THE AXES-LITE VIDEO SEARCH ENGINE

Shu Chen[1], *Kevin McGuinness*[1], *Robin Aly*[2], *Noel E. O'Connor*[1], *Franciska de Jong*[2]

[1]CLARITY: Center for Sensor Web Technology, Dublin City University, Ireland
[2]University of Twente, the Netherlands

ABSTRACT

The aim of AXES is to develop tools that provide various types of users with new engaging ways to interact with audiovisual libraries, helping them discover, browse, navigate, search, and enrich archives. This paper describes the initial (lite) version of the AXES search engine, which is targeted at professional users such as media professionals and archivists. We describe the overall system design, the user interface, and the results of our experiments at TRECVid 2011.

1. INTRODUCTION

The AXES project focuses on bringing together users, content, and technology to build next generation tools for searching, browsing, and discovering multimedia digital libraries. To achieve this goal, the project will develop a series of digital library search and navigation systems. These systems will target different user groups: our first system will target professional users, our second researchers, and our final system home users. This paper describes an initial version of the AXES system for professional users. This is not a final production system, but rather was designed as a platform for integrating new computer vision and multimedia indexing algorithms, for developing and experimenting with novel interaction techniques, and for performing user testing and gathering feedback. The current version of the system integrates several state-of-the-art content based multimedia search techniques, and is a refinement of our TRECVid 2011 system. The remainder of this paper describes the overall system and its components, the user interface, and the outcomes of known-item search (KIS) and instance search (INS) user experiments that we carried out for TRECVid.

The paper is organized as follows: Section 2 describes the search engine and relevant components. Section 3 describes the user interface design. Section 4 discusses the results and findings of the TRECVid experiments. Section 5 concludes this paper.

2. SEARCH ENGINE DESCRIPTION

We used the following rationale for the development of the AXES-lite search engine: (1) it should allow easy integration of existing and novel search methods, (2) it should be adaptable for innovative user interaction models, and finally (3) it should provide a test-bed for trialling these methods.

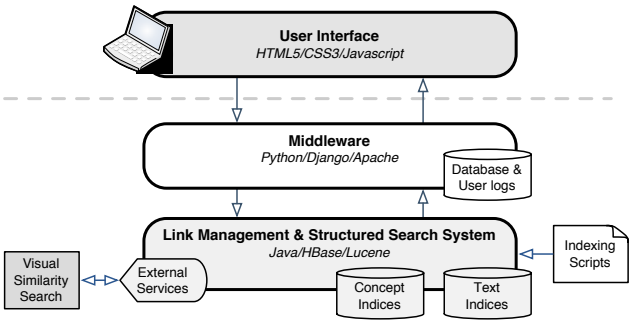

Fig. 1. System Architecture of the AXES-lite System.

2.1. System Architecture

Based on the above rationale, we chose to use a service-oriented architecture that is adaptable to future needs. Figure 1 shows an overview of the architecture's components, which we describe starting from the top. The user interface converts query entered by user to an JSON formatted request that is sent to the middleware; the middleware logs this request for future analysis and passes it on to a Java servlet in the back-end. The servlet uses function calls to communicate with the core library. This library supports three principle retrieval components: (1) text-based retrieval, (2) visual concept-based retrieval, and (3) visual similarity-based retrieval using sample images. The relevant parts of the request are forwarded to the individual retrieval components, and the returned scores are fused to produce a final result list. This list is sent back to middleware where it is logged and forwarded to the user interface. The following describes the currently used retrieval components and the fusion method.

2.2. Text Retrieval

The AXES-lite system stores the available text for each retrieval unit in a text index. Our system uses Apache Lucene (version 3.1.2) for all text based indexing and ranking. We indexed both the provided ASR and several metadata fields

(title, description, keywords, subject, and uploader) for our TRECVid 2011 KIS experiments. We indexed custom ASR for the INS collection.

2.3. Visual Concept Classifiers

The user interface allows the user to select one or more predefined concept classifiers and have the results re-ranked based on the confidence of these classifiers. The classifiers are applied at the keyframe level; for scene-like concepts, we represent the keyframes using a Pyramid Histogram of Visual Words (PHOW) [1] descriptor, and rank each keyframe in the collection using a non-linear χ^2 SVM. We used the following scene-like concepts for our TRECVid experiments: airplane, boat/ship, cityscape, demonstration, female-human-face-closeup, nighttime, and playing instrument.

2.4. Similarity Search

Users can also drag images into the similarity search area in the interface to have the system find visually similar videos in the collection. Again, similarity is computed at the keyframe level, and is based on the *Video Google* approach [2, 3]. The aim is to retrieve keyframes containing a certain specific object despite changes in scale, viewpoint, and illumination. Interest points are located based on Hessian-Affine regions and SIFT descriptors are used to characterize the elliptical region surrounding each interest point. These interest points are quantized to visual words by finding representative words for the collection using k-means, and each keyframe is represented by the visual words. With these visual words, standard efficient text retrieval methods can be employed to enable object retrieval in a Google-like manner. Similarity search is coupled with a fast spatial re-ranking method [3] to improve retrieval quality.

2.5. Fusion

Since the AXES-lite system is a basis to be extended for future experiments, we chose a relatively simple algorithm to fuse the scores from above retrieval components. First, we normalized the scores of each component to the interval $[0, 1]$ and then fused them using a linear combination as follows:

$$score = \lambda_1 score_{text} + \frac{\lambda_2}{n} \sum_{i=1}^{n} score_{c_i} + \frac{\lambda_3}{m} \sum_{j=1}^{m} score_{sim_j}, \quad (1)$$

where $score$ is the final score, $\lambda_1 \in [0, 1]$ is the mixture component for textual score $score_{text}$, $\lambda_2 \in [0, 1]$ is the weight of the n selected concepts, $score_{ci}$ is the confidence score for concept i, $\lambda_3 \in [0, 1]$ is the weight of the image similarity, m is the number of images used in similarity search, and $score_{sim_j}$ is the similarity score of the j example image to current image.

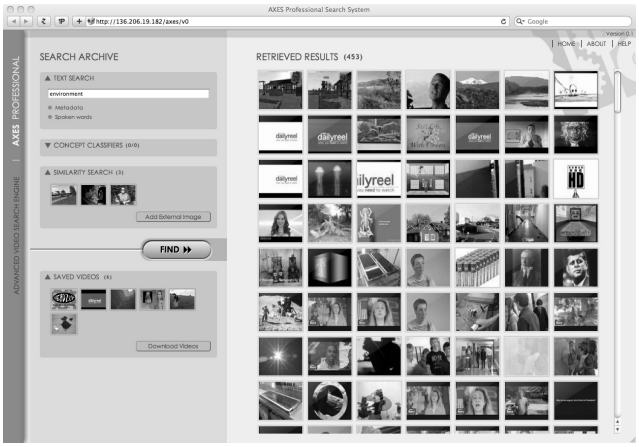

Fig. 2. Screenshot of the user interface of AXES-lite system.

We set the weights λ_1, λ_2, and λ_3 uniformly, modeling a situation where text, concepts, and image similarity are equally important. In future research, we plan to replace this straightforward fusion scheme with a more sophisticated scheme, such as the probabilistic scheme described in [4]

3. USER INTERFACE DESCRIPTION

Figure 2 contains a screenshot of the AXES-lite user interface. There are three main panels in the interface. The text search panel is located in the top-left of the interface and includes a text input box and some checkboxes, which allow the user to specify whether they wish to search the video metadata, spoken words (from ASR), or both. Below the text search area is a concept classifier area that allows user to select from several predefined predefined concepts. The similarity search area below this allows users to drag and drop videos from the result list to add them as query terms in a visual similarity search. Users can also upload custom image files or input external URLs here.

The right part of interface displays all retrieved videos based on the text query, selected concepts, and visual similarity search. Videos are represented in a scrollable thumbnail grid-based result list. Double-clicking on a result displays an video playback overlay with fast-seek support built in. The total number of retrieved results is shown in the top-right.

The saved videos area, located at the lower-left of interface, allows users to save video shots for subsequent use. As with the similarity search area, videos can be saved using drag-and-drop. Users can review saved videos by double clicking their thumbnails to play back corresponding video. The top-left of the area displays the number of saved videos. The *download videos* button, located at the bottom of the saved videos area, packages all saved shots into an archive file and downloads it to the users computer.

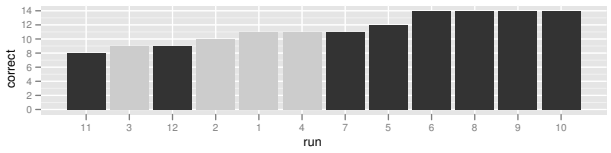

Fig. 3. Number of correct videos found by all groups for KIS task. Runs 1...4 (highlighted) are submitted by AXES

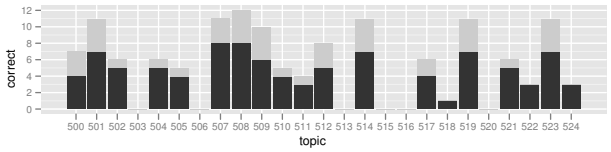

Fig. 4. Number of correct videos found by all participating groups by topic. AXES runs are highlighted.

4. EXPERIMENTS

We benchmarked our system by participating in the two interactive search tasks at TRECVid 2011: known-item search (KIS), which models the situation in which someone is searching for a particular video, has seen it before, and knows that it is contained somewhere in the collection [5]; and instance search (INS), which models the situation where the user wants to find more instances of a specific person, object, or place given a visual example [5]. Our experiments were carried out over two days with participants from a Dutch media company and archivists from the Netherlands Institute for Sound and Vision. We used the previously described system for both the KIS and INS tasks with a slightly different user interface, which included several TRECVid specific components such as timers and topic descriptions.

4.1. Known-item Search

A total of 14 media professionals participated in the KIS experiments. Each participant was assigned 10 of the 25 topics and given five minutes to complete each topic. We evaluated our system for all four TRECVid runs. Each run used an identical search system and user interface, varying only in the users who performed the search. Each user was randomly assigned an equal number of topics. Figure 3 shows the number of correct videos found by all groups that attended TRECVid 2011 with AXES results highlighted. The best AXES run found 11 videos; the best-submitted run found 14. It is clear from the figure that our best performing runs were around the median. Our worst performing run found 9 videos: a variation due to user search performance alone.

Figure 4 shows the number of correct videos found in each topic run with AXES runs highlighted. There is considerable variation in topic difficulty. No submission contained the correct video for the six topics: 503, 505, 513, 515, 516, and 520. Only one submission found the correct video for topic 518. The figure also shows that at least one of our users was able to find the correct video for most topics that any other participating groups were also able to find, the exceptions being the three most difficult topics (in terms of number of correct results submitted): 518, 522, and 524. The figure highlights the high-variation in user performance: a combined run containing our best performing user for each topic would have found 16 of the 25 videos, whereas only 5 of the 25 individual topic videos were found by all our users.

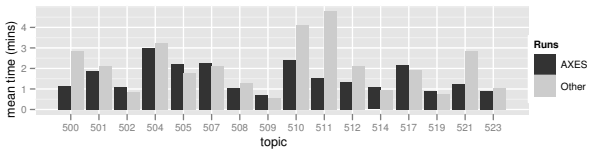

Fig. 5. Mean time (in minutes) to find the correct video. Topics where the correct answer was not found by any AXES runs are not shown.

Figure 5 shows the mean time in minutes spent by participants finding the correct video for each topic, in which at least one other group also successfully found the correct video. The figure shows that the AXES users were often faster than average at finding the correct video.

4.2. Instance Search

A total of 30 visiting media students participated in the INS experiments. Each participant was assigned five topics and was allowed 15 minutes to complete each topic. We submitted four runs for the INS task, ordered by the number of saved videos, i.e. participants that saved more videos were placed in the first runs. AXES was the only group to submit runs for interactive INS search at TRECVid 2011.

The following table shows precision, recall, mean average precision (MAP), the bpref measure [6], the average number of relevant videos (rel), and the average number of videos judged to be non-relevant (non-rel):

run	precision	recall	MAP	bpref	rel	non-rel
1	0.74	0.36	0.33	0.34	26.40	8.68
2	0.73	0.28	0.26	0.27	20.80	5.60
3	0.81	0.26	0.25	0.25	18.76	3.12
4	0.81	0.21	0.21	0.21	14.76	2.68

The first run was clearly the best in terms of MAP and bpref, implying that users that saved more videos performed better than users that saved less. The table also suggests that the effect of searcher on MAP and bpref can be quite large.

Figure 6 shows average precision for each topic for our best run, and mean average precision over all runs. The results

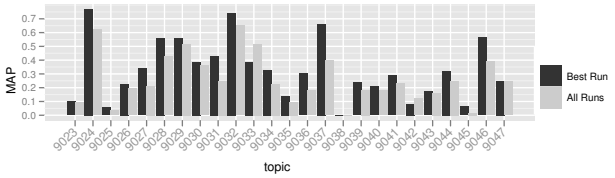

Fig. 6. Average precision by topic for our best run (run 1) and mean average precision (MAP) over all runs (INS task).

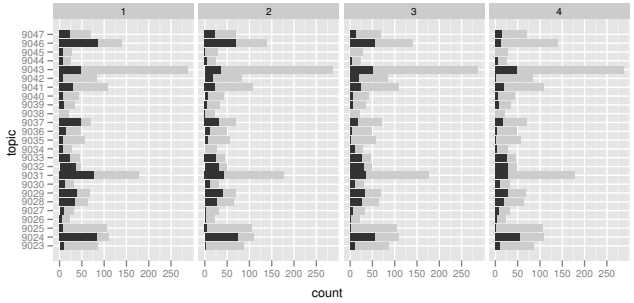

Fig. 7. Comparison of the number of relevant videos (dark bars) with the number of saved videos (light bars) for each of the four AXES runs.

indicate some tasks were much more difficult than others. In particular, none of our participants successfully found any of 21 relevant videos for topic 9038 "Female presenter X," even though an average of 12 videos had been saved. The result implies that the participant may have misunderstand topic. Participants might have searched female presenters in general, or saved shots of a person they believed to be the female presenter featured in the example image. A similar conclusion can be inferred for topic 9042 "Male presenter Y," from the low precision of submitted videos. For example, only 9 out of all 46 saved videos were judged relevant in run 1.

Figure 7 shows the proportion of relevant videos found by all participants in each of the four runs. Each bar in this plot represents the performance of a single user on a single topic. There is a high variation in performance from the user who found almost 100 relevant videos for topic 9046 to another who only found less than 25 relevant videos.

Figure 8 shows the relative proportions of relevant and non-relevant videos saved by each participant. Topic 9038 and 9042, most easily misunderstood by participants, again stand out as having many non-relevant videos saved across all participants. The recall-oriented group (run 1) clearly has more false-positives than the other three groups, but performed better in terms of MAP and bpref.

5. CONCLUSION

This paper described AXES-lite system developed by the EU AXES project, and described the design decisions, system

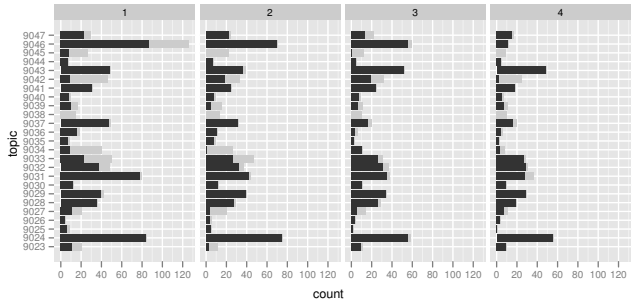

Fig. 8. Plot showing the relative proportions of relevant (dark bars) and non-relevant (light bars) videos saved by each participant by topic.

architecture, and user interface. The novelty of the system is not in the individual indexing and searching components, but rather their combination in a highly usable system. We tested and benchmarked our system at TRECVid 2011, from which we received considerable user feedback on the KIS and INS task. Participants commented that the system was intuitive and responsive, and provided other valuable advice on how to improve the system. In the future we plan to incorporate suggested user feedback, such as displaying more information to the user on *why* a particular result was judged to be relevant, and also plan experiment with alternative interaction mechanisms and to incorporate more multimedia search and indexing algorithms.

6. ACKNOWLEDGEMENT

This work was funded by the EU FP7 Project AXES ICT-269980.

7. REFERENCES

[1] A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forests and ferns," in *Proceedings of ICCV*, 2007.

[2] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proceedings of ICCV*, 2003.

[3] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proceedings of CVPR*, 2007.

[4] Rong Yan, *Probabilistic Models for Combining Diverse Knowledge Sources in Multimedia Retrieval*, Ph.D. thesis, Carnegie Mellon University, 2006.

[5] "Guidelines for TRECVid 2011," NIST, `http://www-nlpir.nist.gov/projects/tv2011/tv2011.html`.

[6] Chris Buckley and Ellen M. Voorhees, "Retrieval evaluation with incomplete information," in *Proceedings of ACM SIGIR conference on Research and development in information retrieval*, 2004, pp. 25–32.