PSPGC: Part-Based Seeds for Parametric Graph-Cuts

Bharat Singh, Xintong Han, Zhe Wu and Larry S. Davis

University of Maryland, College Park

Abstract. PSPGC is a detection-based parametric graph-cut method for accurate image segmentation. Experiments show that seed positioning plays an important role in graph-cut based methods, so, we propose three seed generation strategies which incorporate information about location and color of object parts, along with size and shape. Combined with low-level regular grid seeds, PSPGC can leverage both low-level and high-level cues about objects present in the image. Multiple-parametric graph-cuts using these seeding strategies are solved to obtain a pool of segments, which have a high rate of producing the ground truth segments. Experiments on the challenging PASCAL2010 and 2012 segmentation datasets show that the accuracy of the segmentation hypotheses generated by PSPGC outperforms other state-of-the-art methods when measured by three different metrics(average overlap, recall and covering) by up to 3.5%. We also obtain the best average overlap score in 15 out of 20 categories on PASCAL2010. Further, we provide a quantitative evaluation of the efficacy of each seed generation strategy introduced.

1 Introduction

From the perspective of image labeling - accurately segmenting and labeling a set of known objects in an image - the goal of image segmentation is to discover a set of image regions that correspond to those objects. Since the appearance of an object may be based on a combination of color, texture and shape and is also a function of apparent size (scale), classic segmentation algorithms that construct a single partition of the image typically fail to recover segments that correspond to objects. So, vision systems have utilized families of segmentations - scale space representations and/or alternative segmentations constructed by varying parameters of bottom-up segmentation algorithms [1–3]. For example, by varying the merging parameter of multi-parametric graph-cuts (MPGC) [4–6] one can construct a pool of image segmentations. Additionally, over the past several years image segmentation has been augmented through data-driven methods. These methods can be based, for example, on examining how segmentation algorithms fragment object regions and then learning how to merge [7], or on using shape priors to bias segments selected from parametrically varied segmentation algorithms [5].

The MPGC algorithm is based on generating many seed regions - for example by sampling pixels on a rectangular grid. Enlarging the pool of seed regions increases the chance of "hitting" ground-truth objects, but at the expense

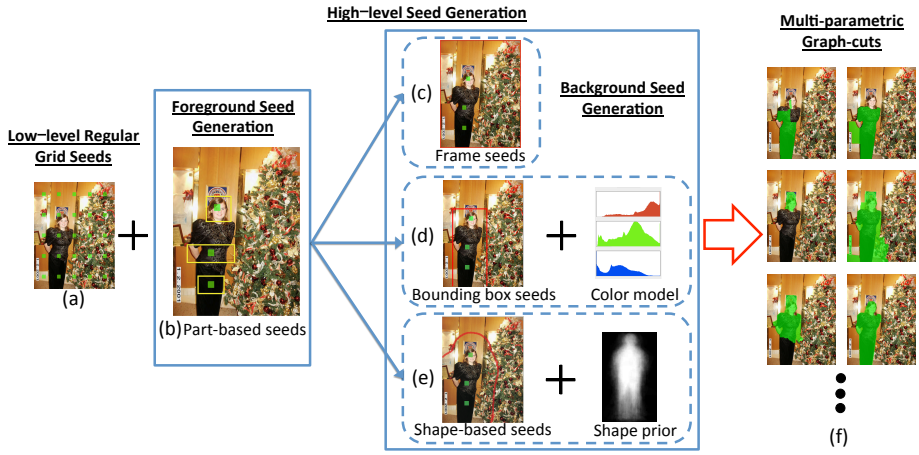


Fig. 1. Overview of PSPGC. (a) Low-level regular grid seeds are generated. (b-e) High-level seeds are generated using three different strategies. (b) Part-based foreground seeds: we apply DPM to detect parts with high filter responses (marked as yellow boxes) in a detected object, and a rectangular seed region (marked as a green rectangle) is generated inside each part. These seeds are used in all three strategies as foreground seeds. (c) Frame background seeds: the frame pixels are chosen as background seeds (marked in red). (d) Bounding box background seeds + Color model: pixels lying on the frame of the bounding box (marked as red) are set as background seeds. The weight of non-seed pixels belonging to the foreground or background depends on the color distribution of pixels lying on the bounding box frame and the foreground seed pixels. (e) Shape Prior + Shape-based background seeds: a shape prior for the detected object is selected and shape-based seeds are generated depending on this prior (marked as a red curve). The weight of non-seed pixels belonging to the foreground depends on the shape prior. (f) Segmentation hypotheses are generated by multi-parametric graph-cuts.

of increasing the complexity of subsequent image labeling. However, the main challenge methods like these face is that the segmentations they produce depend critically on how well the seed regions sample the statistics of the object. If the seed regions are too small, they cannot provide sufficient information (e.g., color distribution) about the foreground and background, so objects will be over- or under-segmented, and when they are too large, object and background statistics are merged in the seed and again segmentation fails.

These problems can be ameliorated if some seeds are chosen using data-driven methods that capture high-level priors about the object. Good seeds should capture the location of parts, color, size and shape of an object to generate a good prior about the foreground. As a function of imaging conditions, (for example: different scales), each of these characteristics have differential utility in producing an accurate segmentation of an object. To this end, we describe a detection-based, multi-parametric graph-cut method for image segmentation.

Given an image, we augment seeds chosen by sampling over a regular grid of square seeds with three complementary detection-based seeding methods. In all three methods, the foreground seeds remain the same - they are chosen by sampling rectangular regions within high ranking parts (parts with a high filter response) obtained by running deformable part-based models (DPM) [8] detectors. We refer to these foreground seeds as part based seeds. The background seeds differ for each method. For the first method, they are set to image frame pixels, to compensate for situations where the bounding box is highly inaccurate. In the second method, they are the pixels lying on the frame of the bounding box (to impose a size constraint on the object), while in the third method we train a class- and size- specific shape model to obtain a shape prior for this detected object, and the background seed is generated based on the contour of this shape prior; see Fig. 1(e). For the last two methods, the weights of non-seed pixels belonging to the foreground/background depend on the color distribution of part-based seeds (the green rectangles in Fig.1(b-e)) and the background seed pixels, while no such bias is used in the first method. While using shape priors, the weights of pixels belonging to the foreground also depend on the shape (along with the color distribution of seeds) projected onto the detection box. Finally, we solve multi-parametric graph-cuts to generate segmentation proposals based on these seeds.. An overview of our approach is shown in Fig. 1.

Experimental results on the PASCAL2010, 2012 datasets [9, 10] show that our approach outperforms state-of-the-art methods. Furthermore, we show how the three different seeding methods (shown in Fig.1 (b-d)) improve segmentation accuracy in a complementary way. Finally, we illustrate the effect of the number of part-based seeds on the quality of segmentation.

The contributions of our approach are as follows:

1. We combine grid based seeds with high level detection based seeds by adding information obtained from high scoring parts in DPM. Therefore, our method leverages both low-level and high-level cues.

2. Using detection-based seeds, we can capture the multi-modal color distribution of the object and enforce a spatial constraint on the size and pose of the segmentation hypothesis. A category dependent shape prior further enhances segmentation. A statistical evaluation quantifying the efficacy of each of the priors introduced is presented.

3. Experimental results on challenging image segmentation datasets show that the proposed method is superior to the state-of-the-art methods as measured by three different evaluation metrics.

2 Related work

Current algorithms providing a single bottom-up segmentation [11–14] are not reliable. A common and highly successful approach that offers improvement over them is to generate a large set of segmentation hypotheses by using multi-parametric graph-cuts [4, 6]. CPMC [4] generates segmentation hypotheses by sampling points on a grid using a rectangular basis which are used to seed the

foreground color model for segmentation. The border of the image is used to seed the background, and a pixel-wise segmentation is generated with graph-cuts over simple color cues. Object proposals [6] uses a similar pipeline, but chooses seeds from a hierarchical segmentation, and learns an affinity measure between superpixels. However, these methods rely heavily on local bottom-up cues (e.g. color, texture, contour strength). As a result, their performance deteriorates in situations where color consistency, contour information, etc. are insufficient to form a good segment. Moreover, seed positioning plays an important role in graph-cut based methods, but their seed generation schemes are not informed by high-level information. They often result in mixed or under-sampled color and texture distributions for the foreground and background. In contrast, we propose detection-based seed generation strategies based on deformable part-based models (DPM), which improve the segmentation accuracy by incorporating high-level cues in the seed generation process.

Several recent methods have attempted to obtain more accurate segmentation given detection bounding boxes [15–17, 7, 18, 19]. However, they [16, 17, 15] trust the class assignments of detection, which makes them inaccurate in situations where the detection is inaccurate. In contrast, PSPGC uses detection priors along with low-level grid based seeds, so even when the detector fails, the performance of PSPGC does not drop significantly. Further, most of these approaches [19, 15, 7, 18] assume the detection bounding boxes are accurate. Therefore, when the detection bounding box is inaccurate, for example, the bounding box only covers a portion of the object, these methods only segment parts of the object inside the bounding box. Unlike these methods, PSPGC not only uses the bounding box to generate background seeds, but also takes advantages of the part-based model from DPM to generate foreground seeds. These foreground seeds provide a partial spatial and color distribution of the foreground, which results in a more accurate segmentation than just using the bounding box.

Also, our work is related to methods leveraging shape priors by transfer-based approaches. These approaches match regions in the test image with similar regions in the training examples using k Nearest Neighbors or SVM, and project a shape mask over the object. Some transfer-based methods [20, 7] first detect objects and then project an average mask of the training examples onto the detected object using a linear/non-linear transformation. Transferring category independent shape masks by matching regions in an image with training examples has proved effective in generating a pool of segments [5]. In addition to the above methods, category-specific shape priors from a mixture model of deformable parts have been combined with bottom-up cues [17] for segmenting objects. Inspired by SCALPEL [7], we use a class- and size-specific method to form the shape priors, which affect the probability of a pixel belonging to the foreground. We should also note that part-based seeds play a complementary role to shape priors, since when using a global shape prior it is hard to account for articulations and deformations present in an object class. However, in PSPGC, seeds generated using part scores from DPM are robust to such changes, hence can handle situations where global shape priors fail.

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

Fig. 2. In Image (a), the foreground seeds chosen by the ranker are displayed in green. The foreground unary potentials when 1, 2 and 3 seeds are selected as foreground are shown in figure (b), (c) and (d) respectively.

3 PSPGC

Given an image and a set of object categories for which DPM models have been trained, our goal is to generate a set of segmentation hypotheses such that the overlap between one of the hypothesis and each of the objects present in the image is high. To this end, we extend the existing framework of Constrained Parametric Min-Cuts (CPMC) [4] by incorporating seeds based on detections from DPM and shape priors. In order to explain our framework, we give a brief overview of CPMC.

3.1 Constrained Parametric Min-Cuts

CPMC solves multiple min cut problems with different seeds and unary terms. The selection of foreground seeds is done by placing a rectangular grid over the image and sampling pixels around the points on the grid using a rectangular basis. Frame pixels are set as background seeds. The unary cost for seed pixels is set to infinity while the unary cost for non-seed pixels is based on the color distributions of the foreground and background seeds. The pairwise potential depends on the contours obtained from the multi-cue contour detector globalPb [21]. Multiple min-cut problems are solved by varying the degree of foreground bias to generate different segmentation hypotheses. Once the segmentation hypotheses are generated, segments of very small size and those which have a high degree of similarity are discarded in a fast rejection step. Finally, the segments are ranked by graph, region and Gestalt properties.

Although CPMC is quite successful in generating a good pool of segments, the final set of segmentation hypotheses is biased towards the initial seeds. Due to its bottom up category independent approach, it is not able to utilize higher level features (for example, location of parts or shape of an object) present in the image. To this end, we augment CPMC with detection-based seeds that capture the multi-modal color distribution of the objects in the image, enforce a spatial

constraint on the size of the segmentation hypothesis and a category dependent shape prior for better localization.

3.2 Segmentation Prior using Detection Based Seeds

Given an image $I(\mathcal{V}) \rightarrow R^3$, defined over a set of pixels $\mathcal{V}$, a 4 connected weighted grid graph $G = (\mathcal{V}, \mathcal{E})$ is constructed, where edge weights quantify the similarity between neighboring pixels. In order to generate a binary partition of the image, two nodes s and t are added to $\mathcal{V}$. These nodes represent the foreground and background labels respectively, which are connected to all other nodes in the graph. The weights corresponding to these edges represent the unary cost of assigning each pixel as a foreground or a background pixel. Given foreground and background seed pixels $\mathcal{V}_f$ and $\mathcal{V}_b$, the aim is to make a label assignment $\{x_1, ..., x_n\}$, $x_i \in \{0, 1\}$, where n is the total number of pixels in the image, such that the following energy function is minimized,

$$E^{\lambda}(X) = \sum_{u \in \mathcal{V}} D_{\lambda}(x_u) + \sum_{(u,v) \in \mathcal{E}} V_{uv}(x_u, x_v)$$

where $\lambda \in \mathbb{R}$. The unary potential is defined as follows:

$$D_{\lambda}(x_u) = \begin{cases} 0 & \text{if } \; x_u = 1, u \notin \mathcal{V}_b \\ \infty & \text{if } \; x_u = 1, u \in \mathcal{V}_b \\ \infty & \text{if } \; x_u = 0, u \in \mathcal{V}_f \\ f(x_u) + \lambda & \text{if } \; x_u = 0, u \notin \mathcal{V}_f \end{cases}$$

The function $f(x_u)$ is either set to 0 or is computed as $\ln p_f(x_u) - \ln p_b(x_u)$, where p_f and p_b are estimates of the RGB color distributions of the foreground and background respectively. The pairwise term, V_{uv}, is set to zero if the labels are same otherwise the cost is defined as $\exp\left[-\frac{max(gPb(u), gPb(v))}{\sigma^2}\right]$, where gPb is the response of the contour detector globalPb at each pixel, while σ is a constant.

The result of graph-cut based segmentation as formulated above heavily depends on the placement of seeds. The color distribution of the seeds directly affects the unary cost throughout the graph. As is evident in Fig. 2, the unary cost which is computed using multiple seeds placed over different parts of an object results in a much better prior about the foreground object.

Part-Based Seeds: In order to localize objects, we run discriminatively trained part-based models (DPM) [8] for each class in the training set. DPM not only provides us a bounding box on the detected objects, but also scores corresponding to parts of an object. Since the distributions of the positioning of parts are learned while training, the model accounts for deformation and articulation which may be present in the test image. In our framework, we rank the part scores based on the filter response. Finally, the top k non-overlapping parts lying inside the image are selected and a rectangular basis around the center of these parts is used to sample the foreground seed pixels. Below, we describe

three different strategies adopted to select the background seeds and determine weights for non-seed pixels.

Frame Seeds (FS): In the first method, the frame pixels are set as background seeds. No information about the bounding box or the color distribution of the foreground or the background is used, i.e., $f(x_u)$ is set to 0 resulting in a uniform foreground bias. The part based seeds provide a spatial prior about the location of different parts of an object. Even though the detection bounding box may be inaccurate (e.g., covering a portion of the object), they are likely to cover a reasonable portion of the foreground, which can result in an accurate segmentation.

Bounding Box Seeds + Color Model (BBSC): In the second method, pixels lying on the frame of the detection bounding box are used as background seeds. Here, we do add a foreground bias for non-seed pixels and $f(x_u)$ is computed as $\ln\, p_f(x_u) - \ln\, p_b(x_u)$. Since the background seeds are hard, the detection bounding box also enforces a size constraint on the object. Whenever detection is more accurate or the color distribution of the object is multi-modal (as in Fig. 2), these seeds provide a better segmentation than the previous ones.

Shape Prior + Shape-Based Seeds: In the final method, the average shape mask is projected on the bounding box and is first thresholdeded to create a silhouette. Finally, pixels which lie at a constant distance d from the silhouette are assigned as background seeds. An additional term depending on the intensity of the average mask projected on the image is added to the unary cost for non-seed pixels, i.e., $f(x_u)$ is computed as $\ln\, p_f(x_u) - \ln\, p_b(x_u) + c * (S(x_u))$, where S is the shape mask projected onto the image and c is a positive constant. The process for creating shape masks and predicting the correct mask for a detection bounding box is described in the next section.

3.3 Shape Prior

For the creation of shape masks, we build on SCALPEL [7]. First, for each category, silhouettes obtained from the training data are clustered based on the aspect ratio of their bounding boxes using k-means clustering. Silhouettes in each aspect ratio are further clustered again to account for pose variability present in each class. An average mask of all images within a cluster is constructed for use as a shape prior. Clusters below a certain cardinality are discarded. We employ a different strategy than SCALPEL to project the shape masks. Once we have a mapping between images and cluster ids, the bounding boxes corresponding to the silhouettes are extracted and HoG is computed over the window. We learn an SVM on these HoG features with cluster ids as labels. Given a detection box in a test image, HoG is computed over the detection window. The nearest aspect ratio is chosen and the shape prior to be used is predicted by the SVM learnt for that aspect ratio.

4 Experiments

The experiments have three goals: (1) to demonstrate the effectiveness of our approach by comparison with other state-of-the-art approaches, (2) to evaluate the impact of each seed generation strategy in our method by showing the accuracy gain for each strategy for objects of different sizes, and (3) to study the performance of our method when different numbers of parts are used.

Dataset: We use the PASCAL2010 segmentation training dataset to train our method. This training dataset has pixel-level annotations for 2,075 objects in 964 images from 20 classes. Our approach is evaluated on the PASCAL2010 segmentation validation dataset, which contains 2,128 objects in 964 images. A comparison of the proposed approach with other algorithms which have presented results on the validation set of PASCAL2010 is shown. Additionally, we also test our method on the PASCAL2012 dataset (3,422 objects in 1,449 images), and compare the results with other methods. Note that for both datasets, we use the PASCAL2010 training dataset to train our method.

Implementation details: In our implementation, we use Version 5 (Sept. 5, 2012) of discriminatively trained deformable part models (DPM) [22], to train a detector for every class on the train set of the detection challenge in the PASCAL 2010 dataset. We sample part-based seed regions with 3 values of k - 1, 2 and 3. For the maximum k (3 in our case), we also add one seed by sampling pixels on the clique connecting the center of the parts, which improves segmentation for small objects. The detection threshold in DPM for each class is chosen such that 1.75 times the total number of objects present in the training images are detected. For clustering of shapes, we choose 4 aspect ratios and the number of clusters for each aspect ratio is set to 8. Our method takes about 4 minutes to generate segmentation proposals per image: 1 min for detection + 3 mins for graph-cuts with an unoptimized MATLAB code running on a 64-bit 2.2Ghz single core Linux machine. The running time for detection can be reduced to 10 seconds by running detectors in parallel on a 6 core processor. So the runtime of PSPGC is comparable with CPMC which takes 3 minutes. SCALPEL's runtime was reported as 2.5-4.5 minutes on a 2.8GHz machine. Shape sharing takes 7-8 minutes on our 2.2GHz machine. Moreover, all methods use the globalPb contour detector as a preprocessing step which takes 4-5 minutes to run. In short, every method takes around 7-12 minutes in total and we believe that addition of 1 minute (or 10 seconds in parallel) would not be a large computational overhead.

Evaluation: To evaluate segmentation quality, we use three metrics: *IoU* [7], *covering* [4, 21, 5], and *recall as a function of overlap* [6].

• IoU: For each proposed segment and ground truth object, the overlap score is computed, which is the sum of the intersection of the two masks divided by their union (IoU). To evaluate a pool of segments with respect to a given object, we report the best IoU across all segments.

• Covering: For a given pool of segments and objects, the covering metric is the average best overlapping score between ground-truth and proposed segments, weighted by the size of each object. Since covering penalizes incorrect segmen-

Method	Covering(%)	IoU(%)	Recall(%)	Num Segments
FS	84.20	74.51	83.88	707
FS + BBSC	84.97	75.35	84.16	756
PSPGC	**85.21**	**75.70**	**84.67**	788
Object Proposals* [6]	82.8	71.2	82.5	650
CPMC [4]	83.01	72.37	81.39	643
Shape Sharing [5]	83.69	70.9	78.26	1132
Shape Sharing* [5]	84.3	-	-	1448
SCALPEL [7]	83.09	73.77	83.46	658
SCALPEL* [7]	84.4	-	-	1456

Table 1. Segmentation results on PASCAL2010 validation set. Recall is computed at 50 percent overlap. The last column presents average number of segments generated per image. The first three rows show the results obtained by different strategies of PSPGC. * Results reported in SCALPEL [7].

tation of large objects greater than small objects, it is a good complementary metric to IoU for evaluation of segmentation methods.

• Recall as a function of overlap: We calculate the percentage of objects recalled at a given overlap score in order to evaluate the overall quality of proposals.

4.1 Segmentation Pool Quality

To compare our approach with existing approaches, we run the publicly available implementation of CPMC [4], Shape Sharing [5], and SCALPEL [7] with their default parameters on the PASCAL2010 validation dataset. We also provide a comparison with the published results of Object proposals [6] mentioned in SCALPEL [7]. Table 1 shows that our approach outperforms all existing methods in all 3 metrics. Note that to study the effect of different seed generation strategies in PSPGC, we list the results when only using parts of PSPGC: FS (frame seeds), FS + BBSC (bounding box background seeds + color model), along with the full model, i.e., PSPGC. From Table 1, we make the following observations:

1. PSPGC outperforms other state-of-the-art approaches on IoU, covering, and recall. Since it is based on CPMC, we find that we improve the accuracy of segmentation of CPMC significantly by only adding a small number of hypotheses. Shape sharing has reported a covering of 84.3%, however the number of segments generated were 1448. Similarly, SCALPEL has reported a covering score of 84.4% with 1456 segments. It is to be noted that PSPGC generates a comparable number of segmentation proposals to CPMC, and much fewer than several methods like SCALPEL and Shape Sharing, while providing a better covering score with only half the segments. Further, PSPGC provides a better overlap than CPMC for 870 out of 2128 segments in the dataset, which implies that more that 40% of the time, the best segment in the generated pool comes from the seeds added by PSPGC.

	CPMC(%)	SCALPEL (%)	Shape Sharing(%)	PSPGC(%)	Increase (%)
Aeroplane	83.04	80.86	78.64	**83.90**	0.86
Bicycle	48.29	**50.93**	42.71	50.53	2.24
Bird	85.64	83.601	84.65	**86.46**	0.82
Boat	73.30	76.06	72.26	**76.15**	2.85
Bottle	73.09	78.08	69.05	**79.06**	5.97
Bus	77.14	**82.39**	79.08	79.64	2.50
Car	54.75	62.71	51.66	**63.48**	8.53
Cat	89.72	87.94	90.16	**90.46**	0.74
Chair	69.34	66.13	66.18	**69.97**	0.63
Cow	85.22	84.54	86.19	**87.11**	1.89
Table	76.90	76.98	**82.04**	76.56	-0.34
Dog	87.66	87.63	86.59	**88.26**	0.6
Horse	79.39	78.67	79.44	**80.94**	1.55
Mobike	73.96	76.93	76.71	**77.98**	4.02
Person	64.11	66.65	62.36	**70.14**	6.03
Plant	66.32	67.18	66.87	**69.42**	3.10
Sheep	68.33	**74.45**	66.87	71.64	3.31
Sofa	85.44	80.50	**86.80**	85.69	0.25
Train	83.07	82.76	84.84	**85.24**	2.17
Monitor	82.04	80.61	80.79	**84.09**	2.05

Table 2. Segmentation results on PASCAL2010 validation set. The average IoU for each class is reported. The increase in percentage (by PSPGC) is measured over CPMC.

2. PSPGC, to the best of our knowledge, is the first method after CPMC which improves the state of the art in segment pool generation significantly and consistently over all metrics. SCALPEL reported an increase in IoU, however covering improved only by 0.1%; Shape Sharing improved covering, while IoU and recall dropped. We show an improvement of 3.3% in IoU, 3.3% in recall and 2.2% in covering over CPMC on the same dataset (PASCAL2010).

3. While comparing the accuracy when utilizing different strategies of PSPGC (i.e., the results in the first three rows in Table 1), we find that by only adding 50 more segmentation hypotheses, the location information of part-based seeds can improve the segmentation accuracy significantly. Furthermore, adding almost the same number of hypotheses, by leveraging the bounding box and color model, we obtain a considerable improvement in all three metrics. However, with the addition of the shape prior, accuracy does not improve significantly. It is to be noted that small errors in the positioning of the seeds does not affect the unary cost of non-seed pixels; however using a global prior like the shape of an object is sensitive to translation and its projection needs to be accurate. We will further discuss the effects of these parts of the algorithm in Section 4.3 for objects of different sizes.

Additionally, Table 2 shows IoU for different methods for each category in the dataset. PSPGC obtains the best score in 15 out of the 20 categories. It is evident that we obtain high gains for categories in which clear distinctions can be

Method	Covering(%)	IoU(%)	Recall(%)	Num Segments
PSPGC	84.74	74.12	82.74	791
CPMC [4]	82.51	70.48	78.82	646

Table 3. Segmentation results on PASCAL2012 validation set. Recall is computed at 50 percent overlap. The last column presents average number of segments generated per image.

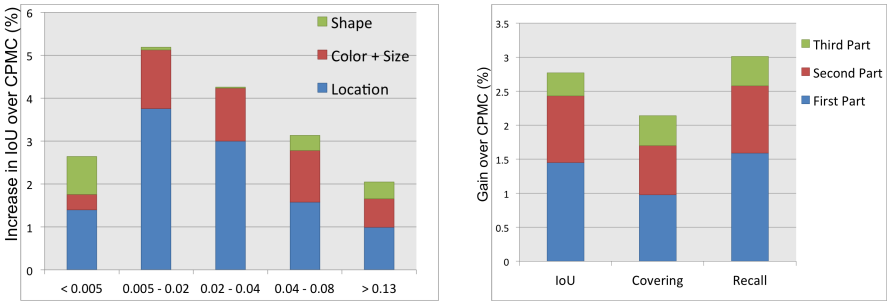
made about parts of an object. PSPGC obtains significant improvement (>6%) for humans over CPMC, in which there is intraclass variation in the form of deformation and articulation between parts. Further, the color distribution in the case of humans is multi-modal, which can be captured by placing multiple seeds at appropriate positions. It also outperforms every other method for animals, which are likely to have significant deformation and intraclass color variations. We also obtain a noticable improvement in the case of rigid objects in which parts can be distinctly identified like bikes, bottle, bus, car, motorbike, potted plant.

It is likely that the detector might get confused because many groups of categories like animals or vehicles share significant visual properties with each other. However, in our method, even though the detection category is inaccurate, the positioning of the detected parts can provide sufficient information about the foreground. For example, in the sixth row of Fig. 5, even though the sofa was detected as a chair, the positioning and color distribution of the seeds provided significant information to improve segmentation.

In order to prove the robustness of PSPGC, we also show results on the PASCAL2012 validation dataset. The results are summarized in Table 3. PSPGC was run with the same parameters on the PASCAL2012 dataset as used in PASCAL2010. We note that the improvement in performance is 3.6% in average overlap, 3.9% in recall and 2.25% in covering.

4.2 Impact of Different Seed Generation Strategies in PSPGC

In Fig. 3 (a), we present a graph to analyze the effectiveness of different seed generation strategies of PSPGC. This figure shows the gain over CPMC when different seed generation strategies of our algorithm are applied. Additionally, it can be seen that correct localization of seeds plays a very important role in segmentation, irrespective of the size of the object. For very small objects ($<$ 800 pixels), the improvement by estimating the color distribution from the seeds for the foreground and background is not very substantial, because these objects are very small and there are not sufficient statistics available to estimate the foreground color distribution. However, shape is quite useful in improving the quality of segmentation for very small objects. As the size of the object increases, the color distribution about the foreground/background object helps significantly to improve the segmentation accuracy. Moreover, detection accuracy improves when objects are of reasonable size and the size constraint enforced by



(a) (b)

Fig. 3. (a) Gain in IoU (in percentage) due to part localization, color + size and shape for objects of different sizes is shown. The x-axis denotes the size of the object relative to the image size. The range of different sizes was chosen such that each bin contains approximately the same number of objects. (b) Gain in accuracy (in percentage) over CPMC on different metrics as we increase number of parts. No shape information is used when we calculate these metrics.

the bounding box becomes more reliable. We observe that the improvement due to addition of part-based seeds is largest in the case of mid-sized objects, where there is maximum scope of improvement. We attribute this to the fact that there is not much part information in small objects, while CPMC is reasonably successful in computing the correct segmentation of large objects. Since FS and BBSC also add a spatial prior about the object and performance is measured after their addition, the effect of shape is not very evident in the combined results presented. However, only using the seeds generated by shape priors (without FS and BBSC), gives the following results: 84.4% covering (+1.4%), 74.1% IoU (+1.7%) and 82.9% recall (+1.5%) with 692 segments, which is a noticeable improvement over CPMC. Moreover, the number of objects in articulated and deformable classes outnumber objects in rigid classes in PASCAL (person alone comprising of 40% of all segments). It can be seen in the results shown in the supplementary material, that shape prior helps significantly in rigid classes like Monitor, Plant, Sofa, Sheep, Motorbike, Boat and Car.

4.3 Effect of Number of Part-Based Seeds Used

We also analyze the effect of the number of part-based seeds used in PSPGC on different metrics. In this study, we do not use any prior shape information, since we aim to evaluate the effect of part-based information. From Fig. 3(b), it can be observed that we get a big improvement for the first added part because new detection based background and foreground seeds are added for the first part, while in the other two cases, only the foreground seeds are changed. We notice that although the improvement in IoU is not significant for the third added part, it helps more in improving covering. This demonstrates that more part-based seeds are helpful in segmenting larger objects.

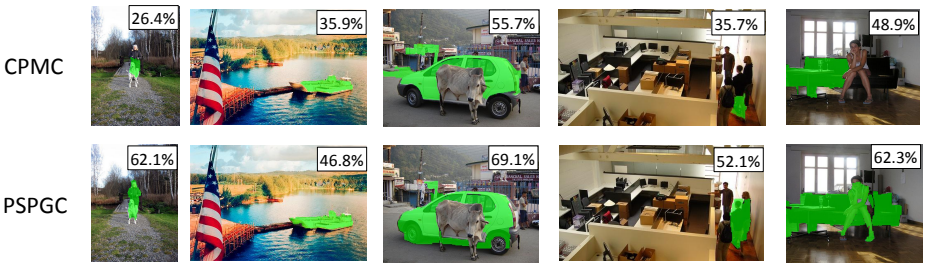


Fig. 4. Failure cases of CPMC (first row) and PSPGC (second row). We can observe that even though PSPGC is better than CPMC, there is still a scope of improvement when objects are occluded, or when they are near to each other.

5 Discussion

PSPGC combines grid based seeds with three higher level detection-based seed generation strategies, which help to capture more information about the location, color, size and shape of an object. Quantitative and qualitative results on challenging datasets show that these seed generation methods improve the quality of segmentation when measured by IoU, covering, and recall.

Among three seed generation strategies, the positioning of seeds plays the most important role in improving the segmentation accuracy regardless of the size of objects. Color information helps when sufficient statistics are available. Shape prior is helpful in segmenting rigid objects. Since we observe that seeds play an important role in the segmentation process, a promising step would be to incorporate relationships between objects in the seed generation process so that segmentation is more accurate in occluded situations.

References

1. Arbeláez, P., Hariharan, B., Gu, C., Gupta, S., Bourdev, L., Malik, J.: Semantic segmentation using regions and parts. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2012) 3378–3385
2. Carreira, J., Caseiro, R., Batista, J., Sminchisescu, C.: Semantic segmentation with second-order pooling. In: European Conference on Computer Vision–ECCV 2012. Springer (2012) 430–443
3. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv preprint arXiv:1311.2524 (2013)
4. Carreira, J., Sminchisescu, C.: Cpmc: Automatic object segmentation using constrained parametric min-cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence **34** (2012) 1312–1328
5. Kim, J., Grauman, K.: Shape sharing for object segmentation. In: European Conference on Computer Vision–ECCV 2012. Springer (2012) 444–458
6. Endres, I., Hoiem, D.: Category independent object proposals. In: European Conference on Computer Vision–ECCV 2010. Springer (2010) 575–588

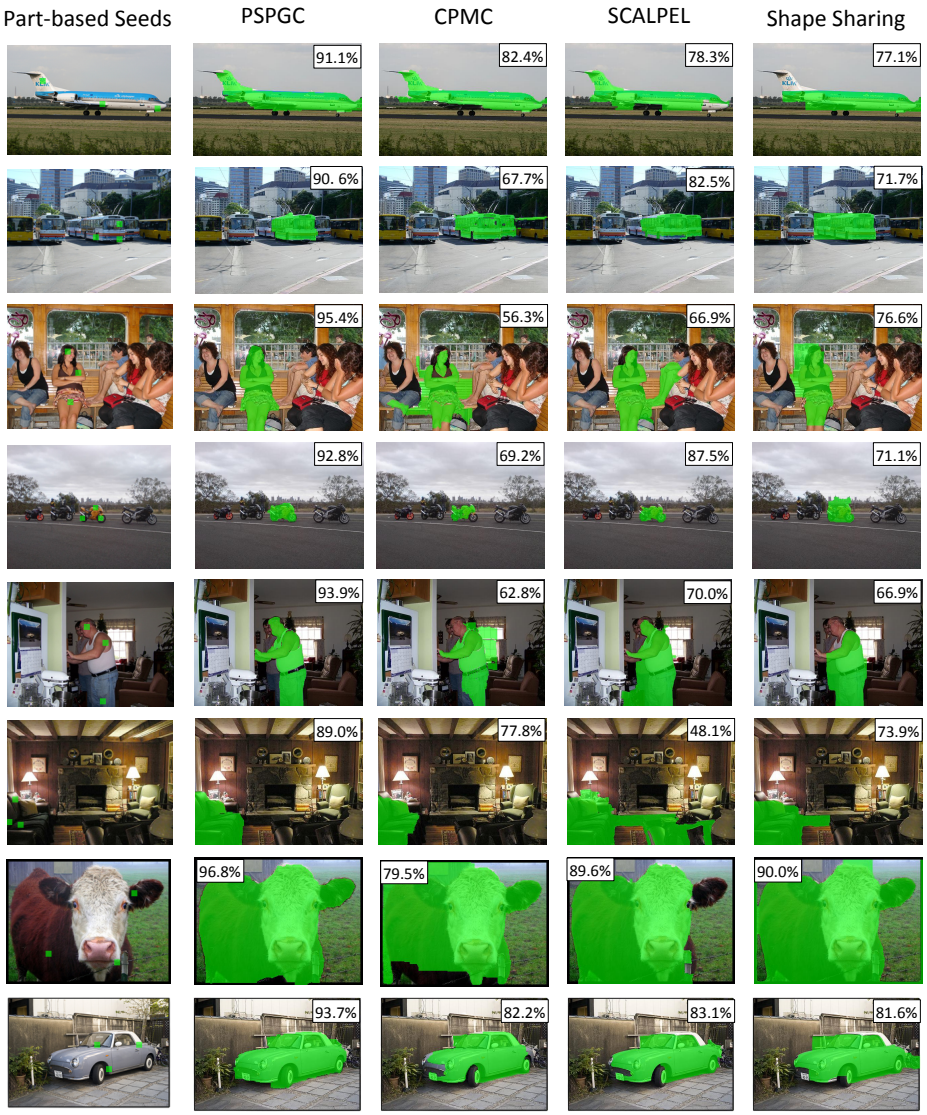


Fig. 5. Comparison among PSPGC (second column), CPMC [4](third column), SCALPEL [7] (forth column), and Shape Sharing [5] (fifth column) is shown, the first column presents the seeds used by PSPGC (marked as green rectangles). In PSPGC, since the placement of foreground seeds is appropriate, they provide a good estimate about the color distribution of parts of an object and their location, which results in a better segmentation.

7. Weiss, D., Taskar, B.: Scalpel: Segmentation cascades with localized priors and efficient learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2013) 2035–2042
8. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence **32** (2010) 1627–1645
9. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. (http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html)
10. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. (http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html)
11. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 888–905
12. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. International Journal of Computer Vision **59** (2004) 167–181
13. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on. Volume 1., IEEE (2001) 105–112
14. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. In: ACM Transactions on Graphics (TOG). Volume 23., ACM (2004) 309–314
15. Xia, W., Song, Z., Feng, J., Cheong, L.F., Yan, S.: Segmentation over detection by coupled global and local sparse representations. In: European Conference on Computer Vision–ECCV 2012. Springer (2012) 662–675
16. Dai, Q., Hoiem, D.: Learning to localize detected objects. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2012) 3322–3329
17. Yang, Y., Hallman, S., Ramanan, D., Fowlkes, C.C.: Layered object models for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **34** (2012) 1731–1743
18. Brox, T., Bourdev, L., Maji, S., Malik, J.: Object segmentation by alignment of poselet activations to image contours. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2011) 2225–2232
19. Lempitsky, V., Kohli, P., Rother, C., Sharp, T.: Image segmentation with a bounding box prior. In: IEEE 12th International Conference on Computer Vision. (2009) 277–284
20. Gu, C., Arbeláez, P., Lin, Y., Yu, K., Malik, J.: Multi-component models for object detection. In: European Conference on Computer Vision–ECCV 2012. Springer (2012) 445–458
21. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **33** (2011) 898–916
22. Girshick, R.B., Felzenszwalb, P.F., McAllester, D.: Discriminatively trained deformable part models, release 5. (`http://people.cs.uchicago.edu/~rbg/latent-release5/`)