list of topics, ordered according to their general class:

- *General setting*: a crowd in urban environment (1), aerial view of buildings (2), road with vehicles (3), snow-covered mountains (4), flames (5).
- *Specific object and/or events*: the mercedes logo (6), the white house (7), tomb of the unknown soldier at Arlington (8), the sphinx (9).
- *Generic objects and/or events*: airplane taking off (10), tank (1), cup of coffee (12), locomotive approaching you (13), basketball passing down a hoop (14), cats (15), helicopter (16), rocket taking off (17).
- *Specific persons*: Pope John Paul II (18), Yassar Arafat (19), Morgan Freeman (20), Osama bin Laden (21), Mark Souder (22).
- *Generic people and/or events*: person diving into water (23), view from behind cather while pitcher is throwing the ball (24).

The system has been evaluated by having 44 students perform the interactive search task in groups of two. Before the start of the experiment they were asked to fill in a questionnaire about their prior experiences with searching for information in general and searching for multimedia items. In summary, the overall experience with searching is high. All students search for information at least once a week and 92% have been searching for information for two years or more. All students search for multimedia items at least once a year, and 65% does this once a week or more. 88% of the students have been searching for multimedia for at least two years [6].

3.2. Evaluation criteria

Traditional evaluation measures from the field of information retrieval are precision and recall. For evaluation within TRECVID the *average precision*, AP, is used. This single-valued measure corresponds to the area under an ideal precision-recall curve and is the average of the precision values obtained after each relevant camera shot is retrieved. This

metric favors highly ranked relevant camera shots. Let $L^i = \{l_1, l_2, \ldots, l_i\}$ be a ranked version of the answer set A. At any given index i let $R \cap L^i$ be the number of relevant camera shots in the top i of L, where R is the total number of relevant camera shots. Then AP is defined as:

$$AP = \frac{1}{R} \sum_{i=1}^{A} \frac{R \cap L^i}{i} \lambda(l_i) \quad (1)$$

where $\lambda(l_i) = 1$ if $l_i \in R$ and 0 otherwise. We use the AP as the basic metric for the conducted experiments.

4. RESULTS

The interactive search task has been performed by 22 groups of 2 students all searching for 12 topics. To show the overall quality of our system, we evaluated all runs that were made by the different students. Results are shown in figure 3. Compared to the overall TRECVID result the system performed well for the categories that involved finding specific or generic objects involved in some event. It also does pretty well for finding persons which is remarkable as we did not make use of VideoOCR or face recognition.

The system is outperformed by some other systems. Partly this is because of the long existence of especially the best performing system Informedia [2]. It should be noted, however, that this system was operated by the operator and not by novice users as in our case. It should further be noted that the results for UvA are slightly pessimistic in the sense that they are based on the pooled ground truth used in the TRECVID, only two out of the 22 results were evaluated directly. Elements in the output of the others that were not judged were considered not relevant to the topic.

Questionnaires indicate that students found the system easy to use, and were neutral with respect to the ease of learning. The topics and the system were both not similar to what the students had been using so far. Spearman's rank correlation test showed a correspondence between familiarity of the topics and ease of use ($\rho = 0.35; p = 0.05$) and between familiarity of the topics and ease of learning ($\rho = 0.34; p = 0.05$).

5. CONCLUSION

We have developed an interactive search mechanism which starts with high-level concepts and then lets the user browse through the data using advanced visualization mechanisms to find examples. Final results are found using query-by-example.

The systems perform well, but also has a number of limitations. Concerning the visualization, students often relied on the simple array of images, rather than the structure based visualization. This was mostly due to the fact that images were overlapping in the display, so it was difficult to select images. We are currently optimizing the overlap of images in 2D display, while at the same time we pursue the use of 3D display mechanisms. Finally, query-by-example was based on global histograms, but a region based approach, where users select the proper region to use, would be more suited. [2]

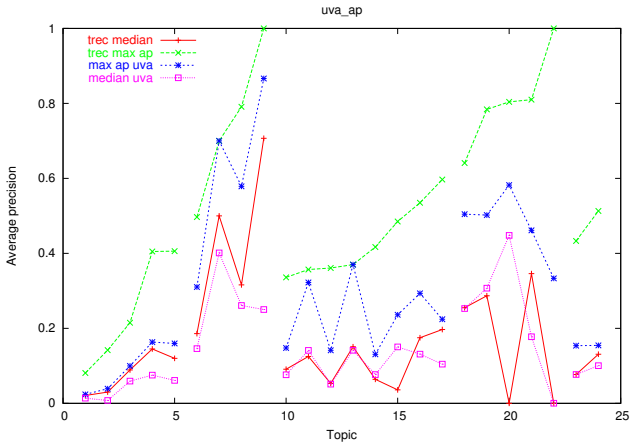

Fig. 3. *Plots of average precision, for the 24 topics defined earlier. Within each category the elements are ordered according to the best result achieved by any system within the TRECVID evaluation.*

6. REFERENCES

[1] J. Gauvain, L. Lamel, and G. Adda. The LIMSI broadcast news transcription system. *Speech Communication*, 37:89–108, 2002.

[2] A. Hauptman *et. al.* Informedia at TRECVID 2003 : Analyzing and searching broadcast news video. In *Proc. of TRECVID*, 2003.

[3] A. F. Smeaton, W. Kraaij, and P. Over. TRECVID-an introduction. In *Proc. of TRECVID*, 2003.

[4] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[5] J. Tenenbaum, V. Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2322, 2000.

[6] M. Worring *et.al.* Interactive search using indexing, filtering, browsing and ranking. In *Proceedings of TRECVID*, 2003.

[2]This work is partly sponsored by the IOP MMI Project I'mIK (MMI0136). We would like to thank F. Verster, J. van Rest, G. Schreiber, T. Pham, A. Diplaros and CMU for their efforts