AXES at TRECVid 2011

Kevin McGuinness[1], Robin Aly[2], Shu Chen[1], Mathieu Frappier[1]
Martijn Kleppe[3], Hyowon Lee[1], Roeland Ordelman[2,4]
Relja Arandjelović[5], Mayank Juneja[6], C. V. Jawahar[6]
Andrea Vedaldi[5], Jochen Schwenninger[7] Sebastian Tschöpel[7]
Daniel Schneider[7], Noel E. O'Conner[1], Andrew Zisserman[5]
Alan Smeaton[1], Henri Beunders[3]

[1]CLARITY: Center for Sensor Web Technology, Dublin City University, Ireland
[2]University of Twente, the Netherlands
[3]Erasmus University Rotterdam, the Netherlands
[4]National Institute for Sound and Vision, the Netherlands
[5]Oxford University, United Kingdom
[6]International Institute of Information Technology, India
[7]Fraunhofer-Gesellschaft, Germany

The AXES project participated in the interactive known-item search task (KIS) and the interactive instance search task (INS) for TRECVid 2011. We used the same system architecture and a nearly identical user interface for both the KIS and INS tasks. Both systems made use of text search on ASR, visual concept detectors, and visual similarity search. The user experiments were carried out with media professionals and media students at the Netherlands Institute for Sound and Vision, with media professionals performing the KIS task and media students participating in the INS task. This paper describes the results and findings of our experiments.

1 Introduction

This paper describes the first participation of the EU Project AXES at TRECVid. The AXES project aims to link users and multimedia content together by means of technology. The project partners involved in this year's participation (with references to earlier participations) were: 1) Dublin City University (CLARITY: Center for Sensor

Web Technologies) [6, 17]; 2) University of Twente [2, 3]; 3) the National Institute for Sound and Vision (NISV/Netherlands); 4) Erasmus University Rotterdam; 5) Oxford University [16]; 6) Fraunhofer-Gesellschaft.

Since AXES is about bringing users, content, and technology together, we focused this year on interactive user experiments in the known-item search (KIS) and instance search (INS) tasks. The collaboration with industry partners gave us a unique opportunity to conduct experiments in a realistic environment with professional users engaging with and testing the system. We limited ourselves to using state-of-the-art technology in the search components, rather than investigating new search methods, as our focus is intended to be more on the users than the technology this year.

This paper is structured as follows: Section 2 describes the system we developed for this year's TRECVid participation, including the system architecture and the user interface. Section 3 describes the experiments and discusses the results and findings. Section 4 concludes this paper.

2 System Overview

In this section we describe the system we developed for this year's TRECVid participation.

2.1 Architecture

We used a service-oriented architecture for this year's TRECVid participation, which we plan to adapt for future research. Figure 1 shows an overview of the components involved. The central component of the system is a Java library that manages indexing, handles search requests (queries), and connects the various subsystems and external services. At search time, the user interface, described in Section 2.4, takes each query entered by the user and converts it to a JSON formatted request that is sent to the middleware. The middleware logs the request and forwards it to a Java servlet in the backend. The servlet then communicates with the core library by means of function calls. The relevant parts a query (e.g. keywords for ASR based search, or sample images for visual similarity search) are forwarded to the individual retrieval components described in Section 2.2. The scores returned by each of these components are then fused (combined) according to the score function described in Section 2.3, producing a list of retrieval units (videos or shots). This list is then send back to middleware as a JSON document where it is logged and forwarded to the user interface.

2.2 Retrieval Components

To rank retrieval units, we compute a score based on three primary components: 1) text retrieval scores based on matching user provided text against ASR and metadata indices (where available), 2) confidence scores based on matching concepts selected by the user

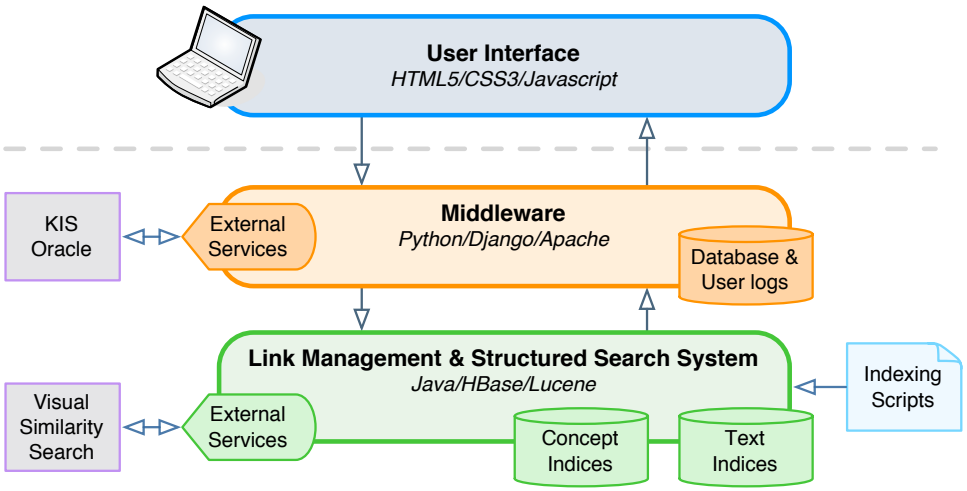

Figure 1: Overview of the system architecture. The client-side technologies are above the dashed line, the server-side technologies below.

with automatically detected concepts, and 3) similarity scores based on visual similarity search using example images provided by the user. The following describes these text search, concept classifier, and visual similarity search components.

Text Retrieval We stored the available text for each retrieval unit in a text index. Both our KIS and INS search engines used ASR data; we used the provided ASR for KIS and extracted custom ASR for the INS task. We also used five metadata fields from the provided metadata XML files for the KIS task: title, description, keywords, subject, and uploader. At query time, the standard Lucene retrieval function was used to calculate a text retrieval score for each retrieval unit if the query contained any text terms. We used Lucene version 3.1.2 [13] in our experiments.

Image Level Classifiers For scene-like concepts the keyframes are represented by a Pyramid Histogram of Visual Words (PHOW) [4], and ranked using a non-linear χ^2 SVM. The scene-like concepts that we used included: airplane, boat/ship, cityscape, demonstration, female-human-face-closeup, nighttime, and playing instrument.

In more detail, the PHOW descriptor involves computing visual words on a dense grid. Here visual words are vector quantized SIFT descriptors [8] that capture the local spatial distribution of gradients. In order to represent the spatial layout, an image is tiled into regions at multiple resolutions. A histogram of visual words is then computed for each image sub-region at each resolution level [7].

To train a large-scale SVM efficiently we use the PEGASOS stochastic gradient descent algorithm [10] (as implemented in the VLFeat library [14]). While PEGASOS is a linear SVM solver, we use the explicit feature map for the χ^2 kernel [15] to extend it efficiently to use a χ^2 (non-linear) kernel. The whole setup is fast and efficient compared to traditional SVM techniques that do not use the feature map idea. For example, in our framework training a SVM using 100K frames requires only 2 minutes and classifying 100K frames requires only 1 minute on an Intel Xeon CPU clocked at 1.86 GHz.

Similarity Search The similarity search is based on the "Video Google" approach [12, 9]. The aim is to retrieve key-frames containing a specific object despite changes in scale, viewpoint and illumination. The visual query is specified at runtime by an image containing the object or place of interest. The query image can be provided by downloading from a URL or taken from the corpus.

For the visual features, we use Hessian-Affine regions, which are invariant to changes in object illumination, scale, rotation and viewpoint. We then use SIFT descriptors [8] to describe each elliptical region in the image. The SIFT descriptors for these appearance regions are vector quantized into visual words, to give us a visual words representation for each key-frame. With this representation, standard efficient text retrieval methods can be employed to enable object retrieval in a Google-like manner. Searching the 100k key-frames of the corpus requires only a fraction of a second.

The vector quantization is carried out using approximate K-means method, which allows us to use very large, highly discriminative visual vocabularies. This search is coupled with a fast spatial re-ranking method [9] to improve retrieval quality.

2.3 Fusion

One of the main focuses of this year's participation was to develop the basis of a system that can be extended for future experiments. We therefore chose a relatively simple algorithm to fuse the scores from the above components. We first normalized the scores of each component to the interval $[0, 1]$ and then fused them using a linear combination as follows (see also [11]):

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

where $score$ is the final score, $\lambda_1 \in [0, 1]$ is the mixture component for textual scores, $\lambda_2 \in [0, 1]$ is the weight of the n selected concepts, $score_{c_i}$ 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 the similarity search, and $score_{sim_j}$ is the similarity score of the j example image to the current image.

In this year's experiments, we set the weights $\lambda_1, \lambda_2, \lambda_3$ equally, modeling a situation where text, concepts, and image similarity are equally important. In future participa-

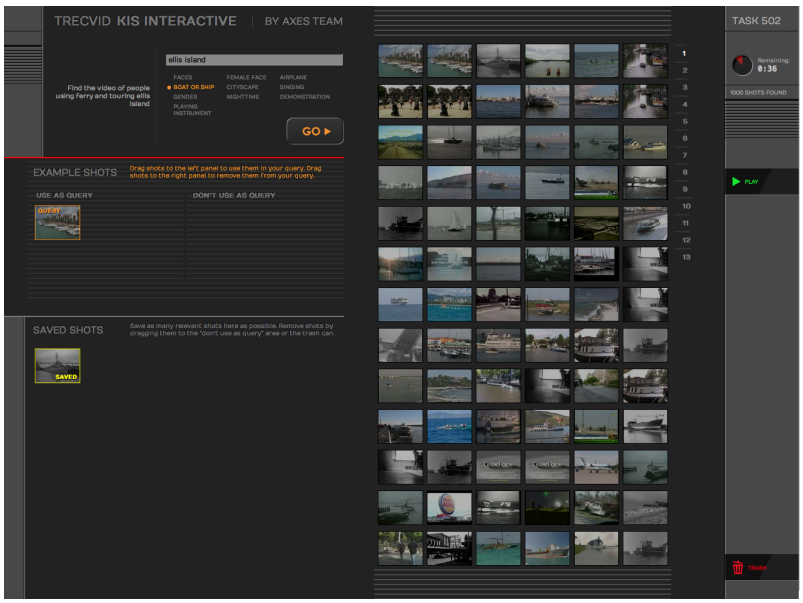

Figure 2: Screenshot of the user interface for the KIS task.

tions we plan to replace this straightforward fusion scheme with a more sophisticated scheme, such as the probabilistic scheme described in [18], or using a scheme that models the uncertainty of the detected objects (for example, words in transcripts or concept occurrences) [1].

2.4 User Interface

We developed a single web browser-based user interface for both the KIS and INS task, targeted at traditional desktop based interaction. The client side interface was developed using a combination of HTML5, CSS3, and Javascript, and used AJAX to communicate asynchronously with a server-side middleware. We used JSON as the document format for this communication, since it is both lightweight and simple to parse using the client browser. Figure 2 shows a screenshot of the user interface during a KIS task; Figure 3 shows the interface for the INS task.

On startup, the interface asks the experiment participant to enter a preassigned identification number and some contact details. This ID is sent to the middleware, which determines which task (KIS or INS) and topics have been assigned to that participant. The middleware responds with the task and topic details, and the user interface shows the appropriate UI elements. Each topic assignment is then displayed to the participant

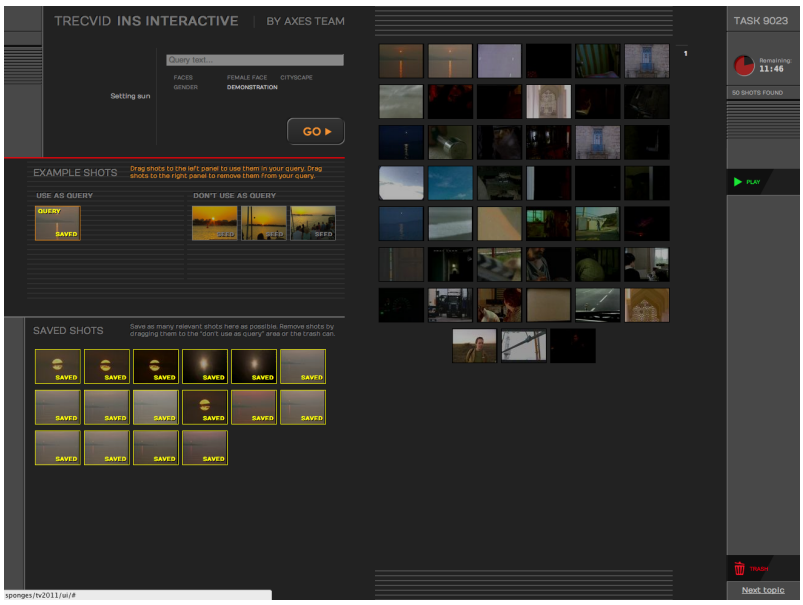

Figure 3: Screenshot of the user interface for the INS task.

in turn until they have completed all assignments.

The text associated with the topic is displayed in the top left of the user interface. For the INS task, this is just the short description of the object, place, person, or character that the user is required to find. For the KIS task, this is the full description of the topic. The participant can enter query terms in the search area on the right of where the topic is displayed. The query terms are used to search the metadata and ASR associated with the videos in the target collection (depending on what is available for the collection). Below this, the user can select from several visual concepts. Again, the available concepts depend on the collection being searched.

The middle-left of the user interface provides an area where the user can select images for the visual similarity search. There are two panels in this area; on the left is the *Use as query* area, where the user can place up to six shots to be added to the query. The example images for the INS task are automatically added to this area when the topic begins. On the right of this is the *Don't use as query* area: a holding area for shots that the user would like to keep for later use. The *saved shots* area on the bottom left is where the participant is required to place shots that they believe are relevant. For the KIS task, any shot added to this area is immediately sent to the KIS oracle for verification. If the shot is correct, the topic ends and the user proceeds to the next topic. For the INS task, the participant is required to add as many relevant shots to this area as possible.

Figure 4: Photograph of the user experiments at Netherlands Institute for Sound and Vision in Hilversum.

The centre-right of the interface displays the fused results of a search. These are displayed as keyframe thumbnails at the shot level. The participant can playback the video associated with a particular shot by double clicking on the corresponding thumbnail. Playback automatically starts at the relevant part of the video. The top-right of the interface shows topic information including the time remaining and the number of shots returned by the last search.

The user interface emphasizes *drag and drop*-based interaction: participants are free to drag shots from the between the results area, the query shots area, the holding area, and the saved shots area. As shots are moved between the areas, logging messages are sent to the middleware recording the user actions. When the time elapses for a topic (or if the user finds the correct shot in the KIS task) the shots saved and elapsed time are sent to the middleware, and the system displays a dialog prompting users to proceed to the next task when ready. The middleware is notified when all topics are completed and the participant is prompted to close the browser window.

3 Experiments

The user experiments were carried out over two days at the Netherlands Institute for Sound and Vision, Hilversum during early September. Figure 4 shows a photograph of the some of the media professionals participating in the KIS experiments in NISV, Hilversum. The following describes the experiment setup and discusses the results and findings.

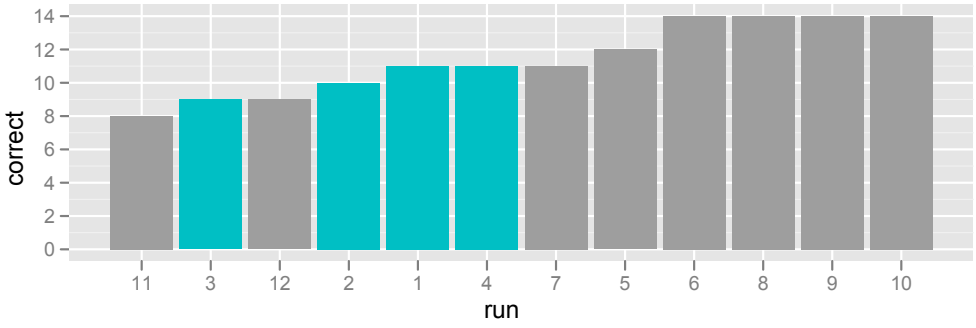

Figure 5: Number of correct videos (out of 24) found by KIS participants in each of the KIS runs. The graph is ordered from left to right by the number of videos found. Runs 1 . . . 4 (highlighted) are the runs submitted by the AXES team.

3.1 Known Item Search

A total of 14 media professionals (researchers from a Dutch media company and archivists from the Netherlands Institute for Sound and Vision) participated in the KIS experiments. Both types of media professionals are professionally trained in searching for audiovisual materials in different archival systems. Before the experiment, participants were briefed on the purpose of the experiment, and given a quick tutorial on using the user interface. Each participant was assigned ten topics and had 5 minutes to complete each topic. Participants were allowed to take short breaks between topics if desired. After the experiment, participants were asked to provide us with some freeform feedback on the task, interface, and system in general.

We submitted four runs of our system for evaluation. Each run used an identical search system and user interface, varying only in the users that actually performed the search. The participating users were randomly assigned to a single run for evaluation. Figure 5 shows the number of correct videos found in each of the runs submitted by all participants for evaluation. The AXES runs are highlighted. Our best runs (1 and 4) found 11 of the 25 correct videos; the best submitted runs by other groups found 14 correct videos. It is clear from the figure that our best performing runs performed around the median. Our worst performing run, which used precisely the same system, found 9 correct videos: a variation due to user search performance alone.

Figure 6 shows the number of correct videos found for each topic in all submitted runs, with the number found by AXES runs highlighted. The figure shows considerable variation in topic difficulty: all 12 submitted runs found the correct video for topics 501 and 508, whereas no submissions contained the correct video for the six topics: 503, 505, 513, 515, 516, and 520. Only one submitted run found the correct video for topic 518. The

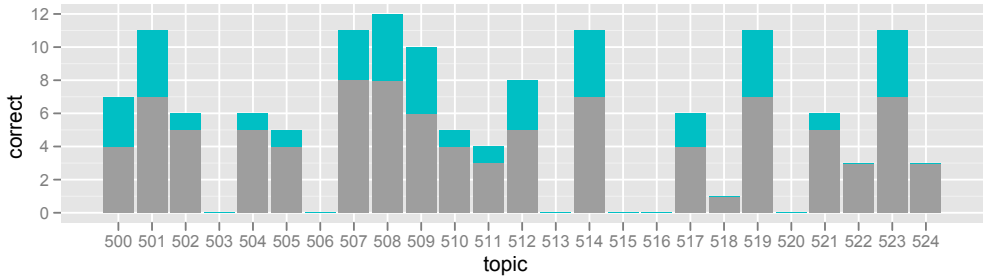

Figure 6: Number of correct videos found by KIS participants for each topic over all submitted runs. The total height of each bar indicates total number of correct videos found over all runs; the highlighted region is videos found in AXES submitted runs.

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.

Figure 7 shows the mean time in minutes spent by participants finding the correct video for topics in which the correct video was found for AXES runs and at least one other run. The figure shows that the AXES system participants were often faster than average at finding the correct video.

3.2 Instance Search

In total, 30 visiting media students participated in the INS experiments. Each participant was assigned five topics and had 15 minutes to complete each topic. As with the KIS task, participants were briefed on the experiment and given a tutorial on the user experiment before the task, and asked to provide freeform feedback after.

We submitted four runs of our system for evaluation. As with the KIS runs, all runs used exactly the same underlying system. This time we ordered the runs by number of saved videos: users that saved the most videos (recall-oriented users) were submitted in run 1 and users that submitted the least (precision-oriented users) in run 4. Unfortunately, this year AXES was the only group to submit runs for the interactive INS search task, so we cannot compare our results with other groups.

The following table shows the overall values for precision, recall, mean average precision (MAP), and the bpref measure [5] for each of the four submitted runs. Also shown is the

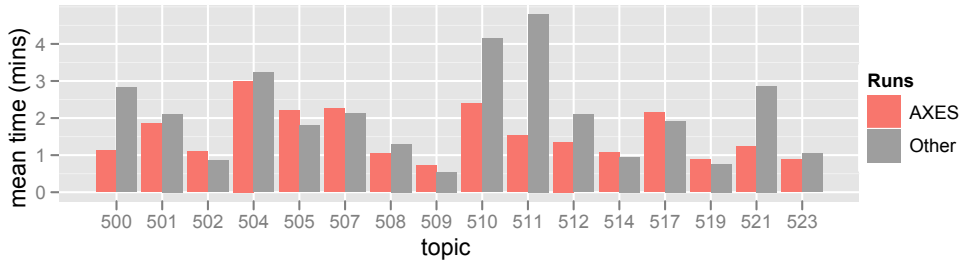

Figure 7: Mean time (in minutes) required to find the correct video for each topic by AXES runs and other runs. Topics where the correct answer was not found by any AXES runs are not shown.

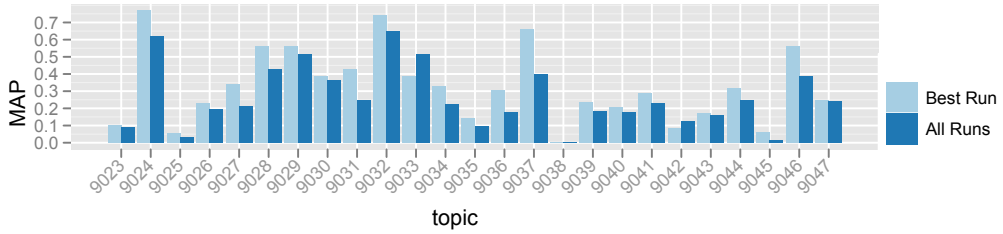

Figure 8: Average precision by topic for our best run (run 1) and mean average precision (MAP) over all runs (INS task).

average number of relevant videos saved by a user (rel) and the average number of videos judged to be non-relevant (non-rel) saved by a user:

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

Figure 8 shows average precision for each topic for our best run and mean average precision over all our runs. Some tasks were clearly more difficult for our system than others. In particular, none of our searchers found any of the 21 relevant videos for topic

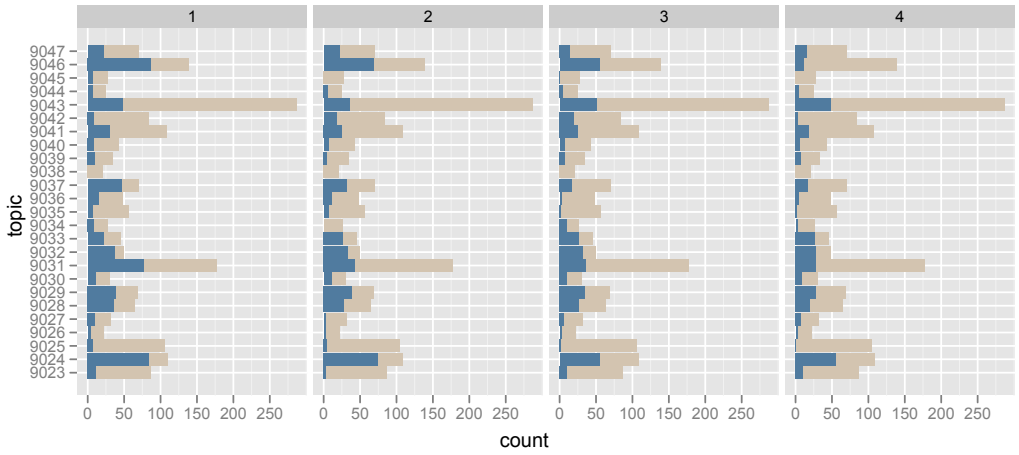

Figure 9: Comparison of the number of relevant videos with the number of saved (returned) videos for each of the four AXES runs. The number of saved videos are shown as dark blue bars; the total number of relevant videos are shown as light brown bars.

9038 "Female presenter X." These searchers did, however, save an average of 12 videos for this topic, implying that searchers may have misunderstood the topic (i.e. they were searching for videos of female presenters in general), or may have been saving shots of a person that they believed to be the female presenter featured in the topic example images. A similar conclusion can be inferred from the results for topic 9042 "Male presenter Y" – for example, in run 1 the participant saved 46 separate videos, of which only 9 were judged relevant.

Figure 9 shows a more detailed plot of the proportion of relevant videos found by the experiment participants in each of four runs. Each bar in this plot represents the performance of a single user on a single topic. Clearly, there can be dramatic variation in user performance: one participant found almost 100 relevant videos for topic 9046 (run 1), another found less than 25 (run 4).

Figure 10 shows the relative proportions of relevant and non-relevant videos saved by each participant. Topic 9038 and 9042 (male and female presenter X and Y) again stand out as having many non-relevant videos saved across all participants. The recall-oriented group (run 1) clearly have more false-positives than the other three groups, but nevertheless performed better in terms of MAP and bpref.

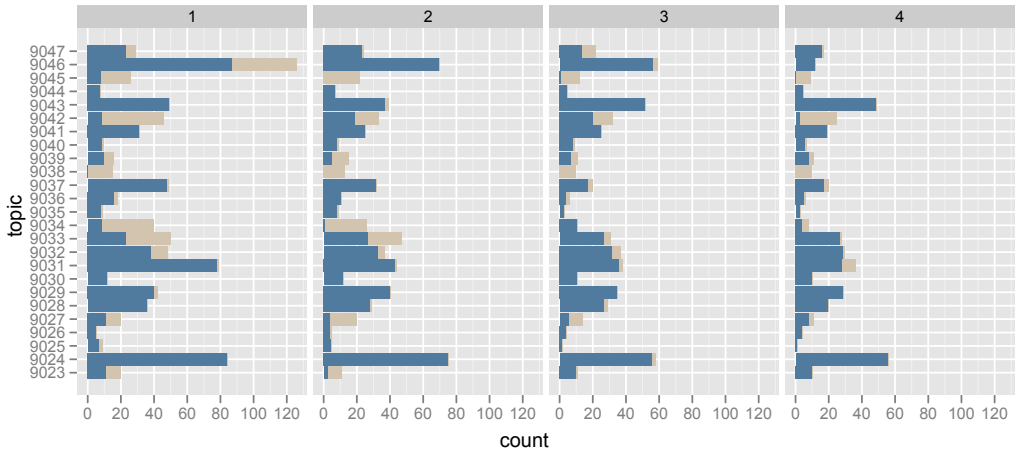

Figure 10: Plot showing the relative proportions of relevant and non-relevant videos saved by each participant by topic. The dark blue bars represent the number of relevant videos; the light brown bars represent the number of non-relevant videos.

3.3 User Feedback

Survey forms were filled in after both the KIS and INS tasks. The survey consisted of a blank form in which respondents could describe their experiences. The following summarizes the user comments and feedback.

Both the respondents of the KIS and INS were very positive about the interface. It responded quickly, the design was perceived as intuitive, and we received very positive comments on the drag-and-drop based interaction style. Several remarks were made about the size of the videos: participants remarked that they preferred to be able to adjust the size to fit their needs. Participants also commented that it was unclear whether Boolean search terms could be used, and that the operation of the videos could be more intuitive (users had to click a close button to finish watching the video; several commented that they would rather simply click somewhere outside the video frame). While the participants of the KIS experiment were experienced archival searchers, participants in the INS task were students and not particular familiar with searching in archives. Several INS respondents remarked that they did not fully understand visual similarity search or concept search. Some participants commented that interface lacked some features that could make the work easier. For example, it was not possible to select several clips at once and it was unclear which clips had already been watched. For some it was unclear how many shots could be saved and the functionality of the trash area was unclear.

Several KIS and INS participants commented that they would prefer if the system gave

them a better idea of how the results were created and why certain items appeared in the ranked list. Typical questions included: 1) is every item that is shown relevant? 2) why are some results ranked higher than others? 3) why is this particular result judged as relevant by the system? Some participants remarked they did not know how the system worked and were not able to learn the system to adjust their search strategy. In line with this remark, several users noted that when more search terms were added, more results were shown, but that they had expected that the number of results would decrease. This is likely based on their experiences with other media retrieval systems that use Boolean logic for textual search. One user noted that it would be useful if available metadata were shown besides a clip. Participants that tended to search using text found it difficult to judge results without contextual information.

4 Conclusions

This paper described the AXES participation in the interactive KIS and INS tasks for TRECVid 2011. Our system for both tasks used a near identical user interface and the same system backend. Both INS and KIS made use of text search, visual similarity search, and visual concepts, and used a simple linear fusion mechanism to combine the results. We used Apache Lucene for the text search engine, a non-linear χ^2 SVM trained on key-frames represented by a pyramid histogram of visual words for concept classification, and an engine based upon SIFT visual words for the visual similarity search. The user experiments were carried out at NISV in Hilversum with 14 media professionals participating in the KIS task and 30 media students participating in the INS task.

Our best run for the KIS task performed around the median, with our participants finding 11 of the 25 videos. AXES was the only group to submit runs for the interactive INS task; our best run MAP was 0.33. We got considerable user feedback on the task and user interface; participants commented that the system was intuitive and responsive, and gave valuable feedback on how the system could be improved. Our experiments showed considerable variation in topic difficulty and user performance for both the INS and KIS tasks. We plan to improve the interface in accordance with user feedback and use a more sophisticated fusion scheme in future TRECVid participation.

Acknowledgements

We would like to thank Erwin Verbruggen and Roeland Ordelman for organizing the experiment at the Netherlands Institute for Sound and Vision and recruiting the participating media professionals, and to thank Franciska de Jong and Marta Stachowiak of the Erasmus Studio for recruiting the students to participate in the experiments. We would also like to acknowledge everyone that participated in the experiments, both the media professionals and the visiting students. This work was funded by the EU FP7 Project

AXES ICT-269980. We are grateful to the UK-India Education and Research Initiative (UKIERI) for financial support.

References

[1] R. Aly. *Modeling Representation Uncertainty in Concept-Based Multimedia Retrieval*. PhD thesis, University of Twente, Enschede, July 2010.

[2] R. Aly, C. Hauff, W. Heeren, D. Hiemstra, F. de Jong, R. Ordelman, T. Verschoor, and A. de Vries. The lowlands team at TRECVid 2007. In *Proceedings of the 7th TRECVid Workshop*, Geithesburg, U.S., February 2007. NIST.

[3] R. Aly, D. Hiemstra, A. P. de Vries, and H. Rode. The lowlands team at TRECVid 2008. In *Proceedings of the 8th TRECVid Workshop*, 2008.

[4] A. Bosch, A. Zisserman, and X. Munoz. Image classification using random forests and ferns. In *International Conference on Computer Vision*, 2007.

[5] C. Buckley and E. M. Voorhees. Retrieval evaluation with incomplete information. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 25–32, 2004.

[6] C. Foley, J. Guo, D. Scott, P. Wilkins, C. Gurrin, A. F. Smeaton, P. Ferguson, K. McCusker, E. S. Diaz, X. Giro-i-Nieto, F. Marques, K. McGuinness, and N. E. O'Connor. TRECVID 2010 Experiments at Dublin City University. In *Proceedings of the 10th TRECVid Workshop*, Gaithersburg, USA, 2010.

[7] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Conference on Computer Vision and Pattern Recognition*, June, 2006.

[8] D. Lowe. Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision, 60(2):91-110*, 2004.

[9] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[10] S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for svm. In *Proc. of the International Conference on Machine Learning*, 2007.

[11] J. A. Shaw, E. A. Fox, J. A. Shaw, and E. A. Fox. Combination of multiple searches. In *The Third Text REtrieval Conference (TREC-3)*, pages 243–252, 1994.

[12] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the IEEE International Conf. on Computer Vision (ICCV)*, 2003.

[13] L. C. D. Team. *Lucene 3.2.*

[14] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

[15] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[16] S. Vempati, M. Jain, O. M. Parkhi, C. V. Jawahar, M. Marszalek, A. Vedaldi, and A. Zisserman. Oxford-iiit trecvid 2009 notebook paper. In *Proceedings of the 5th TRECVid Workshop*, 2009.

[17] P. Wilkins, D. Byrne, G. J. F. Jones, H. Lee, G. Keenan, K. McGuinness, N. E. O'Connor, N. O'Hare, A. F. Smeaton, T. Adamek, R. Troncy, A. Amin, R. Benmokhtar, E. Dumont, B. Huet, B. Merialdo, G. Tolias, E. Spyrou, Y. Avrithis, G. T. Papadopoulous, V. Mezaris, I. Kompatsiaris, R. Mörzinger, P. Schallauer, W. Bailer, K. Chandramouli, E. Izquierdo, L. Goldmann, M. Haller, A. Samour, A. Corbet, T. Sikora, P. Praks, D. Hannah, M. Halvey, F. Hopfgartner, R. Villa, P. Punitha, A. Goyal, and J. M. Jose. K-Space at TRECVid 2008. In *Proceedings of the 8th TRECVid Workshop*, Gaithersburg, USA, 2008.

[18] R. Yan. *Probabilistic Models for Combining Diverse Knowledge Sources in Multimedia Retrieval*. PhD thesis, Carnegie Mellon University, 2006.