BUPT-MCPRL at TRECVID 2014*

Zhicheng Zhao, Wenhui Jiang, Qi Chen, Jinlong Zhao, Yuhui Huang,
Xiang Zhao, Lanbo Li, Yanyun Zhao, Fei Su, Anni Cai

Multimedia Communication and Pattern Recognition Labs,
Beijing University of Posts and Telecommunications, Beijing 100876, China
{zhaozc, zyy, sufei}@bupt.edu.cn

Abstract

In this paper, we describe BUPT-MCPRL systems and evaluation results for TRECVID [16] 2014. Our team participated in two tasks: instance search and surveillance event detection. This year, our systems show superior performance in both tasks compared with the results of last year.

Instance Search (INS): We submit three runs for automatic INS and one run for interactive search, and a brief description is as follows:

- **F_D_BUPT_MCPRL_1**: three local features + CNN
- **F_D_BUPT_MCPRL_3**: three local features + selective search + CNN
- **F_D_BUPT_MCPRL_4**: three local features + average query expansion + CNN
- **I_D_BUPT_MCPRL_2**: three local features + average query expansion (interactive)

Surveillance Event Detection (SED): We focus on five events: Embrace, PeopleMeet, PeopleSplitUp, PersonRuns and Pointing.

- **Embrace and Pointing**: CNN (pedestrian detection + key-posture detection) + rules
- **PeopleMeet, PeopleSplitUP**: pedestrian tracking + trajectory detection + HMM + rules
- **PersonRuns**: pedestrian tracking + trajectory detection + linear regression + rules

1 Instance Search

This year, we propose a similar search framework for both automatic and interactive search tasks. Figure 1 shows the overview of our INS system. Firstly, video keyframes with a sample rate of 2.5 fps are extracted, and then, local and global features are extracted to describe the image content. In our experiment, three local features, Harris-Laplace detector with HSV-SIFT descriptor, MSER detector with RootSIFT descriptor, and Hessian detector with RootSIFT descriptor are adopted. As for global features, we extract deep learning features based on CNN learned from ImageNet to represent different instances. Subsequently, two re-ranking schemes are followed to improve the initial retrieval performance. The first scheme is based on weighted query expansion. In the second method, we apply a re-ranking step on the top 100 frames of initial search by localized search. Finally, we consider the maximum frame score as the shot score and rank the video shots as the evaluation results, which are shown in Table 1. More details will be given in the following sections.

*This work is supported by Chinese National Natural Science Foundation (61101212, 61372169, 61471049), National High and Key Technology Research and Development Program (2012AA012505).

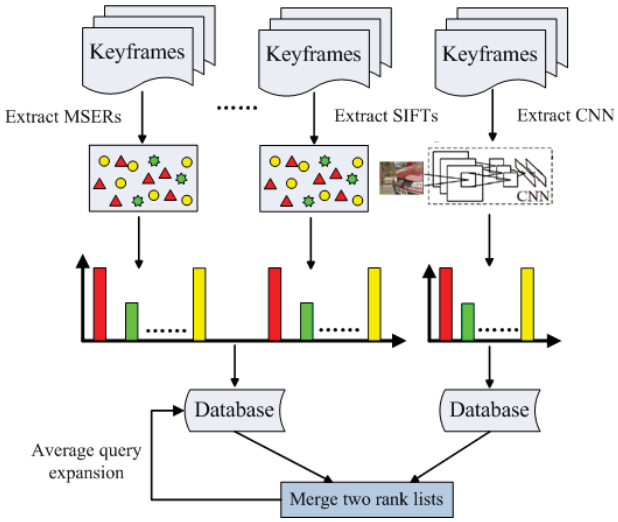

Figure 1. An overview of our system in instance search task

Table 1. Results for each run

Run ID	mAP
F_D_BUPT_MCPRL_1	22.7
F_D_BUPT_MCPRL_3	22.1
F_D_BUPT_MCPRL_4	21.6
I_D_BUPT_MCPRL_2	23.7

1.1 Ranking with local features

We extract three kinds of local features from each keyframe: MSER detector with RootSIFT descriptor [1], Hessian detector with RootSIFT descriptor, and Harris-Laplace detector with HSV-SIFT descriptor [2]. Three large codebooks comprising 1M visual words are subsequently trained by AKM [3]. Each descriptor is projected into 3 neighboring visual words. It is applied on both queries and dataset images. We finally use the inverted file system to efficiently index each descriptor.

Assume two images X and Y are described by feature histograms $\boldsymbol{X}$ and $\boldsymbol{Y}$, the similarity between $\boldsymbol{X}$ and $\boldsymbol{Y}$ can be represented by

$$\mathrm{sim}(\boldsymbol{X},\boldsymbol{Y}) = N(\boldsymbol{X})N(\boldsymbol{Y})F(\boldsymbol{X},\boldsymbol{Y}) \tag{1}$$

where the function $F(\boldsymbol{X},\boldsymbol{Y})$ determines the similarity between two images, and $N(\cdot)$ is the normalization factor. In our system, we adopt inner product as the similarity function and SSR [1] for normalization since it gives superior performance than L2 normalization.

We take advantage of *E-Idf* to improve the discrimination between visual words.

$$eidf_i = \log\left(\frac{e^{-\frac{n_i}{\alpha}}(N - n_i + e^{\frac{n_i}{\alpha}} - e^{-\frac{n_i}{\alpha}} + 1)}{(e^{\frac{n_i}{\alpha}} - e^{-\frac{n_i}{\alpha}} + 1)(n_i + e^{-\frac{n_i}{\alpha}})}\right) \tag{2}$$

Besides, as discussed in some related works [4], the context around the mask in the query image is helpful, thus we utilize both the information provided by the regions-of-interest and the contexts around the ROIs for retrieval. In order to emphasis the importance of ROIs, larger weights are set on these regions, which further improve the accuracy of our system.

Table 2. Performance of different local features on INS2013

local features	points per image	mAP
MSER + RootSIFT	around 150	16.308
Hessian + RootSIFT	around 500	12.739
Harris + HsvSIFT	around 250	12.967
Total	around 900	21.731

1.2 Ranking with deep learning features

Techniques based on deep neural networks have substantially improved the state-of-the-art in many recognition tasks such as image classification and object detection, thus we expect deep neural networks will be helpful to enhance the performance of instance search. Because there is no training data for instance search, we are not able to train a model specific for our instance search task. Inspirited by [5], CNN model [9] can be used as a feature extractor. Therefore, in our system, we extract deep learning features by CNN pre-trained on ImageNet dataset. Convolutional neural features serve as global features in our tasks. We made experiments on features extracted from different layers of CNN on INS2013 task. From Table 3, we can see that the output of fc6 without Relu gives the best performance. This result is consistent with [6].

Table 3. Performance of convolutional neural features on INS2013

Layer	Dim	Metric	mAP
Fc6	4096	L2	3.84
Fc6 + Relu	4096	SSR	3.43
Fc7 + Relu	4096	L2	3.07
Fc7 + Relu	4096	SSR	2.67
Fc8	1000	SSR	1.34

After fusing the results from local features and convolutional neural features, we got the result of 22.7% in INS2014 task.

1.3 Re-ranking

Re-ranking procedure is subsequently followed to improve initial ranking results. Two methods are designed to re-rank the initial result. The first method is based on weighted query expansion as described in [7]. This gives a mAP of 21.6%. Compared with initial ranking, the re-ranked results seem worse, and we will analysis the result in the future. In the second method, we apply a re-ranking step on the top 100 frames of initial search by localized search, but this time, the initial result is obtained using only local features. The re-ranking method is similar to [8], and our result is 22.1%.

1.4 Conclusion

We extract both local and global features for instance search. Three kinds of local features are extracted and fused shows that rich features are important in improving the performance. CNNs have recently been substantially improving upon the state of the art in image recognition tasks. However, CNNs cannot be directly employed to instance search task. In the next year, we will further study on deep convolutional networks.

2 Surveillance Event Detection

This year we pay more attention to the events of Embrace, PeopleMeet, PeopleSplitUp, PersonRuns and Pointing. In our system different algorithms are adopted to detect events accordingly. Our system mainly includes two parts: the retrospective part and the interactive part. The retrospective part consists of pedestrian detection, pedestrian tracking and event detection. The interactive part is an extension of the retrospective part.

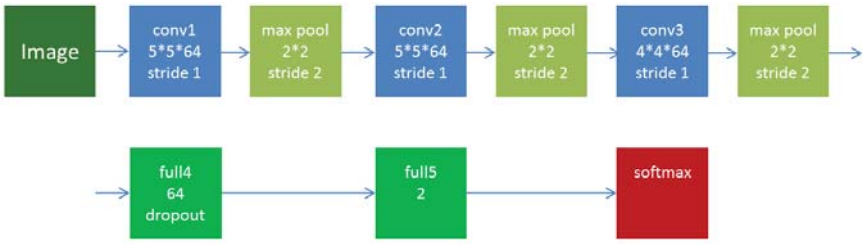

Figure 2 The architecture of CNNs for pedestrian recognition.

2.1 Pedestrian Detection

Convolutional neural networks (CNNs) achieved outstanding performance on image classification tasks [9], thus we apply it to pedestrian recognition. Compared to complex architecture of [9], we design a relatively simple CNN for pedestrian recognition on a small training dataset. The architecture of our CNN is shown in Figure 2.

We collect 11538 positive images from TRECVID 2008 dataset for training and 4946 ones for testing. As shown in Figure 3, we only deal with the head-shoulder part of human body to suppress the effects of occlusion, and meanwhile, randomly sampled background windows from non-head-shoulder parts as negative samples.

During the course of pedestrians detecting, a set of candidate pictures for each image of video in a sliding window is firstly generated [10], and then is sent to CNN to classify. Finally we refine the results from CNN by non-maximal suppression (NMS). Some results are shown in Figure 4.

Figure 3 Training samples. Left are positive samples and Right are negative samples.

Figure 4 Pedestrian detection results of four scenes

2.2 Pedestrian Tracking

According to pedestrian detection results, we exploit a multi-target pedestrian tracking by online learning of non-linear motion patterns and robust appearance models [11], a method of hierarchical association of detection responses, which processes detection result at different levels, where a non-linear motion pattern and robust appearance models for each tracked target could be learned. With this hierarchical association framework, we obtain a robust tracking result under the condition of occlusion. Finally, Gaussian process regression is used to optimize primitive trajectory to obtain smooth one. A tracking result is shown in Figure 5.

Figure 5 Left is the primitive trajectory and Right is the smoothed trajectory

2.3 Event Detection

Our SED detection method in this year mainly depends on the application of CNN and trajectory analysis (Hidden Markov Model and Motion History Image).

2.3.1 Embrace and Pointing

Pointing and Embrace both have a key-pose as shown in Figure 6, so we adopt CNN, similar pedestrian recognition method, to recognition the event of Pointing and Embrace. We use the same architecture shown in Figure 2 to train models of Embrace and Pointing events. The pedestrian detection results with 1.5-fold expansion are regarded as the input of the network.

Figure 6 Left is the samples of Pointing and Right is the samples of Embrace. We normalize the pointing direction of left and right to one direction.

2.3.2 PeopleMeet, PeopleSplitUp

We divide PeopleMeet into 3 subevents: walking closely, slowing down and stay, and then use HMM(Hidden Markov Model) to model the event[13]. In our implement, trajectories of a pair of persons are detected, and then their distance and speed are used as features to train HMM [14] with 3 hidden states. Besides, Forward-algorithm is used to calculate the probability of an observation sequence. Finally, based on constraints of some rules such as time, place etc. PeopleMeet event is determined.

PeopleSplitUp could also split into 3 subevents: stay, speeding up, walking away, thus a similar method as PeopleMeet is used.

2.3.3 PersonRuns

Through computing the velocity parameter of tracked objects from trajectories, we distinguish PersonRuns event.Firstly, we choose the fast-moving pedestrian tracks from the large number of trajectories by Forward-backward Motion History Image (MHI)[15], which can filter trajectories belonging to stationary and walking people. Then, among the remained trajectories, a velocity threshold is set to distinguish running objects from the others. In order to get the velocity exactly, we adjust the camera distortion by linear regression.

2.4 Interactive System

The framework of the interactive system is similar to our proposed framework in last year[12]. For the automatic detection results of each kind of events, a manual intervention is applied to select the correct detections and eliminate the false positives within 25 minutes. During interactive procedure, we correct some wrong event labels. This process reduces the false alarm significantly, but contributes little for the missing activities.

2.5 Experimental Results

We show our primary run results on retrospective task in Table 4. The Embrace and Pointing results are detected by CNN. The PeopleMeet and PeopleSplitUp results are determined by trajectory analysis based on HMM. And the PersonRuns results are discriminated by MHI and trajectory analysis.

Table 4: The actual DCR and minimum DCR of the 2014 retrospective result

Event	#CorDet	#FA	#Miss	ActDCR	MinDCR
Embrace	26	44	112	0.8318	0.8318
PeopleMeet	6	128	250	1.0354	1.0018
PeopleSplitUp	19	158	133	0.9476	0.9455
PersonRuns	8	139	43	0.9070	0.9038
Pointing	21	57	774	0.9998	0.9953

2.6 Conclusion

This year we use CNN and trajectory analysis as our main methods. The method of CNN can work very well and it detects a small number of false alarms and a relatively big number of correct detection.

The methods of trajectory analysis and Motion History Image are also potential.

References

[1] R. Arandjelovic and A. Zisserman, “Three things everyone should know to improve object retrieval,” *CVPR’12*, pp. 1–8, 2012. [2] K. E. A. Van De Sande, S. Member, T. Gevers, and C. G.

[2] K. E. A. Van De Sande, S. Member, T. Gevers, and C. G. M. Snoek, “Evaluating Color Descriptors for Object and Scene Recognition,” *TPAMI*, vol. 32, no. 9, pp. 1582–1596, 2010.

[3] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” in CVPR’07, 2007.

[4] C.-Z. Zhu and S. Satoh, “Large vocabulary quantization for searching instances from videos,” *ICMR’12*, pp. 1–8, 2012.

[5] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, T. Darrell, T. Eecs, and B. Edu, “DeCAF : A Deep Convolutional Activation Feature for Generic Visual Recognition,” in ICML 2014, 2014.

[6] A. Sharif, R. Hossein, A. Josephine, and S. Stefan, “CNN Features off-the-shelf : an Astounding Baseline for Recognition,” in *CVPR workshop*, 2014.

[7] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, “Total Recall: Automatic Query Expansion with a Generative Feature Model for Object Retrieval,” in *ICCV’07*, 2007.

[8] R. Tao, E. Gavves, C. G. M. Snoek, and A. W. M. Smeulders, “Locality in Generic Instance Search from One Example,” in *CVPR 2014*, 2014, pp. 1–8.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton. Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems 25, 2012.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In IEEE CVPR, volume 1, pages 886–893, 2005.

[11] Bo Yang and Ram Nevatia. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In IEEE CVPR 2013.

[12] Zhixuan Li, Yuhui Huang, Kaiqi Zhang, Zhicheng Zhao et al. BUPT-MCPRL at TRECVID 2013.In: Proceedings of TRECVID 2013 Workshop.

[13] Chan M T, Hoogs A, Schmiederer J, et al. Detecting rare events in video using semantic primitives with HMM. Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on IEEE, 2004, 4: 150-154.

[14] Xiaolin Li, et al. Training hidden Markov models with multiple observations–A combinatorial method. Pattern Analysis and Machine Intelligence, IEEE Transactions on Volume:22, Issue: 4.

[15] Zhaozheng Yin, Robert Collins. Moving object localization in thermal imagery by forward-backward motion history images. Augmented Vision Perception in Infrared. Springer London, 2009: 271-291.

[16] Paul Over, George Awad, Martial Michel, Jonathan Fiscus, Greg Sanders, Wessel Kraaij, Alan F. Smeaton, Georges Quéenot. TRECVID 2014 -- An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics. Proceedings of TRECVID 2014.