Technische Universität Chemnitz and Hochschule Mittweida at TRECVID Instance Search 2017

Stefan Kahl[1], Daniel Richter[2], Christian Roschke[1], Manuel Heinzig[1], Danny Kowerko[1], Maximilian Eibl[2], and Marc Ritter[1,3]

[1]Junior Professorship Media Computing, Technische Universität Chemnitz, D-09107 Chemnitz, Germany
[2]Chair Media Informatics, Technische Universität Chemnitz, D-09107 Chemnitz, Germany
[3]Professorship Media Informatics, University of Applied Sciences Mittweida, D-09648 Mittweida, Germany

Abstract. With our submission to the 2017 TRECVID Instance Search task (Awad et al., 2017b), we focused on the usage of dedicated CNN models. We limited the available video training data to only three sources: The EastEnders episode 0, the given location examples and the given person examples. Our main workflow consists of three steps: First, we identify and crop persons from the training videos. Secondly, we train person and location classifiers using CNNs. Finally, we apply an ensemble strategy with prediction pooling to find matches for the given topics. In this contribution, we will provide some insights into our strategies and discuss results. Additionally, we present a novel way of interactive result annotation using HTC Vive VR headsets.

1 Structured Abstract

1. *Briefly, list all the different sources of training data used in the creation of your system and its components.*

- We used the given master shot reference, the first episode with ID 0 (also denoted as DEV0 in this contribution) from the provided *BBC EastEnders* video footage as well as the location and person video examples. Additionally, we used textual metadata crawled from the BBC website containing episode descriptions. No other external training data was used.

2. *Briefly, what approach or combination of approaches did you test in each of your submitted runs?*

- F_E_TUC_HSMW_1: Dedicated model ensemble for person and location classification. The results of this run are re-ranked by similarity group scores.
- F_E_TUC_HSMW_2: Dedicated model ensemble like our first run, no re-ranking.
- F_E_TUC_HSMW_3: Faster-RCNN for person detection and classification. This is our best system from 2016.
- I_E_TUC_HSMW_4: Our only interactive run. Result re-ranking of our first run. We used a novel VR environment for the annotation process.

Correspondence to: Stefan Kahl
stefan.kahl@informatik.tu-chemnitz.de

3. *What if any significant differences (in terms of what measures) did you find among the runs?*

- In contrast to last year's submission, similarity group scores did not benefit the results.
- We managed to retrieve 20% more relevant shots compared to our 2016 system.
- Interactive re-ranking of results did not boost the overall scores as much as we expected.

4. *Based on the results, can you estimate the relative contribution of each component of your system/approach to its effectiveness?*

- Person and location detection can be done using small, dedicated datasets for a closed domain environment like the EastEnders universe.
- Simple CNN architectures are easy to train on small data sets but lack high generalization performance.
- CNN ensembles perform significantly better than single models.
- Interactive VR environments are suitable for interactive result-re-ranking but require more time than traditional annotation tools.

5. *Overall, what did you learn about runs/approaches and the research question(s) that motivated them?*

- Increasing the generalization performance on small training sets is very challenging and requires extensive hyper-parameter tuning.
- Di erent learning strategies aside from classification tasks might improve the performance (e.g. matching images of persons for similarities).
- Ensemble strategies are strong and render some other techniques (e.g. similarity clustering) obsolete.

The remainder of the paper is organized as follows: First, we provide a short workflow overview in section 2. After that, section 3 contains some insights into our training dataset. In section 4 we give a summary of our training process using artificial neural networks. Section 5 presents a novel approach for interactive evaluation on the EastEnders dataset. Finally, in section 6 we discuss the results of our submission.

2 Workflow Overview

Our workflow for this year's submission to the TRECVID Instance Search task (Awad et al., 2017a), (Awad et al., 2017b) derives directly from our approach presented in 2016 (Kahl et al., 2016b). We follow the idea of dedicated models trained solely on images and videos extracted from the DEV0 episode an the given task examples even further. We wanted to use Open Source frameworks and toolkits as well as consumer hardware to explore the possibilities and limitations of this attempt for real-world applications. Our workflow consists of three main steps

- First, we train a custom human detection network based on Faster-RCNN (Ren et al., 2015) and crop persons from training and test samples.
- Secondly, we train person and location classifier based on a custom CNN architecture.
- Finally, we apply an ensemble strategy with prediction pooling to find matches for the given topics. Some of our runs contain an additional re-ranking of the result list based on similarity clusters for groups of frames.

Our approach is to some degree very straight forward and does not feature sophisticated strategies for person similarity matching. However, we show that even simple techniques can provide satisfying results if the setting for training and prediction is chosen carefully.

3 Dataset

We strictly limited our video and image training data to EastEnders samples only. This also excludes pre-trained networks for Faster-RCNN trained on ImageNet. By choosing the right training parameters, we can overcome most of the di culties that arise with that restriction. However, we might not be able to achieve top performing results within these boundaries.

We added one additional, text-based metadata source to this year's attempt. The BBC hosts episode descriptions of every single EastEnders episode contained in the 244 omnibus episodes of the dataset on their homepage. These descriptions include listings of characters present in each episode. Most listings are not completely accurate but can be used to determine the range of episodes in which a character can be detected before his or her departure. For future reference, we include a visualization of the metadata in the appendix of this contribution (see Figure 4).

In summary, our training data includes following sources (all annotations were done manually):

- DEV0 episode containing 33 notable locations and 28 re-appearing characters. This episode misses the locations 'cafe2', 'pub' and 'market'. It also misses three of the eight targeted characters: Janine, Ryan and Archie.
- Example videos for ten locations and eight characters provided by the organizers. We merged the given examples with our annotations of the DEV0 episode.
- Textual episode descriptions crawled from the BBC website containing lists of character appearances per episode. We used these metadata to determine start and end episodes for each character.

We extracted the person samples using our custom human detection (more details in section 4). We decided to use three frames per second from every given video file. Our final training set contains the following amount of samples per class:

- Person crops: Archie (107), Billy (908), Ian (1098), Janine (105), Peggy (1420), Phil (1687), Ryan (25), Shirley (1171), Other (2782)
- Location frames: Cafe1 (2373), Cafe2 (2579), Foyer (1568), Kitchen1 (2752), Kitchen2 (577), Laundrette (1008), Livingroom1 (1398), Livingroom2 (1955), Market (1674), Pub (1270), Other (8499)

In some cases it might be helpful to add a class containing negative samples to lower confusion with unknown test samples. We decided to merge all person and location samples not part of the given training data into one class by randomly picking frames and crops from those classes.

4 Training

Due to the constraint to use only limited training data, we had to overcome some di culties during training. In the end, we decided to train three di erent classifiers. The first classifier is a custom Faster-RCNN model which retrieves

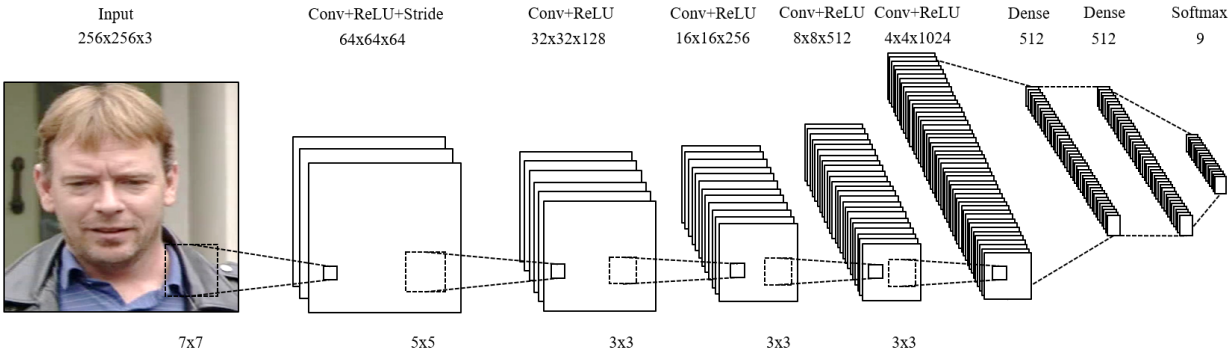

Figure 1. CNN architecture of our approach. Simple CNN layouts are easier to train with small datasets. Most state-of-the-art networks need pre-training to get good generalization with limited training data. Our net contains eight weighted layers, every convolution has batch normalization, is ReLu activated and followed by 2x2 max pooling.

bounding boxes of humans from the EastEnders dataset. Secondly, we used those person detections to train a simple, custom CNN for character recognition. The third classifier recognizes locations based on the example data described earlier.

Training custom Faster-RCNN:
As described in our 2016 working notes, we annotated all main characters in episode DEV0 to train a Faster-RCNN model which can classify individual characters. This year, we merge those annotations into just one class to train a person detection model[1]. Since we did not use any pre-trained model and no additional training data, this approach does not achieve maximum performance. However, results are still very good, containing only few false detections. In total 25856 bounding box annotations were used to train this model.

Person and location classification:
Simple CNN architectures are easier to train and fine-tune than most of the state-of-the-art models with multiple tens of layers for small datasets. Additionally, using a wide layout with many filters in every convolution benefits regularization with dropout. We experimented with such simple (or better: traditional) architectures for our BirdCLEF2017 submission (Kahl et al., 2017). Figure 1 provides a detailed visualization of the net we used for this contribution. The main characteristics of our model are:

- An input size of 256×256 pixels. We re-scale person crops to fit the input size. We use batches of 16 random quadratic crops from the source image for the training of the location classifier.
- Large receptive fields (kernel sizes of 7×7 and 5×5) in the first two convolutional layers.
- Large number of filters, doubling in size with every convolution with a maximum of 1024 feature maps in the last convolutional layer.
- We decided to go with batch normalization (Io e and Szegedy, 2015) and ReLU activation (Nair and Hinton, 2010) which is current best practice for numerous image recognition tasks. All layers are He-initialized (He et al., 2015) and we applied dropout to every dense layer. Every convolutional layer is followed by a 2×2 max pooling layer.

We use the Adam optimizer (Kingma and Ba, 2014) with a starting learning rate of 0.001. We interpolated the learning rate after each epoch to drop it to 0.000001 after 75 epochs of training. We augmented all input images at run time using horizontal flips and random crops.

Our implementation was done using Theano (Theano Development Team, 2016) and Lasagne (Dieleman et al., 2015) on a single machine with a NVIDIA P6000 GPU for training and Titan X GPU for inference.

5 Interactive Evaluation

Over the course of the last years, our approach to the interactive runs proved itself very successful. Therefore, we decided positively on a submission for this years competition. Analogous to our prior e orts, the evaluation module we used in the preceding year was brought to a new iteration. The basic concept proposed in (Ritter et al., 2015) and (Kahl et al., 2016a) is still untouched, but this time we added VR in form of using an HTC Vive headset and corresponding controllers.

Former approaches focused on displaying a maximum number of images on screen and enabled annotators to interact with them in an easy and task-adjusted way, so that a large number of evaluated images per time slot could be achieved. We then ported the key aspects of the system to a web application, which allowed us to collaborate as a team,

[1] Faster-RCNN automatically uses rejected region proposals as background samples, this actually makes this a two-class task.

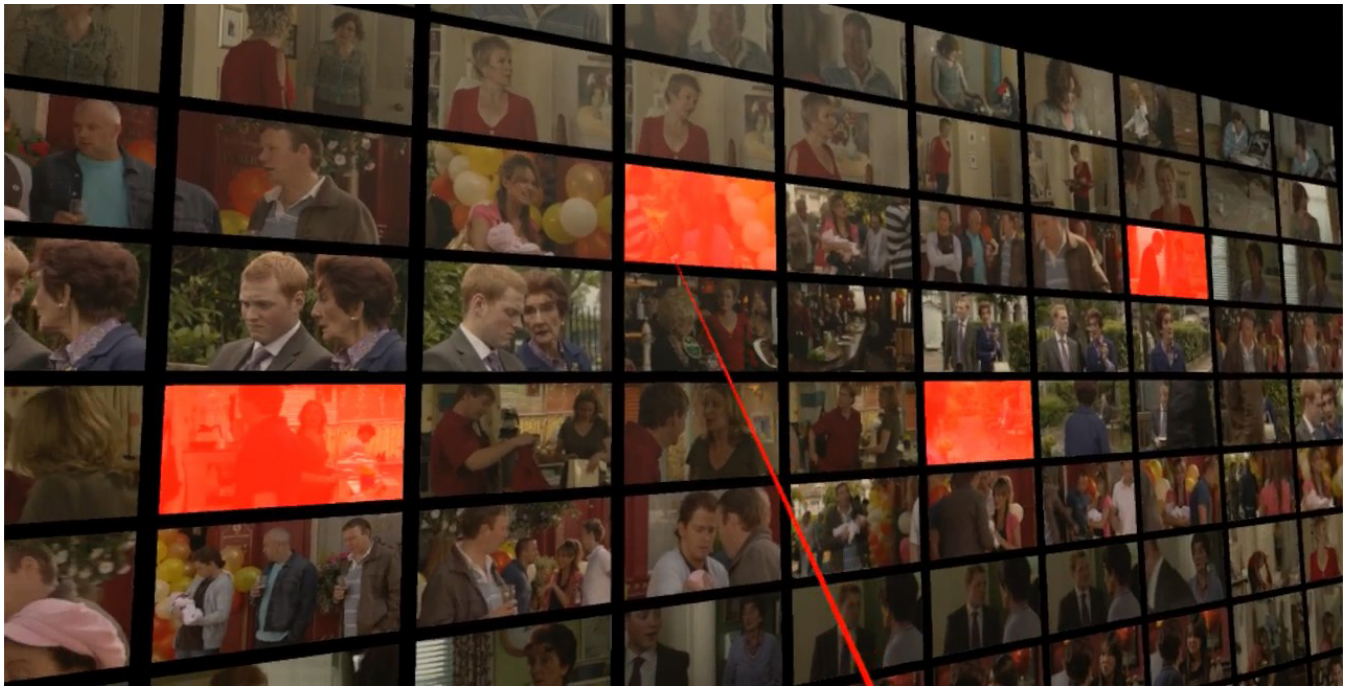

Figure 2. Interactive evaluation interface. The person annotating the dataset faces a 2D-projection of 144 preliminary results and uses a point-and-shoot control to mark false detections. Head movement is tracked and is used for 3D-navigation, hand-held controllers are used as pointers.

easily administrate the whole process and aggregate results in a very e cient way. Consequently, we wanted to achieve a further improvement of the evaluation rate this year. We saw that the limits in 2D-Space were reached. With an evaluation speed of approx. 200ms per image and a nearly optimal ratio of displayed image size to available resolution, the classical way of visualizing and interacting with optical information on-screen has seen its final call.

The logical consequence was to pursue the emerging topic of virtual reality. Work conducted in (Heinzig, 2017) provided a solid foundation on the topic of image evaluation in this new domain of visualization and interaction methodology.

5.1 Basic idea

A virtual reality system such as the used HTC Vive excels in displaying contents to users with a big degree of immersion. This e ect is mainly achieved by placing two screens directly in front of the users eyes, hence widening the utilized amount of the users natural field of view (FOV). In the works of (Heinzig, 2017) the theoretical foundation of exploiting this feature for the application of fast image evaluation has been investigated with promising results. We therefore decided to use a similar system in our first attempt to further improve the quality and speed of our interactive testing with the help of this emerging technology.

There is one main advantage of the enhanced FOV compared to a standard computer screen. While the human visual system can normally perceive a field of 160° (horizontal) × 135° (vertical) originating from the eyes, a standard 24" display only covers up a 45° × 30° area when viewed from the recommended distance of 85 cm. This greatly limits the space in which we can project relevant information such as images. By using the HTC Vive however, those limits are exceeded by a magnitude, providing a FOV of 145° × 100° . The idea is to use this FOV to perceive all images "as a whole", not requiring a great deal of eye movement while scanning them for specific contents. This concept is backed by the findings in (Heinzig, 2017), where peripheral vision was used by many of the testers to judge the given images, without actually requiring their vision system to do the timely expensive process of refocusing eyesight on single pictures. The fact that this system is only applicable to one user at a time is of no consequence, because the rules of the interactive task changed over the course of the last year, so that only one person is allowed to do the manual evaluation anyways.

5.2 The evaluation process

Our tests were done with the standard setting from the best experimental setup of the mentioned underlying thesis. We displayed 144 images on a 2-dimensional projection screen in virtual space at the same time. In order to enable the user to interact with elements in this virtual environment, we also displayed the controller and a laser beam originating from it. The laser operated as a mouse pointer in 3-dimensional

space and enabled the user to select an image and therefore assign it to the classes ”true” or ”false”. During the first annotation attempts it quickly became evident that the test person was overtaxed with this setting and thus the visualization with given parameters was unsuitable. This is partly due to the fact that only simple forms and black and white images were used in (Heinzig, 2017). In this case, however, color images and complex shapes had to be recognized. With the peripheral vision described in the thesis, this is not applicable to the current context. In order to capture all images, several head movements and the constant refocusing of the eyes were necessary. Furthermore, it was not possible to display fine-granular di erentiations, since the Vive depicts images partly blurred due to hardware limitations. For this purpose, we empirically figured out a suitable configuration, where the user was feeling comfortable of doing a longer annotation session. This led us to a configuration with 12 or 9 images in a 4:3 or 3:3 grid and correspondingly adjusted viewing distances. Those numbers are rather low when compared to the targeted counts, but still a bit higher than the 2-dimensional approaches. Therefore we still had hope for at least a small improvement.

5.3 Conclusion on VR

In retrospective, we have to conclude that the VR approach did not live up to the expectations. This obviously originates from the much more complex structure of information found in the TRECVid-Dataset when compared to data that was used in the foundational work. Originating from this we, on the one hand, were not able to make serious use of the basic principle that an expanded FOV raises the number of images simultaneously perceivable by the human vision system. On the other hand, limitations that come with using a new kind of (only recently developed) hardware significantly interfered with our plans. All VR Headsets on the market struggle to deliver a crystal clear image when the user tries to focus on a certain point. This occurs mainly due to the lack of eye-tracking in such devices, which hinders the capability of software tools to fully adapt to the users behaviour. Furthermore, the display technology used in such devices is also not perfect and introduces a subtle but visible and distracting grid of black squares when focusing eyesight on a particular detail.

As a consequence, evaluation speeds significantly decreased and even notably undercut those of last years two-dimensional iteration. However, the increased time spent on checking images actually led to a reduction of the human error, thus increasing the quality of acquired result data. This was measured as a sideline by the administrator supervising the evaluation process.

6 Results

Considering the (self-imposed) limitations of our training process, results were to some degree as expected. We managed to get high scores for topics containing characters present in the DEV0 episode as those characters had far more training samples than the ones not present in the DEV0 episode. We also achieved good results for obvious person/location combinations (e.g. Peggy in Livingroom2 in topic 9190 as the character's home contains this room). As expected, we did not perform well for topics containing Archie, Janine and Ryan (topics 9199 through 9208). Those characters did have only a few training samples and topics contained rare person/location combinations (e.g. Ryan at Kitchen2 in topic 9208).

6.1 Run 1: Model ensemble and re-ranking

With our first run, we were able to return 4206 of 10604 relevant shots at a mean average precision (MAP) of 0.151. This run scored a precision@100 of 0.2983 after re-ranking the results using similarity groups of frames. This is only our second best run, in contrast to last year, similarity groups did not improve the results as expected. More than half of the topics (13) do not contain more than 100 true positives in the ground truth, our model seems to persistently fail to rank those hits at a high position. Ensemble strategies do help to improve the results. Overall, our dedicated dataset might not be large and diverse enough to give good generalization results with simple model architectures.

6.2 Run 2: Model ensemble

The best run from this year's contribution does not feature similarity groups but equals run 1 in every other aspect. We were not able to return as much relevant shots as in run 1 with 4109 out of 10604, but managed to raise the MAP to 0.159. This run achieves a precision@100 of 0.3263 which is the highest of all our runs, even better than our interactive run.

6.3 Run 3: Faster-RCNN and re-ranking

This year's results might not be comparable to previous years due to the new topics. Therefore we included our last year's system in our 2017 submission. This attempt is trained exclusively on episode DEV0 for person recognition and is by design not capable to recognize characters not present in episode DEV0. This run returned 3409 of 10604 relevant shots, scoring a MAP of only 0.106 and a precision@100 of 0.2697. This indicates, that we were able to improve results for 'unknown' characters at least for some topics despite the few training samples. This year's ensemble strategy might have also benefited the quality of location recognition capabilities which greatly impacts the result ranking.

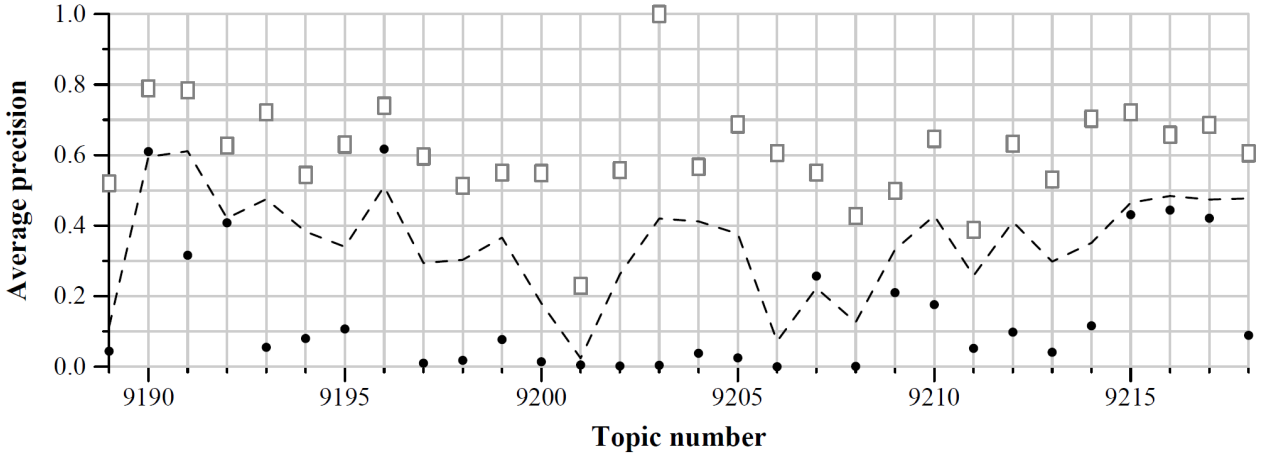

Figure 3. Results of our best fully automated submission (Run 2) provided by the organizers. Dots are our scores, the dashed line is the median score and boxes are best scores. We achieved compatible results for eight topics, most notably topic 9207 'Ryan at Laundrette' despite only 25 training samples for this character. Our worst performing topics mostly include rare person/location combinations such as topic 9197 'Ian at Laundrette' or characters not present in episode DEV0.

6.4 Run 4: Model ensemble and interactive re-ranking

Interactive result evaluation tends to boost the final scores by a large margin. However, this year's topics for the interactive task feature most of the rare person/location combination and misses two prominent characters. Our VR approach is more time-consuming than simple keyboard interaction (and time is limited to only 300 seconds per topic), which might be the reason for the lack of impact on the final results. This approach is based on our run 1 and retrieved 2549 of 7091 relevant shots with a MAP of 0.185 which is the highest of all of our runs. However, the precision@100 is lower than in our best run (run 2) with 0.3145 as the interactive topics feature 11 of 20 topics with less than 100 results in the ground truth.

7 Future Work

We will build upon our dedicated model approach in next year's Instance Search task but will shift our focus from classification towards similarity matching based on deep features. Systems like OpenFace (Amos et al., 2016) use deep feature vectors extracted from faces detected via facial key points to recognize people from only one source image. This approach is very robust for standard web-cam applications and should be a good starting point. However, we will try to overcome two limitations: First, the need of frontal faces to recognize a person and secondly the need for extensive training data.

As for the interactive evaluation, plans are to maybe take a step back by rekindling our 2D system. Since there is only little room for improvement on a standard PC-Monitor, we might be able to e ciently increase the number of perceivable images by using a wider screen or a high resolution video projector to better make better use of the users FOV. The idea of using Augmented Reality also came to our minds, but the available hardware currently has even more limitations for proper and fast interaction than the VR Headset used this year.

8 Additional Material

We published the metadata used for this work freely on Github. You can find episode descriptions crawled from the BBC website and character appearances based on the episode descriptions here:

https://gist.github.com/kahst

You can find videos demonstrating the interactive evaluation process in VR here:

https://youtu.be/Za_p77sM5gU
https://youtu.be/r43SQHFA4zo

For more information concerning metadata or the interactive evaluation process, please do not hesitate to contact the authors.

Acknowledgements. The European Union and the European Social Fund for Germany partially funded this research. This work was also partially funded by the German Federal Ministry of Education and Research in the program of Entrepreneurial Regions InnoProfileTransfer in the project group localizeIT (funding code 03IPT608X). Program material is copyrighted by BBC. We want to thank the organizers of this task, especially George Awad, for the hard work they put into the annotation, evaluation and organization of this challenge.

References

Amos, B., Ludwiczuk, B., and Satyanarayanan, M.: OpenFace: A general-purpose face recognition library with mobile applications, Tech. rep., CMU-CS-16-118, CMU School of Computer Science, 2016.

Awad, G., Butt, A., Fiscus, J., Joy, D., Delgado, A., Michel, M., Smeaton, A. F., Graham, Y., Kraaij, W., Qunot, G., Eskevich, M., Ordelman, R., Jones, G. J. F., and Huet, B.: TRECVID 2017: Evaluating Ad-hoc and Instance Video Search, Events Detection, Video Captioning and Hyperlinking, in: Proceedings of TRECVID 2017, NIST, USA, 2017a.

Awad, G., Kraaij, W., Over, P., and Satoh, S.: Instance search retrospective with focus on TRECVID, International Journal of Multimedia Information Retrieval, 6, 1–29, 2017b.

Dieleman, S., Schlueter, J., Ra el, C., Olson, E., Snderby, S. K., Nouri, D., Maturana, D., Thoma, M., Battenberg, E., Kelly, J., Fauw, J. D., Heilman, M., de Almeida, D. M., McFee, B., Weideman, H., Takcs, G., de Rivaz, P., Crall, J., Sanders, G., Rasul, K., Liu, C., French, G., and Degrave, J.: Lasagne: First release., doi:10.5281/zenodo.27878, http://dx.doi.org/10.5281/zenodo.27878, 2015.

He, K., Zhang, X., Ren, S., and Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE international conference on computer vision, pp. 1026–1034, 2015.

Heinzig, M.: Entwurf und Implementierung eines interaktiven Systems zur Evaluation und Optimierung von maschinellen Lernverfahren in der virtuellen Realität, Ph.D. thesis, Technischen Universität Chemnitz, master Thesis, 2017.

Io e, S. and Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International Conference on Machine Learning, pp. 448–456, 2015.

Kahl, S., Roschke, C., Rickert, M., Hussein, H., Manthey, R., Heinzig, M., and D. Kowerko, M. R.: Technische Universität Chemnitz at TRECVID Instance Search 2015, in: Proceedings of TRECVID Workshop, 2016a.

Kahl, S., Roschke, C., Rickert, M., Richter, D., Zywietz, A., Hussein, H., Manthey, R., Heinzig, M., Kowerko, D., Eibl, M., et al.: Technische Universitat Chemnitz at TRECVID Instance Search 2016, in: Proceedings of TRECVID Workshop, Gaithersburg, MD, USA, 2016b.

Kahl, S., Wilhelm-Stein, T., Hussein, H., Klinck, H., Kowerko, D., Ritter, M., and Eibl, M.: Large-scale bird sound classification using convolutional neural networks, Working notes of CLEF, 2017.

Kingma, D. and Ba, J.: Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980, 2014.

Nair, V. and Hinton, G. E.: Rectified linear units improve restricted boltzmann machines, in: Proceedings of the 27th international conference on machine learning (ICML-10), pp. 807–814, 2010.

Ren, S., He, K., Girshick, R., and Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, in: Advances in Neural Information Processing Systems (NIPS), 2015.

Ritter, M., Rickert, M., Juturu-Chenchu, L., Kahl, S., Herms, R., Hussein, H., Heinzig, M., Manthey, R., Richter, D., Bahr, G. S., and Eibl, M.: Technische Universität Chemnitz at TRECVID Instance Search 2015, in: Proceedings of TRECVID Workshop, Gaithersburg, Maryland, USA, 2015.

Theano Development Team: Theano: A Python framework for fast computation of mathematical expressions, arXiv e-prints, abs/1605.02688, http://arxiv.org/abs/1605.02688, 2016.