AXES @ TRECVid 2014: Instance Search

Shu Chen[1], Kevin McGuinness[1], Robin Aly[2], Noel O'Connor[1], Tinne
Tuytelaars[3]

[1]Dublin City University [2]University of Twente [3]KU Leuven

Abstract. The AXES project submitted three runs for instance search
at TRECVid 2014 [1]: two interactive runs and one automatic run. These
runs used exactly the same instance search engine as was used by AXES
in 2013, but incorporate a custom interface for the instance search tasks,
pseudo relevance feedback, and result list expansion. Unfortunately, these
changes did not improve accuracy significantly over our 2013 results in
the interactive task.

1 Introduction

The goal 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. Over the course of the project, we
have developed two integrated video retrieval systems: one aimed at professional
users (AXES PRO [2]), and one aimed at researchers and journalists (AXES
RESEARCH [3]). The project is currently in the process of implementing a third
system, AXES HOME, targeting the home user. These systems all allow users to
browse and navigate content is a variety of ways, including text search on auto-
matic speech recognition, on the fly visual concept [4] and face classification [5],
and query by example for faces and object instances [6].

Interactive instance search is an integral and heavily used component in all
three systems. As such, the AXES project has participated in the interactive
instance search task at TRECVid 2011 [7], 2012 [8], and 2013 [9], so as to bench-
mark our implementation against the state-of-the-art. In 2012 and 2013, we used
our AXES PRO and AXES RESEARCH systems directly for participation in
the task. This approach allowed our experiment participants to take advantage
of various other tools, like text search on metadata and on the fly visual concept
classification, in addition to our instance search feature. We observed that al-
though users did occasionally use some of these features, direct instance search
using the provided examples combined with instance search as a form of relevance
feedback, was by far the most popular way of interacting with the system.

As we had already evaluated the PRO and RESEARCH interfaces in previous
years we decided this year to use a custom interface tailored to the TRECVid
INS task that uses only our instance search technology. This technology, which is
based on INRIA's BigImbaz engine [6], is exactly the same as we used in 2013.
The idea was to check if a custom tailored TRECVid interface, combined with
some other optimizations like pseudo relevance feedback and subsequent result

list expansion, could improve our performance over previous years. Unfortunately, neither of these optimizations significantly improved results over previous years.

The paper is structured as follows. Section 2 briefs the design of AXES INS submissions. Section 3 provides the details of query expansion implementation for INS task. Section 4 describes the user interface design for interactive INS task. Section 5 presents the experiment analysis and Section 6 concludes the paper.

2   Instance Search

This section describes the system for this year's INS participation. Our instance search system uses query expansion strategy for both interactive and automatic runs. The central component of the INS system is BigImbaz, which implements visual instance retrieval as described in [6]. To improve search quality, a query expansion strategy is used to create a fused retrieval list. In the interactive task, the list is presented to users for further annotation. In the automatic task, the list directly creates the submission runs. In the following, we describe the query expansion strategy and interactive user interface design.

3   Query Expansion

The central component in all AXES systems is the link management and structured search engine (LIMAS), which is responsible for maintaining metadata about indexed videos, and federating search to other services such as concept classifiers and query by example engines. Although supporting multiple functionalities to search videos, this year, we only use query by example based instance search (BigImbaz).

We divide the INS search into two steps to implement query expansion. The first step is initial retrieval. Using the example images provided by the topic description, instance search is implemented by BigImbaz by extracting interest points from keyframes using a Hessian detector and computing a CS-LBP descriptor for each keyframe. It refines the searching by involving a quantization index, Hamming Embedding and burstiness [10]. For each run, all sample keyframes are used as queries. Moreover, we use the mask information provided by TRECVid to extract expanded mask keyframe, shown in Figure 1. The mask image indicates the rectangular area from the sample keyframe query and the expanded mask keyframe is created according to the corresponding area. Both the sample keyframe and expanded mask keyframe are used for run queries. The initial retrieval stage involves multiple queries for each run. The retrieval list is created by fusing all ranked lists. For each run R_i, define its query set $Q = [q_1, q_2, \ldots, q_i]$, which containing both sample and expanded mask keyframes. The score of a retrieved video segment s according to each query is defined as instance(q_i, s). Fusion function to calculate the ranking score is Equation 1. Sorting the ranking score of each video segment creates the initial retrieval list.
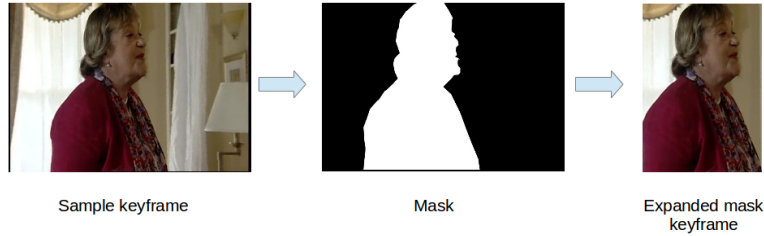
Sample keyframe   Mask   Expanded mask keyframe

Fig. 1: Create a mask keyframe from a sample keyframe. Both of them are used as INS queries

$$rankingScore(s) = \sum_{1}^{i} instance(q_i, s) \tag{1}$$

The second step is query expandion on initial retrieval list. We select the top-N results which are assumed to be relevant to the current topic as additional queries. The INS system uses these top-N results to retrieve a new group of scored lists. No mask information is associated with these expanded queries. For each run, define the keyframe of top-N results in the initial retrieval list as $K = [k_1, k_2, \ldots, k_N]$. The fusion function to calculate the ranking for query expansion retrieval uses Equation 2. The submission results are created using the query expansion retrieval list. The query expansion strategy is used in both interactive and automatic task. The former automatically determines input query according to sample videos and top-N results in the initial retrieval list, while the latter allows the user to select additional query images.

$$rankingScore(s) = \sum_{j=1}^{N} instance(k_j, s) \tag{2}$$

4 User Interface Design

The user interface used for INS interactive task was developed using the Python Django framework[1]. The implementation of user interface uses HTML5, CSS3, and Javascript. The client-side is hosted using nginx[2]. Figure 2 show the screenshot of AXES INS interactive interface. It is composed of four panels: the *information panel*, the *query panel*, the *retrieved result panel*, and the *saved result panel*.

1) Information Panel The information panel presents the INS topic information to the user including the task description and instance type. A total of 4 sample videos are presented on the right side. Each video is represented by its
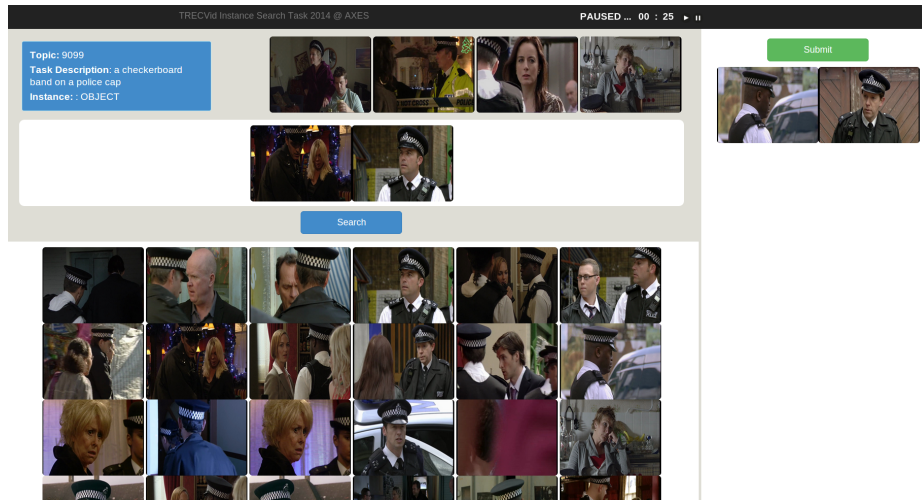
---

[1] https://www.djangoproject.com/
[2] http://nginx.org/

Fig. 2: AXES INS User Interface

keyframe. When the mouse moves over, the interface shows "Add" and "Play" buttons floating over the corresponding keyframe. The "Play" button allows users to watch the full video segment. Moreover, the "Add" button allows users to save the current segment as a new query.

2) Query Panel The query panel (white background under the information panel) contains all queries users have been saved for further searching. Each video segment is represented using its keyframe. The query panel can save as many additional relevant images as the user finds. The AXES INS system supports multiple query searching, using the strategy discussed in Section 2. The user interface provides "Remove," "Save," and "Play" button for each saved keyframe. Users can check the video content, remove those no longer required or save this keyframe for submission.

3) Retrieved Results Panel The retrieved result panel presents search results according to the query keyframes. All retrieved videos are represented by their keyframes. The user interface provides "Add," "Save," and "Watch" buttons when the mouse hovers over a keyframe.

4) Saved Result Panel The saved results panel stores all the video segments supposed to be relevant to the topic in term of video keyframe. Users can save a keyframe from the query panel or retrieved result panel. The user interface provides "Remove" and "Play" button associated with each keyframe. The "Submit" button is located at the top of saved result panel. Users are free to submit all saved results and jump to the next topic at any time. In the experiment, we encouraged each user use the entire 15 minutes on each INS topic.

There is the timer over the information panel. Users are allowed to pause or continue the INS task at any time. The system status is presented at the left side, labelled as "Paused," "Searching," and "Saving". No labelled information
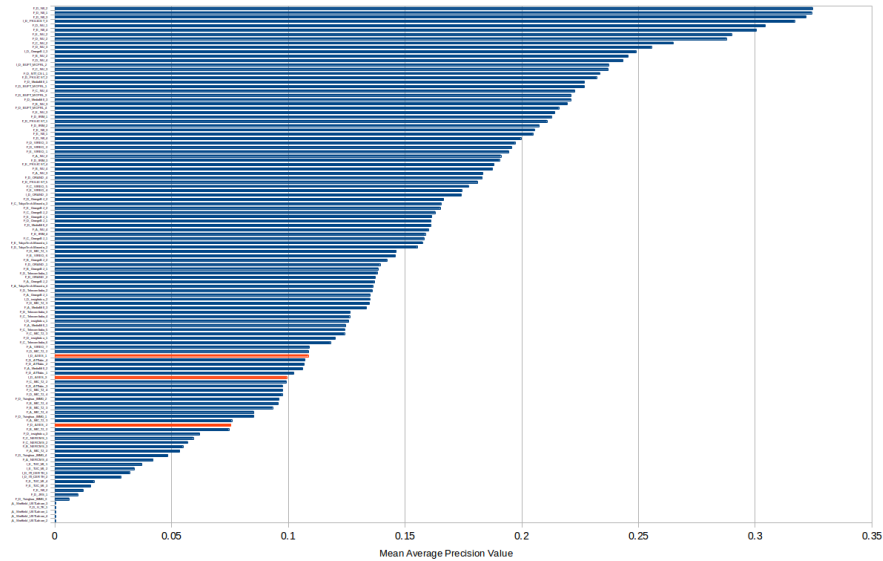
Fig. 3: Mean average precision over all submitted INS runs. AXES runs are shown in red.

means the system is running INS task. Under any labelled status, the system locks all buttons to avoid some conflicts. The time limitation for each topic is set to be 15 minutes. The system can automatically submit the saved results presented in the saved result panel when users run out the time.

Each participant in INS interactive task is automatically guided to next available topic. When a new topic starts, the interface reads pre-process automatic INS results from the server and presents all ranked keyframes in the retrieved result panel. Then users can continue searching until the time elapses or by manually submitting the saved results.

5   Experiments

The instance search experiments were carried out at Dublin City University. A total of 12 people participated in the experiments. Participants are primarily research assistants, students, and postdoctoral researchers. Each participant had 15 minutes to finish each topic and was assigned 6 topics in total. Participants were briefed on the purpose of the experiment and shown how to operate the user interface. A total of three runs were submitted:

1. **F_D_AXES_2** a run for automatic searching evaluation;
2. **I_D_AXES_1** the first run for interactive searching evaluation;
3. **I_D_AXES_3** the second run for interactive searching evaluation.

All the evaluation results are created in terms of the metrics described in [11].
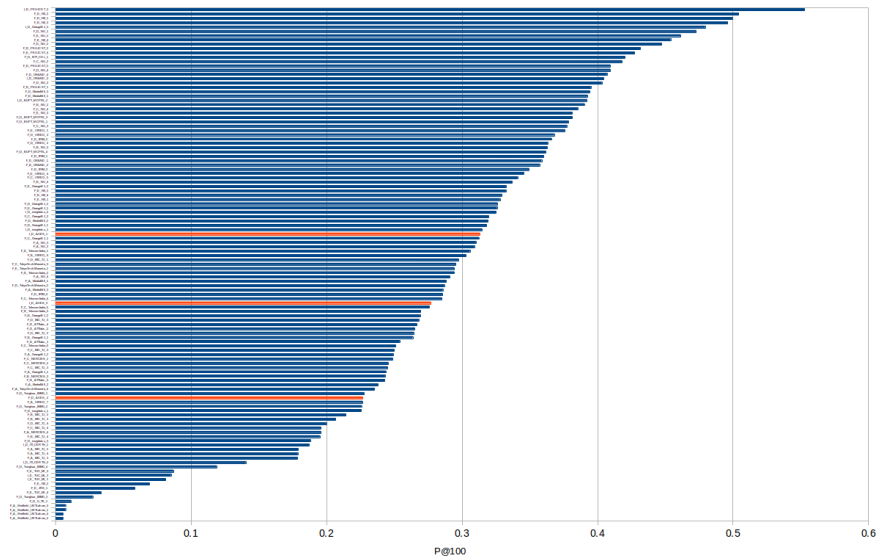
Fig. 4: Number of correct results out of the first 100 results returned for all submitted INS runs. AXES runs are shown in red.

Figure 3 illustrates the Mean Average Precision (MAP) value of all submissions, with AXES results are marked in red and others are marked in blue. Our best run, I_D_AXES_1, achieved an mAP of 0.108 (median 0.11). In previous years, our results were near to or above the median in terms of mAP. This year, our results are below. This is due to a general improvement in the INS results overall, and due to more submissions by other institutes. Our overall mAP values for our interactive runs this year (0.108 and 0.099) are similar to our runs from previous years (0.135, 0.086, 0.079). Figure 4 plots the P@100 value of all submissions. Also, AXES results are presented in red and others are in blue. Run I_D_AXES_1 achieves the median performance in terms of P@100.

Figure 5 illustrates the relative proportion of results returned by the experiment participants in each run. Each topic in the runs is completed by a single participant, represented by each bar. Considering Figure 4, the retrieval results of both interactive and automatic runs can achieve relatively high precision. The plot indicates mAP decreasing is caused by low recall. The topic difficulty varies significantly according to Figure 5, especially **TOPIC 9118**, in which there are a total of 4 relevant video shots defined in the ground truth. All AXES submissions found only 2 of them.

Our interactive results (mAP 0.108 and 0.099) this year did not improve significantly over last year (mAP 0.135, 0.086, 0.079) which suggests our simple query expansion mechanism, result list supplementation, and tailored user interface do not improve results much, at least in terms of mAP. The variation across runs is due to differences in user skill alone. Interestingly, our automatic run this
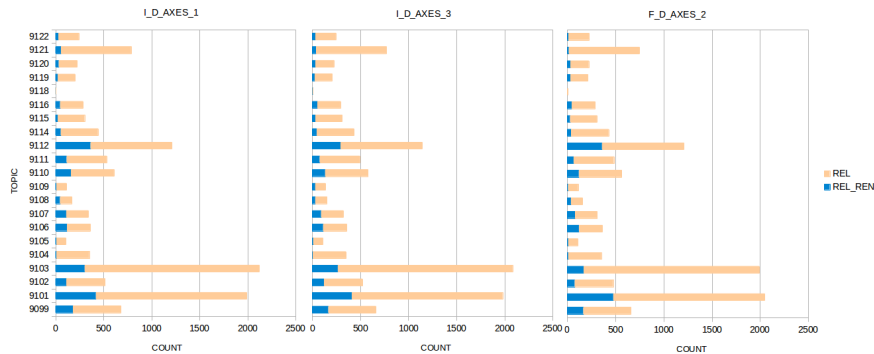
Fig. 5: Comparison of the number of relevant videos (Label: REL) with the number of saved (returned) videos (Label: REL_REN) for each of the three AXES runs. The number of saved videos are shown as light blue bars; the total number of relevant videos are shown as light brown bars.

year (mAP 0.075) performed about as well as our worst user group did last year in the interactive task.

6 Conclusion

This paper described the AXES project's participation in the instance search task at TRECVid 2014. This year we submitted two interactive runs and one automatic run. We used the same instance search engine as last year with a custom UI tailored for instance search, and augmented the results with pseudo-relevance feedback and result list supplementation. Unfortunately, these changes did not improve our results significantly over previous years. Our query expansion strategy was, however, very simple, and a more sophisticated approach would likely improve results more. Our automatic run this year used a similar strategy to our interactive runs, and performed about as well as our one of our interactive user groups did last year.

Acknowledgements

References

1. P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quenot, "TRECVID 2014 – an overview of the goals, tasks, data, evaluation mechanisms and metrics," in *Proceedings of TRECVID 2014*, NIST, USA, 2014.

2. K. McGuinness, N. E. O'Connor, R. Aly, F. De Jong, K. Chatfield, O. M. Parkhi, R. Arandjelovic, A. Zisserman, M. Douze, and C. Schmid, "The AXES PRO video search system," in *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, pp. 307–308, 2013.

3. K. McGuinness, R. Aly, K. Chatfield, O. Parkhi, R. Arandjelovic, M. Douze, M. Kemman, M. Kleppe, P. Van Der Kreeft, K. Macquarrie, *et al.*, "The AXES research video search system," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2014.

4. K. Chatfield and A. Zisserman, "VISOR: Towards on-the-fly large-scale object category retrieval," in *ACCV 2012*, vol. 7725 of *Lecture Notes in Computer Science*, pp. 432–446, 2013.

5. O. M. Parkhi, A. Vedaldi, and A. Zisserman, "On-the-fly specific person retrieval," in *Image Analysis for Multimedia Interactive Services (WIAMIS), 2012 13th International Workshop on*, pp. 1–4, 2012.

6. H. Jégou, M. Douze, and C. Schmid, "Improving Bag-of-Features for Large Scale Image Search," *International Journal of Computer Vision*, vol. 87, pp. 316–336, Feb 2010.

7. K. McGuinness, R. Aly, S. Chen, M. Frappier, K. Martijn, H. Lee, R. Ordelman, R. Arandjelovic, M. Juneja, C. Jawahar, *et al.*, "AXES at TRECVid 2011," 2011.

8. D. Oneata, M. Douze, J. Revaud, S. Jochen, D. Potapov, H. Wang, Z. Harchaoui, J. Verbeek, C. Schmid, R. Aly, K. Mcguiness, S. Chen, N. O'Connor, K. Chatfield, O. Parkhi, R. Arandjelovic, A. Zisserman, F. Basura, and T. Tuytelaars, "AXES at TRECVid 2012: KIS, INS, and MED," in *Proceedings of TRECVID 2012*, NIST, USA, Nov 2012.

9. R. Aly, R. Arandjelovic, K. Chatfield, M. Douze, B. Fernando, Z. Harchaoui, K. McGuinness, N. E. O'Connor, D. Oneata, O. M. Parkhi, *et al.*, "The AXES Submissions at TRECVid 2013," 2013.

10. H. Jégou, M. Douze, and C. Schmid, "On the Burstiness of Visual Elements," in *Conference on Computer Vision & Pattern Recognition*, jun 2009.

11. A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and TRECVid," in *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, (New York, NY, USA), pp. 321–330, ACM Press, 2006.