Region Trajectories for Video Semantic Concept Detection

Yuancheng Ye[1], Xuejian Rong[2], Xiaodong Yang[3], and Yingli Tian[1,2]
[1]The Graduate Center, City University of New York
yye@gradcenter.cuny.edu

[2]The City College, City University of New York
{xrong, ytian}@ccny.cuny.edu

[3]NVIDIA Research
xiaodongy@nvidia.com

ABSTRACT

Recently, with the advent of the convolutional neural network (CNN), many CNN-based object detection algorithms have been proposed and achieved encouraging results. In this paper, we introduce an algorithm based on region trajectories to establish the connections between object localizations in individual frames and video sequences. To detect object regions in the individual frames of a video, we enhance the region-based convolutional neural network (R-CNN), by incorporating EdgeBox with the Selective Search to generate candidate region proposals and combining the GoogLeNet with the AlexNet to improve the discriminability of the feature representations. The DeepMatching algorithm is employed in our proposed region trajectory method to track the points in the detected object regions. The experiments are conducted on the validation split of the TRECVID 2015 Localization dataset. As demonstrated by the experimental results, our proposed approach improves the object detection accuracy in both temporal and spatial measurements.

Keywords

Object detection, convolutional neural network, region trajectory algorithm

1. INTRODUCTION

Nowadays, with the success of deep learning technologies, many computer vision areas including object detection have ushered a new era which has witnessed the significant improvements compared with previous methods [18, 17]. Inspired by region-based convolutional neural network (R-CNN) [5], many other approaches have been proposed to exploit the advantages of deep learning methods for object detection and achieved a great success. However, most of these approaches only focus on object detection at the image level, and few methods have been proposed to detect objects at the video level. In this paper, we propose a novel region trajectory algorithm to connect the object regions, localized by the imaged-based object detection algorithm, from the individual frames to generate object trajectories in videos.

ICMR'16, June 06 - 09, 2016, New York, NY, USA

ISBN 978-1-4503-4359-6/16/06…$15.00

DOI: http://dx.doi.org/10.1145/2911996.2912046

1.1 Related Work

Since convolutional neural network (CNN) achieved the impressive image classification accuracy on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 [10, 8, 12], many approaches have tried to incorporate CNN to tackle the problem of object detection at the image level. Region-based convolutional neural network (R-CNN) is one of the first successful attempts, which utilizes one of the state-of-the-art region proposal algorithms and a simple technique (affine image warping) to resize each candidate object region to the uniform size required by the CNN model [5, 3]. Then discriminative features are extracted from the CNN model and fed into a linear SVM. Despite its straightforward structure, R-CNN enjoys a great success compared with the traditional object detection approaches [13, 1, 4, 2], which rely on the sliding windows and hand-crafted features.

Although the image-based object detection has attracted more and more interests, few methods have been proposed to tackle the object detection in videos [9, 21, 24, 11, 15]. Inspired by [6], our proposed method first detects the object regions in individual frames and then try to connect them in the temporal domain. However, the linking algorithm adopted by [6] makes an assumption that an action keeps being presented from the first frame to the end, which is usually not the case for object detection due to occlusion and viewpoint change. The approach developed in the paper [25, 26] is similar to our method, which keeps tracking the action regions localized by the image-based object detection algorithm. However, the tracking mechanism in [25] depends on the neighboring windows around the detected regions, and cannot handle the situation of large displacements of object motion which is common in wild videos.

1.2 Our Proposed System

To bridge the gap between image-based and video-based object detection, a novel region trajectory algorithm is presented in this paper. Moreover, two modifications are made to the standard R-CNN method to improve the performance of object detection at the image level. One is to combine the EdgeBox [28] with the Selective Search [22] to obtain a

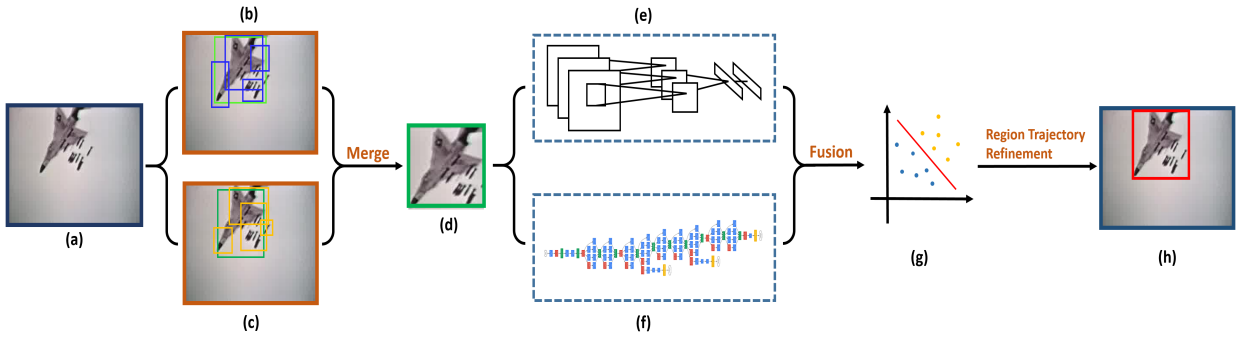

Figure 1: Overview of the Proposed System Structure. (a) Input image. (b) Region proposals by Selective Search. (c) Region proposals by EdgeBox. (d) Region proposals fused by (b) and (c). (e) Extracting features of region proposals by AlexNet. (f) Extracting features of region proposals by GoogLeNet. (g) Applying linear SVM classifier to the vector concatenating by the ℓ_2 normalization of (e) and (f). (h) Final region box obtained by the region trajectory algorithm.

higher coverage rate for the generated region proposals. Another is to incorporate the GoogLeNet [20] with the AlexNet [10], which is the only CNN model employed in the R-CNN, to produce a more discriminative feature representation of each region proposal.

To track the object locations detected by the image-based object detection method, the state-of-the-art point-matching algorithm, DeepMatching [27], is employed. After performing the tracking algorithm on the candidate object regions in different frames, several object region trajectories are produced. If the tracking process is based on a falsely detected object region, the obtained corresponding object region trajectory may also be incorrect. To prune these false trajectories, two thresholds are applied: the average SVM score of the regions in the trajectory and the ratio of the number of regions detected by R-CNN and the total number of the regions in the trajectory. The structure of our system is illustrated in Figure 1.

The rest of the paper is organized as follows. Section 2 presents the enhanced R-CNN method employed in our system. Section 3 introduces the details of our proposed region trajectory algorithm. In Section 4, experimental results on the validation split of TRECVID 2015 Localization dataset are presented, and an ablation study of the parameters in our region trajectory algorithm is also provided. Finally, our paper is concluded in Section 5.

2. ENHANCED R-CNN

R-CNN is one of the first successful algorithms that apply deep learning techniques to the object detection task. It decomposes the detection task into three aspects: 1) generating sufficient category-agnostic region proposals to cover as many objects as possible in the image; 2) applying a fine tuned convolutional neural network to extract a discriminative feature representation for each proposed region; 3) training a set of category-specific linear SVMs based on the extracted feature representation to classify those region proposals. In our implementation, two modifications are made when compared with the method described in [5] to further improve the object detection accuracy at the image level.

Selective Search [22] and EdgeBox [28] are combined in our implementation to generate region proposals [7], which can improve the coverage rate of the objects in the image. Another modification is that, in addition to the AlexNet [10], the GoogLeNet [20] is also employed in our method. By concatenating the ℓ_2 normalization of the features extracted from the last fully connected layers of these two models, a more discriminative feature representation can be generated. After applying the generated feature to the trained class-specific SVMs, the scores of the region for each class can be obtained. Given all scored regions in an image, a greedy non-maximum suppression is independently applied for each class to reject the regions which have a large overlap with the region with higher SVM score.

3. REGION TRAJECTORIES FOR OBJECT DETECTION

To extend the image-based object detection algorithm into the video domain, one important step is how to establish the connections among the object regions in different frames. In this paper, we propose the region trajectory algorithm based on the state-of-the-art point-matching algorithm, DeepMatching, to establish the object trajectories for the video-based object detection. Our region trajectory algorithm is illustrated in Figure 2.

Inspired by the dense trajectory method for action recognition [23], we propose the region trajectory algorithm to establish the temporal connections of object regions in different frames. For a set of frames $\{f_1, \ldots, f_i, \ldots, f_n\}$ in a video, suppose that R-CNN detects an object region b_i in the frame f_i, then an object trajectory, T_i, can be generated based this detected region b_i. The tracking procedure is performed both forwardly and backwardly to determine the start and end regions for the object trajectory T_i. In the following discussions, we only take the forward tracking procedure as the example to illustrate how our region trajectory algorithm works.

Firstly, a set of points $\{p_k\}$ in the region b_i can be computed by applying the DeepMatching algorithm, and a set of matched points, $\{p'_k\}$, is also generated in the frame f_{i+1}. Based on these two sets of matched points, $\{p_k\}$ and $\{p'_k\}$, an affine model can be acquired , which consists of a linear

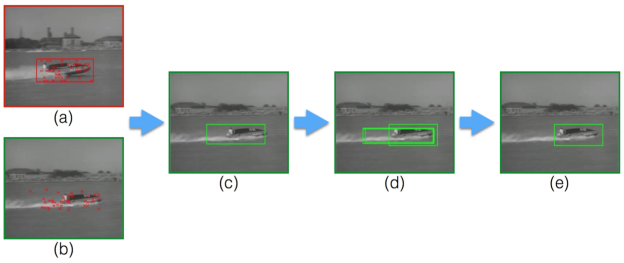

Figure 2: The illustration of the proposed region trajectory algorithm. (a) The first frame with the detected object region by R-CNN. The star marks are the matched points inside the localized object bounding box. (b) The second frame in which the R-CNN fails to locate the object region. (c) The second frame with the object region predicted by the affine model. (d) The bounding boxes with IOU values larger than 0.5 with the bounding box singled out in (c). (e) The bounding box with the highest SVM score from the set of boxes noted in (d).

transformation matrix, $A \in \mathbb{R}^{2\times 2}$, and a translation vector, $\vec{b} \in \mathbb{R}^{2\times 1}$. Then by applying the coordinates $(x_i^j, y_i^j)^T$ of each corner of the object region, b_i, into the affine model:

$$(x_{i+1}^j, y_{i+1}^j)^T = A(x_i^j, y_i^j)^T + \vec{b}, \quad (1)$$

the corresponding points of the four corners of b_i in the next frame can be obtained. However, these four locations, $\{(x_{i+1}^j, y_{i+1}^j)\}_{j=0,1,2,3}$, may be distorted or fall out of the image. To obtain the reasonable predicted object region, b'_{i+1}, in the frame f_{i+1}, a correction procedure should be applied, which is defined as:

$$\alpha_{i+1} = \max(0, \min_j(\{x_{i+1}^j\})), \quad (2)$$

$$\beta_{i+1} = \max(0, \min_j(\{y_{i+1}^j\})), \quad (3)$$

$$\mu_{i+1} = \min(width, \max_j(\{x_{i+1}^j\})), \quad (4)$$

$$\nu_{i+1} = \min(height, \max_j(\{y_{i+1}^j\})), \quad (5)$$

where $(\alpha_{i+1}, \beta_{i+1})$ and (μ_{i+1}, ν_{i+1}) stand for the coordinates of the upper-left corner and bottom-right corner of the predicted region respectively. *width* is the horizontal size of the image, and *height* is vertical size of the image.

Furthermore, to improve the spatial accuracy and reduce repeating tracking process, the predicted region, b'_{i+1}, is replaced by the object region, b_{i+1}, which is detected by the R-CNN algorithm in the frame f_{i+1}, if the value of intersection-over-union (IOU) is larger than a threshold (0.7 in our implementation).

If the region, b'_{i+1}, predicted by the affine model is not replaced by an object region detected by R-CNN, a set of bounding boxes, generated by Selective Search [22] and EdgeBox [28] algorithms, are selected if their IOU values with b'_{i+1} are larger than 0.5. Then the bounding box, b''_{i+1}, with the highest SVM score is chosen from the set of bounding boxes plus the region b'_{i+1} as the predicted object region. The SVM score of b''_{i+1} should then be compared with a threshold to determine whether this region is a reasonable object bounding box. Specifically, if the SVM score is smaller than a threshold, this region should not be assigned to the object trajectory T_i, and the tracking process should also be terminated on this direction.

Run	iframe_fscore	mean_pixel_fscore
1	0.7447	0.4723
2	0.7682	0.4542
3	0.7309	0.5085
4	0.7661	0.4591
MediaMill*	0.7662	0.6557
PicSOM*	0.6643	0.3944
TokyoTech*	0.6699	0.6688
Trimps*	0.7357	0.4760

Table 1: The results of Mean_Per_Run for our four submitted runs in the TRECVID competition. $*$ indicates the best results of other teams among all of their submitted runs.

After performing the tracking process in both forward and backward directions based on the algorithm described above, the object trajectory T_i can be obtained. However, if the object region, *e.g.* , b_i, is falsely detected by the R-CNN, the trajectory T_i which is generated based on b_i may also be incorrect. To prune these falsely produced object trajectories, two kinds of thresholds are applied: the average SVM score of the regions contained in the object trajectory and the ratio of the number of regions detected by R-CNN and the total number of the regions in the trajectory.

We observe that only using the average SVM score of the trajectory as the threshold cannot improve the performance. The reason is that some falsely detected regions have high SVM scores, which makes the average score of false trajectories higher than the average score of correct object trajectories. On the other hand, the value of the ratio of the number of regions detected by R-CNN and the total number of the regions in the trajectory can reflect how many regions in the trajectory are localized by the R-CNN algorithm and how many are inferred by the affine model, which implies that the higher ratio indicates a higher probability that the trajectory is correct.

The score function described in [6] is also employed in our approach to prune the plausible object trajectories. It measures the correctness of a trajectory based on both SVM scores and the IOU values of the regions. If the computed trajectory score, S_T, is below a threshold, the trajectory T will be deleted. Otherwise, this object trajectory can be considered as a reasonable one.

4. EXPERIMENTAL RESULTS AND DISCUSSION

Our proposed video-based object detection system is evaluated on the TRECVID 2015 Localization dataset [16], which is comprised of ten concepts: airplane(1003), anchorperson(1005), boat_ship(1015), bridges(1017), bus(1019), computers(1031), motorcycle(1080), telephones(1117), flags(1261), and quadruped(1392). The numbers in the parentheses represent the code names of the corresponding concepts. There are four other teams submitted final results for this task: MediaMill (University of Amsterdam Qualcomm), PicSOM (Aalto University and University of Helsinki), TokyoTech (Tokyo Institute of Technology), Trimps (Third Research In-

Params	Metrics	1003	1005	1015	1017	1019	1031	1080	1117	1261	1392	Mean
pos_1.0_neg_0.0	iframe_fscore	0.7848	0.9991	0.6779	0.5788	0.6687	0.9051	0.6788	0.6843	0.8396	0.7775	0.7595
	mean_pixel_fscore	0.6671	0.9567	0.4692	0.4433	0.5643	0.8168	0.4698	0.5940	0.7331	0.7054	0.6420
pos_0.7_neg_0.1	iframe_fscore	0.8433	0.9991	0.7404	0.6349	0.6667	1.0000	0.6519	0.7634	0.8873	0.8184	0.8005
	mean_pixel_fscore	0.6203	0.9567	0.3980	0.4349	0.4815	0.8136	0.4054	0.4690	0.6419	0.6281	0.5849
fusion	iframe_fscore	0.8433	0.9991	0.7406	0.6298	0.6667	1.0000	0.6543	0.7634	0.8892	0.8188	0.8005
	mean_pixel_fscore	0.6292	0.9567	0.4220	0.4746	0.4816	0.8699	0.4202	0.4759	0.6603	0.6435	0.6034

Table 2: The results of the three sets of parameters for each concept detected by R-CNN.

Params	Methods	Metrics	Mean
pos_1.0_neg_0.0	without traj	iframe_fscore	0.7595
		mean_pixel_fscore	0.6420
	with traj	iframe_fscore	0.7926
		mean_pixel_fscore	0.6143
fusion	without traj	iframe_fscore	0.8005
		mean_pixel_fscore	0.6034
	with traj	iframe_fscore	0.8823
		mean_pixel_fscore	0.6348

Table 3: The comparison of the overall performance of the video-based object detection system with and without region trajectory algorithm.

System	Params	Metrics	Mean
TRECVID	pos_1.0_neg_0.0	iframe_fscore	0.7724
		mean_pixel_fscore	0.5926
	fusion	iframe_fscore	0.8110
		mean_pixel_fscore	0.5867
RegionTraj	pos_1.0_neg_0.0	iframe_fscore	0.7926
		mean_pixel_fscore	0.6143
	fusion	iframe_fscore	0.8823
		mean_pixel_fscore	0.6348

Table 4: The results of the two systems can achieve on the two sets of parameters with their own tracking algorithms for temporal and spatial measurements respectively.

stitute of the Ministry of Public Security, P.R. China). The overall results of the TRECVID 2015 Localization competition are shown in Table 2. Our team submitted four runs for this task, and the overall results of the TRECVID 2015 Localization competition are showed in Table 2. These results are obtained by another video-based object detection system, which will be compared with the one described in this paper in detail.

Since the ground truth of the testing data is not available, we report the results of the video-based object detection system described in this paper on the validation split of the training data.

The measurements of the results are in two folds. The first is the temporal accuracy, *iframe_fscore*, and the second is the spatial accuracy, *mean_pixel_fscore*. *fscore* is calculated by the formula:

$$fscore = \frac{2 \times Precision \times Recall}{Precision + Recall}, \tag{6}$$

where *Precision* is the ratio of the number of true positives detected by the system and the total number of positives in the test dataset, while *Recall* stands for the ratio of the number of true positives and the total number of positives both detected by the system.

The CNN models of AlexNet and GoogLeNet are both pre-trained on the PASCAL VOC 2012 detection dataset as well as the ImageNet dataset. In practice, we apply two sets of parameters to train the SVM. The first, denoted as pos_1.0_neg_0.0, only employs the ground truth as the positive training data and uses the regions which have no overlap with the ground truth as the negative data. The second set, denoted as pos_0.7_neg_0.1, employs not only the ground truth regions as the positive training data, but also the regions which have the IOU values larger than 0.7 with the ground truth. For the negative training data, it uses the regions with the IOU values smaller than 0.1 and larger than 0 with the ground truth. The results of these two sets of parameters are also fused to form a third set of results, which is denoted as fusion. The evaluation of these three sets of parameters is provided in Table 2.

After obtaining the object regions in individual frames, the object trajectories can then be generated by applying the region trajectory algorithm. The comparison of the results with and without region trajectory algorithm is provided in the Table 3.

The differences between the system submitted in the task of Localization of TRECVID 2015 and the one described in this paper are mainly in two aspects. Firstly, the Kanade-Lucas-Tomasi Feature Tracker (KLT) [14, 19] is employed as the tracking algorithm in previous system, while the DeepMatching algorithm is utilized in the second one. Secondly, in the previous system the object regions predicted by the affine model are not replaced by the object regions detected by R-CNN, even if they have large IOU values. The comparison of these two systems is presented in Table 4.

5. CONCLUSION

In this paper, we have presented a video-based object detection algorithm, which is based on the image-based R-CNN method and the state-of-the-art point-matching algorithm, DeepMatching. Specifically, a novel region trajectory algorithm is proposed to generate object trajectories in the videos based on the object regions detected in individual frames. Then these object trajectories are pruned according the confidence scores associated with them. The experiments demonstrate that our proposed video-based object detection algorithm can robustly and effectively locate the object regions in the video both in the temporal and spatial domain.

6. ACKNOWLEDGEMENT

This work was supported in part by NSF grants EFRI-1137172, IIP-1343402, and IIS-1400802.

7. REFERENCES

[1] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. IEEE, 2005.

[3] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision (IJCV)*, 88(2):303–338, 2010.

[4] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence (PAMI)*, 32(9):1627–1645, 2010.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 580–587. IEEE, 2014.

[6] G. Gkioxari and J. Malik. Finding action tubes. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015.

[7] J. Hosang, R. Benenson, P. Dollár, and B. Schiele. What makes for effective detection proposals? *arXiv preprint arXiv:1502.05082*, 2015.

[8] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[9] A. Klaser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *British Machine Vision Conference (BMVC)*, pages 275–1. British Machine Vision Association, 2008.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[11] I. Laptev and P. Pérez. Retrieving actions in movies. In *IEEE Intenational Conference on Computer Vision (ICCV)*, pages 1–8. IEEE, 2007.

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[13] D. G. Lowe. Object recognition from local scale-invariant features. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 1150–1157, 1999.

[14] B. D. Lucas, T. Kanade, et al. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence (IJCAI)*, volume 81, pages 674–679, 1981.

[15] D. Oneata, J. Verbeek, and C. Schmid. Efficient action localization with approximately normalized fisher vectors. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2545–2552. IEEE, 2014.

[16] P. Over, G. Awad, M. Michel, J. Fiscus, W. Kraaij, A. F. Smeaton, G. QuÃl'enot, and R. Ordelman. Trecvid 2015 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2015*. NIST, USA, 2015.

[17] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 91–99, 2015.

[18] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *International Conference on Learning Representations (ICLR)*, 2013.

[19] J. Shi and C. Tomasi. Good features to track. In *Computer Vision and Pattern Recognition (CVPR)*, pages 593–600. IEEE, 1994.

[20] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[21] Y. Tian, R. Sukthankar, and M. Shah. Spatiotemporal deformable part models for action detection. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2642–2649. IEEE, 2013.

[22] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *International Journal of Computer Vision (IJCV)*, 104(2):154–171, 2013.

[23] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Action Recognition by Dense Trajectories. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3169–3176, Colorado Springs, United States, June 2011.

[24] L. Wang, Y. Qiao, and X. Tang. Video action detection with relational dynamic-poselets. In *European Conference on Computer Vision (ECCV)*, pages 565–580. Springer, 2014.

[25] P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Learning to track for spatio-temporal action localization. June 2015.

[26] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *IEEE Intenational Conference on Computer Vision (ICCV)*, Sydney, Australia, Dec. 2013.

[27] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *IEEE Intenational Conference on Computer Vision (ICCV)*, 2013.

[28] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *Computer Vision–ECCV 2014*, pages 391–405. Springer, 2014.