Exploring Fisher Vector and Deep Networks for Action Spotting

Zhe Wang[1] Limin Wang[1,2] Wenbin Du[1] Yu Qiao[1]

[1]Shenzhen key lab of Comp. Vis. & Pat. Rec., Shenzhen Institutes of Advanced Technology, CAS, China

[2]Department of Information Engineering, The Chinese University of Hong Kong

buptwangzhe2012@gmail.com, 07wanglimin@gmail.com, wb.du@siat.ac.cn, yu.qiao@siat.ac.cn

Abstract

This paper describes our method and attempt on track 2 at the ChaLearn Looking at People (LAP) challenge 2015. Our approach utilizes Fisher vector and iDT features for action spotting, and improve its performance from two aspects: (i) We take account of interaction labels into the training process; (ii) By visualizing our results on validation set, we find that our previous method [10] is weak in detecting action class 2, and improve it by introducing multiple thresholds. Moreover, we exploit deep neural networks to extract both appearance and motion representation for this task. However, our current deep network fails to yield better performance than our Fisher vector based approach and may need further exploration. For this reason, we submit the results obtained by our Fisher vector approach which achieves a Jaccard Index of 0.5385 and ranks the 1^{st} place in track 2.

1. Introduction

Recognizing action in videos is one of the most challenging problems in computer vision [12, 16, 17, 18, 19, 20]. The goal is to automatically classify the action ongoing in a video into a predefined category. It has a wide range of applications including surveillance, human computer interaction, and content-based retrieval. However, most of the existing research works concentrate on action dataset with trimmed videos, such as HMDB51 [9] and UCF101 [13], which focus on classification tasks. In this paper, we describe our approach developed based on the untrimmed dataset from the ChaLearn Looking at People (LAP) challenge [5], which handle action temporal spotting in a continuous video stream simultaneously.

Video representation plays an important role in the action recognition and detection. From a recent study work [11], the Fisher vector representation with improved Dense Trajectory features proves to be very effective for action recognition and has obtained the state-of-the-art performance on HMDB51 [9] and UCF101 [13]. Meanwhile, convolutional neural networks (CNNs) have shown great success in image classification, object detection and so on. CNNs have also been applied on action recognition [12, 20] and obtain even better performance. In this paper, we also make an attempt to apply CNN with appearance information and motion information to action spotting.

Track 2 focuses on action/interaction recognition from 9 untrimmed videos with 11 action classes, such as wave, point, clap, hug and kiss [5]. The main challenge comes from the fact that there are multiple action instances in a single video stream. The task also requires to perform the actor prediction for each action instance in the input video.

The reset of this paper is organized as follows. We give a detailed description about our method on Track 2 in Section 2. Then, we will give some description on our attempt using deep learning methods in Section 3. And we will report the performance of the proposed method in Section 4. Finally, we conclude our paper in Section 5.

2. Method

Our method is mainly based on our previous work [10] which used Fisher vector for action classification. Compared with [10], we make two main modification with this work. Figure 1 demonstrates the pipeline of our method. Our approach is composed of 3 steps: (i) feature extraction (ii) temporal pooling and segmentation (iii) clip classification and post processing. Detailed descriptions are as follows.

2.1. Feature Extraction

Improved Dense trajectory [16] is very popular in action recognition due to its robustness to background clutter and independence on detection and tracking techniques. It tries to select locations and scales in video by dense sampling strategy. To describe the extracted region, several hand-crafted features such as HOG [3], HOF [4], MBH [15] are also extracted. In [16], it set sample stride as 5 frames and sample length as 15 frames. Thus the trajectory with length of 15 frames is extracted. As stated in [11], sampling with smaller stride and shorter length can yield bet-

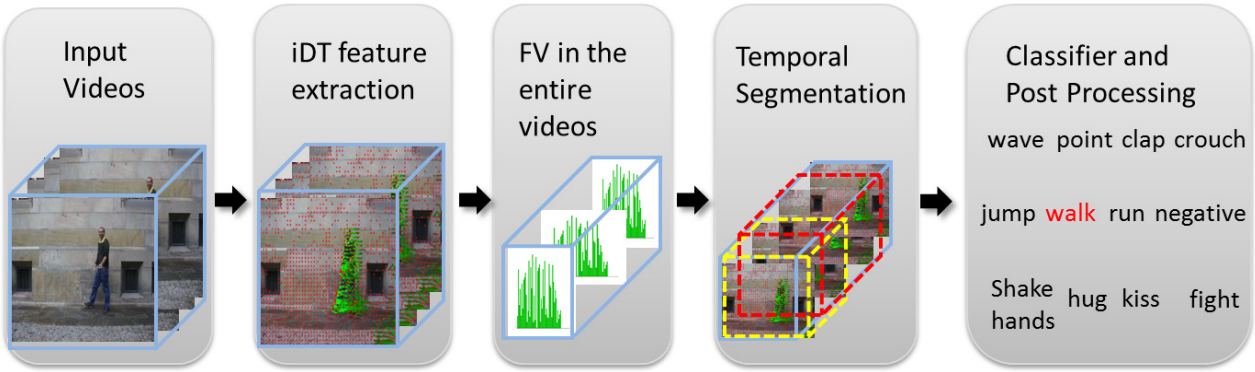

Figure 1. The pipeline of our action spotting system based on Fisher vector. (i) feature extraction, (ii) temporal pooling and segmentation, (iii) clip classification and post processing.

ter performance but at the cost of computation and memory demand. We set stride as 3 frames and sample length as 9 frames. The size of extracted feature for 1 video is 3 times as large as former one.

2.2. Temporal Pooling and Segmentation

As the videos provided for us is a continuous stream with multiple action instances, we need to conduct action localization first. Inspired by the sliding window strategy in object detection, we resort to a temporal sliding window scheme to conduct action localization. We set the sliding window length as 15 frames and stride as 6 frames based on the average length of actions in ground truth. We use a different stride in comparison with [10] which leads to better performance.

For the convenience of extracting features from the length and stride set above, we encode the improved dense trajectory features using Fisher vector and pool them over the whole video. Thus we calculate the integral Fisher vector of the whole video and can get any length video representation from any time point for clip classification.

2.3. Clip Classification and Post Processing

For clip classification, we mainly follow the work of [16] which includes the following steps: (i) extracting the improved Dense Trajectory features from videos, (ii) using the PCA and Whiten technique to remove the correlations among different dimensions and normalize the variance, (iii) learning a generative Gaussian Mixture Model (GMM), (iv) generating the Fisher vector code based on the GMM [21], (v) training SVMs [14] for classification.

Another improvement in comparison with [10] is we introduce a multi-threshold strategy in the post processing stage. By visualizing the results on the validation dataset. we found the former threshold is weak in detecting action class 2, which is illustrated in Figure 2. Thus we identify a threshold with good evaluation performance for action

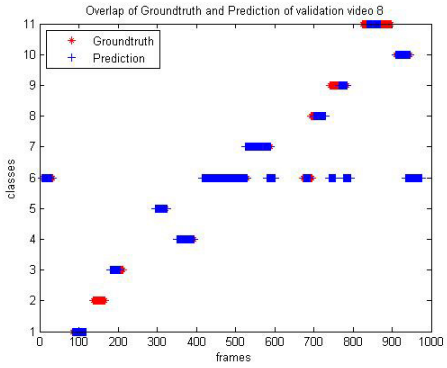

Figure 2. The visualization of ground truth and prediction of validation video 8, the blue represent the prediction while the red represent the ground truth. In the Figure we can conclude that this post-processing threshold is weak in detecting action 2.

class other than 2, and find another good threshold for action class 2. Finally, we combine the results of these two thresholds.

3. Attempt on Deep Learning Method

Lots of deep learning attempts have been conducted on computer vision topic since [8]. Karpathy *et al.* [7] constructed a large database Sports-1M and used video frames to train their net, Krizhevsky *et al.* [12] combined appearance information and motion information by feeding CNN with video frames and optical flow. Wang *et al.* [20] proposed Trajectory-pooled Deep-convolutional Descriptors by using the CNN and achieves state-of-art performance on UCF101 and HMDB51. Following their work [1], we make an attempt with both appearance and motion net on the ChaLearn LAP dataset [5].

[1]The code and model is available at webpage: https://wanglimin.github.io/tdd/index.html

3.1. Spatial Convolutional Neural Network

We hope the spatial convolutional neural network could capture the appearance information of the action in the video. For the attempt on spatial convolutional neural network, we first extract frames from videos. And treat video label as picture label. Our convolutional neural network has the same structure as that of [2], and use the pre-trained model on ImageNet dataset [8]. Then, we fine tune the model for task of action/interaction recognition. The architecture of the network is illustrated in Figure 3.

3.2. Temporal Convolutional Neural Network

We hope the temporal convolutional neural network could capture the motion information of the action in the video. Thus, we follow the work of [20], and extract optical flow [22] from videos and store them as gray images, which can be used as the input for temporal convolutional neural network. We treat video action class label as optical flow label as well. We use the structure in [2] and use the model pre-trained on UCF101 [13]. It is worth noticing that the fine tuning details is not the same as the spatial convolutional neural network. As described in [12], optical flow is always pooled over several frames to better represent an action. Thus, we pool 10 frames of optical flow here and treat them as the input to a single temporal CNN. Then we fine tune both temporal CNN and spatial CNN with video category information. We implement and train the CNNs with Caffe [6]. Some example frames from the training videos are illustrated in Figure 4.

4. Experiments

In this section, we first describe the dataset and the evaluation measurement of action/interaction recognition at the ChaLearn LAP Challenge 2015 [1]. Then we give detailed description of the implementation details of training spatial convolutional neural networks, temporal convolutional neural network and Fisher vector. Finally, we present the experimental results of proposed method on the testing dataset.

4.1. Dataset and Measurement

There are 11 action classes such as wave, point, clap and so on in track 2 of action/interaction recognition dataset [5]. For the training data, there are 5 video sequences, containing 136 action instances. For validation data, there are 2 videos including 44 action instances.

For measurement, it uses the Jaccard Index to evaluate the performance of action/interaction spotting. The Jaccard Index is defined as follows:

$$J_{s,n} = \frac{A_{s,n} \cap B_{s,n}}{A_{s,n} \cup B_{s,n}}$$

Where $A_{s,n}$ is the ground truth of action n at sequence s, and $B_{s,n}$ is the prediction for action n in sequence s. Our method is evaluated with the mean Jaccard Index among all action classes.

4.2. Implementation Details

Evaluation using Fisher Vector: When training the SVM classifier using Fisher vector representation, we not only use the 11 given action classes, but also mine some negative clips which represent static background or noisy motion. During test phase, if a video sub-window is predicted as the background class, we will remove it from the detection results. We use a simple method to determine the user of action class. We firstly label all actions with user 1, and add user 2 when interaction occurs. Then the time label for user 1 will be copied for user 2. For post processing, we set threshold as 0.39 for classifier without action class 2 and set threshold as 0.55 for action class 2. Then we combine the results from two post processing thresholds.

Evaluation using Spatial Convolutional Neural Network: There are 4,718 frames in training data and 1,827 frames in validation set. The network weights are learnt using the mini-batch stochastic gradient descent with momentum (which is set as 0.9). At each iteration, a mini-batch of 256 samples is constructed by randomly sampling and sent to network. During training phase, all the images are resized to 256×256 and a 227×227 sub-image is randomly cropped from the image. Then they are manipulated with horizontal flipping. The dropout ratio for last two fully-connected layer is set to 0.5. The learning rate is initially set to 10^{-2} and decreases to 10^{-3} after 8k iteration, to 10^{-4} after 16k iterations. It stops at 30k iterations. During testing phase, we do the same manipulation as in the training phase.

Evaluation using Temporal Convolutional Neural Network: As the smallest number of frames we can feed temporal neural network is 10, the data we can use for training is 4,668 frames, for validation is 1,807 frames. The training strategy is almost the same except that the batch size is 64 and the dropout ratio is 0.8. We feed 10 successive frames of optical flow on x axis and on y axis to temporal convolutional neural network each 1/64 mini-batch.

4.3. Experimental Results

We report the action spotting performance on track 2 of ChaLearn Looking at People (LAP) challenge, and the results with Fisher vector approach are shown in Table 1. We also find the our current CNN based method don't obtain good results. This might be ascribed to two facts. Firstly, we do not have sufficient data to train the temporal and spatial nets for actions. Secondly, the supervised information

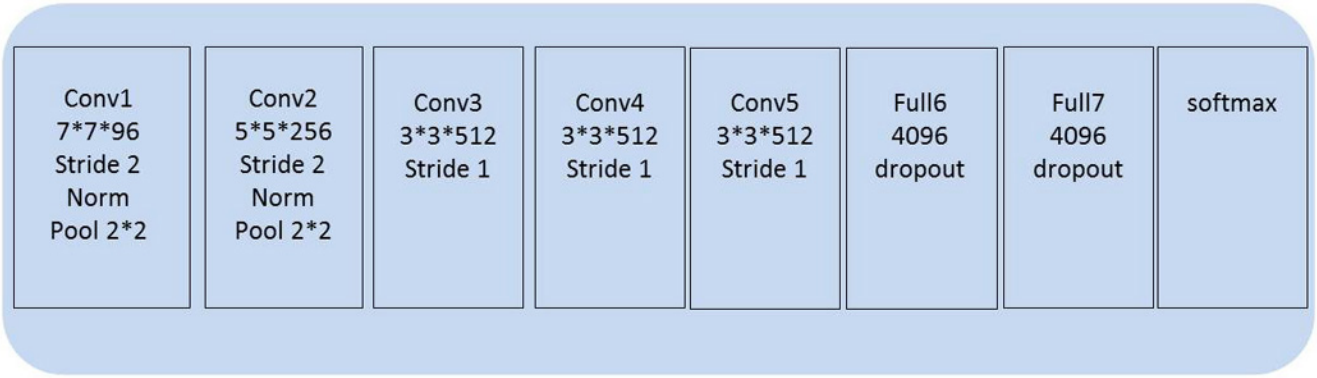

Figure 3. The architecture of Spatial and Temporal Convolutional Neural Network for action/interaction recognition.

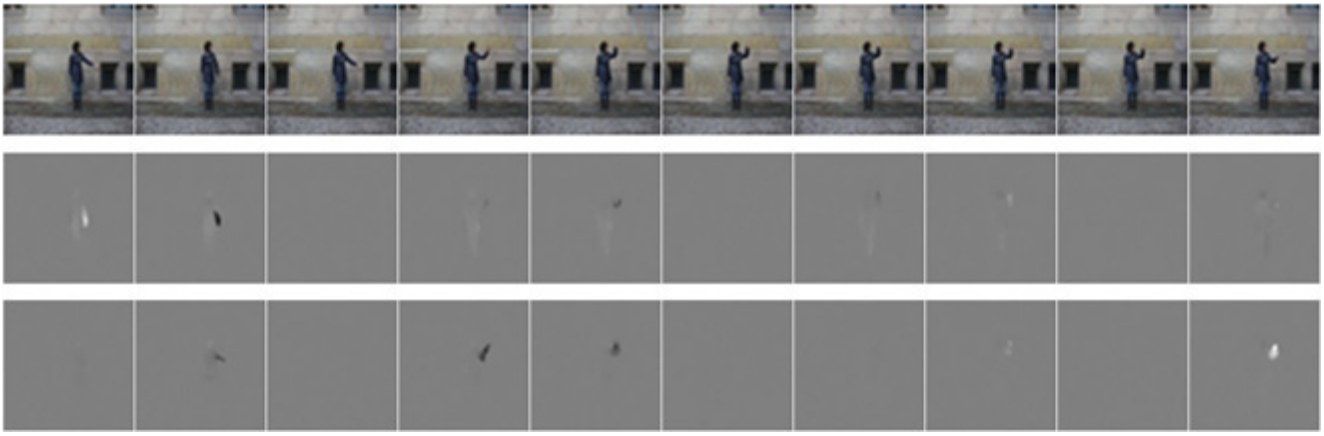

Figure 4. In the first row are the samples of action/interaction recognition dataset at the ChaLearn LAP Challenge 2015. In the second and third row are the optical flow for the corresponding frames in the first row, and the second row is for flow on x axis while the third row for flow on y axis. The optical flow is the source for temporal convolutional neural network and we feed 10 successive frames of optical flow on x axis and y axis to temporal net each 1/64 mini-batch.

Rank	Team	Score
1	Ours	0.5385
2	FKIE	0.5239

Table 1. Comparison between the performance of our Fisher vector spotting system and that of other team.

for CNN training can be noisy, which harms the final performance. Our approach obtains the best performance for track 2.

5. Conclusions

We have presented our method designed for track 2 at the ChaLearn Looking at People challenge. We explore both Fisher vector and deep networks for this task. The performance of our method is effective for action/interaction recognition and ranks 1st in the challenge. In the future, we may tune the convolutional neural networks and combine the with the Fisher vector representation to further improve the performance.

Acknowledgement

This work is supported by a donation of Tesla K40 GPU from NVIDIA corporation. Limin Wang is supported by Hong Kong PhD Fellowship. Yu Qiao is supported by National Natural Science Foundation of China (91320101, 61472410), Shenzhen Basic Research Program (JCYJ20120903092050890, JCYJ20120617114614438, JCYJ20130402113127496), 100 Talents Program of CAS, and Guangdong Innovative Research Team Program (No.201001D0104648280).

References

[1] X. Baro, J. Gonzlez, J. Fabian, M. A. Bautista, M. Oliu, I. Guyon, H. J. Escalante, and S. Escalers. Chalearn looking at people challenge 2015: Dataset

and results: action spotting and cultural event recognition. In *CVPR, ChaLearn Looking at People workshop*, 2015. 3

[2] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, pages 1–11, 2014. 3

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

[4] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, pages 428–441. 2006. 1

[5] S. Escalera, X. Baro, J. Gonzalez, M. A. Bautista, M. Madadi, M. Reyes, V. Ponce, H. J. Escalante, J. Shotton, and I. Guyon. Chalearn looking at people challenge 2014: Dataset and results. In *ECCV, ChaLearn Looking at People Workshop*, pages 459–473, 2014. 1, 2, 3

[6] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM MM*, pages 675–678, 2014. 3

[7] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, pages 1725–1732, 2014. 2

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012. 2, 3

[9] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: A large video database for human motion recognition. In *ICCV*, pages 2556–2563, 2011. 1

[10] X. Peng, L. Wang, Z. Cai, and Y. Qiao. Action and gesture temporal spotting with super vector representation. In *ECCV, ChaLearn Looking at People Workshop*, pages 518–527, 2014. 1, 2

[11] X. Peng, L. Wang, X. Wang, and Y. Qiao. Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. *CoRR*, abs/1405.4506, 2014. 1

[12] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576. 2014. 1, 2, 3

[13] 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. 1, 3

[14] A. Vedaldi and B. Fulkerson. Vlfeat: an open and portable library of computer vision algorithms. In *ICM*, pages 1469–1472, 2010. 2

[15] H. Wang, A. Kläser, C. Schmid, and C. Liu. Dense trajectories and motion boundary descriptors for action recognition. *IJCV*, 103(1):60–79, 2013. 1

[16] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, pages 3551–3558, 2013. 1, 2

[17] L. Wang, Y. Qiao, and X. Tang. Mining motion atoms and phrases for complex action recognition. In *ICCV*, pages 2680–2687, 2013. 1

[18] L. Wang, Y. Qiao, and X. Tang. Motionlets: Mid-level 3D parts for human motion recognition. In *CVPR*, pages 2674–2681, 2013. 1

[19] L. Wang, Y. Qiao, and X. Tang. Video action detection with relational dynamic-poselets. In *ECCV*, pages 565–580, 2014. 1

[20] L. Wang, Y. Qiao, and X. Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *CVPR*, pages 1–10, 2015. 1, 2, 3

[21] X. Wang, L. Wang, and Y. Qiao. A comparative study of encoding, pooling and normalization methods for action recognition. In *ACCV*, pages 572–585, 2012. 2

[22] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l1 optical flow. In *PR*, pages 214–223. 2007. 3