

Waseda Meisei at TRECVID 2017: Ad-hoc Video Search

Kazuya Ueki^{1,2}, Koji Hirakawa¹, Kotaro Kikuchi¹,
Tetsuji Ogawa¹, and Tetsunori Kobayashi¹

¹ Faculty of Science and Engineering, Waseda University,
Room 40-701, Waseda-machi 27, Shinjuku-ku, Tokyo, 162-0042, Japan

² Department of Information Science, Meisei University,
Room 27-1809, Hodokubo 2-1-1, Hino, Tokyo, 191-8506, Japan

kazuya.ueki@meisei-u.ac.jp

Abstract. The Waseda Meisei team participated in the TRECVID 2017 Ad-hoc Video Search (AVS) task [1]. For this year’s AVS task, we submitted both manually assisted and fully automatic runs. Our approach used the following processing steps: building a large semantic concept bank using pre-trained convolutional neural networks (CNNs) and support vector machines (SVMs), calculating each concept score for all test videos (IACC_3), manually or automatically extracting several search keywords based on the given query phrases, and combining the semantic concept scores to obtain the final search result. Our best manually assisted run achieved a mean average precision (mAP) of 21.6%, which ranked the highest among all the submitted runs. Our best fully automatic run achieved a mAP of 15.9%, which ranked second among all participants.

1 System Description

1.1 Concept bank

To provide good coverage for the given query phrases, we built a large concept bank consisting of more than 50,000 classifiers as shown in Fig. 1. Concept scores for the test videos were calculated using two methods: directly using features from the output layers of the CNN, or calculating the scores from the SVMs.

1. TRECVID346

In this model, concept scores were calculated using a CNN/SVM tandem connectionist architecture. Firstly, at most ten frames from each shot were selected at regular intervals, and the corresponding images were input to the pre-trained GoogLeNet model [9] trained on the ImageNet database to obtain the respective 1,024-dimensional feature vectors from pool5 layers. These feature vectors (a total of ten at most) were then bound to one feature vector using element-wise max-pooling. Next, we trained SVMs for each concept using a collaborative annotation [2]. The shot score for each concept was calculated as the distance to the hyperplane in the SVM model. The score for each semantic concept was normalized over all test shots using a min-max normalization, that is, the maximum and the minimum scores were 1.0 (most probable) and 0.0 (least probable), respectively.

2. FCVID239 and UCF101

This year, we added concepts from two further databases, the Fudan-Columbia Video Dataset (FCVID) [4] and the UCF101 action recognition dataset [8], to

Table 1. Concept bank used in our systems.

Name	Database	# of concepts	Concept type(s)
TRECVID346	TRECVID (ImageNet)	346	Object, Scene, Action
FCVID239	FCVID [4] (ImageNet)	239	Object, Scene, Action
UCF101	UCF101 [8] (ImageNet)	101	Action
PLACES205	Places [10]	205	Scene
PLACES365	Places	365	Scene
HYBRID1183	Places, ImageNet	1,183	Object, Scene
IMAGENET1000	ImageNet	1,000	Object
IMAGENET4000	ImageNet	4,000	Object
IMAGENET4437	ImageNet	4,437	Object
IMAGENET8201	ImageNet	8,201	Object
IMAGENET12988	ImageNet	12,988	Object
IMAGENET21841	ImageNet	21,841	Object

improve the performance of action recognition. Concept scores for FCVID239 and UCF101 were calculated using the same procedure as that for TRECVID346, namely, a CNN/SVM tandem connectionist architecture. The score for each semantic concept was also normalized using the min-max normalization.

3. PLACES205, PLACES365, and HYBRID1183

As concepts that we needed to detect comprised not only objects but also scenes in most query phrases, we selected three types of so-called Places-CNNs [10]: the Places205-AlexNet and the Places365-GoogLeNet models, which were trained on 205 and 365 scene categories with 2.5 and 1.8 million images, respectively; and the Hybrid-AlexNet model, which was trained on 1,183 categories (205 scene categories and 978 object categories) with 3.6 million images. The shot scores were obtained directly from the output layer (before softmax was applied) of the CNNs. The score for each semantic concept was also normalized over all test shots using the min-max normalization.

4. IMAGENET1000, IMAGENET4000, IMAGENET4437, IMAGENET8201, IMAGENET12988, and IMAGENET21841

To increase the number of object categories, we also used pre-trained ImageNet models. The IMAGENET1000 model was provided by the Caffe development team [3]. This network (AlexNet) was trained using the ImageNet dataset containing 1.2 million images and 1,000 categories and was used in the ILSVRC2012 benchmark. The IMAGENET4000, IMAGENET4437, IMAGENET8201, and IMAGENET12988 models were provided by the University of Amsterdam [6]. The full ImageNet model IMAGENET21841 was downloaded from the model zoo. Again, the shot scores were calculated directly from the output layer of the CNNs and normalized using the min-max normalization.

1.2 Manual search keyword selection

Given a query phrase, we manually selected some visually important keywords. For example, given the query phrase “one or more people driving snowmobiles in the snow”, we picked out the keywords “people”, “snowmobile”, and “snow”. If there was no concept name matching a given search keyword, the most semantically similar concept was chosen using the word2vec algorithm [7]. We only used a concept having cosine

similarity ≥ 0.7 , and thus, when a given search keyword did not have a semantically similar concept, the keyword was not used.

1.3 Automatic search keyword selection

For the automatic search, we first used the word2vec algorithm to pick out search keywords from the query phrase. To calculate the score for a search keyword, we tested two keyword matching methods to select classifiers. The first keyword matching method used words from both the classifier’s name and its synsets in WordNet: this is referred to as the *with-synset method*. The other keyword matching method was based only on words from the classifier’s name: this was named the *without-synset method*.

For the with-synset method, we first listed all the words from the classifier’s name and its synsets. We then created a correspondence table between listed words and the concept names in advance. To obtain a score for a specific search keyword, we looked at the correspondence table and used classifiers with concept names corresponding to that keyword. In contrast, for the without-synset method, we only used a classifier that had exactly the same name as a search keyword.

1.4 Score calculation

Given keywords manually or automatically selected from a query phrase, we first calculated the score for each keyword. If there was more than one classifier for a given search keyword, we took the average of the scores of multiple classifiers. For the fusion of multiple keywords, we simply summed or multiplied scores depending on the submitted run.

2 Submissions

This year, we submitted four manually assisted runs and four fully automatic runs to the TRECVID 2017 Ad-hoc Video Search (AVS) task.

2.1 Manually assisted runs

We submitted the following four manually assisted runs (Manual1, Manual2, Manual3, and Manual4). The differences between these four runs were the fusion methods used and whether or not fusion weights were applied.

– Manual1

For a query phrase and a test video, the total score was calculated by multiplying the weighted scores of the selected concepts. We supposed that a rare keyword was of higher importance than an ordinary keyword. For example, in the query phrase “one or more people driving snowmobiles in the snow”, it is rarer for a “snowmobile” to be seen in a video than for a video to feature “people”. In this case, we would like to assign a higher weight to “snowmobile”. Therefore, the total score S_{M1} was calculated by

$$S_{M1} = \prod_{i=1}^N s_i^{w_i}, \quad (1)$$

where N is the number of selected concepts, s_i is the normalized concept score, and w_i is the fusion weight. We used the IDF values calculated from the Microsoft COCO database [5] as the fusion weights.

This criterion focuses more on shots that include all the selected concepts. Therefore, shots having all the selected concepts will tend to appear in the higher ranks. However, if one of the concepts is not correctly detected or the performance of the concept detection model is low, this criterion can have a harmful effect on the final performance.

– **Manual2**

This run is almost the same as the Manual1 run except for not using a fusion weight. For this run, we simply calculated the total score S_{M2} by multiplying the scores of the selected concepts:

$$S_{M2} = \prod_{i=1}^N s_i. \quad (2)$$

– **Manual3**

The total score was calculated by summing the weighted scores of the selected concepts with fusion weights:

$$S_{M3} = \sum_{i=1}^N w_i \cdot s_i. \quad (3)$$

The final score S_{M3} was calculated under somewhat looser conditions than in the Manual1 and Manual2 runs. Under this criterion, all the selected concepts are not necessarily included in a shot, but we expect shots including as many concepts as possible to be found in the higher ranks.

– **Manual4**

The total score S_{M4} was calculated simply by summing the scores of the selected concepts without fusion weights:

$$S_{M4} = \sum_{i=1}^N s_i. \quad (4)$$

2.2 Fully automatic runs

We submitted four types of fully automatic runs. All the fully automatic runs were based on the multiplication of the weighted scores of the selected concepts; this is the same method as used in the Manual1 run. The total scores S_A were calculated by:

$$S_A = \prod_{i=1}^N s_i^{w_i}. \quad (5)$$

Each run differed only in how classifiers were selected.

– **Automatic1**

For the Automatic1 run, we used the with-synset method as described in Section 1.3. Given a query phrase, we split a sentence into individual words or compound words. A search keyword was then selected based on the criterion of cosine similarity = 1.0, that is, a search keyword has exactly the same name as a

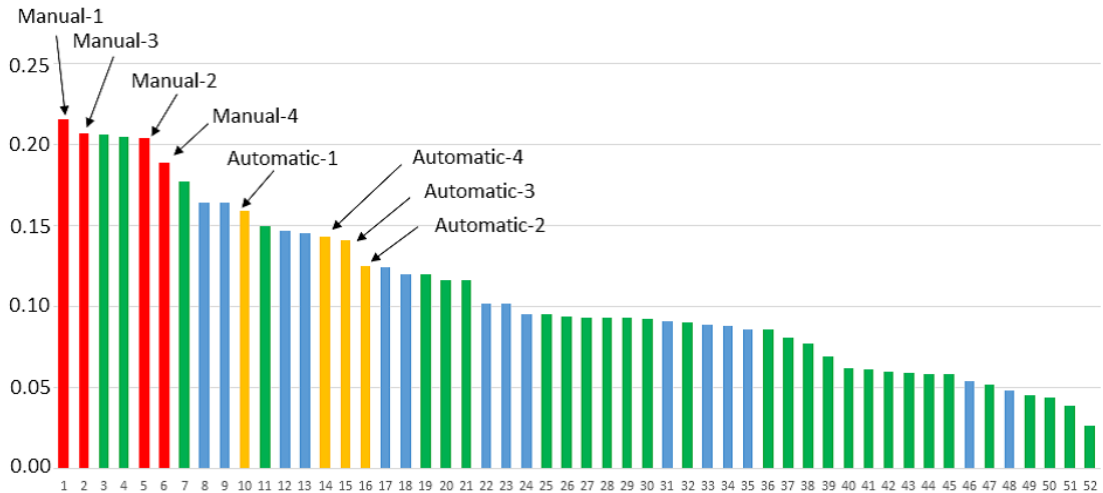


Fig. 1. Comparison of Waseda.Meisei runs with the runs of other teams for all the submitted runs.

word in a query. Finally, by looking at the corresponding table, we chose corresponding classifiers to the search keyword. In this run, we selected classifiers from TRECVID346, PLACES205, PLACES365, HYBRID1183, IMAGENET1000, IMAGENET4000, IMAGENET4437, IMAGENET8201, IMAGENET12988, and IMAGENET21841.

– Automatic2

The Automatic2 run was also performed using the with-synset method. This run differed from the Automatic1 run in how search keywords were selected. Semantically similar search keywords were selected based on the criterion of cosine similarity ≥ 0.7 , and the number of selected search keywords was therefore essentially larger than the ones in the Automatic1 run. If all the search keywords had cosine similarity < 0.7 , those keywords were not used. All other settings were the same as the Automatic1 run.

– Automatic3

The Automatic3 run was based on the without-synset method; the concept classifier’s name was simply used. In this run, we selected semantically similar concepts to a split word in a query phrase based on the criterion of cosine similarity ≥ 0.7 . In this run, we selected concepts from all the models shown in Table 1.

– Automatic4

The Automatic4 run also used the without-synset method, but we excluded classifiers of three models, FCVID239, UCF101, and IMAGENET21841, namely, we used classifiers of TRECVID346, PLACES205, PLACES365, HYBRID1183, IMAGENET1000, IMAGENET4000, IMAGENET4437, IMAGENET8201, IMAGENET12988, and IMAGENET4000; these concepts used were the same as those used in last year’s submission.

2.3 Results

Fig. 1 shows the results of all runs. The mAPs of our submitted manually assisted runs (Manual1, Manual2, Manual3, and Manual4) were 21.6%, 20.4%, 20.7%, and 18.9%,

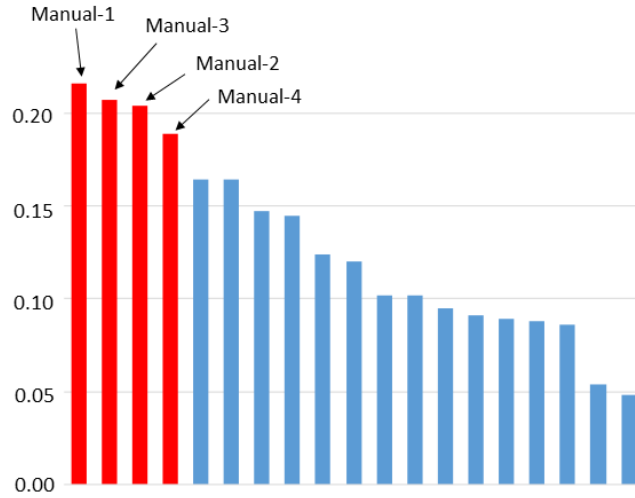


Fig. 2. Comparison of Waseda.Meisei runs with the runs of other teams for all submitted manually assisted runs.

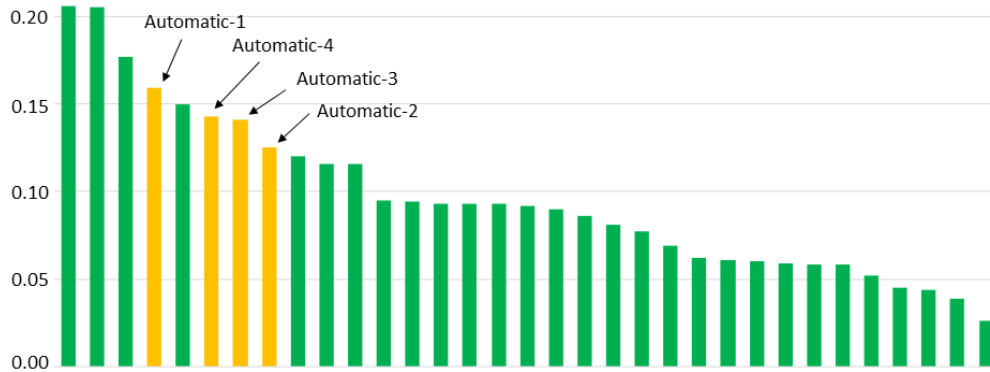


Fig. 3. Comparison of Waseda.Meisei runs with the runs of other teams for all the fully automatic runs.

which ranked 1st, 5th, 2nd, and 6th among the 52 runs, respectively. The mAPs of our fully automatic runs (Automatic1, Automatic2, Automatic3, and Automatic4) were 15.9%, 12.5%, 14.1%, and 14.3%, which ranked 10th, 16th, 15th, and 14th, respectively.

Figs. 2 and 3 show the results of the manual and automatic runs, respectively. Our manually assisted runs ranked 1st through the 4th overall. Our fully automatic runs ranked 4th, 6th, 7th, and 8th, ranking us 2nd overall among all participants.

Fig. 4 shows the average precision of the Manual1 run for each semantic concept. For some concepts, our runs achieved the best average precision. This high performance was achieved by using a relatively large number of semantic concept classifiers, which exceeded 50,000. It can be seen that the gap between the high and low performance significantly widened; average precisions for several query phrases were almost zero.

For the fusion method, the result of the comparison of Manual1 (or Manual2) and Manual3 (or Manual4) shows that the fusion weights worked effectively, that is, rarely seen concepts are much more important for the video retrieval task.

For the fully automatic runs, the average precisions were mostly worse than those of the manually assisted runs. Two reasons were considered to be the cause of this

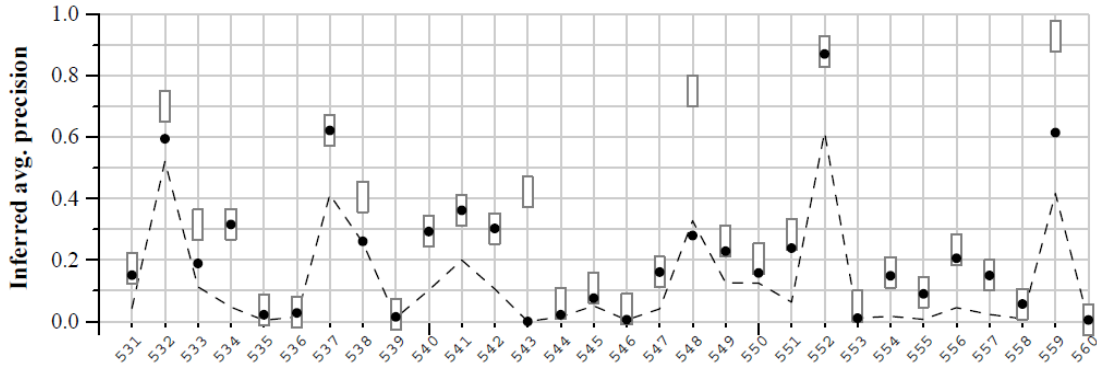


Fig. 4. Average precision of our best manually assisted run (Manual1) for each query. Run score (dot), median (dashes), and best (box) by query.

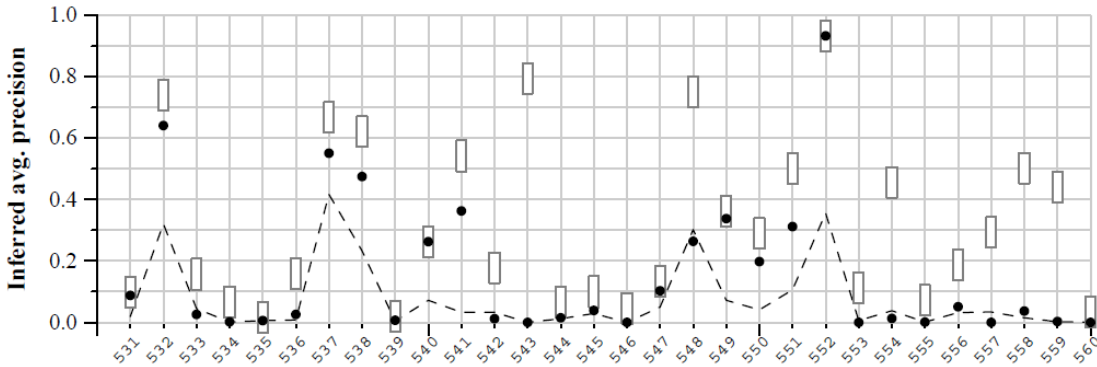


Fig. 5. Average precision of our best fully automatic run (Automatic1) for each query. Run score (dot), median (dashes), and best (box) by query.

poor performance: words that were not useful in searching for videos were used, and concepts were excessively selected by the word2vec method. To further improve search performance, we need to eliminate visually useless concepts and add more effective concepts.

3 Conclusion

For this year’s submissions, we solved the problem of ad-hoc video search using a combination of many semantic concepts in the same manner as in last year’s submission. We achieved the best performance among all the submissions for the manually assisted runs. However, the performance was still extremely poor for some query phrases.

Our future work will be focused on improving the performance of the fully automatic search system by effectively selecting visually valuable concepts. We would also like to consider a method of directly searching for videos without decomposing the query phrases into words.

Acknowledgments

This work was partially supported by JSPS KAKENHI Grant Number 15K00249 and 17H01831.

References

1. G. Awad, A. Butt, J. Fiscus, D. Joy, A. Delgado, M. Michel, A. F. Smeaton, Y. Graham, W. Kraaij, G. Quénot, M. Eskevich, R. Ordelman, G. J. F. Jones, and B. Huet. Trecvid 2017: Evaluating ad-hoc and instance video search, events detection, video captioning and hyperlinking. In *Proceedings of TRECVID 2017*. NIST, USA, 2017.
2. S. Ayache and G. Quénot. Video corpus annotation using active learning. In *30th European Conference on Information Retrieval (ECIR'08)*, pages 187–198, 2008.
3. Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.
4. Y. Jiang, Z. Wu, J. Wang, X. Xue, and S. Chang. Exploiting feature and class relationships in video categorization with regularized deep neural networks. *CoRR*, abs/1502.07209, 2015.
5. T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision (ECCV)*, Zürich, September 2014.
6. P. Mettes, D. C. Koelma, and C. G. Snoek. The ImageNet Shuffle: Reorganized Pre-training for Video Event Detection. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval, ICMR '16*, pages 175–182, New York, NY, USA, 2016. ACM.
7. T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.
8. K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.
9. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.
10. B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*, 2014.