Segmentation-aware Deformable Part Models

Eduard Trulls[1], Stavros Tsogkas[2,3], Iasonas Kokkinos[2,3], Alberto Sanfeliu[1], Francesc Moreno-Noguer[1]
[1]Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain
[2]Center for Visual Computing, Ecole Centrale de Paris, France [3]Galen, INRIA Saclay, France
[1]{etrulls, sanfeliu, fmoreno}@iri.upc.edu, [2]{stavros.tsogkas, iasonas.kokkinos}@ecp.fr

Abstract

In this work we propose a technique to combine bottom-up segmentation, coming in the form of SLIC superpixels, with sliding window detectors, such as Deformable Part Models (DPMs).

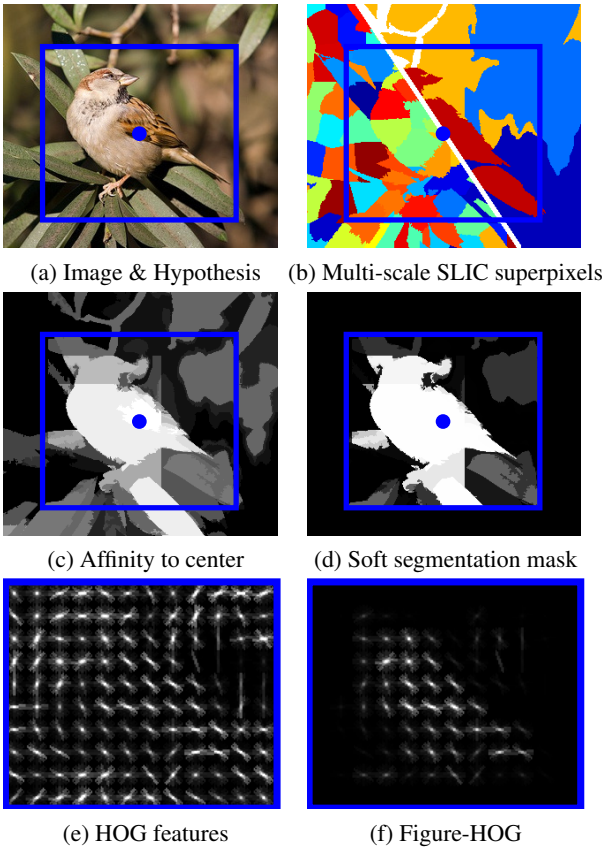
The merit of our approach lies in 'cleaning up' the low-level HOG features by exploiting the spatial support of SLIC superpixels; this can be understood as using segmentation to split the feature variation into object-specific and background changes. Rather than committing to a single segmentation we use a large pool of SLIC superpixels and combine them in a scale-, position- and object-dependent manner to build soft segmentation masks. The segmentation masks can be computed fast enough to repeat this process over every candidate window, during training and detection, for both the root and part filters of DPMs.

We use these masks to construct enhanced, background-invariant features to train DPMs. We test our approach on the PASCAL VOC 2007, outperforming the standard DPM in 17 out of 20 classes, yielding an average increase of 1.7% AP. Additionally, we demonstrate the robustness of this approach, extending it to dense SIFT descriptors for large displacement optical flow.

1. Introduction

Sliding window classifiers are the method of choice for object detection in the high-recall regime, as these ensure that no objects 'fall through the cracks' of a segmentation front-end. However, even if a putative detection window is tightly surrounding the object, background structures can creep into the low-level features extracted from the image, adversely increasing the variability of the input signals. This is typically the case for highly deformable or non-convex objects (e.g. tables, cats, or dogs), that do not naturally occupy a rectangular area, and therefore background structures often appear in the rectangular box con-

This work has been partially funded by Spanish Ministry of Economy and Competitiveness under projects PAU+ DPI2011-27510, TaskCoop DPI2010-17112, and ERA-Net Chistera project ViSen PCIN-2013-047; by the EU project ARCAS FP7-ICT-2011-28761; by grant ANR-10-JCJC-0205; and by the EU Project MOBOT FP7-ICT-2011-600796.



(a) Image & Hypothesis (b) Multi-scale SLIC superpixels
(c) Affinity to center (d) Soft segmentation mask
(e) HOG features (f) Figure-HOG

Figure 1. Method overview: We take as input a detection hypothesis (bounding box) and image (a), and a set of SLIC superpixels at different scales (b)—we show two scales. We pick the superpixels which contain the center of the bounding box and rank them by how well they fit the box, using intersection over union; in (c) we show their aggregated response. We use the highest-ranked superpixels to build a soft segmentation mask (d), which is then used to split Histograms of Gradients (HOG) features (e) into a figure-HOG channel (f) and a background channel (the remainder). These are processed by a DPM-based classifier.

taining them.

Our work proposes a simple and efficient method to exploit segmentation information for object detection. Our main technical contribution consists in improving the performance of low-level, gradient-based features such as Histograms of Gradients (HOG) [10] or SIFT [29], and is inspired by recent advances in appearance descriptors [38, 40] and sliding-window detectors [32, 15]. As shown in Fig. 1, we construct a foreground mask 'on the fly', namely for each putative window position, and use it to split the HOG features into foreground and background features. Rather than constructing a binary foreground mask through some discrete optimization procedure, as e.g. in [16], we do this in a soft manner, building on the recent work of [40] on constructing 'segmentation-aware' descriptors. Namely, as in [40] we use a segmentation 'hardness' design parameter, which we adapt per class with cross-validation.

Other than simplicity, a main advantage of our method is its computational efficiency. In particular, we exploit SLIC superpixels [1], which have a computational overhead of a fraction of a second per image. We use intersection over union to rank superpixels according to how well they match every detection hypothesis. We then use a subset of these segments to build soft masks measuring the affinity between pixels (or HOG blocks) and the putative object position; this is then used to split the image measurements into a foreground and background channel. Unlike [15, 16], our approach extends naturally to both root and part filters, while incurring a minimal additional computational cost. The segmentation process outlined in Fig. 1 is fast enough to be performed 'on the fly' for all object hypotheses in sliding window detection.

We validate our approach by applying it to the standard Deformable Part Models (DPM) paradigm [14]; keeping all other modelling parameters identical, our segmentation-based variant of HOG delivers consistent improvements in detection performance on PASCAL VOC 2007. We also apply our approach to dense SIFT matching for large-displacement optical flow; there we attain results comparable to those in our earlier work [40], but in a fraction of the processing time used therein. The code for this paper will be available in [39].

2. Prior Work on Segmentation & Recognition

In the previous decade several works extended segmentation techniques such as curve evolution [35, 41, 9, 8, 20] and graph cuts [23, 25, 24] to combine model-based information with region- and contour-based terms. However, with the exception of the rigid model of [25], these works assume that a shortlist of object 'proposals' is available beforehand, and can thus help object detection only by pruning false positives, rather than helping objects 'pop up'.

A tighter coupling of segmentation and recognition is pursued in semantic segmentation, where object-specific appearance terms influence image labelling, e.g. [37, 17, 22], without necessarily relying on the outputs of an object detection module. Even though the latest techniques [45, 6] deliver compelling results, their impact on recognition performance has only very recently been explored [15]. Finally, such techniques can be computationally demanding, involving some form of discrete optimization for segmentation, or object-tailored cascades [45], meaning a substantial overhead for multi-category detection.

Coming to using a segmentation front-end for detection, originally [36, 30, 33] used multiple image segmentations to obtain a rich set of object hypotheses in the context of learning. The current state-of-the-art techniques [42, 31] deliver a compact, yet sufficient set of proposals at a rate of multiple frames per second, thereby guiding the application of more demanding classifiers, such as bag-of-words. A more recent thread of works, relevant to the 'objectness' idea [2], is that of learning to segment in an object-independent manner [12, 7]. Still, these works can harm recall if object positions are missed by the segmentation front-end.

Turning to sliding-window variants, which are more similar in spirit to ours, Ramanan [34] applies local figure-ground segmentations post-hoc to prune false positives in pedestrian detection; this however is not taking segmentation into account when training a classifier.

In [44] a model which explicitly accounts for truncated objects in both training and detection was shown to provide increased performance in detection.

Gao et al. [16] consider forming a binary segmentation mask per bounding box hypothesis using graph-cuts; they accelerate detection using branch-and-bound, but this still takes a couple of seconds for single root filters, while it is not straightforward how to extend their method to part-based models.

In [32] the Fisher criterion is used to create a per-patch soft figure-ground segmentation, which is then summarized through a HOG descriptor. By contrast we bring superpixels into play, and also learn to detect from segmentation-sensitive HOG features.

Most recently, Fidler et al [15] combine semantic segmentation results with DPM-based detection, by constructing additional features that measure the overlap of a putative bounding box and the region assigned to an object hypothesis by semantic segmentation. This yields substantial improvements in performance, yet requires running first the semantic segmentation algorithm of [6], which requires multiple seconds per frame, on a 6-core machine, for feature extraction. Our approach yields more modest improvements, but incurs a negligible additional computational cost.

3. Method

We first briefly describe DPMs, and then turn to combining them with segmentation information coming in the form of SLIC superpixels.

3.1. DPMs for object detection

Deformable Part Models [14] represent objects as a star-shaped graphical model of parts, with the 'root' node at the center corresponding to the entire object domain and the 'leaf' nodes indicating the deformable object parts. The score for a specific arrangement of a root filter x_0 and n part filters $x_1, \ldots, x_n$ is given by:

$$S(x_0, x_1, \ldots, x_n) = \sum_{p=0}^{n} \langle w_p, G(x_p) \rangle + \sum_{p=1}^{n} D_p(x_p, x_0), \quad (1)$$

where $G(x_p)$ indicates the image-based features at position x_p, w_p is the template for part p, $\langle w_p, G(x_p) \rangle$ is the score obtained for placing part p in position x_p, and $D_p(x_p, x_0)$ is a quadratic function that measures the spatial compatibility between the positions of part p and the root.

3.2. Superpixel-grounded DPMs

Our contribution lies in modifying the local features $G(x)$ used in the first term of Eq. 1 so as to exploit segmentation information. In particular, inspired by the recent success of integrating segmentation and image descriptors [38, 40], we apply a similar approach to feature extraction for object recognition. For this, as illustrated in Fig. 1, we efficiently compute a large pool of image segments which are then combined to build segmentation masks for any putative object hypothesis. These foreground and background masks allow us to decouple the effects of background changes from class-specific appearance variability.

Our segment hypotheses are obtained using SLIC superpixels [1] with the implementation of [43] in a fraction of a second. We extract superpixels over 7 scales, ranging from 200-250 to < 10 superpixels per image, and for five different regularisation values (we will make our code available, and therefore omit exact parameter values). This provides us with a large pool of candidate segments of different size, valid both for objects that can take up the whole image and also for small image parts.

For every candidate detection hypothesis we only consider superpixels which contain the center of the hypothesis' bounding box. We then use the intersection over union as a matching metric to select the top $k = 15$ matching superpixels out of these. Averaging these provides us with an affinity measure that ranges in $[0, 1]$, indicating how likely it is that two pixels or blocks belong to the same region.

Indexing HOG blocks (or 'cells') by i, we denote this affinity measure with $f[i]$, where i ranges over the filter size (e.g. $i \in [1, 6] \times [1, 6]$ for a 6×6 part filter).

We use this affinity to build segmentation masks over the window using a sigmoid function parameterized by a 'segmentation hardness' parameter λ:

$$M[i] = \frac{1}{1 + \exp\left(-\frac{10}{1-\lambda}(f[i] - \lambda)\right)}. \quad (2)$$

This expression ensures that for $f[i] = 1$ we will have $M[i] \approx 1$ regardless of λ, so $\lambda \in [0, 1)$ can be determined in a per-category manner through cross-validation. Fig. 2 shows how this approach works over multiple scales and object categories, along with some failure cases. More results are provided in Fig. 7.

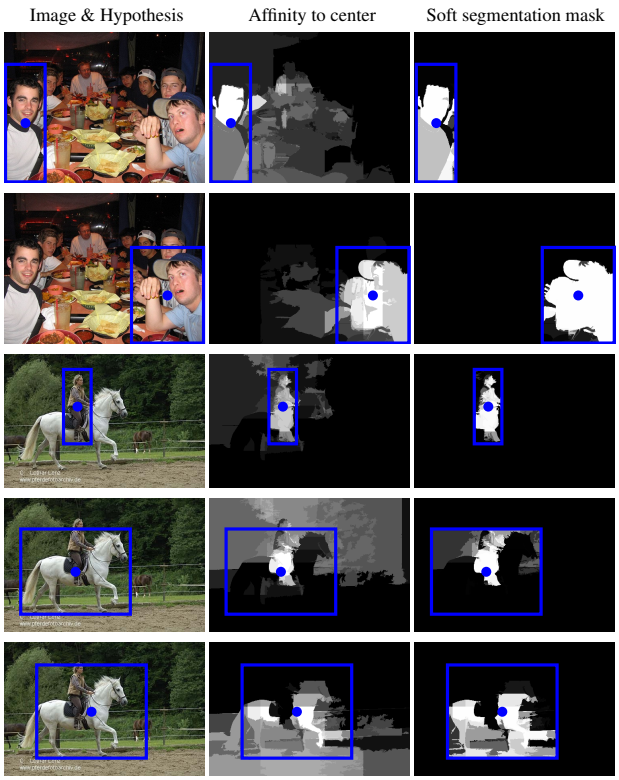


Figure 2. Examples of soft masks over different scales and object categories. Note that even when the center of the window hypothesis does not contain the object, as is the case for row 2, ranking the superpixels by how well they match the window can help us recover. This can still be problematic over extreme examples such as those on rows 4 and 5, where the rider correctly detected in row 4 occludes the middle part of the horse, breaking the object in two. We deal with these scenarios by keeping the original HOG features. Note that we use pixel data for illustration purposes, but for DPMs our masks are computed over HOG blocks.

We use these soft masks as weights over the HOG features, to pick those that share an affinity with the center of the candidate window: $G^+[i] = M[i] \cdot G[i]$. As

the background can be informative for some object categories, we also consider the complementary set of features, $G^-[i] = (1 - M[i]) \cdot G[i]$. Our extended feature array is the concatenation of (i) the original features (ii) the figure-ground channels and (iii) the mask itself:

$$G^{seg}[i] = \left[G[i], G^+[i], G^-[i], M[i]\right]. \quad (3)$$

This feature array can be applied directly to standard DPMs—the cost of computing and scoring the superpixels and building the masks is small compared to the actual cost of the convolution. Our implementation extends the fast convolution with SSE instructions of [14], and will be released as open source, while we also consider exploiting recent advances on fast DPM detection [18, 19].

3.3. Segmentation mask 'Alpha-blending'

The assumptions behind our segmentation method may not hold for certain categories—for instance for bicycles the center of a bounding box often does not belong to the object, bottles may contain transparencies or specularities, and people or man-made objects like vehicles are often composed of diverse components which are hard to segment together, as illustrated in the last examples of Fig. 7. As suggested by Table 1, using segmentation features for such categories may result in a performance drop.

We address this problem with a strategy similar to that of 'alpha-blending' for images. Namely, given a design parameter $\alpha \in [0, 1]$, we define new masks M_α as:

$$M_\alpha[i] = (1 - \alpha)1[i] + \alpha M[i], \quad (4)$$

where $1[i]$ indicates the unit function, and apply these over the feature array G as before. For $\alpha = 1$ (or 100%) we have our full-blown segmentation-sensitive features, while for α tending towards 0, the foreground-HOG channel G^+ becomes equal to the HOG features G, while G^- tends to 0; for intermediate values of α we work with features that blend between these two extremes.

For both λ and α we use cross-validation to separately fix the right parameter values per object category.

3.4. Superpixel-grounded descriptors

Having described our method on using superpixels to decompose HOG features into foreground and background channels, we now describe how we can use similar ideas to address the problem we had originally considered in [40].

There we introduced a methodology to build soft segmentation masks for dense SIFT and SID [21] descriptors based on the soft segmentations of [26]. These soft segmentations served as low-dimensional embeddings $e(x)$ for every pixel x, which we used in turn to measure the affinity between a pair of pixels, i, j in terms of $\|e(x_i) - e(x_j)\|_2$. In particular, around a point x_i we built segmentation masks according to:

$$w_i = \exp\left(-\lambda \|e(x_i) - e(x_j)\|_2\right) \quad (5)$$

where λ is a 'hardness' design parameter analogous to the one used in this paper. The last step amounted to 'gating' the SIFT features with these soft segmentation masks, and is again analogous to what we have been doing so far on the HOG channel.

When it comes to using SLIC superpixels to compute the soft affinity masks for descriptors certain things change with respect to object detection: our goal is still to separate foreground and background, but we cannot rely on having a single 'bounding box' per pixel (as was the case for detection). In other words, we want to treat all pixels equally, rather than adapt our masks for those pixels that may contain whole objects.

To do this we adjust the technique of Sec. 3.2 by (i) using SLIC superpixels that are approximately the same size or larger (at least 50%) than the image patch, ensuring that the mask borders are at the same scale as the image patch, and (ii) by using all the superpixels that contain the current pixel, rather than using intersection-over-union for superpixel ranking.

Qualitative results of this method are shown in Fig. 3, while a quantitative evaluation follows in the next section.

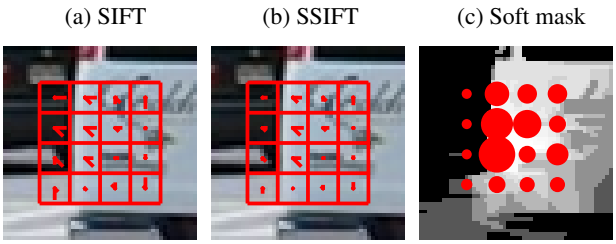


Figure 3. SIFT (a) and segmentation-aware SIFT (b). The response of the background pixels is greatly attenuated. (c) shows the soft segmentation mask computed from SLIC superpixels, and its magnitude at the SIFT grid coordinates, which are the weights applied over the descriptor bins.

4. Results

We present two experiments: the DPM-based object detector introduced in the previous section, and an extension of the work presented in [40] to enhance SIFT descriptors with segmentation masks.

4.1. Object detection on the PASCAL VOC

We evaluate the performance of our approach on the PASCAL VOC 2007, with standard DPM as a baseline, using the evaluation kit provided by [13]. For our approach

we use five different values for $\lambda \in [0.3, 0.4, 0.5, 0.6, 0.7]$, where λ determines of the 'hardness' of the segmentation, and pick the best value for each object category with two-fold cross-validation. As explained in Sec. 3.2, we can apply 'alpha-blending' to determine how much of the soft segmentation mask is desirable over different object categories—in particular, after we determine λ we follow the same procedure for 'alpha-blending', with α values 100% (i.e. no blending), 75%, 50%, and 25%. Our approach outperforms standard DPM on 17 out of 20 classes, for an average improvement of 1.7% AP (1.3% without 'alpha-blending'). We report the results in terms of average precision (AP) in Table 1, and the precision-recall curves in Fig. 4. We display the per-class increase in performance over DPM in Fig. 5. Fig. 7 shows some examples of the soft masks generated by our filters on the PASCAL VOC.

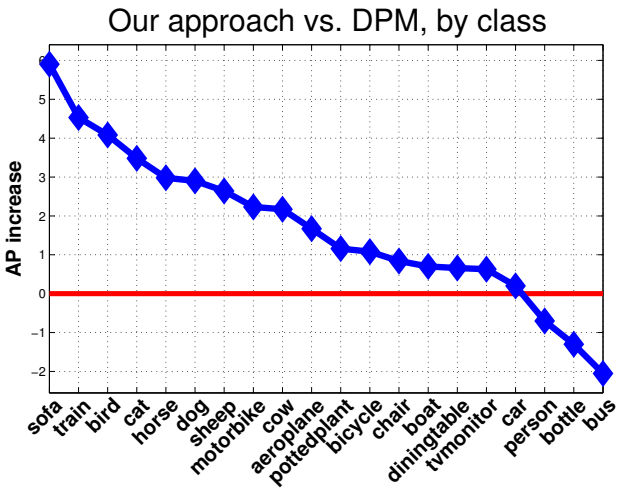


Figure 5. Increase in performance (AP in %) between our segmentation-sensitive DPM, with 'alpha-blending', and standard DPM, for every class.

4.2. Large displacement SIFT-flow

We use SIFT-flow [28] to compute large displacement optical flow estimates with enhanced dense SIFT descriptors (DSIFT). To evaluate descriptor performance we use 31 image pairs with ground truth object segmentation annotations from the Berkeley Motion Dataset [5], all of which feature multi-layered motion. Our metric is the Dice overlap coefficient [11] between the ground truth mask for the first frame and the ground truth for the k-th frame warped over the first frame with the flow estimates. The results are shown in Fig. 6, for different sizes of the descriptor bin—we include only the best λ for every case. There are ground truth annotations every ten frames, and the results are accumulated—e.g. the first bin contains every pair, and second bin contains every frame pair separated by 20 or more frames, and so on. We report experimental results using both the exponential of Eq. 5 and the sigmoid of Eq. 2 to build the masks.

Our approach shows better performance than SIFT, and closely matches that of [40]. However, our SLIC-based masks are faster to compute. Furthermore, when using superpixel segmentations the affinity between pixels can be computed through binary membership operations, rather than euclidean distances in Eq. 5; this results in yet another acceleration. As before, we refrain from providing all the details and will make the code available instead [39].

5. Conclusions

We have presented a simple technique to combine bottom-up segmentation with object detection, using SLIC superpixels computed at different scales to build soft segmentation masks. This process is fast enough that we can compute it for every hypothesis of a multi-scale sliding-window detector. We apply it to DPM, using the segmentation masks to 'clean up' the HOG features, for both the root and part filters. We evaluate it on the PASCAL VOC and demonstrate consistent improvements. We also extend the same design principle to build background-invariant SIFT descriptors, removing the features which share little affinity with the center of the descriptor, and thus making them more robust against background motion and occlusions—again, this process is fast enough that we can use it to compute dense descriptors, i.e. for every pixel in the image.

Regarding future work, an obvious extension would be to try to pick the right λ for every object instance, instead of object category—it is unclear how to do this. A possible criticism of our method is that it takes as a reference the center of a bounding box, which may not actually contain the object—we could consider the statistics of the superpixels to design a richer model. We also intend to consider alternatives to SLIC superpixels [26, 3, 27, 7], which may provide us with increased performance at a computational cost.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *PAMI*, 34(11):2274–2282, 2012.

[2] B. Alexe, T. Deselaers, and V. Ferrari. What is an object? In *CVPR*, 2010.

[3] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *PAMI*, 33(5):898–916, 2011.

[4] T. Brox and J. Malik. Berkeley motion segmentation dataset. `http://lmb.informatik.uni-freiburg.de/resources/datasets/moseg.en.html`, 2010.

[5] T. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, 2010.

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbik	pers	plant	sheep	sofa	train	tv	Avg.
DPM	31.5	54.9	7.8	11.1	**29.9**	**51.2**	53.9	23.3	22.2	25.1	29.2	10.6	56.8	44.5	**40.4**	13.6	20.2	30.2	42.2	42.0	32.0
Ours/λ	**33.2**	55.9	**11.9**	**12.3**	28.9	49.1	53.8	26.6	**23.1**	25.4	**29.9**	**14.1**	**59.8**	45.8	39.4	**14.7**	21.2	34.3	44.4	**42.6**	33.3
Ours/α	**33.2**	**56.0**	**11.9**	11.8	28.6	49.1	**54.1**	**26.7**	**23.1**	**27.3**	**29.9**	13.5	**59.8**	**46.8**	39.8	**14.7**	**22.8**	**36.1**	**46.7**	**42.6**	**33.7**

Table 1. AP performance (%) on the PASCAL VOC 2007, for DPM (first row) and for our method, with cross-validated λ (second row), and with 'alpha-blending' (third row). We cross-validate α only for the best λ per category (i.e. not a full α/λ sweep). Entries where the second and third rows are equal correspond to those where $\alpha = 100\%$, i.e. no 'alpha-blending'.

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

[7] J. Carreira and C. Sminchisescu. Constrained parametric min-cuts for automatic object segmentation. In *CVPR*, 2010.

[8] D. Cremers. Dynamical Statistical Shape Priors for Level Set-Based Tracking. *PAMI*, 28(8):1262–1273, 2006.

[9] D. Cremers, F. Tischhauser, J. Weickert, and C. Schnorr. Diffusion Snakes: Introducing Statistical Shape Knowledge into the Mumford-Shah Functional. *IJCV*, 50(3):295–313, 2002.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[11] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3), 1945.

[12] I. Endres and D. Hoiem. Category-independent object proposals. In *Proc. ECCV*, 2010.

[13] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010.

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

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

[16] T. Gao, B. Packer, and D. Koller. A segmentation-aware object detection model with occlusion handling. In *CVPR*, 2011.

[17] S. Gould, T. Gao, and D. Koller. Region-based segmentation and object detection. NIPS, 2009.

[18] I. Kokkinos. Rapid deformable object detection using dual-tree branch-and-bound. In *NIPS*, 2011.

[19] I. Kokkinos. Shufflets: shared mid-level parts for fast object detection. In *ICCV*, 2012.

[20] I. Kokkinos and P. Maragos. Synergy Between Image Segmentation and Object Recognition Using the Expectation Maximization Algorithm. *PAMI*, 31(8):1486–1501, 2009.

[21] I. Kokkinos and A. Yuille. Scale invariance without scale selection. In *CVPR*, 2008.

[22] M. P. Kumar and D. Koller. Efficiently selecting regions for scene understanding. In *CVPR*, 2010.

[23] P. Kumar, P. H. Torr, and A. Zisserman. Objcut: Efficient segmentation using top-down and bottom-up cues. *PAMI*, 32(3):530–545, 2010.

[24] L. Ladicky, P. Sturgess, K. Alahari, C. Russell, and P. H. S. Torr. What, where and how many? Combining object detectors and CRFs. In *ECCV*, 2010.

[25] V. Lempitsky, A. Blake, and C. Rother. Image segmentation by branch-and-mincut. In *ECCV*, 2008.

[26] M. Leordeanu, R. Sukthankar, and C. Sminchisescu. Efficient closed-form solution to generalized boundary detection. In *ECCV*. 2012.

[27] J. J. Lim, C. L. Zitnick, and P. Dollar. Sketch tokens: A learned mid-level representation for contour and object detection. In *CVPR*, 2013.

[28] C. Liu, J. Yuen, and A. Torralba. SIFT flow: dense correspondence across difference scenes. *PAMI*, 33(5), 2011.

[29] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[30] T. Malisiewicz and A. A. Efros. Improving spatial support for objects via multiple segmentations. In *BMVC*, 2007.

[31] S. Manén, M. Guillaumin, and L. V. Gool. Prime object proposals with randomized Prim's algorithm. In *ICCV*, 2013.

[32] P. Ott and M. Everingham. Implicit color segmentation features for pedestrian and object detection. In *ICCV*, 2009.

[33] C. Pantofaru, C. Schmid, and M. Hebert. Object recognition by integrating multiple image segmentations. In *ECCV*, 2008.

[34] D. Ramanan. Using segmentation to verify object hypotheses. In *CVPR*, 2007.

[35] M. Rousson and N. Paragios. Shape priors for level set representations. In *ECCV*, 2002.

[36] B. C. Russell, W. T. Freeman, A. A. Efros, J. Sivic, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, 2006.

[37] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *ECCV*, 2006.

[38] E. Tola, V. Lepetit, and P. Fua. A fast local descriptor for dense matching. In *CVPR*, 2008.

[39] E. Trulls. Code release. `https://github.com/etrulls`.

[40] E. Trulls, I. Kokkinos, A. Sanfeliu, and F. Moreno-Noguer. Dense segmentation-aware descriptors. In *CVPR*, 2013.

[41] Z. W. Tu, X. Chen, A. Yuille, and S. C. Zhu. Image Parsing: Unifying Segmentation, Detection, and Recognition. In *ICCV*, 2003.

[42] K. E. A. van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders. Segmentation as selective search for object recognition. In *ICCV*, 2011.

[43] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org`, 2008.

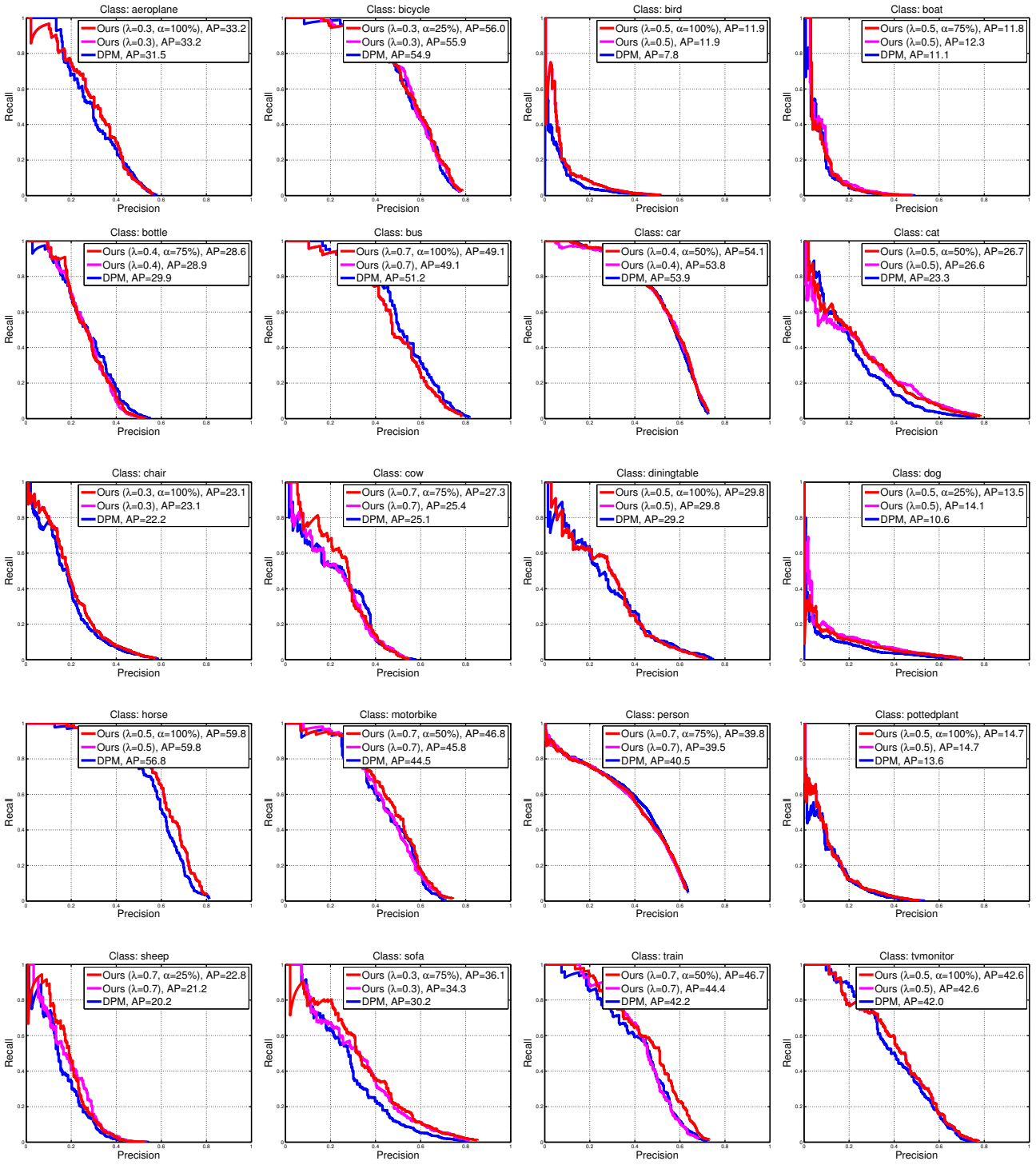


Figure 4. Precision/Recall curves on the PASCAL VOC 2007, for (1) the standard DPM (blue), (2) segmentation-aware DPM with cross-validated λ (magenta), and (3) segmentation-aware, alpha-blended DPM (red). For the latter, we cross-validate α for the cross-validated λ only. Note that (2) and (3) are the same filter if $\alpha = 100\%$.

[44] A. Vedaldi and A. Zisserman. Structured output regression for detection with partial truncation. In *NIPS*, 2009.

[45] D. Weiss and B. Taskar. Scalpel: Segmentation cascades with localized priors and efficient learning. In *CVPR*, 2013.

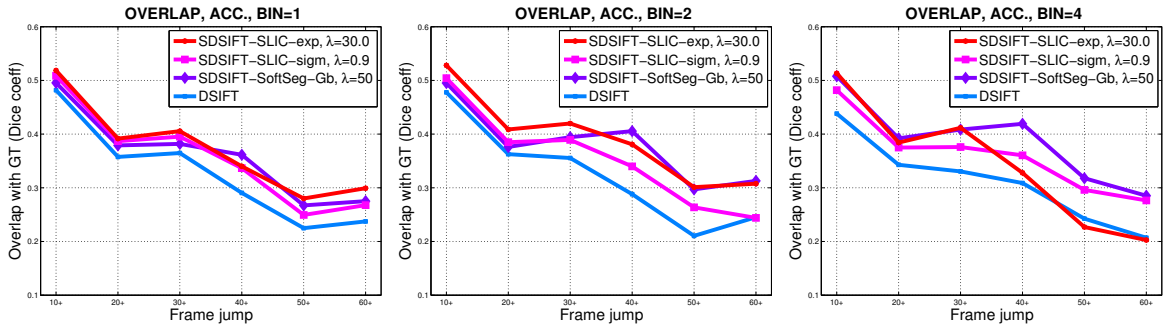


Figure 6. Overlap with the ground truth annotations of [4], for DSIFT (blue), and DSIFT with the soft segmentation masks of [40] (purple) and our SLIC-based masks (magenta and red, for masks built with Eqs. 2 and 5, respectively), at different scales.

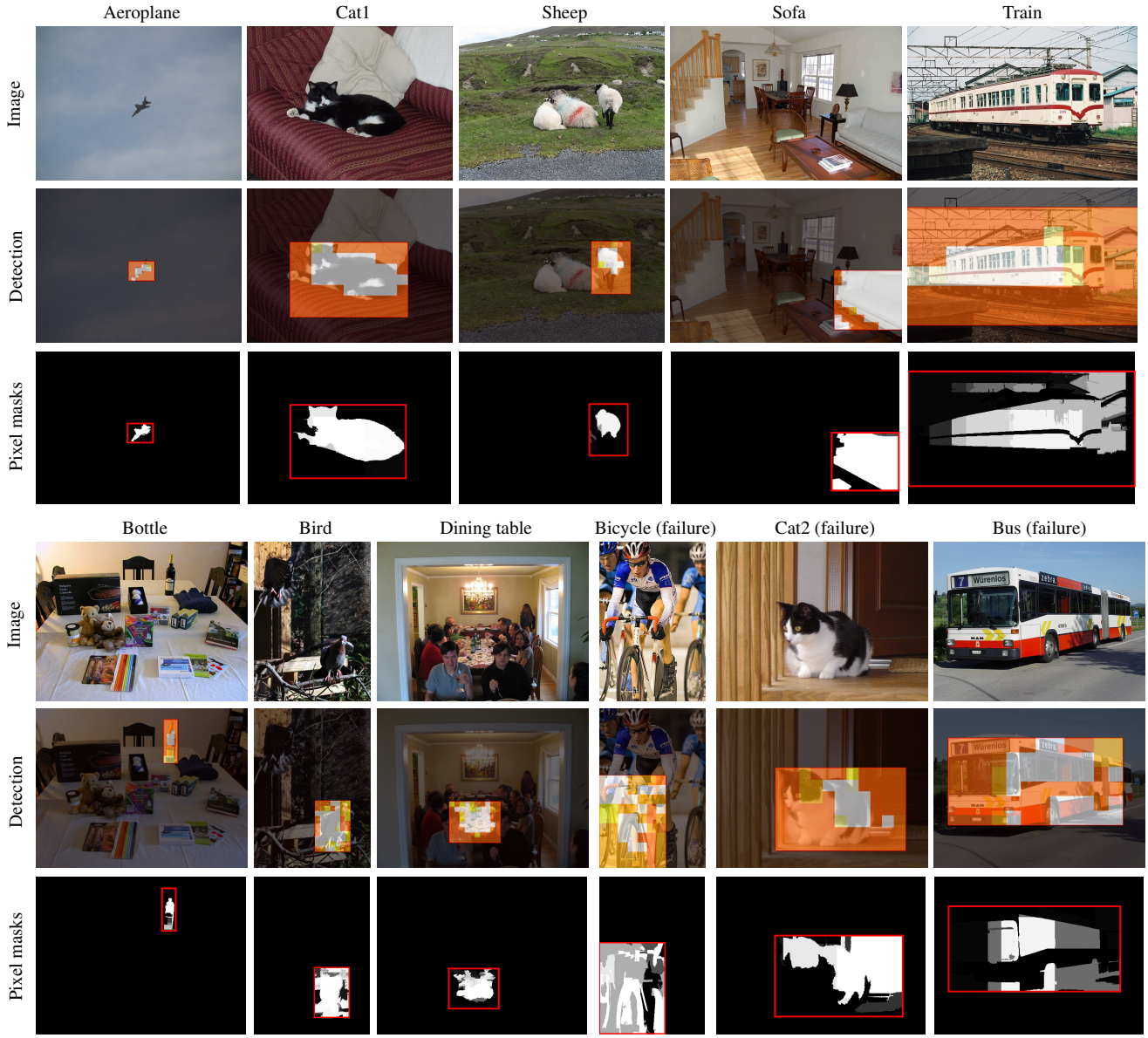


Figure 7. Top row: input image. Middle row: detection hypothesis, with the segmentation mask generated by our filters overlaid on top—only the root filter is shown. Bottom row: masks computed at a pixel level—the actual filter response is in HOG blocks (row 2). Note that our approach can fail when the center of the bounding box has a different appearance (cat2, bus), or the object is hard to segment (bicycle).