Bottom-up Segmentation for Top-down Detection

Sanja Fidler[1] Roozbeh Mottaghi[2] Alan Yuille[2] Raquel Urtasun[1]
[1]TTI Chicago, [2]UCLA
{fidler, rurtasun}@ttic.edu, {roozbehm@cs, yuille@stat}.ucla.edu

Abstract

In this paper we are interested in how semantic segmentation can help object detection. We propose a novel deformable part-based model which exploits region-based segmentation algorithms that compute candidate object regions by bottom-up clustering followed by ranking of those regions. Our approach allows every detection hypothesis to select a segment (including void), and scores each box in the image using both the traditional HOG filters as well as a set of novel segmentation features. Thus our model "blends" between the detector and segmentation models. Since our features can be computed very efficiently given the segments, we maintain the same complexity as the original DPM [5]. We demonstrate the effectiveness of our approach in PASCAL VOC 2010, and show that we outperform the original DPM [5] in 19 out of 20 classes, achieving an improvement of 8% AP. Furthermore, we outperform the previous state-of-the-art on VOC'10 test by 4%.

1. Introduction

In this paper we are interested in exploiting semantic segmentation in order to improve object detection. While bottom-up segmentation has often been believed to be inferior to top-down object detectors due to its frequent over- and under- segmentation, recent approaches [3, 1] have shown impressive results on difficult datasets such as the PASCAL VOC challenge. Here, we take advantage of region-based segmentation approaches [2], which compute a set of candidate object regions by bottom-up clustering, and produce a segmentation by ranking those regions using class specific rankers. Our goal is to make use of these candidate object segments to bias sliding window object detectors to agree with these regions.

Deformable part-based models (DPM) [5] are arguably the leading technique to object detection. However, so far, there has not been many attempts to incorporate segmentation into DPMs. In this paper we propose a novel DPM model, which exploits region-based segmentation by allowing every detection hypothesis to select a segment (including void) from a small pool of segment candidates. We derive simple features, which can capture the essential information encoded in the segments. Our detector scores each box in the image using both, HOG filters as well as the set of novel segmentation features. Our model "blends" between the detector and the segmentation models by boosting object hypotheses that overlap with the segments. Furthermore, it can recover from segmentation mistakes by exploiting a powerful appearance model.

We demonstrate the effectiveness of our approach in PASCAL VOC 2010, and show that we outperform the original DPM [5] by 8%. Furthermore, we outperform the previous state-of-the-art on VOC2010 by 4%. We believe that these results will encourage new research on bottom-up segmentation as well as hybrid segmentation-detection approaches, as our paper clearly demonstrates the importance of segmentation for object detection. Details of our method can be found in [6].

2. A Segmentation-Aware DPM (segDPM)

Our approach takes advantage of region-based segmentation approaches, which compute object regions by bottom-up clustering, and rank those regions to estimate a score for each class. We frame detection as an inference problem, where each detection hypothesis can select a segment from a pool of candidates (as well as void).

Following [5], let p_0 be a random variable encoding the location and scale of a bounding box in a HOG pyramid, and $\{p_i\}_{i=1,\cdots,P}$ be a set of parts. Denote with h the index over the set of candidate segments returned by the segmentation algorithm. We frame the detection problem as inference in a Markov Random Field (MRF), where each root filter hypothesis can select a segment from a pool of candidates. We write the score of a configuration as

$$E(\mathbf{p},h) = \sum_{i=0}^{P} w_i^T \cdot \phi(x,p_i) + \sum_{i=1}^{P} w_{i,def}^T \cdot \phi(x,p_0,p_i) + \\ + w_{seg}^T \phi(x,h,p_0) \quad (1)$$

Note that $h=0$ implies that no segment is selected.

We derive simple features which encourage the selected segment to agree with the object detection hypothesis. Most of our features employ integral images which makes them extremely efficient, as this computation can be done in constant time. We briefly summarize the features, and refer the reader to Fig 1 for visualization.

	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	Avg.
VOC 2010 val, no post-processing																					
CPMC [2]	53.3	19.5	22.8	15.7	8.1	42.7	22.1	**51.3**	4.3	18.9	10.5	28.1	30.5	38.3	20.9	6.0	19.2	18.6	35.4	21.1	24.4
DPM [5]	46.3	49.5	4.8	6.4	22.6	53.5	38.7	24.8	14.2	10.5	10.9	12.9	36.4	38.7	**42.6**	3.6	26.9	22.7	34.2	31.2	26.6
segDPM	**55.7**	**50**	**23.3**	**16.0**	**28.5**	**57.4**	**43.2**	49.3	**14.3**	**23.5**	**17.7**	**32.4**	**42.6**	**44.9**	42.1	**11.9**	**32.5**	**25.5**	**43.9**	**39.7**	**34.7**
VOC 2010 test																					
segDPM	**61.4**	53.4	**25.6**	**25.2**	**35.5**	51.7	**50.6**	**50.8**	19.3	**33.8**	26.8	**40.4**	48.3	54.4	47.1	**14.8**	**38.7**	**35.0**	**52.8**	**43.1**	**40.4**
NLPR_HOGLBP [9]	53.3	**55.3**	19.2	21.0	30.0	54.4	46.7	41.2	**20.0**	31.5	20.7	30.3	48.6	55.3	46.5	10.2	34.4	26.5	50.3	40.3	36.8
MITUCLA_HIERARCHY [10]	54.2	48.5	15.7	19.2	29.2	**55.5**	43.5	41.7	16.9	28.5	26.7	30.9	48.3	55.0	41.7	9.7	35.8	30.8	47.2	40.8	36.0
NUS_HOGLBP_CTX [4]	49.1	52.4	17.8	12.0	30.6	53.5	32.8	37.3	17.7	30.6	**27.7**	29.5	**51.9**	**56.3**	44.2	9.6	14.8	27.9	49.5	38.4	34.2
UOCTTI_LSVM_MDPM [8]	52.4	54.3	13.0	15.6	35.1	54.2	49.1	31.8	15.5	26.2	13.5	21.5	45.4	51.6	**47.5**	9.1	35.1	19.4	46.6	38.0	33.7

Table 1. AP performance (in %) on **VOC 2010 val** (top Table), and performance **VOC 2010 test** (bottom Table).

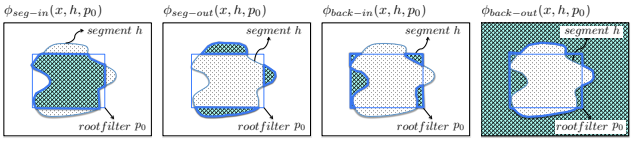


Figure 1. The first two features encourage the box to contain as many segment pixels as possible. This pair alone could result in boxes that "overshoot" the segment. The second pair tries to minimize the number of background pixels inside the box and maximize its number outside. In synchrony, these features try to tightly place a box around the segment.

Segment-In: Given a segment $S(h)$, our first feature counts the percentage of pixels in $S(h)$ that fall inside the bounding box defined by p_0. This feature encourages the bounding box to contain the segment.

Segment-Out: This feature counts the percentage of segment pixels that are outside the bounding box. This feature discourages boxes that do not contain all segment pixels.

Background-In: This feature counts the amount of background inside the bounding box. It captures the statistics of how often the segments leak outside the true bounding box vs how often they are too small.

Background-Out: This feature counts the amount of background outside the bounding box. It tries to discourage boxes that are too big and do not tightly fit the segments.

Overlap: This feature penalizes bounding boxes which do not overlap well with the segment. In particular, it computes IOU between the candidate bounding box p_0 and a bounding box around a segment.

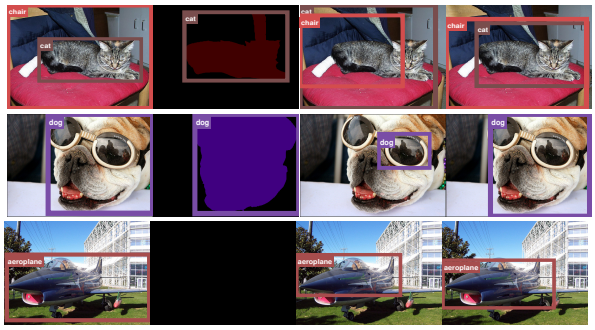
Background bias: The value of all of the above features is 0 when $h = 0$. We incorporate an additional feature to learn the bias for the background segment ($h = 0$). This puts the scores of the HOG filters and the segmentation potentials into a common referential.

Learning and Inference We learn a different weight for each feature using a latent structured-SVM [7]. Inference is carried out via dynamic programming [5].

Segments We use the final segmentation output of CPMC [2] to get the candidate segments.

3. Experimental Evaluation

We evaluate our approach on PASCAL VOC 2010 `val` detection dataset in Table 1 (top). We train all methods, including the baselines on `train`. We use the standard 50%



(a) GT (b) CPMC (c) DPM (d) segDPM

Figure 2. For each method, we show top k detections for each class, where k is the number of boxes for that class in GT.

IOU overlap criterion for detection and report average precision (AP). Our model outperforms the CPMC baseline by 10% and achieves a significant boost of 8% AP over DPM, which is a well established and difficult baseline to beat.

We evaluate our approach on VOC 2010 `test` in Table 1 (bottom). We outperform the competitors by 3.6%, and achieve the best result in 13 out of 20 classes.

References

[1] P. Arbelaez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik. Finding animals: Semantic segmentation using regions and parts. In *CVPR*, 2012. 1

[2] J. Carreira, R. Caseiroa, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In *ECCV*, 2012. 1, 2

[3] J. Carreira, F. Li, and C. Sminchisescu. Object Recognition by Sequential Figure-Ground Ranking. *IJCV*, 2011. 1

[4] Q. Chen, Z. Song, Y. Hua, Z. Huang, and S. Yan. Hierarchical matching with side information for image classification. In *CVPR'12*. 2

[5] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32(9), 2010. 1, 2

[6] S. Fidler, R. Mottaghi, A. Yuille, and R. Urtasun. Bottom-up segmentation for top-down detection. In *CVPR*, 2013. 1

[7] R. Girshick, P. Felzenszwalb, and D. McAllester. Object detection with grammar models. In *NIPS*, 2009. 2

[8] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *ICCV*, 2009. 2

[9] Y. Yu, J. Zhang, Y. Huang, S. Zheng, W. Ren, C. Wang, K. Huang, and T. Tan. Object detection by context and boosted hog-lbp. In *ECCV w. on PASCAL*, 2010. 2

[10] L. Zhu, Y. Chen, A. Yuille, and W. Freeman. Latent hierarchical structural learning for object detection. In *CVPR*, 2010. 2