

WHU-NERCMS at TRECVID2016: Instance Search Task

Zheng Wang¹, Yang Yang¹, Shuosen Guan¹, Chenxia Han¹,
Jiamei Lan¹, Rui Shao¹, Jinqiao Wang², Chao Liang^{1*}

¹National Engineering Research Center for Multimedia Software, School of Computer, Wuhan University

²National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences
cliang@whu.edu.cn

Abstract

This report introduces our work at the instance search task of TRECVID 2016. This year INS task asks systems to retrieve specific persons in specific locations. For the new task, the key points include: (1) We exploit effective face recognition method, which detect faces by a scale-adaptive deconvolutional regression network and then recognize detection results by a deep embedding network. (2) We search for specific scene from both of local view (hand-crafted system) and global view (deep learning system). (3) We omit a large amount of unrelated shots, such as outdoor scenes, non-face shots and some previous groundtruth shots. (4) We adjust the score of missed low-score shots among adjacent high-score shots. Based on these improvements, our team acquires promising results.

1 Introduction

The instance search (INS) is a special content based multimedia retrieval task. Given one or more visual examples of a specific item, which can be a person, an object, or a plane, the aim of the task is to find more video segments of the certain specific item [1]. The INS is always a hot topic in multimedia community [2]. In the past, the TRECVID INS task paid attention to searching different specific instances **separately**, such as a certain person (Fig.1 (a)), or a specific object (Fig.1 (b)). With the effort of participants, the BoW (Bag-of-Words) [5] based frameworks obtained good performances in most cases. Tab.1 shows some typical results of traditional methods. It is easy to conclude that the effectiveness of non-rigid objects (such as persons, or animals) is not good enough as that of the rigid ones (such as a washing machine). Since 2014, participants started to utilize CNN (Convolutional Neural Networks) [6] to improve the object's representing ability. The PKU-ICST team [7] in 2015, fused the BoW feature with the outputs of pre-trained CNN. Compared with their previous results [4], they made an obvious improvement. However, the effectiveness of person topics is still worse than that of the rigid objects (as shown in Tab.1). Previous works lead us to making the following conclusions: (1) BoW based models are very useful for the rigid object retrieval. (2) CNN model increasingly plays an important part in INS task. (3)

*Corresponding author

Seeking out certain person from videos is still very difficult, and far from being solved by previous BoW based models or CNN model.

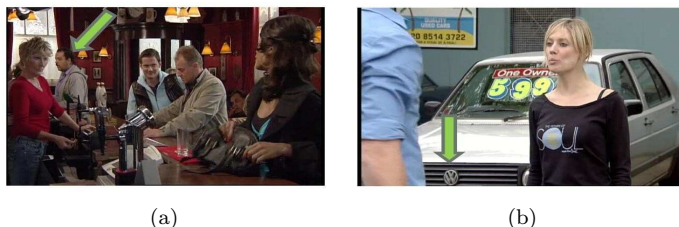


Figure 1: As the green arrow indicates: (a) the topic is to find *this man with moustache* (topic 9138 in 2015), (b) the topic is to find *a VW logo* (topic 9087 in 2013).

		Target(topic)	Average AP [1,4]	Max AP [1,4]
BoW	rigid objects	a no smoking logo (9069)	0.29	0.88
		this David magnet (9085)	0.24	0.81
	non-rigid objects	this man (9084)	0.03	0.29
		Aunt Sal (9096)	0.01	0.04
BoW+CNN	rigid objects	this starburst wall clock (9153)	0.42	0.91
		this picture of flowers (9157)	0.44	0.88
	non-rigid objects	this bald man (9143)	0.04	0.19
		this shaggy dog (9139)	0.01	0.01

Table 1: Some results based on BoW and CNN.

In this year, the TRECVID organizer further enhances the difficulty of the INS task, which asks participants to search video shots **simultaneously** identifying a specific target person in a specific target location. Given a collection of test videos, a master shot reference, a set of known location/scene example videos, and a collection of topics (queries), the task is to delimit a person in some example videos, locate for each topic up to the 1000 shots most likely to contain a recognizable instance of the person in one of the known locations [?]. In this condition, four challenges should be mainly considered:

- Previous BoW or CNN based methods demonstrated their unsatisfied results on person topics. All the topics of this year ask for seeking out certain person, hence exploiting effective face recognition method becomes particularly important.
- Following previous object-oriented frameworks, we can search for a specific scene by its landmark objects. However, those landmark objects are often occluded as viewpoints change, which would lead to scene missing retrieval.
- After obtaining the person retrieval results and the scene retrieval results, fusing the results together will be a crucial task for the final performance.
- In general, there are totally 471,526 shots to be ranked. So many noisy data will further deteriorate the final results.

Compared with traditional methods, the contributions of the proposed method are as follows:

- For specific person recognition, we recognize a certain person by his/her face in two steps: First, we utilize a Scale-Adaptive Deconvolutional Regression (SADR) Network [8] to detect faces in video shots. Second, the detection results are recognized by a Deep Embedding Network.
- For specific scene retrieval, on one hand, based on a hand-crafted system (BoW), through identifying landmark objects in certain topic scenes by previous methods, we seek out several target scenes. On the other hand, based on a deep learning system, we further take the output of a pre-trained CNN to be the global scene feature to find more given scenes.
- For extensive noisy data, we adopt several different tricks to delete unrelated shots: (1) All the given topics are indoor scenes, so outdoor scenes can be filtered. For example, based on the category results of ResNets [9], we filter shots with the vehicle categories of the ImageNet 1000 categories [10], which may only appear outside. (2) All the given topics include persons, hence the shots without any target person should be filtered. (3) Several previous topics are independent to the topics of this year, so corresponding groundtruth shots of previous years can be omitted.
- Scene retrieval may be lost due to viewpoint change, discovering that scene retrieval scores presenting a “L” shape, we find high-score shots with high slope of the score curve, and adjust those missed low-score shots among adjacent high-score shots. Then, face result is combined with global scene result and local scene result respectively. At last, these two ranking list are across combined.
- What’s more, we utilize useful methods to optimize results: (1) In addition to the use of video content, we take advantage of caption information, which include important keywords, such as names and places, and speaker’s voice, where speaker identification technology is applied to our work. (2) Those certain previous topics can be used to enhance accuracy of this year. For example, *a necklace on the fatboy* is an old topic, the groundtruth shots of this necklace can help us find the fatboy.

Based on above approaches, our team acquires promoting results on INS of TRECVID 2016.

2 Our Framework

The proposed framework of INS task is shown in Fig.2. It consists of four parts: shots filtering, person retrieval, scene retrieval and result optimization. The shots filtering part removes the unrelated shots. The person and scene retrieval parts search for target person and scene respectively. The result optimization part refines the combined results and obtains the final ranking list. The details of each part and related key technologies are demonstrated below.

Face recognition mainly consists of two steps. (1) Based on Faster R-CNN [12], we propose Scale-Adaptive Deconvolutional Regression (SADR) face detection network [8]. We use the pre-trained VGG16 model [13] to initialize the proposed network. By computing the classification and regression loss, we integrate multi-layer outputs of CNN network to boost the detection performance. (2) A Deep Embedding Network is utilized to conduct face identification after face detection and alignment with 78 landmarks. This network includes 9 convolutional layers, 5 pooling layers and

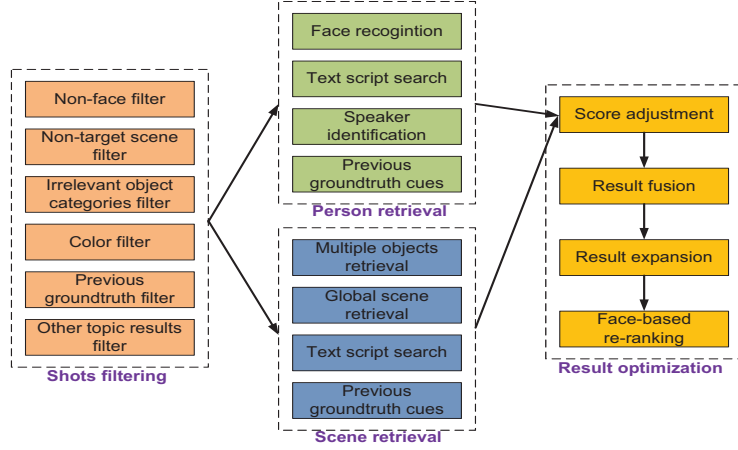
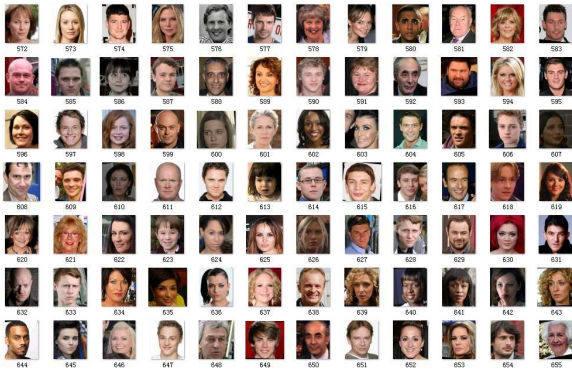


Figure 2: Our framework of INS task.

2 fully connected layer. Following [14], Softmax and triplet cost are combined to construct the objective function. The network is trained in our collected IVA-WebFace with 80 thousand identities and each has about 500-800 face images.



(a) Part of the non-target faces

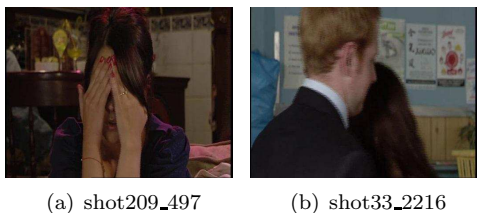
Name	Faces with different angles in the face library
Brad	
Dot	
Fatboy	
Jim	
Pat	
Patrick	
Stacy	

(b) Target faces

Figure 3: Face library for INS task.

For face recognition, we built our own face library, which includes not only target persons we are searching for, but also the other persons in the TV series *Easterners*. We search the keyword *Easterners* in Bing. And then, we choose about 750 different face images from the results, which construct the non-target faces (Fig.3(a)). Meanwhile, an efficient regression approach for face alignment [15] is conducted to regularize these non-target faces. In addition, we select multiple face images of different appearances to represent each target person, as shown in Fig.3(b). In this way, once a face image in shot is identified as one of target faces in the library, it will be considered to be the certain target. Our own face library includes 815 face images.

Non-target face filter. With the help of face recognition, we can not only seek out the target person, but also filter unrelated shots without target faces. We filter shots where the recognition score values zero directly. It is counted that **217,894** shots are deleted by this way. Nevertheless, as shown in Fig.4, due to non-front and occlusion, some groundtruth shots are filtered by mistake. In total, there are **851** groundtruth shots deleted. However, with expanding shots forward and backward, **822** of them are recovered. By means of non-face filter, up to 46% of original video shots are filtered.



(a) shot209_497

(b) shot33_2216



(a) living room3

(b) kitchen3

Figure 4: Examples of shots deleted by mistake.

Figure 5: Non-target scenes

Global scene retrieval. When an image is input into the CNN, the output of the fully connected layer can be taken as the global feature of the image. It is reported that ResNets [9], with a depth of 152 layers, achieved state-of-the-art results on ImageNet [9]. We adopt the Facebook’s 152-layer model [11] in scene retrieval, and the output of the model is denoted as the global scene feature. For each specific probe scene, we utilize multiple related images in different angles. For example, we select 12 images for *pub* to retrieve. The scene retrieval score of each shot is the maximum value of its similarities to all the probe images of the scene.

Non-target scene filter. Meanwhile, we exploit global feature to filter shots by their high similarities to those non-topic scenes. In fact, the shots of *cafe1*, *cafe2*, *kitchen2*, *living room2*, *market* can be filtered. In addition, we select two other scenes to be omitted, which are named *living room3* and *kitchen3* (shown in Fig.5). We respectively select top 1000 of ranking results of each scene based on global feature. By taking the union of these results, we filter **5592** shots and 4 shots are filtered incorrectly. Actually, when we retrieve one specific target scene, shots with other target locations can be filtered as well. For example, if our target location is *foyer*, shots with *pub* can be filtered.

Irrelevant object categories filter. As we know, all the given topics are indoor scenes, so we can filter outdoor scenes. We filter shots with the vehicle categories of the ImageNet 1000 categories [10]. From the results of ImageNet classification based on ResNets [11], there are 37 categories about vehicles, such as *ambulance*, *minibus* and *police van*. When the score of classification result of any category is more than 0.3, the image is judged to include vehicles. If all key frames in the shot have retrieved vehicles, the shot is considered as an out-door scene and filtered out. Analogously, shots with other 52 categories (such as *hippopotamus*, *Indian elephant* and *castle*) only appear outdoor, and should be filtered as well. By this step, we totally delete **19,244** shots, and 45 of which are groundtruth shots. Then by expending shots forward and backward, 42 groundtruth shots are recovered.¹

Multiple objects retrieval. Through identifying typical objects in a certain topic scene, we

¹We reckon that there are two filtered groundtruth shots (shot159_1643, shot226_1352) probably incorrectly marked.





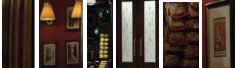
Topic scene	Number	Object samples
foyer	9	
kitchen1	23	
laundrette	19	
living room1	19	
pub	20	

Figure 6: Selected objects for topic scene.

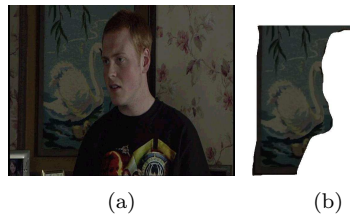


Figure 7: The picture on the wall is split out.

can seek out shots of this scene indirectly. As shown in Fig.6, for each target scene, we select several different landmark objects. BoW model is utilized to conduct object search. We employ hessian-affine SIFT features [16] to express each key frame, and Root-SIFT features are then calculated. Then, we adopted Approximate K-Means (AKM) [17] algorithm to train the 1 million dimensional codebook. With the trained codebook, we can quantize each 128 dimensional SIFT features into one of the codes ranging from 1 to 1000000. Hard assignment method [18] is used to quantize SIFT features of key frames and soft assignment method is exploited to quantize SIFT features of query images. After getting the BoW features of both query images and key frames, we adopt Query Adaptive Similarity Measure [19] to measure the similarity between them. The detailed descriptions are in [20] and compared with the BoW model that we used last year, we update several configurations (shown in Tab.2). To improve BoW model, we introduce some tricks as well. For each of target object, we cut it out from images manually in preprocessing work. For example, in a frame of *living room1* (Fig.7), the picture on the wall is split out irregularly. Traditional Bow model may lose spatial information, for which we apply spatial verification in our module, based on the Delaunay Triangulation (DT) [21] technique.

	2016	2015
Machine memory	256G	48G
SIFT feature extraction	1 in every 10 frames based on original videos	1 in every 15 frames based on images
Number of SIFT points for codebook training	100 million clustered without unrelated shots	50 million clustered with all shots

Table 2: Different configurations for BoW model.

Previous groundtruth filter. Several previous topics are independent to the topics of this year, so corresponding groundtruth shots of previous years can be omitted. (1) Some landmark objects only appear in a specific location. If such an object appears in a non-target scene, the corresponding groundtruth shots can be filtered. For instance, *This picture of flowers* (topic 9157) is in *living room2* and *This cash register* (topic 9148) only appears in *cafe2*. But *living room2* and *cafe2* are not target locations, so we can filter groundtruth shots of these two topics. (2) Some objects must not be contained in the topics of this year, such as *a BMW logo* (topic 9082) and *this*

wooden bench with rounded arms (topic 9090), thus groundtruth shots of these objects can also be filtered. In total, there are 40 such topics from TRECVID 2013 to 2015. Obviously, the groundtruth shots of these topics can be filtered. In this way, we filter **12,006** shots, 17 of which are groundtruth shots. However, we reckon that some groundtruth shots are probably incorrectly marked, such as shot164_1651, shot202_1407, shot97_845 and shot81_1429. What’s more, video 0 is not included in the retrieval scope. And as for video 138 and video 226, their shots detection documents have some mistakes. Thus, these accordingly mistaken shots can be filter. By this means, we filter **2993** shots. However, we reckon that shot138_1993 and shot138_2140 are probably incorrectly marked.

Color Filter. By extracting basic color feature, we get the area of image within the scope of color. If the Area > threshold, the image will be filtered. For instance, the walls of *kitchen2* are yellow, but all target locations are not yellow. So when we retrieve scenes with large yellow area, the corresponding shots can be filtered. Concretely, we retrieve scenes with regard to the color of yellow. Then we filter the top **3000** shots, where there are 2 groundtruth shots filtered.

Previous groundtruth cues. Previous topics can be used to enhance accuracy of this year. For example, *a Primus washing machine* (topic: 9101) appears only in *laundrette*, *this jukebox wall unit* (topic: 9145) appears only in *pub* and *this ceramic cat face* (topic: 9073) appears only in *kitchen1*. In this case, the groundtruth shots of *a Primus washing machine* help us find *laundrette*, those of *this jukebox wall unit* and *this ceramic cat face* help us find *pub* and *kitchen1* separately. Similarly, with the help of groundtruth shots of topics like these, we can find scene shots more accurately.

Text script retrieval and Speaker identification. In addition to the use of image information, the caption and voice in the video shots are applied in our work. The process of text script retrieval is the same as that we performed last year. We search keywords of every topic, and then obtain several results after retrieving keywords. For example, for the target person *Jim*, the retrieval keywords are *Brads*, *Stace*, *Stacey*, *Bradley*, *Dot*, because they are family. In our implementation, taking keyword *Jim* as an example, we find several shots such as shot5_1269, shot8_734 (shown in Fig.8). The process of speaker identification is decomposed into training and testing parts. Training part aims to built a voice library and testing part to conduct speech recognition. Similar to face library, we build our own voice library. For those seven target persons, we intercept 6 voice segments of each person. For the rest 93 people, we intercept 4 voice segments of each person. Finally, there are 412 audio in the library. Then, we extract MFCC feature of all voice segments and train audio modules for each of them. On the testing stage, shots with target persons can be found by calculating the voice similarity between audio information in each shot and trained audio module. For example, utilizing audio information of target person *Patrick*, we find shots such as shot63_1614, shot76_147 (shown in Fig.9) via speaker identification by voice.

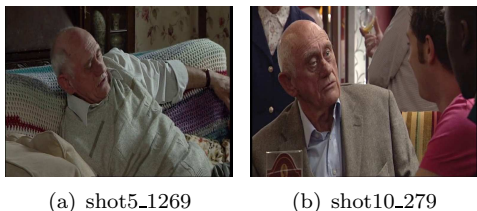


Figure 8: Found shots for keyword *Jim*.

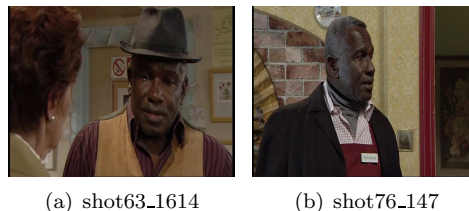


Figure 9: Found shots for voice of *Patrick*.

Score adjustment. The scene in TV series is likely to be blocked by the person, which causes

the similarity scores of such shots are not high. In this case, we adjust the scene retrieval scores of these shots, where the similarity score of each neighbour shots is high but that of itself is low. For example, we retrieve *laundrette* based on CNN, we find that scores of shot192_1199 and shot192_1205 (shown in Fig.10) are high. However, as the viewpoint focus is on specific person, large area of the scene is blocked, which makes the score of shot192_1201 low. Considering the continuity of video shots playing, shot192_1201 is likely to be the target scene, the score of adjacent shot192_1199 is assigned to them. To find high-score shots, we rank the results, and find that the scores of retrieval result are “L” shaped [22] (shown in Fig.10(d)). That is to say, a small amount target shots lie in high slope area, while a large amount of un-target shots lie other areas. So, we find high-score shots with high slope of the score curve, and adjust those missed low-score shots among adjacent high-score shots.



Figure 10: (a) and (c) have high scene retrieval scores, while (b) has a low score. However, if we want to search the scene *laundrette*, shot192_1201 should be recovered. (d) “L” shaped curve. The top 30000 scores are shown in the figure.

Result fusion. The implementation of result fusion is shown in Fig.11. In the process of retrieval, we obtain three score vectors which have values from 0 to 1, namely face results based on face recognition, scene results based on BoW feature and scene results based on CNN feature. Actually, these vectors are assigned weights based on above filter or cues. For example, the deleted shots score is set to 0, while the sure shots are set to 1. With both BoW based scene results and CNN based scene results, we get a preliminary topic results by multiplying scene results and face results shot by shot. And then, we combine two ranking lists together alternately.

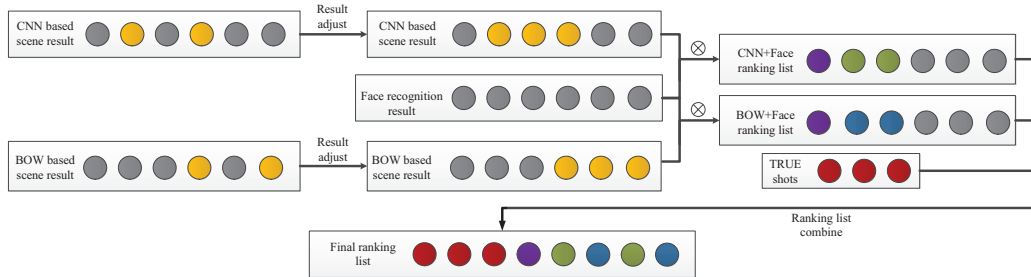


Figure 11: The framework of result fusion. The orange points in CNN based and BoW based scene retrieval indicate the high-score shots. The purple points in the ranking lists indicate the high-score shots in common. The green and blue points indicate the high-score shots respectively in two ranking lists.

Result expansion and Face-based re-ranking. We find that the scores of fused retrieval result are “L” shaped as well. We hold that shots in the high slope are chosen to expand. With expansion, we can find some missed shots. At last, we use face recognition results to re-rank the selected top shots.

3 Results and Analysis

Results of our submitted 4 runs on Instance Search task of TRECVID 2016 are shown in Tab.4.

Compared to our work at TRECVID 2015 [20], we have following useful improvements: (1) Exploiting effective face recognition method becomes particularly important this year. We build our own face library to improve its effectiveness. (2) For extensive noisy data, we adopt several different tricks to delete unrelated shots. (3) “L” shape of results help us to find high-score shots, then score adjustment and result expansion are exploited to fill the missed shots.

Abbreviation	Method
F	Shots Filter
R	Face Recognition
C	CNN Based Scene Retrieval
B	BoW Based Scene Retrieval
A	Score Adjustment and Result Expansion
T	Text script search and Speaker identification
P	Previous Groundtruth Cues

Table 3: Description of our methods

ID	MAP	Method
F_NO_NERCMS_1	0.758	F+R+C+B+A+T+P
F_NO_NERCMS_2	0.632	F+R+C+B+A
F_NO_NERCMS_3	0.135	R+C
F_NO_NERCMS_4	0.172	R+B

Table 4: Results of our submitted 4 runs on Instance Search task of TRECVID 2016

After analysing our results, we get some suggestions and experiences to guide future work: (1) When the speaker identification algorithm is tested on a small-scale dataset, the effect is not too bad. However, when it is applied to INS task, the real effect is not as good as we expected. We suggest increasing the size of audio library and researching to omit the background sounds. (2) The retrieval efficiency needs to be improved, so research and measures to improve retrieval speed are necessary.

Acknowledgement. Thanks for the great support to our work by professor Ruimin Hu and professor Jun Chen, who provide us lots of thoughts and ideas. Thanks for the support by DaQian Information co., Ltd, which provides sources of face recognition. And thanks for former members, who are Mang Ye in Hong Kong Baptist University, Wei Zhu working at Siemens, Lei Yao, Dongshu Xu and Lin Wu in National Engineering Research Center for Multimedia Software, School of Computer, Wuhan University.

References

- [1] Paul Over, et al. Trecvid 2015—an overview of the goals, tasks, data, evaluation mechanisms and metrics, Proceedings of TRECVID, 2015.
- [2] Wei Zhang, Chong-Wah Ngo. Topological spatial verification for instance search, IEEE Transactions on Multimedia, 2015.
- [3] George Awad, Jonathan Fiscus, Martial Michel, David Joy, Wessel Kraaij, Alan F. Smeaton, Georges Qunot, Maria Eskevich, Robin Aly, Gareth J. F. Jones, Roeland Ordeman, Benoit Huet, and Martha Larson. 2016. TRECVID 2016: Evaluating Video Search, Video Event Detection, Localization, and Hyperlinking, In Proceedings of TRECVID 2016. NIST, USA.
- [4] Paul Over, et al. Trecvid 2014—an overview of the goals, tasks, data, evaluation mechanisms and metrics, Proceedings of TRECVID, 2014.
- [5] Sivic Josef, Andrew Zisserman. Video Google: A text retrieval approach to object matching in videos, ICCV, 2003.
- [6] LeCun Yann, et al. Gradient-based learning applied to document recognition, Proceedings of the IEEE, 1998.
- [7] Yuxin Peng, et al. PKU-ICST at TRECVID 2015: Instance Search Task, Participant Notebook Paper of TRECVID, 2015.
- [8] Y. Zhu, J. Wang, C. Zhao, H. Guo and H. Lu. Scale-adaptive Deconvolutional Regression Network for Pedestrian Detection, ACCV, 2016.
- [9] Kaiming He, et al. Deep residual learning for image recognition, arXiv, 2015.
- [10] Olga Russakovsky, et al. Imagenet large scale visual recognition challenge, IJCV 2015.
- [11] Gross, S., and M. Wilber. Training and investigating residual nets, 2016.
- [12] Shaoqing Ren, et al. Faster R-CNN: Towards real-time object detection with region proposal networks, NIPS, 2015.
- [13] Karen Simonyan, Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, arXiv, 2014.
- [14] Haiyun Guo, et al. Multi-View 3D Object Retrieval with Deep Embedding Network, ICIP, 2016.
- [15] Shaoqing Ren, et al. Face alignment at 3000 fps via regressing local binary features, CVPR, 2014.
- [16] Pierre Moreels, Pietro Perona. Evaluation of features detectors and descriptors based on 3d objects, IJCV, 2007.
- [17] James Philbin, et al. Object retrieval with large vocabularies and fast spatial matching, CVPR, 2007.
- [18] Relja Arandjelovic, Andrew Zisserman. Three things everyone should know to improve object retrieval, CVPR, 2012.
- [19] Cai-Zhi Zhu, Herve Jegou, Shin Ichi Satoh. Query-adaptive asymmetrical dissimilarities for visual object retrieval, ICCV, 2013.
- [20] Lei Yao, et al. WHU-NERCMS at TRECVID2015: Instance Search Task, Participant Notebook Paper of TRECVID, 2015.
- [21] Yannis Kalantidis, et al. Scalable triangulation-based logo recognition, ICMR, 2011.
- [22] Liang Zheng, et al. Query-Adaptive Late Fusion for Image Search and Person Re-identification, CVPR, 2015.