PKU-ICST at TRECVID 2016: Instance Search Task

Yuxin Peng, Xin Huang, Jinwei Qi,

Junjie Zhao, Junchao Zhang, Yunzhen Zhao,

Yuxin Yuan, Xiangteng He, and Jian Zhang

Institute of Computer Science and Technology,

Peking University, Beijing 100871, China.

pengyuxin@pku.edu.cn

Abstract

We participated in all two types of Instance Search (INS) task in TRECVID 2016: automatic search and interactive search. This paper presents our approaches and results. In this task, we first conducted two search processes: location-specific search and person-specific search, and then the score fusion of the two processes was performed to get the final results. In the location-specific search process, we adopted two kinds of features: (1) Bag-of-Words (BoW) feature based on Approximate K-means (AKM); and (2) DNN feature based on Convolutional Neural Networks (CNN). In the person-specific search process, we adopted three kinds of methods, including (1) face recognition; (2) person re-identification; and (3) text-based search. After getting the results of location-specific and person-specific search, we conducted instance score fusion with two strategies: searching a given person based on location-specific results, and searching a given location based on person-specific results. In the re-ranking stage, we further applied semi-supervised learning based re-ranking method to improve the search results. The official evaluations showed that our team ranked 1st in both automatic search and interactive search.

1 Overview

In TRECVID 2016[1], we participated in all two types of Instance Search (INS) tasks: automatic search and interactive search. We totally submitted 7 runs including 6 automatic runs and 1 interactive run, and the result of interactive search is based on that of automatic search. The official evaluation results of our 7 runs are shown in Table 1.

In both automatic search and interactive search, our team ranked 1st among all teams. Table 2 gives the detailed explanation of brief descriptions in Table 1. Our system's framework is shown in Figure 1. In the 6 automatic runs, the notations "A" and "E" specify whether the video examples were used or not, and the methods of two runs are the same if the only difference between them is the notation “A” or “E”. The difference between Run1_A/E and Run2_A/E is that Run2_A/E incorporates person re-identification method based on the methods of Run1_A/E. Compared to Run2_A/E, Run3_A/E uses re-ranking strategy to improve the search results.

Table 1: Results of our submitted 7 runs on Instance Search task of TRECVID 2016.

Type	ID	MAP	Brief description
Automatic	PKU_ICST_RUN1_A	0.317	K+D+F+T
	PKU_ICST_RUN1_E	0.349	K+D+F+T
	PKU_ICST_RUN2_A	0.328	K+D+F+I+T
	PKU_ICST_RUN2_E	0.364	K+D+F+I+T
	PKU_ICST_RUN3_A	0.335	K+D+F+I+T+R
	PKU_ICST_RUN3_E	**0.370**	K+D+F+I+T+R
Interactive	PKU_ICST_RUN4	**0.484**	K+D+F+I+T+R+H

Table 2: Description of our methods.

Abbreviation	Description
K	**K**eypoint-based feature based on AKM
D	**D**eep Neural Networks based feature
F	**F**ace Recognition
I	Person Re-**I**dentification
T	**T**ext-based Search
R	**R**e-ranking
H	**H**uman feedback

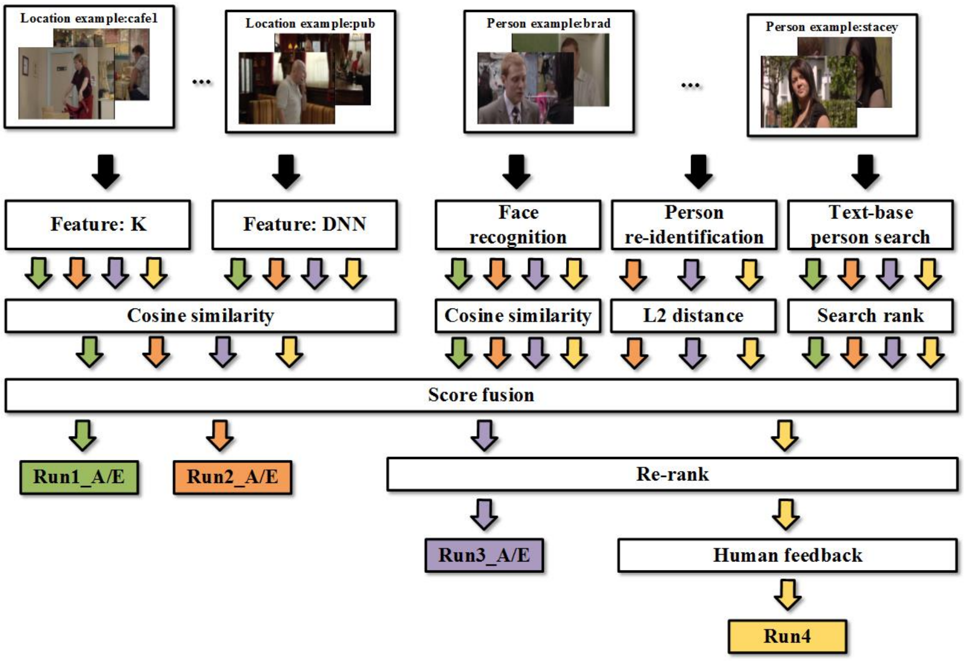

Figure 1: Framework of our instance search approach for the submitted 7 runs.

2 Location-specific search

Two kinds of features were exploited for location-specific search process, namely AKM-based BoW feature and DNN feature respectively.

2.1 AKM-based BoW feature

We represented the video shots by using several kinds of keypoint-based BoW features. The keypoint-based BoW features were generated by the following three steps:

(1) First, we used three detectors which were Harris Laplace[2], Hessian Affine[3] and MSER[4] to detect the keypoints in the keyframes. Then we used two descriptors, namely 128-dimensional SIFT descriptor[5] and 192-dimensional ColorSIFT descriptor[6] for each detector to represent the neighboring regions around those keypoints.

(2) Second, the keypoints were clustered into one-million cluster centroids by the AKM algorithm, and a visual vocabulary was built with the cluster centroids.

(3) At last, we assigned each keypoint of all keyframes in one shot to the nearest centroid, where the word weights were determined by the keypoint-to-word similarity and region of interest (ROI), thus each shot could be quantized into a one-million-dimensional BoW feature, as shown in Figure 2. Because there were three detectors and two descriptors we used, 6 kinds of BoW features in total were generated for each video shot and query topic.

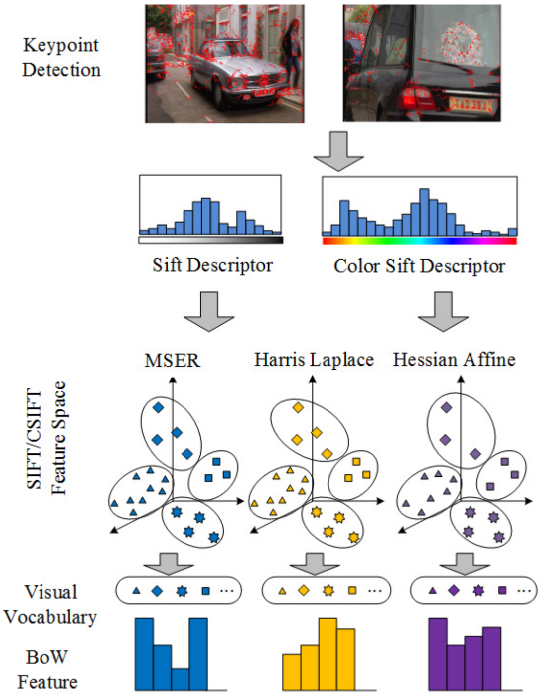

Figure 2: Combination of BoW features based on different detectors and descriptors.

2.2 DNN feature

We also used DNN feature to promote the performance on location-specific search. First, we fine-tuned a publicly available VGG-16 model to fit with the task, which has been trained on Places205[7] dataset. Then we extracted the DNN feature from the fine-tuned model.

In fine-tuning phase, we took two strategies to construct the training dataset. The first was to make various kinds of transformations on the provided location example images (for runs with notation “A”), and the second took video examples into the training dataset (for runs with notation “E”). In the feature extraction phase, based on the fine-tuned model, we took the activations of the first fully-connected layer as feature representation, and represented each example by a 4096-dimensional vector.

2.3 Location-specific search on distance measure

After getting the AKM-based BoW feature and DNN feature of each query location example and test video shot, we calculated the similarity based on cosine distance measure. We conducted two location-specific search procedures by using AKM-based BoW feature and DNN feature respectively, then we performed late fusion of the two kinds of similarity scores to get the final location similarity, and further obtained the retrieval ranking of location-specific search.

3 Person-specific search

In person-specific search process, we adopted three kinds of methods, including face recognition, person re-identification and text-based search.

3.1 Face recognition

First, we detected faces from the video keyframes by using DPM Face detector[8], and generated a 4096-dimensional feature vector for each face image by using the VGG-Face CNN descriptor[9]. The faces in query person examples would also be detected and represented by VGG-Face CNN descriptor. Second, we used cosine distance to calculate the similarity between the query face image and each shot, then we fine-tuned the VGG-Face CNN model with each query person’s top-50 shots. Finally, we extracted a 4096-dimensional feature vector based on the fine-tuned VGG-Face CNN model for each face image, and calculated the similarity between the query person and each shot.

3.2 Person re-identification

Face information in person-specific search has some limitations: (1) When the face deflection angle is large or even no face is shown, the face detection may fail. (2) The pose and clothing information which is helpful for person-specific search is ignored. In order to address these limitations, we adopted person re-identification to improve the search accuracy. Our framework included two phases.

(1) Person detection

In person detection phase, we applied the Faster R-CNN[10] method to detect persons in the video. We only saved the results of person detection whose prediction scores are greater than a given threshold (here we set the threshold as 0.8).

(2) Person retrieval

In person retrieval phase, we first trained the CNN model for recognition. Considering that only several query person examples were provided, we generated more training data through image transformation, such as color or background swap and image quantity change, which enriched the training data. In addition, we also applied the progressive strategy to the model training.

Once the CNN model was trained, we used it to extract features of query person examples and video keyframes. When using CNN as feature extractor, the activations of the first fully-connected layer were output as features. For each image, we extracted 4096-dimensional feature vectors based on the CNN model. Then we adopted the L2 distance for the person-specific search based on the CNN feature vectors.

3.3 Text-based person search

As NIST provided the transcripts of videos this year, we used this information to perform text-based person search. We found that each topic explicitly pointed out the person's name of the instance for searching, and an instance was likely to appear in the shots when the given person's name was included in the transcript. In addition, we used the structured data from related Wikipedia webpage to extend the person's information for search, such as nick name, character name, etc. For each person, the shots whose transcripts included the name would be given a score 1, while the other would be given a score 0.

3.4 Person similarity fusion

After getting the scores of face recognition, person re-identification and text-based search for each query person example and test video shot, we performed late fusion of the 3 kinds of scores. Like the process of location-specific search, the fused scores would be the final person similarity, and the person query rankings could also be obtained. The person similarity would be further fused with the location similarity scores for instance search.

4 Instance score fusion

So far we got the location similarity from the location-specific search, as well as the person similarity from the person-specific search. This year, each query topic was to find a given person in a given place, which required the fusion of both location and person similarity. Two fusion strategies were adopted as follows:

(1) First, we searched the given person from location-specific results. Specifically, we selected top-*N* (set to be 3500 here) candidate shots from location-specific results ranked by the location similarity score, which had a considerable chance of containing the given location. For each candidate shot, we performed late fusion of both the location and person similarity to get the score s_1. For those shots not included in top-*N* location-specific results, $s_1 = 0$.

(2) Second, the given location was searched from person-specific results. We chose top-*M* (set to be 4000 here) candidate shots, from person-specific results according to the person similarity. Person-specific search was more complex than location-specific search, so we selected more candidate shots. Also, late fusion was performed on both the location and person similarity, and then the score s_2 was obtained. Similarly, $s_2 = 0$ for the shots not in the top-*M* person-specific results.

(3) Finally, we adopted late fusion strategy on s_1 and s_2 to get the final instance score ranking list. It could be easily seen that the final instance score preserved information of both location and person aspects, and the fusion of them could improve the instance search

accuracy.

5 Re-ranking

In re-ranking stage, we observed that most of the top-ranked videos were correct and they looked similar with each other. But there existed a few noisy videos, which often contained the right location but the wrong person. To eliminate such noise, we proposed a semi-supervised re-ranking algorithm[11] to refine the top-ranked results as below:

(1) Given the data matrix of 1000 top-ranked videos F and L, where F_i stood for the face feature vector of a keyframe image and L_i stood for the video ID of vector F_i, $i \in \{1, 2, \ldots, n\}$ where $n > 1000$ meant there were n keyframes from 1000 videos.

(2) Initialized the affinity matrix W with all zeros, and updated as following:

$$W_{i,j} = \frac{F_i \bullet F_j}{|F_i| \bullet |F_j|}, i, j \in \{1, 2, \ldots, n\}, i \neq j$$

(3) Generated the k-NN graph:

$$W_{i,j} = \begin{cases} W_{i,j}, & F_i \in kNN(F_j); \\ 0, & otherwise. \end{cases}$$

(4) Constructed the matrix: $S = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, where D was a diagonal matrix with its *(i, i)*-element equal to the sum of the *i*-th row of W.

(5) Iterated $G_{t+1} = \alpha S G_t + (1 - \alpha) Y$ until convergence, where G_t denoted the refined result in *t*-th round and we set $G_0 = Y$. α was a parameter in the range (0, 1). Y was the initial score list of the keyframes of 1000 top-ranked videos, we set the score of each keyframe the same as its original video.

6 Interactive search

This year, we adopted a one-turn interactive search strategy, which was different to the two-turn interactive strategy we adopted in the instance search task of 2015. The user was first shown the ranking list of automatic search for each topic. Then he would manually label the top-ranked results as positive and negative samples. The negative samples would be discarded in the final interactive run, and the positive samples acted as expansion queries. We measured both the location and person similarities between expansion queries and top-5000 shots of each topic in automatic search results, and then used the same method with automatic search to generate a new ranking list. To guarantee the speed of interactive search, if there were more than 10 positive samples selected, only top-10 positive samples for each topic were actually used as expansion queries. The result in Table 1 shows that the human feedback significantly improves the accuracy.

7 Conclusion

By participating in the instance search task in TRECVID 2016, we have the following conclusions: (1) Location-specific and person-specific search processes have different characteristics, so different strategies should be adopted for them. (2) The fusion of location and person similarity is a key factor of the instance search. (3) Video examples are helpful for accuracy improvement, so for the same method, run "E" would achieve higher search accuracy than run "A" (see Table 1).

Acknowledgements

This work was supported by National Hi-Tech Research and Development Program of China (863 Program) under Grant 2014AA015102, and National Natural Science Foundation of China under Grants 61371128 and 61532005.

References

[1] G. Awad, J. Fiscus, M. Michel, *et al.*, "TRECVID 2016: Evaluating Video Search, Video Event Detection, Localization, and Hyperlinking". Proceedings of TRECVID 2016, 2016.

[2] C.G.M. Snoek, K.E.A. van de Sande, O. de Rooij, *et al.*, "The MediaMill TRECVID 2008 Semantic Video Search Engine". *TRECVID*, Maryland USA, November 17-18, 2008.

[3] K. Mikolajczyk, and C. Schmid, "Scale and Affine Invariant Interest Point Detectors". *International Journal of Computer Vision (IJCV)*, vol. 60, no. 1, pp. 63-86, 2004.

[4] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions". *British Machine Vision Conference (BMVC)*, pp. 384-393, 2002.

[5] D. G. Lowe, "Distinctive Image Features from Scale-invariant Keypoints". *International Journal of Computer Vision (IJCV)*, vol. 60, no.2, pp. 91-110, 2004.

[6] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors". *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 27, no.10, pp. 1615-1630, 2004.

[7] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. "Learning Deep Features for Scene Recognition using Places Database". *Advances in Neural Information Processing Systems (NIPS)*, pp. 487-495, 2014.

[8] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object Detection with Discriminatively Trained Part Based Models". *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 32, no. 9, pp. 1627-1645, 2010.

[9] O. M. Parkhi, A. Vedaldi, and A. Zisserman. "Deep Face Recognition". *British Machine Vision Conference (BMVC)*, vol. 1. no. 3, pp. 6, 2015.

[10] S. Ren, K. He, R. Girshick, and J. Sun. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". *Advances in Neural Information Processing Systems (NIPS)*, pp. 91-99, 2015.

[11] Y. Peng, X. Zhai, J. Zhang, *et al.*, "PKU-ICST at TRECVID 2012: Instance Search Task". *TRECVID*, Maryland, USA, November 26-28, 2012.