Object Segmentation by Alignment of Poselet Activations to Image Contours

Thomas Brox[1], Lubomir Bourdev[2,3], Subhransu Maji[2], and Jitendra Malik[2*]

[1]University of Freiburg, Germany
brox@informatik.uni-freiburg.de

[2]University of California at Berkeley
{lbourdev,smaji,malik}@eecs.berkeley.edu

[3]Adobe Systems Inc., San Jose, CA

Abstract

In this paper, we propose techniques to make use of two complementary bottom-up features, image edges and texture patches, to guide top-down object segmentation towards higher precision. We build upon the part-based poselet detector, which can predict masks for numerous parts of an object. For this purpose we extend poselets to 19 other categories apart from person. We non-rigidly align these part detections to potential object contours in the image, both to increase the precision of the predicted object mask and to sort out false positives. We spatially aggregate object information via a variational smoothing technique while ensuring that object regions do not overlap. Finally, we propose to refine the segmentation based on self-similarity defined on small image patches. We obtain competitive results on the challenging Pascal VOC benchmark. On four classes we achieve the best numbers to-date.

1. Introduction

As object detection is getting more and more mature, there is growing interest in precise object localization that goes beyond bounding boxes. Object segmentation provides the means for this. While bottom-up segmentation at the object level is ill-defined in general static images, it becomes a valid problem when supported by object detectors.

In this paper, we contribute to the line of research on how to combine bottom-up cues, as traditionally used in image segmentation, with top-down information as obtained from contemporary object detectors. Early works on this are [4, 13]. We argue that an object detector with a rich part structure, such as the recent poselet-based detector [5], provides an excellent basis for top-down object segmentation. In particular, poselets can deal well with occlusion and allow for competitive segmentation of multiple, partially occluding objects without explicit depth reasoning.

Figure 1. **Left:** Image from the Pascal VOC challenge. **Right:** Multiple object semantic segmentation with a person (light pink) and a horse (magenta).

The detector information needs to be accompanied by bottom-up cues. Object detectors can mark coarsely where an object of a certain class can be expected, but they lack the power to exactly localize the object. This is primarily due to the need of the detector to generalize over object instances, which leads to a loss of precise shape information. This missing information on the exact shape of the detected object instance must be recovered by means of the image itself. In this paper, we present two complementary ways to exploit information in the test image: image edges and self-similarity.

There are two main reasons why shape prediction by a detector is not exact: (1) Due to efficiency reasons of the scanning window approach, contemporary detectors are run on a subsampled grid. Consequently, each detection may be shifted a few pixels from the actual object location. (2) Due to averaging across multiple object instances and articulations, the detector only models a coarse shape that cannot predict the particular shape of the object instance at hand. Nonparametric shape models without such a shortcoming are too expensive and currently not used. Hence, deformations and fine details of the shape are not predicted.

*This work was supported by the German Academic Exchange Service (DAAD), Adobe Systems Inc., Google Inc., and ONR MURI N00014-06-1-0734.

In this paper, we suggest approaching the shift and deformation issue by non-rigidly aligning each poselet activation to the corresponding edge structures in the image. This extends the alignment strategy in [5], where the whole object shape was aligned to the image. By aligning each activation separately, we can allow for larger local deformations and better deal with occlusion and articulation.

As the alignment can only shift and deform the contour, it cannot regenerate holes or strong concavities in the object region, such as the horse legs in Fig. 1. To recover such shape details, we propose a process that can flip the labels of superpixels based on patch-based similarity.

Finally, in multi-class segmentation, we have to deal with multiple objects competing for the occupancy of a pixel. We propose an extension of poselets from people detection to other categories and a process that builds upon the detection scores of the poselet activations and their spatial distribution over the object area. While this process decides on which object part is in the foreground, it also sorts out many false positive detections. We show competitive results on the challenging Pascal VOC 2010 benchmark both with regard to quantitative numbers and visually appealing segmentations.

2. Related work

Shape priors in image segmentation have become popular based on the work by Leventon et al. [15] and the line of works by Cremers et al. [8, 9]. In particular Cremers et al. put much effort in a rich statistical modeling of shapes. While the statistical shape models in these approaches are very sophisticated, they assume that the class and coarse pose of the object in the image as well as its existence as such are already known. This is a strong assumption that, apart from specialized medical applications, can be hardly satisfied in practice.

In Leibe and Schiele [14] the detection of object-specific patches indicates the local existence of an object class, and its shape is derived from assembling the masks of these patches. Shape variation is solely modeled by the assembly of patches. Evidence from the test image is only used for detection but not for segmentation.

Most related to our work is the approach in [19]. Building upon the strong part-based detector by Felzenszwalb et al. [10], they refine the shape predictions of the detector by using color and reasoning about the depth ordering of objects.

Another related line of work is the one of texture-based semantic segmentation, where a texture classifier in combination with a CRF model assigns pixel labels to a restricted number of classes [18, 17]. While such approaches perform well on background classes like sky, water, building, trees, their performance on actual objects is usually significantly lower ('Oxford Brookes' in Table 2). Combined with image classification, this approach has been quite successful [11].

Finally, [16] follow a strategy where a fairly large number of object region hypotheses is generated. Classification is then done based on these segments using a set of color, texture and shape features. Similar in spirit, but focusing on a complex graphical model that makes more use of context, is the work in [12]. [16] performs very well on the Pascal VOC benchmark and is kind of complementary to our approach since detection hypotheses are generated using image segmentation tools and a classifier is applied to the features of these segments, whereas our approach detects and scores hypotheses in a scanning window fashion, and segmentation follows on the basis of these detections.

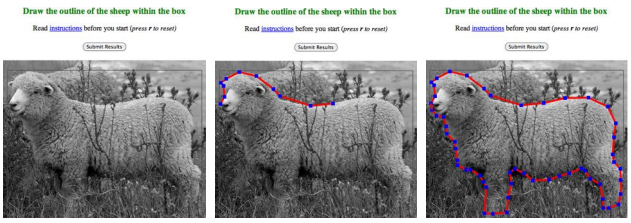
3. A baseline segmentation based on poselets

3.1. Poselets beyond people

We build on the poselet concept introduced in [6], where class and pose specific part detectors are trained by means of extra keypoint annotation. In particular, we use our framework in [5] and extend it to categories beyond person. To this end we must define class-specific keypoints. This is straightforward for animal categories but becomes more complicated for categories, such as chair, boat, or airplane, which show large structural variations. There are chairs with four legs or one stem and a wide base and military airplanes look very different from commercial ones. We split such categories into a few common subcategories and provide separate keypoints for each subcategory. This enables training separate poselets for the pointed front of a military airplane, the round tip of a commercial airliner, and the propeller blades of a propeller plane.

Some categories, such as bottles, do not have a principal orientation, which makes it difficult to assign keypoints in the reference frame of the object. For example, what is the front left leg of a table? Our solution is to introduce view-dependent keypoints. For example, we have a keypoint for the bottom left corner of a bottle, and we define the front left leg of a table based on the current camera view.

In [5] we showed that keypoints can be effective even if defined in 2D space. This helps tremendously when dealing with other visual categories in which there is no easy way to annotate the depth of a keypoint, but can sometimes introduce ambiguities. For example, in 2D configuration space the front view of a bicycle is almost identical to the back view; the only difference is that the left and right handle keypoints, which may not be visible in all examples, should be swapped. This could result in mixing the front and the back aspect, which are visually very different, into the same poselet. To prevent this scenario we made use of the view annotations of Pascal categories – "frontal", "left", "right" and "back". Specifically, we disallow the training examples of a poselet to come from the view opposite to the one in its



(a) (b) (c)

Figure 2. The user interface for annotating the outer boundary of the objects. (a) The user sees this inside the Amazon Mechanical Turk environment. (b) Partial annotation by the user. (c) The user closes the polygon and edits the boundary if needed and clicks the submit button.

seed.

Lastly, the visual categories vary widely in aspect ratios and using poselets of a fixed size and aspect ratio is sub-optimal. We extended the framework to support poselets of variable class-specific aspect ratios, and trained different number of poselets for each category.

3.2. Annotation with AMT

We collected 2D keypoint annotations and figure/ground masks for all training and validation images of the Pascal VOC challenge on Amazon Mechanical Turk [1]. For the keypoint annotation, 5 independent users are shown zoomed images of objects from a category together with a set of predefined keypoints. The users are asked to place these keypoints at the right place on the object or leave them unmarked if they are not visible due to occlusion, truncation, etc. We assume that a keypoint is visible if at least 2 annotators have marked its location.

Figure/ground masks were collected in a similar way. We ask the annotators to mark the outer boundary of the object using a polygon-like tool shown in Figure 2. This simple interface allows to quickly mark outer boundaries of the object. We again collect 5 independent annotations for each object.

3.3. Mask summation

The figure/ground annotation enables us to generate a soft mask $m \in [0, 1]$ for each poselet by averaging the binary segmentation annotation among all example patches used for training the respective poselet classifier (Fig. 3).

At test time, each poselet activation i assigned to a certain object hypothesis j now comes with a soft mask $m_{ij} : \mathbb{R}^2 \rightarrow [0, 1]$ indicating the probability that a certain pixel at the detected location is part of the object j or not. We can build a very simple baseline segmentation by just summing over all soft masks m_{ij} assigned to one object j:

$$M_j(x, y) = \sum_i m_{ij}(x, y) \quad (1)$$

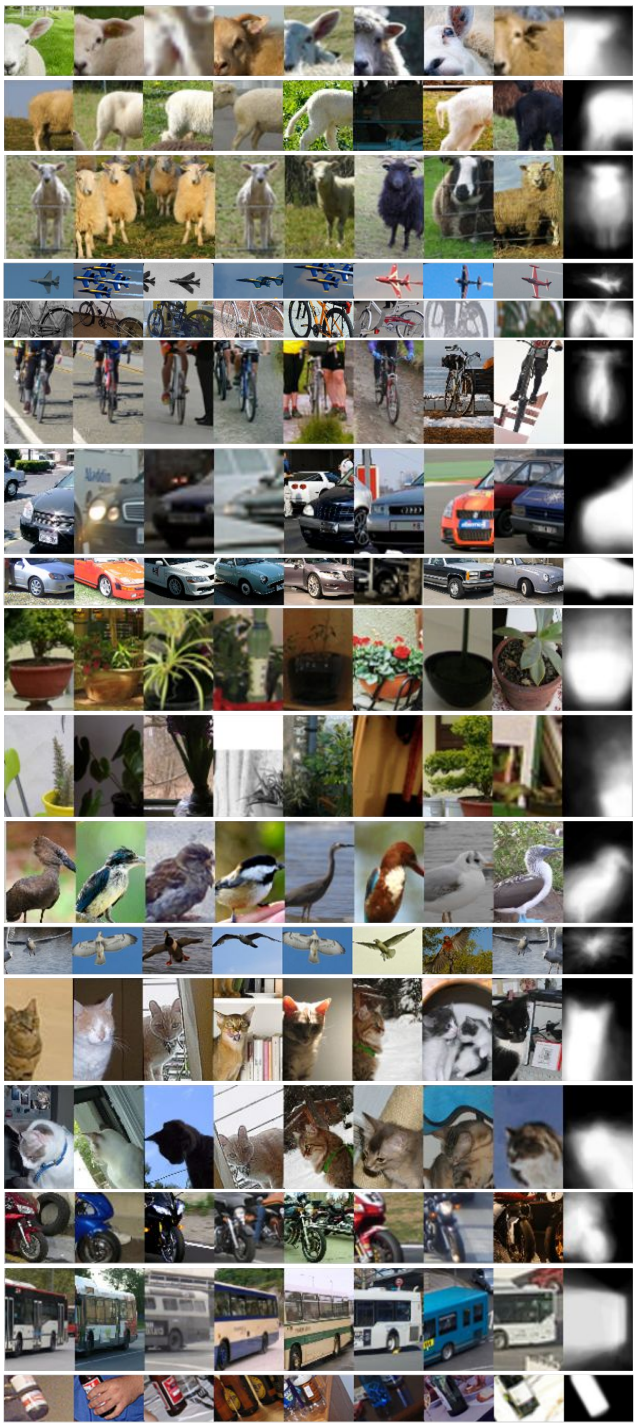


Figure 3. Each row shows some training examples of a particular poselet and its average mask.

and setting all points where the mask is smaller than a threshold θ_m to zero. Since we aim at a disjoint segmentation, i.e., each pixel can only be assigned to one object, we simply select the object with the maximum score:

$$\mathcal{C}(x, y) = \text{argmax}_j M_j(x, y), \quad (2)$$

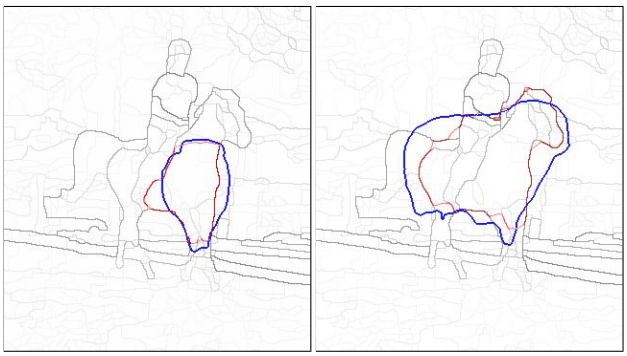


Figure 4. Poselet contour before (blue) and after (red) alignment.

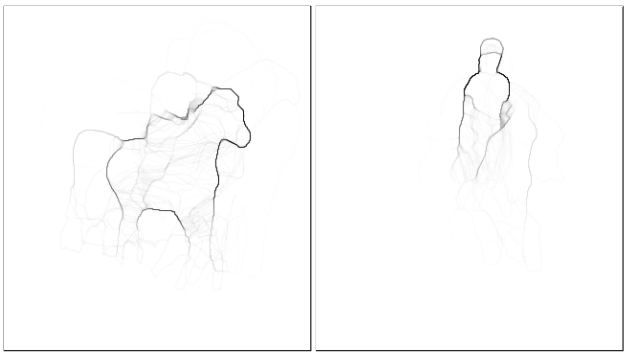


Figure 5. Summed poselet contours after alignment. Thanks to the alignment, almost all contours agree and lead to a good prediction of the object contour.

where we ignore all object hypotheses j with a score less than a threshold θ_c to avoid considering false positive detections in the segmentation. In Table 1 we compare this baseline to the improvements we present in the following three sections.

4. Alignment

As the soft masks m have been obtained by averaging across multiple object instances and articulations in the training data, they correspond only coarsely to the actual shape of the particular object instance in the test image. The information about the precise location of the object contour has been lost in this averaging process. We aim at retrieving this information by aligning the poselet contours to the edge map of the test image. This assumes that (1) the true object contour is a subset of the contours in the edge map (allowing few exceptions), and (2) that the true object contour is close to the poselet prediction.

We take the 0.5 level set of m_{ij} to obtain the poselet contour $g_{ij} : \mathbb{R}^2 \to \{0, 1\}$ as predicted by the classifier (Fig. 4). For the image edge set $f : \mathbb{R}^2 \to [0, 1]$ we use the ultrametric contour map (UCM) from [3], which is among the best performing contour detectors. We then estimate the non-rigid deformation field (u, v) that locally aligns the

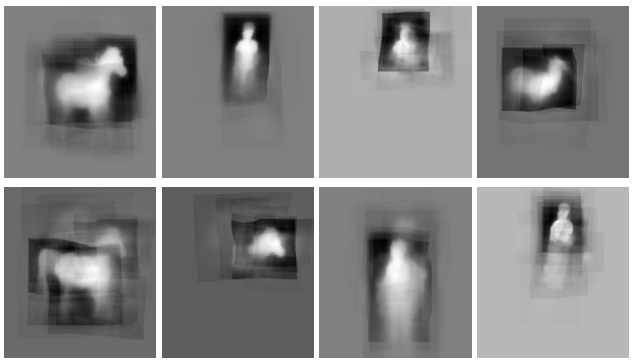


Figure 6. Summed poselet masks after alignment for 8 out of 20 object hypotheses. For visualization, values have been normalized to a [0,255] range. Only the two hypotheses in the top left will survive the competition in Section 5.

predicted silhouette g to the edge map f. This is achieved by minimizing

$$E(u, v) = \int_{\mathbb{R}^2} |f(x, y) - g(x + u, y + v)| \, dxdy + \alpha \int_{\mathbb{R}^2} (|\nabla u|^2 + |\nabla v|^2) \, dxdy. \tag{3}$$

with $\alpha = 100$. This is done by a variational coarse-to-fine minimization technique as used in variational optical flow estimation [7]. The alignment yields the aligned silhouette prediction. Moreover, the field (u, v) can be used to align the soft mask m_{ij} as well. Fig. 4 shows two poselet silhouettes before and after the alignment.

Again the aligned soft masks can be summed to generate a prediction of the whole object. Since in contrast to the baseline in Section 3.3 the masks have been aligned before summing them, they mostly agree on a common contour. This can be seen very well from the sum over the aligned contours g_{ij}, as shown in Fig. 5.

5. Competitive smoothing

After aligning and summing the masks, we are confronted with three challenges:

1. We would like to detect and remove false positives by means of the consistency of the aligned poselet masks.

2. There can be multiple, partially overlapping objects in one image. In such cases we have to decide, which of the detections occupies a certain pixel.

3. Object labels should be spatially consistent. Therefore, we must smooth the masks. In this smoothing process, we would like to preserve the precise localization of boundaries as established by the alignment procedure.

By preprocessing the aligned masks, we aim at the first two objectives to obtain good initial masks for the variational

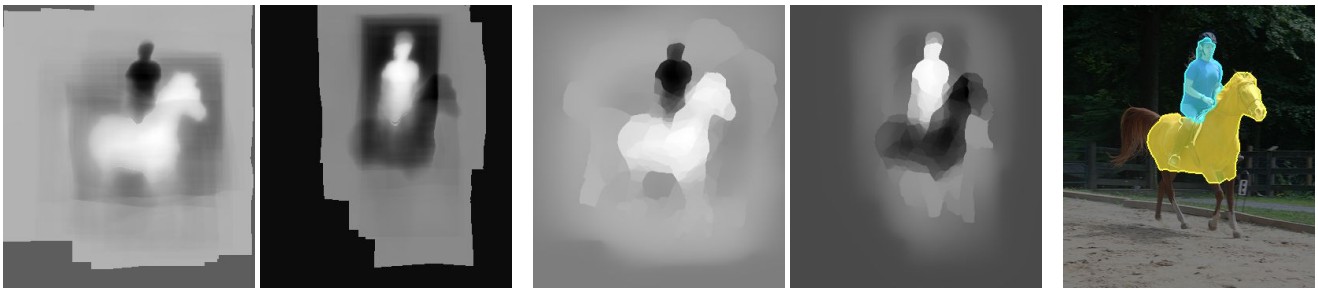


Figure 7. **Left:** Remaining mask predictions M_j'' after competition and damping. **Center:** Masks u_j after spatial aggregation. **Right:** Overlay on the image.

smoothing method that mainly deals with the spatial aggregation while preserving the previously established properties.

Let M_j denote the summed soft masks of object j. First we run a winner-takes-all competition on each pixel independently by setting:

$$M_j'(x,y) = \begin{cases} M_j(x,y), \text{if } M_j(x,y) = \max_k M_k(x,y) \\ M_j(x,y) - \max_k M_k(x,y), \text{otherwise} \end{cases} \tag{4}$$

If two objects j are of the same category and one of them wins at pixel x, y, we add the loser's value to the winner. This ensures that we do not lose object evidence due to an erroneous clustering of poselets. For a few typical confusion cases, such as bus and car or cow and sheep, we bias the decision on the winner by multiplying M_j by a pair-specific value to normalize the relative strength of the poselets of the two categories. Apart from the winner decision, M_j' is not affected by this bias.

For each object mask we compute the accumulated positive mass

$$\chi_j = \int_{\mathbb{R}^2} \delta_{M_j(x,y)>0} M_j(x,y)\, dxdy \tag{5}$$

before and after the competition. Objects j that lose at least half of their mass are removed. Their positive mass is reassigned to the winning object. This ensures that a removed object does not leave a hole, as its mass in winning areas is given back to the best competitor.

Like in the baseline method, we only consider objects with a detection score larger than θ_c. This fast selection of candidate objects is complemented by taking also the summed soft masks M_j' into consideration. A high detection score might have been obtained by having several wide spread poselet activations that did not align well to the same contours. We can detect such cases by considering M_j', which will then tend to be small for all x, y. We build a normalized soft mask

$$M_j''(x,y) = \frac{M_j'(x,y)}{\lambda + \max_{x,y} M_j'(x,y)}, \tag{6}$$

where λ is a damping parameter. This normalizes all soft masks and ensures that the maximum of M_j'' approaches 1 in places of large confidence. Objects with $M_j'' < \frac{1}{2}$ everywhere are removed. Apart from removing more false positives, this procedure also deals with false positive poselet activations that have been erroneously assigned to the object. As their soft mask does not agree with that of other activations, the damping pushes M_j'' to a value close to 0 in these areas, which makes the area likely to be smoothed away.

Finally, we determine smoothed versions u_j of the masks M_j'' with a variational method minimizing

$$E(u_1, ..., u_K) = \sum_j \int (u_j - M_j'')^2 |M_j''| + \frac{2}{C_j + 1} |\nabla u_j|\, dxdy, \tag{7}$$

subject to $\sum_j \delta_{u_j(x,y)>0} \leq 1, \forall x, y$. This energy model consists of an evidence term, taking into account the mask predictions M_j'', and a smoothness term that aggregates information from larger areas to agree on a specific class label. The energy seeks to have the final labels close to the predicted masks, while producing compact areas that align well with the predicted contours C_j. C_j denotes the summed aligned contours g_{ij} of an object normalized to a range [0,255]. In areas where the mask prediction is uncertain, we want the smoothness term to have more impact than in areas where the mask label is already well predicted. This is achieved by weighting the evidence term with the mask magnitude, which is zero if there is no evidence if the pixel belongs to that object or not.

Apart from the additional constraint that ensures disjoint regions, this is a convex optimization problem, i.e., we can compute the global optimum of the unconstrained problem with a variational technique. The Euler-Lagrange equations

$$(u_j - M_j'')|M_j''| + \text{div}\left(\frac{\nabla u_j}{(C_j+1)|\nabla u_j|}\right) = 0 \tag{8}$$

yield a nonlinear system of equations, which we solve using a fixed point scheme with an over-relaxed Gauss-Seidel

Figure 8. **Top row:** Segmentation before refinement. **Bottom row:** Segmentation after refinement. Many details are corrected by means of the object's self-similarity. The second example from left shows that we can also separate multiple instances of the same category. This more challenging problem is not covered in current benchmarks.

solver. The constraint is established by projecting back to the constrained set in each fixed point iteration.

Fig. 7 shows the mask prediction M_j'' and the corresponding mask u_j after this aggregation. As the aligned contour C_j is taken into account, we obtain sharp object boundaries. The l_1 norm in the smoothness term supports this effect by closing gaps in C_j. In contrast, the l_2 norm would lead to leakage. The positive parts of u_j yield the binary object mask. Areas not occupied by an object yield the background mask.

6. Refinement based on self-similarity

While the previous alignment process has improved the consistency of the predicted shape with image boundaries, the shape still lacks most concavities, e.g., the legs of the horse. So far, we have made use of color and texture only indirectly by considering color and texture discontinuities. We suggest further refinement of the shape by means of self-similarity of the objects. This refinement can flip the labels of superpixels if they better fit to another object according to their color and texture.

We start with the object masks we have obtained so far and build a non-parametric appearance model for each object and the background. For the appearance we consider 7×7 patches in the Cielab color space. We multiply the L channel with factor 0.1 to reduce its weight in the patch distances. We do not run a refinement on grayscale images.

Rather than flipping single pixels, we consider superpixels as provided by the UCM that we already used for the alignment. In case the object boundary does not coincide with a UCM boundary, we add such an edge and split the UCM region accordingly. This ensures that the top-down shape knowledge can still hallucinate boundaries that are not visible in the image or have been missed by the UCM.

For each pixel within a superpixel, we find the 100 nearest neighbors in the image with an approximate nearest neighbor method. The labels found at these 100 nearest neighbors vote for the desired label of the superpixel. Formally, we can write this as an approximate density estimation with a nearest neighbor kernel:

$$p(F(x,y)|j) \approx N_j^{-\frac{1}{2}} \sum_{k=1}^{100} \delta_{u_j(x_k,y_k)=1}, \tag{9}$$

where $F(x,y)$ denotes the patch at the pixel of interest and $k = 1, ..., 100$ lists its 100 nearest neighbors. N_j denotes the size of object j. The label of the superpixel R is decided according to the maximum a-posteriori principle:

$$\mathcal{C}(R) = \text{argmax}_j \sum_{l \in R} \left(\log p(F(x_l, y_l)|j) + \log p(j) \right). \tag{10}$$

We use a uniform prior $p(j)$, except for bicycles, which are highly non-convex objects that often show much of the background in the initial object mask. This can be seen in the fourth example of Fig. 8[1]. We determined the optimum prior for this class on the training set.

In order to avoid flipping superpixels far away from the actual object just because of similar color and texture, we only allow superpixels to obtain a label that exists within a 10 pixel distance to this superpixel. Iteration of this process ensures that labels can still propagate over long distances as long as the object is contiguous. We stop iterations if there is no further change in labels anymore.

Fig. 8 shows some results before and after the refinement. In most cases we obtain more precise segmentations, and we can avoid uncontrolled spreading of labels to the background.

[1]Our AMT annotation did not include the holes in the wheels and the frame.

	full model Sec.6	alignment+ smoothing Sec.5	alignment Sec.4	baseline model Sec.3
background	79.23	78.77	78.76	78.58
aeroplane	36.26	33.25	26.14	26.63
bicycle	38.54	36.02	32.98	32.14
bird	16.57	15.78	13.46	12.70
boat	12.14	12.38	13.18	12.74
bottle	30.40	30.45	32.94	31.40
bus	33.20	32.28	28.43	29.24
car	42.15	41.88	39.84	39.25
cat	44.99	42.87	38.67	38.19
chair	10.33	8.99	8.27	7.89
cow	37.21	34.80	29.77	29.24
diningtable	10.69	9.90	11.61	11.37
dog	23.15	21.64	18.04	17.61
horse	43.92	40.71	36.34	35.41
motorbike	32.59	31.53	28.52	27.90
person	49.64	47.78	44.92	44.00
pottedplant	17.60	18.91	18.12	17.07
sheep	37.38	34.23	27.63	26.68
sofa	9.49	9.22	9.97	9.72
train	23.55	22.63	20.23	20.34
tvmonitor	47.50	47.19	38.87	43.51
average	32.21	31.01	28.41	28.17

Table 1. Segmentation results on the combined Pascal VOC 2007 training, validation and test set (632 images).

	ours	Barce-lona	Bonn	Chicago	Oxford Brookes
background	82.2	81.1	84.2	80.0	70.1
aeroplane	43.8	58.3	52.5	36.7	31.0
bicycle	23.7	23.1	27.4	23.9	18.8
bird	30.4	39.0	32.3	20.9	19.5
boat	22.2	37.8	34.5	18.8	23.9
bottle	45.7	36.4	47.4	41.0	31.3
bus	56.0	63.2	60.6	62.7	53.5
car	51.9	62.4	54.8	49.0	45.3
cat	30.4	31.9	42.6	21.5	24.4
chair	**9.2**	9.1	9.0	8.3	8.2
cow	27.7	36.8	32.9	21.1	31.0
diningtable	6.9	24.6	25.2	7.0	16.4
dog	29.6	29.4	27.1	16.4	16.4
horse	**42.8**	37.5	32.4	28.2	27.3
motorbike	37.0	60.6	47.1	42.5	48.1
person	**47.1**	44.9	38.3	40.5	31.1
pottedplant	15.1	30.1	36.8	19.6	31.0
sheep	35.1	36.8	50.3	33.6	27.5
sofa	**23.0**	19.4	21.9	13.3	19.8
train	37.7	44.1	35.2	34.1	34.8
tvmonitor	36.5	35.9	40.9	48.5	26.4
average	34.9	40.1	39.7	31.8	30.3

Table 2. Our segmentation results on the Pascal VOC 2010 test set together with the top performing methods in this challenge. Latest results of more methods are available from [2].

7. Experimental evaluation

We evaluate the detection and segmentation approach on the Pascal VOC challenge. The poselet classifiers have been trained on the training and validation sets of the challenge (excluding images from the 2007 challenge). We also used these sets to optimize the parameters of our approach (e.g. λ). In order to show the impact of the different parts of our technique we removed one after the other until we end up with the baseline method described in Section 3. We compared these different versions on the combined training, validation and test set of the VOC 2007 challenge. None of the 2007 images have been used for training the classifiers or parameter optimization[2].

Table 1 shows the result of this comparison. Clearly, each part of the full model improves the overall performance. Compared to the baseline model we get an improvement of 15%. The alignment of poselets has only a small quantitative effect, as it affects only relatively small areas. Moreover, the alignment has a negative effect on tvmonitors, as the stronger boundary of the screen is preferred over the correct outer boundary of the monitor. Nonetheless, the alignment is very important in our model as it helps the aggregation, which shows the largest boost.

In order to compare to alternative object segmentation techniques we also run the full approach on the test set of the VOC 2010 challenge. Table 2 shows our results next to the top performing methods in this challenge. Our approach ranks third on the average score and shows the best results on 4 categories, among them the important person category. Although in contrast to the other approaches in Tab. 2 we have segmentation annotation on all training images, these segmentations were quite coarse. In contrast to the pixel-accurate VOC segmentations, they can be obtained easily using [1].

Fig. 9 shows some example segmentations. We obtain segmentations that align very well with the true object boundaries and thus also look visually very appealing. From the visual impression alone, one would actually expect even better quantitative numbers. However, as overall segmentation numbers are still relatively weak (even the very best method produces more false positives and false negatives than true positives), correctly detecting just one more big object in the dataset has a larger quantitative effect than boundaries that are more accurate by a few pixels. This is also the reason why in Table 1 we get the largest boost with the competitive smoothing.

8. Conclusions

We presented an approach for object segmentation based on a rich part-based detector combined with image edges

[2]For training horse poselets the 2007 training and validation sets were used.

Figure 9. Results on the Pascal VOC 2010 test set are usually pixel-accurate if the object is well detected. Failure cases, as shown in the last row, are mainly due to problems of the detector. Object completion requires a sufficiently homogenous object or background.

and self-similarity cues and we showed that this leads to very competitive results on the challenging Pascal VOC benchmark, where we could achieve the best numbers on four classes. Even more striking is the visual quality of our results due to the precise alignment of our predicted contours to edges in the image. Another interesting observation is that we obtain significantly better segmentation results than methods building upon the detector in [10], such as [19] or the Chicago entry in Table 2, although poselets overall perform worse than [10] in the detection task.

References

[1] Amazon Mechanical Turk. www.mturk.com. 3, 7

[2] The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. pascallin.ecs.soton.ac.uk/challenges/VOC/voc2010/results. 7

[3] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. From contours to regions: an empirical evaluation. *CVPR*, 2009. 4

[4] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. *ECCV*, 2002. 1

[5] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. *ECCV*, 2010. 1, 2

[6] L. Bourdev and J. Malik. Poselets: body part detectors training using 3D human pose annotations. *ICCV*, 2009. 2

[7] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. *ECCV*, 2004. 4

[8] D. Cremers, T. Kohlberger, and C. Schnörr. Nonlinear shape statistics in Mumford–Shah based segmentation. *ECCV*, 2002. 2

[9] D. Cremers, S. Osher, and S. Soatto. Kernel density estimation and intrinsic alignment for shape priors in level set segmentation. *International Journal of Computer Vision*, 69(3):335–351, 2006. 2

[10] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010. 2, 8

[11] J. Gonfaus, X. Boix, J. van de Weijer, A. Bagdanov, J. Serrat, and J. Gonzàlez. Harmony potentials for joint classification and segmentation. *CVPR*, 2010. 2

[12] S. Gould, T. Gao, and D. Koller. Region-based segmentation and object detection. *NIPS*, 2009. 2

[13] M. P. Kumar, P. H. S. Torr, and A. Zisserman. OBJ CUT. *CVPR*, 2005. 1

[14] B. Leibe and B. Schiele. Interleaved object categorization and segmentation. *British Machine Vision Conference*, 2003. 2

[15] M. E. Leventon, W. E. L. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. *CVPR*, 2000. 2

[16] F. Li, J. Carreira, and C. Sminchisescu. Object recognition as ranking holistic figure-ground hypotheses. *CVPR*, 2010. 2

[17] C. Russell, L. Ladick'y, P. Kohli, and P. H. S. Torr. Graph cut based inference with co-occurence statistics. *ECCV*, 2010. 2

[18] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. *CVPR*, 2008. 2

[19] Y. Yang, S. Hallman, D. Ramanan, and C. Fowlkes. Layered object detection for multi-class segmentation. *CVPR*, 2010. 2, 8