THE MEDIAMILL SEMANTIC VIDEO SEARCH ENGINE

M. Worring , C.G.M. Snoek, O. de Rooij, G.P. Nguyen, A.W.M. Smeulders

Intelligent Systems Lab Amsterdam, University of Amsterdam
www.mediamill.nl

ABSTRACT

In this paper we present the methods underlying the MediaMill semantic video search engine. The basis for the engine is a semantic indexing process which is currently based on a lexicon of 491 concept detectors. To support the user in navigating the collection, the system defines a visual similarity space, a semantic similarity space, a semantic thread space, and browsers to explore them. We compare the different browsers and their utility within the TRECVID benchmark. In 2005, We obtained a top-3 result for 19 out of 24 search topics. In 2006 for 14 out of 24.

Index Terms— Video indexing, visualization, retrieval, performance evaluation, semantic threads.

1. INTRODUCTION

Commercial video search engines such as Google and Blinkx rely mainly on text in the form of closed captions or transcribed speech. This results in disappointing performance when the visual content is not reflected in the associated text. In addition, when the videos originate from non-English speaking countries, such as China or The Netherlands, querying the content becomes even harder as automatic speech recognition results are much poorer. Indexing videos with semantic visual concepts is more appropriate.

In an ideal system there is a lexicon containing thousands of semantic visual concepts accurately detected in the video collection. The semantic gap between the concepts and the data, however, dictates that this is not realistic. Current systems at best have small lexicons with some of the concepts having high accuracy. When concepts are not in the lexicon, or when the accuracy is limited, the burden of finding relevant video fragments remains with the user. The user should interactively find her way through the collection.

In literature various methods have been proposed to support the user beyond text search. Some of the most related work is described here. Informedia uses a limited set of high-level concepts to filter the results of text queries [2]. In [8], the authors employ clustering to improve the presentation of results to the user. Both [2] and [8] use simple grid based visualizations. More advanced visualization tools are proposed in [1], [4], and [3] based on collages of keyframes, dynamically updated graphs, and rapid serial visual presentation respectively, but no large semantic lexicon is used.

In this paper we present our semantic search engine. This system computes a large lexicon of 491 concepts, clusters and threads to support interaction. It provides advanced visualization methods, giving users quick access to the data. To demonstrate the effectiveness of our interactive search engine, it is evaluated in the NIST TRECVID video retrieval benchmark, the de facto standard for this field.

2. STRUCTURING THE VIDEO COLLECTION

The aim of our interactive retrieval is to retrieve from a multimedia archive A, which is composed of n unique shots $\{s_1, s_2, \ldots, s_n\}$, the best possible answer set in response to a user information need. Examples of such needs are ”find me shots of dunks in a basketball game” or ”find me shots of Bush with an American flag”. To make the interaction most effective we add different indices and structure to the data.

The result of speech recognition from the audio stream can provide important information on the content of the video. We use standard information retrieval techniques to compute the similarity S_T between the pieces of transcribed text corresponding to the shots to compare [10]. We then build up the *textual similarity space* in which all pairs of shots are assigned a distance.

The visual indexing starts with computing a high-dimensional feature vector F for each shot s. In our system we use the Wiccest features as introduced in [13] (see also [12]) and Gabor features. Wiccest features combine color invariance with natural image statistics. Color invariance aims to remove accidental lighting conditions, while natural image statistics efficiently represent image data. They are the basis for deriving a set of low level semantic protoconcepts like grass, sky etc.

In the next indexing step we compute a similarity function S_v, allowing comparison of different shots in A. For this we use the function described in [13], which computes the distance to the protoconcepts. The result of this step is the *visual similarity space*. This space forms the basis for visual exploration of the dataset.

This research is sponsored by the BSIK project MultimediaN

We now move on to the more specific topic of adding semantic indexing to the data, which is the process of associating every shot s in the database with a measure of presence P_j of the given concept j.

The central assumption in our semantic indexing architecture is that any broadcast video is the result of an authoring process. For authoring-driven analysis we proposed the semantic pathfinder [11], composed of three analysis steps following the reverse authoring process. Each analysis step in the path detects semantic concepts. In addition, one can exploit the output of an analysis step in the path as the input for the next one. The semantic pathfinder starts in the *content analysis step*. In this analysis step, we follow a data-driven approach, using an optimal fusion of visual and textual information, for indexing semantics. In the *style analysis step* we tackle the indexing problem by viewing a video from the perspective of production. Rather than focussing on consistency in content, we focus here on the consistency in the video production process. Finally, in the *context analysis step*, we analyze the semantics of a shot by taking the scores of other concepts for this shot into account. One would expect that some concepts, like *vegetation*, have their emphasis on content where the style (of the camera work that is) and context (of concepts like *graphics*) do not add much. In contrast, concepts like *airplane*, might profit from an observed high score on *sky* and a low score on *indoor*. The virtue of the semantic pathfinder is its ability to find the best path of analysis steps on a per-concept basis. The generic indexing structure is used to create a lexicon of 491 concepts, using the combined annotated examples of MediaMill [12] and LSCOM [5] as training set. Elements in the lexicon range from specific persons to generic classes of people, generic settings, specific and generic objects etc. Every shot s_i is now described by a probability vector $\{P_1, P_2, ..., P_{491}\}$.

Given two probability vectors, we use similarity function S_C to compare shots, now on the basis of their semantics. S_C resembles histogram intersection, adding the minimum probability of the two shots for concept j over the whole vector P. This yields the *semantic similarity space*.

The semantic similarity space induced by S_C is complex as shots can be related to several concepts. Therefore, we propose to add additional navigation structure composed of a collection of linear paths, called *threads*, through the data. Such a linear path is easy to navigate by simply moving back and forth. The question is how to select the different elements which constitute the path and the ordering of those elements.

When the whole collection is considered, the first obvious ordering is time. So our first thread is the time thread T^t. A complete set of threads $T^l = \{T^l_1, ..., T^l_{491}\}$ on the whole collection is defined by the concepts in the lexicon. The ranking based on P provides the ordering.

The question arises how to proceed if we want to compute semantic threads based on the semantic similarity space, but which are not in 1-1 correspondence with one element in the lexicon? This requires to consider the whole space and to find shots that share similar semantics. To find such groups we perform k-means clustering in the semantic similarity space. The elements of each group define the elements of the set of threads $T^s = \{T^s_1, ..., T^s_k\}$. Ordering of these elements is done by applying a shortest path algorithm inside the cluster. So, shots with similar semantic content are near each other in the thread.

The *Semantic thread space* is composed of T^l and T^s. An overview of all the steps performed in the structuring of the video collection is given in Fig. 1.

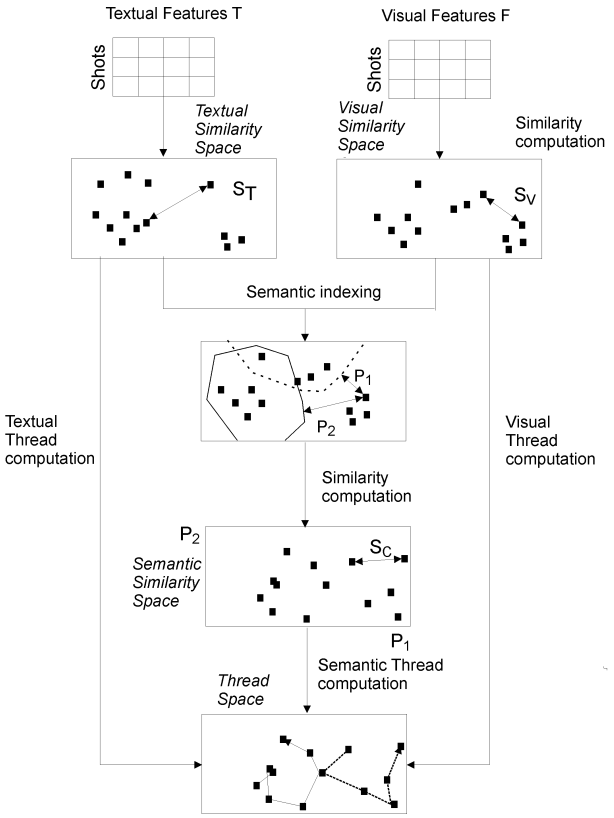

Fig. 1. A simplified overview of the computation steps required to support the user in interactive access to a video collection. Note, that for the vectors T, F and P only two dimensions are shown.

3. INTERACTIVE SEARCH

The visual similarity space and the thread space define the basis for interaction with the user. Both of them require different visualization methods to provide optimal support. We developed four different browsers, which one to use depends on the information need. The CrossBrowser is defined for those cases where there is a direct relation between the information need and one of the concepts in the lexicon. If a more com-

plex relation between the need and the lexicon is present, the user should be provided with more semantic navigation possibilities and the SphereBrowser and RotorBrowser are most appropriate. Finally, when there is no semantic relation, we have to interact directly with visual similarity space and this is supported in the GalaxyBrowser. The different browsers are visualized in Fig. 2.

The *CrossBrowser* visualizes a single thread T_j^l based on a selected concept j from the lexicon versus the time thread T^t [12]. They are organized in a cross, with T_j^l along the vertical axis and T^t along the horizontal axis. Except for threads based on the lexicon, this browser can also be used if the user performs a textual query on the speech recognition result associated with the data, as this leads to a linear ranking also.

In the *SphereBrowser* the time thread T^t is also presented along the horizontal axis [12]. For each element in the time thread, the vertical axis is used to visualize the semantic thread T_j^s this particular element is part of. Users start the search by selecting a current point in the semantic similarity space by taking the top ranked element in a textual query, or a lexicon based query. The user can also select any element in one of the other browsers and take that as a starting point. They then browse the thread space by navigating time or by navigating along a semantic thread.

The *RotorBrowser* takes the notion of semantic threads a step further and exploits the observation that a shot does not only share semantics with other shots, but can share similar speech or similar visual content. The RotorBrowser therefore visualizes the currently active shot and the semantic thread T_j^s in which this shot occurs. Additional threads are then created on demand, starting from the current shot using visual similarity S_V and textual similarity S_T respectively. They provide the directions along which the user can navigate.

Browsing visual similarity space is the most difficult task as there are no obvious dimensions on which to base the display [6][12]. The core of the *GalaxyBrowser* is formed by a projection of the high-dimensional similarity space induced by S_v to the two dimensions of the screen. This projection is based on ISOMAP and Stochastic Neighbor Embedding. However, in these methods an element is represented as a point. When applied to images it leads to visualizations where images are overlapping one another. We therefore extended the methods to assure that images show very limited overlap. The result is a two-dimensional space where images are well visible and images next to each other have similar visual characteristics. A great advantage of this is that similar images are typically all relevant to the information need and can thus be selected by one interaction of the user.

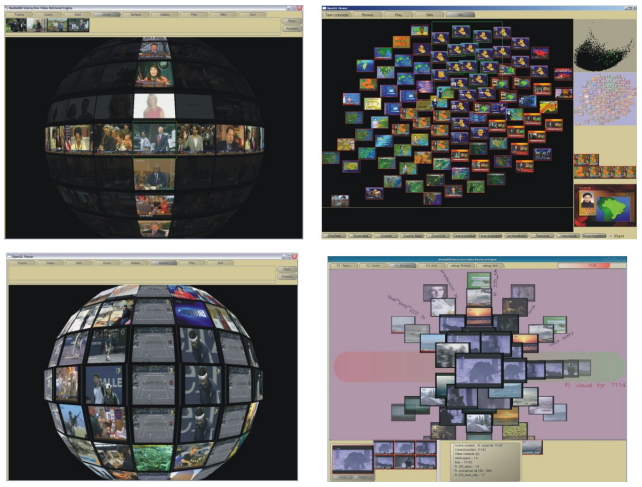

Fig. 2. Browsers in the MediaMill search engine. Top left the CrossBrowser (showing a ranked query result vertically and the timeline of the program horizontally), bottom left the SphereBrowser (with a tennis thread in the center), top right the GalaxyBrowser (several clusters of different maps visible) and bottom right the RotorBrowser (showing several threads containing the current shot) .

Fig. 3. Interactive search results for 24 topics, results for the users of the different browsers are indicated with special markers.

4. EXPERIMENTS

We performed our experiments within the interactive search task of the 2005 and 2006 NIST TRECVID benchmark, due to space limitations we only show results for 2006 here, for 2005 see [12]. The video archive used in 2006 is composed of 320 hours of US, Arabic, and Chinese broadcast news sources, recorded in MPEG-1 during November 2004. The test data contains about 150 hours. Together with the video archive came automatic speech recognition results and machine translations donated by a US government contractor. The Fraun-

hofer Institute provided a camera shot segmentation [7]. The camera shots serve as the unit for retrieval. The semantic pathfinder detects the 491 concepts with varying performance [10]. The goal of the TRECVID interactive search task, is to satisfy an information need. Given such a need, in the form of a search topic, a user is engaged in an interactive session with a video search engine. To limit the amount of user interaction and to measure search system efficiency, all individual search topics are bounded by a 15-minute time limit. The interactive search task contains 24 search topics in total. They became known only a few days before the deadline of submission. Hence, they were unknown at the time we developed our semantic concept detectors. In line with the TRECVID submission procedure, a user was allowed to submit, for assessment by NIST, up to a maximum of 1000 ranked results for the 24 search topics.

Following the standard in TRECVID evaluations [9], we use *average precision* to determine the retrieval accuracy on individual search topics. The average precision is a single-valued measure that corresponds to the area under a recall-precision curve.

4.1. Results

In 2006 two users participated in the search experiment, using the CrossBrowser and RotorBrowser respectively. Results in Fig. 3 indicate that for most search topics, users of the proposed interactive retrieval system score well above average. They obtain a top-3 average precision result for 14 out of 24 topics. Best performance is obtained for 7 topics.

To gain insight in the overall quality of our system, we compare the results of our users with all other users that participated in the retrieval tasks of the 2006 TRECVID benchmark. We visualized the results for all submitted search runs in Fig. 3. It follows that the results are state-of-the-art.

5. DISCUSSION AND CONCLUSION

The success of the Crossbrowser indicates that having a large lexicon of concepts, such that a direct match between information need and a lexicon concept is likely to exist, is the best method for effective search.

The SphereBrowser (2005, results not shown) is successful when multiple semantic concepts are relevant such as *People with banners or signs*, *Meeting* and *Tall building*. It was also successful for topics such as *Airplane takeoff* and *Office setting*. Here there were only a limited number of consecutive valid shots visible in each thread, but because of the combination of both time and semantic threads there was always another valid, but not yet selected, shot visible.

The RotorBrowser is successful for those cases where the result can not only be found along semantic directions, but also through similar text and visual content. Topics for which this was successful are e.g. greeting by a kiss on the cheek which has semantic aspects, but will likely show two faces taken from a close distance, hence visual similarity helps here. The same holds for something burning with flames visible.

Finally, the GalaxyBrowser (also used in 2005 only) works well in case shots for an information need are visually similar e.g. topics related to *tennis*, *car* or *fire*. When topics have large variety in visual settings, for instance *person x* topics, visual features hardly yield additional information to aid the user in the interactive search process.

In conclusion, we have developed a number of different browsing methods based on a lexicon of 491 concepts, where the optimal method follows from the information need and the availability of reliable concepts in the lexicon.

6. REFERENCES

[1] J. Adcock, M. Cooper, A. Girgensohn, and L. Wilcox. Interactive video search using multilevel indexing. In *CIVR, LNCS*, volume 3568, 2005.

[2] M. Christel and A. Hauptmann. The use and utility of high-level semantic features. In *CIVR, LNCS*, volume 3568, 2005.

[3] A. Hauptmann, W.-H. Lin, R. Yan, J. Yang, and M.-Y. Chen. Extreme video retrieval: Joint maximization of human and computer performance. In *ACM Multimedia*, Santa Barbara, USA, 2006.

[4] D. Heesch and S. Ruger. Three interfaces for content-based access to image collections. In *CIVR, LNCS*, volume 3115, 2004.

[5] M. Naphade, J. Smith, J. Tesic, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis. Large-scale concept ontology for multimedia. *IEEE Multimedia*, 13(3):86–91, 2006.

[6] G. P. Nguyen and M. Worring. Interactive access to large image collections using similarity based visualization. *Journal of Visual Languages and Computing*, 2007. *In press*.

[7] C. Petersohn. Fraunhofer HHI at TRECVID 2004: Shot boundary detection system. In *Proc. TRECVID Workshop*, 2004.

[8] M. Rautiainen, T. Ojala, and T. Seppnen. Clustertemporal browsing of large news video databases. In *IEEE International Conference on Multimedia and Expo*, 2004.

[9] A. Smeaton. Large scale evaluations of multimedia information retrieval: The TRECVid experience. In *CIVR*, volume 3569 of *LNCS*, pages 19–27, 2005.

[10] C. Snoek et al. The MediaMill TRECVID 2006 semantic video search engine. In *Proc. TRECVID Workshop*, NIST, 2006.

[11] C. Snoek, M. Worring, J. Geusebroek, D. Koelma, F. Seinstra, and A. Smeulders. The semantic pathfinder: Using an authoring metaphor for generic multimedia indexing. *IEEE PAMI*, 28(10):1678–1689, 2006.

[12] C. Snoek, M. Worring, J. van Gemert, J. Geusebroek, D. Koelma, G. Nguyen, O. de Rooij, and F. Seinstra. MediaMill: Exploring news video archives based on learned semantics. In *ACM Multimedia*, Singapore, 2005.

[13] J. van Gemert, J. Geusebroek, C. Veenman, C. Snoek, and A. W. Smeulders. Robust scene categorization by learning image statistics in context. In *Proceeding of CVPR Workshop on Semantic Learning Applications in Multimedia (SLAM)*, 2006.