TUBE CONVNETS: BETTER EXPLOITING MOTION FOR ACTION RECOGNITION

Zhihao Li, Wenmin Wang⋆*, Nannan Li, Jinzhuo Wang*

School of Electronic and Computer Engineering, Peking University
lizhih@pku.edu.cn, ⋆wangwm@ece.pku.edu.cn, linn@pkusz.edu.cn, jzwang@pku.edu.cn

ABSTRACT

Motion information is a key factor for action recognition and has been eagerly pursued for decades. How to effectively learn motion features in Convolutional Networks (ConvNets) remains an open issue. Prevalent ConvNets often take several full frames of video as input at a time, which can be a heavy burden for network training. In this paper, we introduce a novel framework called Tube ConvNets, by substituting action tubes for full frames to reduce this burden. Tube ConvNets focus on the regions of interest (ROI) where key motions occur, and thus eliminate the distraction of irrelevant objects. Each action tube is a fraction of spatio-temporal volumes, generated by the techniques of object detection and clustering algorithm. We demonstrate the effectiveness of Tube ConvNets for action classification on UCF-101 dataset, and illustrate its potential to support fine-grained localization on UCF-Sports dataset. Source code is available at https://github.com/wangjinzhuo/tubecnn.

***Index Terms*—** action recognition, motion information, ConvNets, action tubes, action localization

1. INTRODUCTION

Action recognition is an active topic in the computer vision community [1, 2]. Compared with recognition of still images, this task is more challenging due to the large intra-class variations, low video resolution, high dimension of video data, and so on.

In earlier years, researchers working on video understanding often employ hand-crafted features that are based on space-time-interest-point (STIP) [3] (e.g., 3D-SIFT [4], 3D-HOG [5], dense trajectory (DT) [6], and its upgraded version, improved dense trajectory (iDT) [1]). By combining advanced statistical models (e.g., Fisher Vector and Gaussian Mixture Model (GMM)), video representations are achieved for detection and classification. However, the performance of such methods relies on human-designed features, which may form a bottleneck [7].

Recently, deep learning models especially convolutional networks (ConvNets) enjoyed such a great success in many image recognition tasks [8, 9] that there is a trend extending deep architectures from image data to video domain [10, 11, 12]. This class of methods often train action recognition systems in an end-to-end manner, from raw video data to the corresponding labels. The authors of [10] first designed 3D Convolutional operations for consecutive frames by treating space and time as equivalent dimensions. While a thorough comparison of different models in [11] indicated that operating on single frames performs even better than considering sequence of frames. This is perhaps due to the heavy burden that deep ConvNets suffer from high dimensional video data, because deep ConvNets in this fashion are expected to learn not only low-level motion features, but also their high-level configurations. To solve this problem, [2] designed temporal stream ConvNets by stacking optical flow displacement fields between consecutive frames as input, and achieved better performance.

However, there is much interferential noise in optical flow signals, which is often produced by irrelevant objects and camera motions. Such noise offers little semantic information but can be a burden for the models. We argue that suppressing such noise can better exploit motion information in deep ConvNets. Our motivation is also inspired by Region-based Convolutional Neural Networks (R-CNN [13]) and its upgraded versions [14, 15]. These methods hold the philosophy that focusing on ROI can take full advantages of deep learning. In other words, we expect to offer ConvNets basic motion atoms that is only provided by key actors, and let them learn high-level configurations of these motion atoms.

In this paper, we propose to embed action tubes into ConvNets for action recognition. Each action tube can be viewed as a subset of the original video in spatio-temporal domain, and used to locate regions of interest (ROI) where key motions have high probabilities to occur. We generate action tubes by the combination of object detection techniques and clustering algorithm. Specifically, object detection provides person proposals, then clustering algorithm separate them into several groups to find key actors. When sending action tubes to ConvNets, we obtain Tube ConvNets that can start from effective atoms to exploit motion information for action recognition. Besides, we find the action tubes can be also used for action localization task, where the performance on UCF-Sports benchmark are competitive to other part-based

This project was supported by Shenzhen Peacock Plan (20130408-183003656).

approaches.

Our contributions can be summarized as follows:

- We propose a method to generate action tubes that can effectively focus on regions of interest (ROI) that contain key motions.
- We embed action tubes into deep ConvNets and show promising performances for action recognition on UCF-101 dataset.
- We demonstrate the potential of action tubes to support a fine-grained analysis of action localization on UCF-Sports dataset and report competitive results.

The remainders of this paper are organized as follows. Section 2 presents our approach including the generation of action tubes, the construction of Tube ConvNets and network training techniques. We report experimental results and comparisons in Section 3. Finally, conclusions and future work are given in Section 4.

2. APPROACH

In this section, we first introduce a procedure to generate action tubes for a given video. Then, we construct deep ConvNets based on the action tubes for action recognition. Finally, we provide some details for training our Tube ConvNets.

2.1. Generation of Action Tubes

Action tubes are expected to contain discriminative motions in video, with which we eliminate distraction from irrelevant motions that are caused by non-human objects, uncritical persons and camera motions. We utilize a three-step procedure to generate action tubes effectively and efficiently, as following:

Object detection. We use Faster R-CNN models [15] trained on PASCAL VOC 2007 and 2012 datasets to detect objects in every frame, due to its good performance in terms of classification accuracy and time cost. Within detection results, we retain the human proposals, discarding non-human ones. However, the models can hardly detect persons whose poses are upside down or lying flat, and those poses are common in UCF-Sports dataset, such as *Diving*. To solve this problem, we augment the datasets by rotating training images 90, 180 and 270 degrees, and flipping the original and the rotated images horizontally. By this way, we obtain 7 more training samples from each original image. We re-train the Faster R-CNN models on this augmented data. The detection get improved results after re-trained, as shown in Fig. 1.

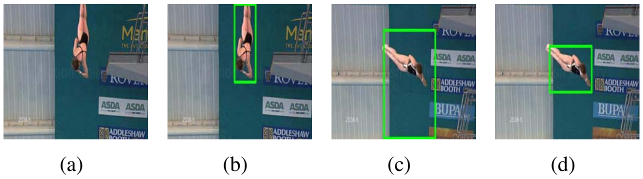
(a) (b) (c) (d)

Fig. 1. Comparison of detection results between the original and re-trained Faster R-CNN models. (a), (c): detections from the original models; (b), (d): detections from the re-trained models. It can be observed that the re-trained models can detect missed person proposals and produce more compact bounding boxes.

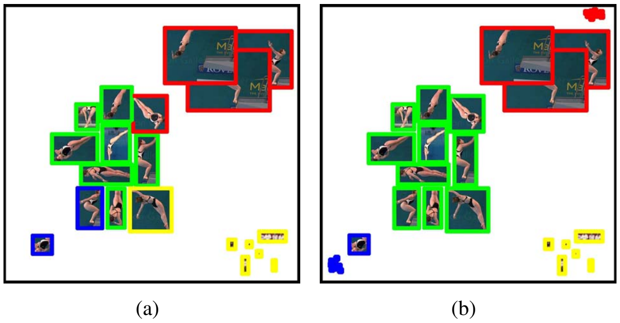
(a) (b)

Fig. 2. Clustering results of proposals with and without isolates. The blue spots in bottom-left corner and the red spots in top-right corner of (b) are isolates. Compared with (a), more proposals are clustered into right groups (each group has a distinct color).

Finding key actors. We cluster the person proposals into several groups. It is reasonable to assume that the proposals clustered in the biggest groups are from the key actors in video. We extract several local features, such as histogram of color and local binary pattern (LBP) [16], and concatenate them into descriptors with the same dimension for regions of different sizes. The DBSCAN clustering algorithm [17] is chosen due to its adaptive group numbers. In practice, we observe that DBSCAN tends to tear proposals apart which should be in the same group, in the case of only a few outliers. We generate some artificial proposals, which we called "isolates". With the appearance of isolates, the proposals that should be in the same group are correctly clustered, as shown in Fig. 2.

Refining tubes. By connecting members from the same clusters in chronological order, we obtain action tube proposals. For each tube proposal, we evaluate its motion saliency by calculating average optical flow magnitude. Then, we discard uncritical tubes of which motion saliency are below a certain threshold α or frame numbers are less than β. After the above processing, the remaining are considered as action tubes. Due to the missed detection of Faster R-CNN models or mis-clustering of DBSCAN, action tubes are discontinuous. Finally, we complete action tubes by utilizing Mean-shift tracking algorithm [18] on the missed frames.

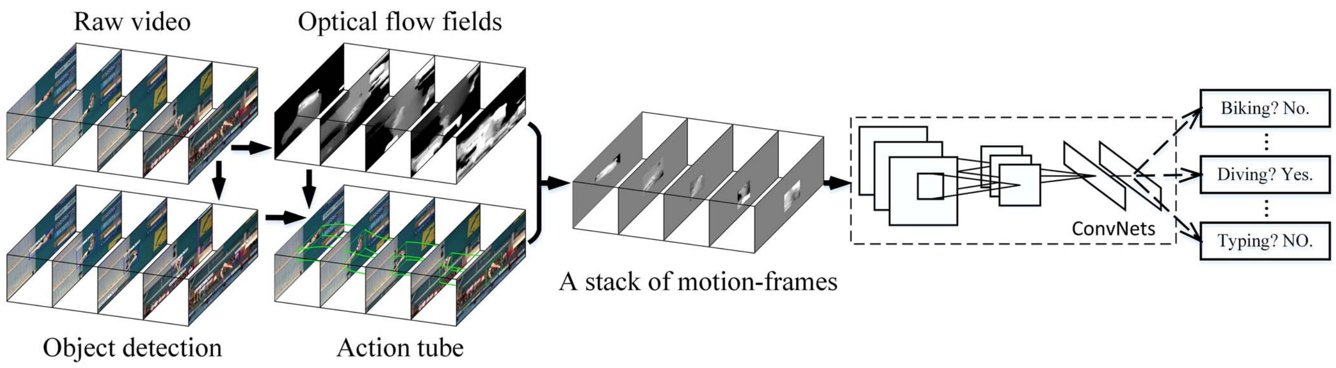

Fig. 3. Illustration of Tube ConvNets. Given a raw video, first Faster R-CNN detects all persons, then DBSCAN algorithm clusters them into several groups for finding key actors, finally action tubes are obtained by discarding motionless groups with low average optical flow magnitude. After suppressing the noise outside the action tubes, we feed a stack of motion-frames into ConvNets to construct Tube ConvNets.

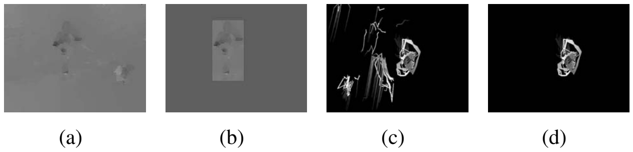
(a) (b) (c) (d)

Fig. 4. Suppress noise outside action tubes. (a), (b): an original optical flow image and the corresponding suppressed one. (c), (d): an original trajectory image and the corresponding suppressed one.

2.2. Tube ConvNets

Here, we introduce our deep ConvNets which takes action tubes as input for action recognition. The extracted action tubes can eliminate the distraction of irrelevant motions and focus on motion information produced by key actors. We achieve this by suppressing the input signals that are outside the action tubes, as illustrated in Fig. 4. In our experiments, we use two implementation schemes, one is the original optical flow fields and the other is improved dense trajectories.

Optical flow fields. We pre-compute optical flow fields for each video, rescale them to a range of $[0, 255]$, and save them as images after compressed using JPEG. We set the magnitude of signals outside the tubes as 128, which is the mean value of the rescaled range and will be 0 after mean substraction.

Improved dense trajectories. We pre-compute the improved dense trajectories (iDT) [1] and also save them as images. In such images, as shown in Fig. 4(c), white lines are drawn to connect two adjacent points along trajectories in a black background. The lines produced by linking more recent points have higher gray values, and will be 255 for the current frame. We set the gray value to 0 for those trajectories outside the tubes as the none-motion areas.

After suppressing the noise, we feed a stack of several consecutive frames of optical flow images or trajectory images, which we call motion-frames, into ConvNets to construct Tube ConvNets, as illustrated in Fig. 3. We use the VGGNet-16 [9] architecture, but with 20 input channels for motion-frames instead of 3 for RGB images.

Network training. We train our networks using Caffe [19] on Tesla K40 GPUs. Due to the absence of training samples in UCF-101, we initialize our networks with pre-trained models as following: average filters of the first layer of ImageNet models across 3 channels, and copy the average results 20 times. We use the corner cropping strategy and multi-scale cropping method introduced by [20], and use high drop out ratios of 0.9 and 0.8 for the fully connected layers, small learning rates starting with 0.005, decreasing to its $1/10$ every $10,000$ iterations, a momentum of 0.9 and a weight decay of 0.005.

3. EXPERIMENTS

In this section, we report the experimental details of our method on UCF-101 and UCF-Sports datasets, and give quantitative results and comparisons with other competitive methods.

3.1. Recognition

The UCF-101 dataset [21] consists of $13,320$ videos belonging to 101 categories that are separated into 5 broad groups: Human-Object interaction, Body-Motion, Human-Human interaction, Playing Instruments and Sports.

On UCF-101 dataset, we conduct experiments to verify the effectiveness of proposed Tube ConvNets. We use optical flow images and trajectory images separately as input for Tube ConvNets and traditional full-frame ConvNets, keeping other conditions consistent, such as GPU configuration,

Table 1. Comparison results between Tube ConvNets and Full-frame ConvNets

	Optical flow fields				Improved dense trajectories			
Architecture	split1	split2	split3	average	split1	split2	split3	average
Full-frame ConvNets	85.7%	88.2%	87.4%	87.0%	61.2%	65.4%	63.9%	63.5%
Tube ConvNets	**88.1%**	**89.4%**	**89.0%**	**88.8%**	**64.4%**	**67.9%**	**66.7%**	**66.3%**

Table 2. Performance comparison on UCF-101 dataset

Method	Year	Accuracy
iDT+FV [1]	2013	85.9%
iDT+HSV [7]	2014	87.9%
MIFS+FV [23]	2015	89.1%
DeepNet [11]	2014	63.3%
Two-stream [2]	2014	88.0%
Two-stream+LSTM [12]	2015	88.6%
Very deep two-stream [20]	2015	91.4%
Tube ConvNets	2016	**92.3%**

training strategies. In our experiments, we extract optical flow fields with $TV\text{-}L^1$ algorithm [22] and use the code released by [1] to get improved dense trajectories. We stack the horizontal and vertical optical flow images of 10 consecutive frames to form a total of 20 input channels; while for trajectory images, we stack 20 consecutive frames for input.

Table 1 shows the performance comparisons. As can be seen, it consistently performs better with action tubes, and the best performance is obtained by optical flow fields with Tube ConvNets. Besides, taking optical flow images as input outperforms using trajectory images. We speculate that Tube ConvNets learn more discriminative motion features than full-frame ConvNets.

Next, we compare our Tube ConvNets with several recent methods. For fair comparison with very deep two-stream ConvNets [20] and the original two-stream ConvNets [2], we take the spatial stream models trained by [20] to make class score fusion with our Tube ConvNets. We follow their testing scheme for action recognition. For a testing video, we sample 25 frames of RGB images for spatial stream ConvNets, and 25 action tubes for Tube ConvNets. The final prediction score is obtained by averaging across the samples. We fuse the spatial and Tube ConvNets by a weighted linear combination, where the weight is set as 2 for Tube ConvNets and 1 for spatial ConvNets.

As Table 2 shows, our recognition accuracy outperforms traditional hand-crafted feature based methods and other ConvNets, due to the good performance of our Tube ConvNets.

3.2. Localization

The UCF-Sports dataset [24] contains 150 videos from 10 action classes, such as diving, horse riding, and skating. The videos are taken from real sports broadcasts. Bounding boxes around the person performing the action of interest in each frame are available for localization measurement.

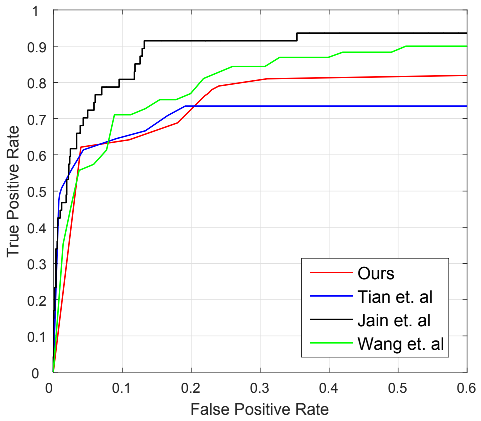

Fig. 5. ROC curves on UCF-Sports dataset for an intersection-over-union threshold of $\sigma = 0.2$. Red shows our proposed method, which achieves a competitive results compared with other approaches.

We apply our proposed tube generation method for the localization task on UCF-Sports dataset, using the same split as other methods [25]. The localization is deemed as correct if the intersection-over-union between detection region and the ground truth is greater than a certain threshold σ. In Fig. 5, we plot the ROC curve for $\sigma = 0.2$, and compare with results from several recent approaches, Jain et al. [26], Wang et al. [27], and Tian et al. [28]. As can be seen, our method achieves a competitive detection result on UCF-Sports dataset.

4. CONCLUSIONS

In this paper, we propose to feed action tubes instead of full frames into deep ConvNets, and obtain a framework called Tube ConvNets for action recognition. Meanwhile, we introduce a simple but effective method to generate action tubes from a given video, which can extract regions of interest (ROI) where key motions occur. We validate the effectiveness of Tube ConvNets for action classification on UCF-101 dataset, and illustrate its potential for action localization on UCF-Sports dataset. In the future, we plan to design improved Tube ConvNets which need no pre-generated tubes but automatically focus on ROI somehow like Faster R-CNN.

5. REFERENCES

[1] Heng Wang and Cordelia Schmid, "Action recognition with improved trajectories," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3551–3558.

[2] Karen Simonyan and Andrew Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

[3] Ivan Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[4] Paul Scovanner, Saad Ali, and Mubarak Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *Proceedings of the 15th international conference on Multimedia*. ACM, 2007, pp. 357–360.

[5] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *B-MVC 2008-19th British Machine Vision Conference*. British Machine Vision Association, 2008, pp. 275–1.

[6] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu, "Action recognition by dense trajectories," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 3169–3176.

[7] Xiaojiang Peng, Limin Wang, Xingxing Wang, and Yu Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice," *arXiv preprint arXiv:1405.4506*, 2014.

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[9] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[10] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu, "3d convolutional neural networks for human action recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 1, pp. 221–231, 2013.

[11] Andrej Karpathy, George Toderici, Sachin Shetty, Tommy Leung, Rahul Sukthankar, and Li Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 1725–1732.

[12] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici, "Beyond short snippets: Deep networks for video classification," *arXiv preprint arXiv:1503.08909*, 2015.

[13] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jagannath Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 580–587.

[14] Ross Girshick, "Fast r-cnn," *arXiv preprint arXiv:1504.08083*, 2015.

[15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99.

[16] Li Wang and Dong-Chen He, "Texture classification using texture spectrum," *Pattern Recognition*, vol. 23, no. 8, pp. 905–910, 1990.

[17] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei X-u, "A density-based algorithm for discovering clusters in large spatial databases with noise.," in *Kdd*, 1996, vol. 96, pp. 226–231.

[18] Robert T Collins, "Mean-shift blob tracking through scale s-pace," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*. IEEE, 2003, vol. 2, pp. II–234.

[19] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678.

[20] Limin Wang, Yuanjun Xiong, Zhe Wang, and Yu Qiao, "Towards good practices for very deep two-stream convnets," *arX-iv preprint arXiv:1507.02159*, 2015.

[21] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[22] Christopher Zach, Thomas Pock, and Horst Bischof, "A duality based approach for realtime tv-l 1 optical flow," in *Pattern Recognition*, pp. 214–223. Springer, 2007.

[23] Zhengzhong Lan, Ming Lin, Xuanchong Li, Alex G Hauptmann, and Bhiksha Raj, "Beyond gaussian pyramid: Multi-skip feature stacking for action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 204–212.

[24] Mikel D Rodriguez, Javed Ahmed, and Mubarak Shah, "Action mach a spatio-temporal maximum average correlation height filter for action recognition," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[25] Tian Lan, Yang Wang, and Greg Mori, "Discriminative figure-centric models for joint action localization and recognition," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2003–2010.

[26] Manan Jain, Jan Van Gemert, Hervé Jégou, Patrick Bouthemy, and Cees GM Snoek, "Action localization with tubelets from motion," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 740–747.

[27] Limin Wang, Yu Qiao, and Xiaoou Tang, "Video action detection with relational dynamic-poselets," in *Computer Vision–ECCV 2014*, pp. 565–580. Springer, 2014.

[28] Yicong Tian, Rahul Sukthankar, and Mubarak Shah, "Spatiotemporal deformable part models for action detection," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 2642–2649.