Boundary Perception Guidance: A Scribble-Supervised Semantic Segmentation Approach

Bin Wang[1,3], **Guojun Qi**[2], **Sheng Tang**[1*], **Tianzhu Zhang**[4], **Yunchao Wei**[5], **Linghui Li**[1,3] and **Yongdong Zhang**[1]
[1]Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, China
[2]Huawei Cloud, China
[3]University of the Chinese Academy of Sciences, China
[4]University of Science and Technology of China, China
[5]University of Technology Sydney, Australia
{wangbin1, ts, lilinghui, zhyd}@ict.ac.cn, guojunq@gmail.com,
tzzhang@ustc.edu.cn, yunchao.wei@uts.edu.au

Abstract

Semantic segmentation suffers from the fact that densely annotated masks are expensive to obtain. To tackle this problem, we aim at learning to segment by only leveraging scribbles that are much easier to collect for supervision. To fully explore the limited pixel-level annotations from scribbles, we present a novel Boundary Perception Guidance (BPG) approach, which consists of two basic components, i.e. prediction refinement and boundary regression. Specifically, the prediction refinement progressively makes a better segmentation by adopting an iterative upsampling and a semantic feature enhancement strategy. In the boundary regression, we employ class-agnostic edge maps for supervision to effectively guide the segmentation network in localizing the boundaries between different semantic regions, leading to producing fine-grained representation of feature maps for semantic segmentation. Experimental results on the PASCAL VOC 2012 demonstrate the proposed BPG achieves mIoU of 73.2% without fully connected Conditional Random Field (CRF) and 76.0% with CRF, setting up the new state-of-the-art in literature.

1 Introduction

Deep learning, especially the deep Convolutional Neural Networks (CNN), greatly advances the state-of-the-art in artificial intelligence and computer vision researches in many fields such as image classification [Krizhevsky *et al.*, 2012; Simonyan and Zisserman, 2015; Szegedy *et al.*, 2015; He *et al.*, 2016; Chen *et al.*, 2018b], object detection [Ren *et al.*, 2015; Liu *et al.*, 2016; Redmon *et al.*, 2016], and semantic segmentation [Long *et al.*, 2015; Chen *et al.*, 2018a; Zhang *et al.*, 2017; Zhang *et al.*, 2019]. For the semantic segmentation task, Fully Convolutional Networks based (FCN-based) architectures [Long *et al.*, 2015; Ronneberger *et al.*, 2015; Chen *et al.*, 2015; Chen *et al.*, 2018a] have achieved competitive segmentation performance with the per-pixel mask labels. However, compared with image classification and object detection, obtaining these pixel-level mask annotations is time-consuming and expensive. In order to alleviate the dependence on expensive high-cost pixel-level labels, weakly-supervised semantic segmentation is much preferred and studied in the literature.

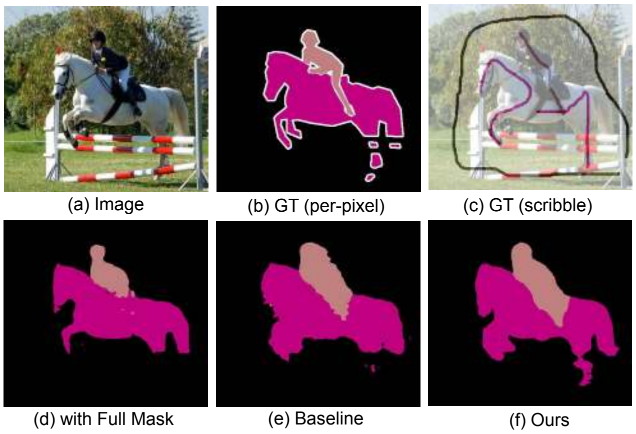


Figure 1: Illustration of a segmentation example: (a) the input image; (b) the corresponding per-pixel annotation; (c) the corresponding scribble annotation; (d) the segmentation result with per-pixel annotation and deeplab-v2 architecture; (e) the segmentation result with scribble annotation and deeplab-v2 architecture; (f) the segmentation result with scribble annotation and the proposed BPG approach.

Weak annotations for semantic segmentation can be roughly divided into the following four categories: image-level tags [Pinheiro and Collobert, 2015], clicks [Bearman *et*

[*] Corresponding author: Sheng Tang (ts@ict.ac.cn).

al., 2016], bounding boxes [Dai *et al.*, 2015] and scribbles [Lin *et al.*, 2016]. For the image-level and click tags, the extremely limited pixel-level information makes it challenging to train a high-accurate segmentation network. In contrast, scribbles and bounding boxes contain more valuable information that can train the segmentation networks more effectively. For bounding-box annotations, iterative training strategy is often adopted by combining proposal masks (e.g. MCG [Dai *et al.*, 2015; Pont-Tuset *et al.*, 2017]) to update the corresponding segmentation masks. These methods improve the precision of the semantic segmentation by using graph algorithms and region proposal methods. However, inaccurate intermediate proposal masks could mislead training by applying the cross entropy loss to uncertain segmentation in bounding boxes. In this paper, we instead choose scribble labels as the weak annotations to train the segmentation network.

For scribble-based weakly-supervised methods, ScribbleSup [Lin *et al.*, 2016] uses an iterative method to update the segmentation mask via graph cuts. Tang *et al.* [Tang *et al.*, 2018a; Tang *et al.*, 2018b] improve the segmentation performance by designing several useful regularized losses. Nevertheless, these methods do not fully explore the characteristics of the scribble annotations and do not take into account of the network structure design to improve the segmentation performance. We find that the scribble annotations can be used as a kind of supervised information to train a model which can segment different objects roughly. In the meanwhile, the edges, which indicate the information of boundaries among semantics, lead the network to grow/shrink the semantic regions, so as to revise the segmentation boundaries. Motivated by this observation, we design a novel network architecture to take advantage of both the scribble annotations and the edges.

As illustrated in Figure 1, scribble annotations are simply drawn in few strokes to tag a small part of the object or the background. The model naively trained by scribbles only produces coarse segmentation results (Figure 1-e). This is mainly because scribbles only contain partial semantic information and no fine-grained boundary is provided to guide the model to accurately segment each object. To this end, we design a novel segmentation model, namely Boundary Perception Guidance (BPG), to effectively leverage weakly supervised segmentation in scribbles by incorporating edge structures from images. The experiments demonstrate that the resulting architecture can produce more accurate segmentation results with clearer boundaries in an unprecedented resolution (Figure 1-f).

Our main contributions are summarized below.

- We propose a novel BPG model to address the scribble-based weakly-supervised semantic segmentation task. The BPG model includes two components: 1) The Prediction Refinement Network (PRN), which combines both the high semantic information and low edge/texture information to produce the fine-grained feature maps by an iterative upsampling strategy instead of a brute-force 8× upsampling operation directly; 2) The Boundary Regression Network (BRN), which guides the network to obtain clear boundaries between regions of different semantics.
- We evaluate the proposed model on the PASCAL VOC 2012 segmentation benchmark. The experiments demonstrate that the refinement sub-network and the boundary regression sub-network can improve 1.5% and 2.5% mIoU (in terms of mean Intersection-Over-Union, i.e. mIoU) respectively. The proposed components combined successfully make 3.3% mIoU improvement and set up the new state-of-the-art for scribble-based weakly-supervised semantic segmentation.

2 Related Work

Supervised semantic segmentation takes the cross-entropy term of each spatial position on the CNN output feature maps as the loss function and expensive mask-level annotations should be provided. Under this circumstance, some Weakly-Supervised Semantic Segmentation (WSSS) methods are proposed which only take some weak annotations as inputs. Training with image-level tags [Pinheiro and Collobert, 2015], clicks [Bearman *et al.*, 2016], bounding boxes [Dai *et al.*, 2015] and scribbles [Lin *et al.*, 2016], has attracted many researchers.

2.1 Image-level and Click based WSSS

Pinheiro *et al.* [Pinheiro and Collobert, 2015] propose a CNN-based model which pays more attention to pixels that are important for classifying during training. Saleh *et al.* [Saleh *et al.*, 2016] present a novel algorithm which can extract markedly more accurate masks from its own pre-trained model instead of external objectness modules. Wei *et al.* [Wei *et al.*, 2017] propose a simple to complex learning method to gradually enhance the segmentation network. In summary, image-level annotations are an order of magnitude cheaper but result in less accurate models. Bearman *et al.* [Bearman *et al.*, 2016] collect click annotation by asking the annotators to click anywhere on a target and the click annotation gives rough location information of each object which can improve the segmentation accuracy to a certain extent.

2.2 Bounding-box based WSSS

Bounding-box annotations are often used as the labels of object detection, compared to pixel-wise annotation task, the workload of labeling object locations can be 15 times less [Lin *et al.*, 2014]. Dai *et al.* [Dai *et al.*, 2015] train the segmentation network combined with region proposals and the prediction masks iteratively, this method can produce competitive segmentation results. Khoreva *et al.* [Khoreva *et al.*, 2017] propose a modified version of GrabCut (GrabCut+) to produce some mask regions to train the segmentation model iteratively with the bounding-box mask.

2.3 Scribble based WSSS

Scribble annotations can be easily drawn and a certain category is given for each scribble. Lin *et al.* [Lin *et al.*, 2016] build the scribble dataset for PASCAL VOC 2012. Combined with these scribble annotations and region proposals that are generated via graph cuts, they train the segmentation network iteratively. Tang *et al.* [Tang *et al.*, 2018a] propose a novel loss for segmentation which combines with partial cross entropy term and the normalized cut.

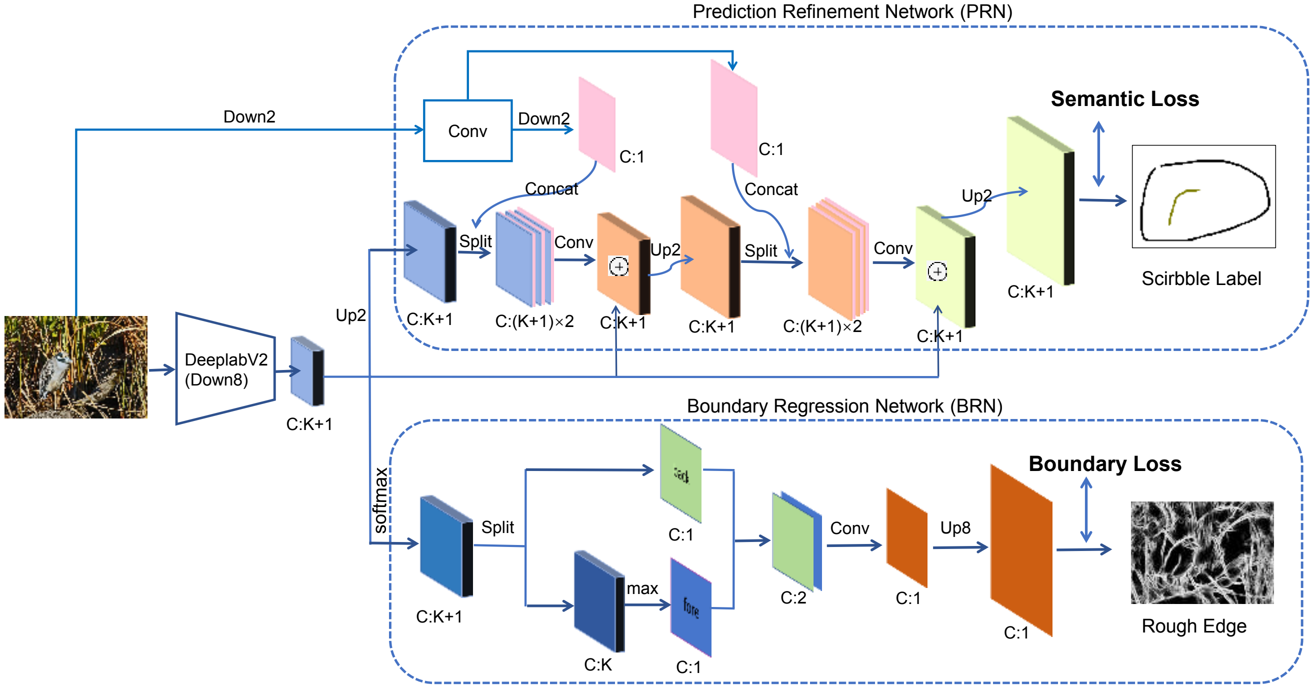


Figure 2: The segmentation architecture (BPG) proposed in this paper. The network backbone is based on deeplab-v2(ResNet-101); Two subnetworks are added into the segmentation architecture: (1) PRN which can refine the segmentation result by fusing the high-level semantic features and low-level features; (2) BRN which can guide the network to extract edge features. The "⊕" symbol means element-wise addition operation between current feature maps and the corresponding feature maps of deeplab. "C:k" means current feature maps contain k channels.

3 Approach

As illustrated in Figure 2, the segmentation architecture proposed in this paper is mainly divided into three parts. The first is a feature encoding network using the deeplab-v2 structure as its backbone. After that, instead of upsampling the feature maps with a simple direct scaling-up operation, we design two sub-network branches to improve the performance of weakly-supervised semantic segmentation: they are the Prediction Refinement Network (PRN) and Boundary Regression Network (BRN), which can greatly refine the segmentation features by combining weakly-supervised semantics in coarser scribbles with rough image edges yet in fine-grained resolution.

3.1 Prediction Refinement Network

Because of the effectiveness of the deeplab architecture [Chen *et al.*, 2015; Chen *et al.*, 2018a], many researchers take it as their backbone network for the weakly-supervised semantic segmentation task [Khoreva *et al.*, 2017; Lin *et al.*, 2016; Tang *et al.*, 2018a]. For a fair comparison with these methods, we use it as our backbone too. However, we find that this network structure has two shortcomings: 1) The last convolutional feature only contains high-level information about semantic segmentation, which is insufficient for segmenting small hard regions with fine details; 2) It directly upsamples the convolutional features by a factor of eight to predict the final pixel labels, yielding quite coarse semantic region boundaries. To address these problems, we design the PRN to produce clear boundaries for those hard regions.

As shown in the top of Figure 2, the PRN branch implements three key ideas. First, we use a shallow network to extract high-resolution convolutional features from an input image and concatenate them with each individual semantic channel to enhance features with more low-level details. Since the low-level features contain high-resolution details, the resulting representation combined with high-level semantics can enable fine-grained segmentation of different semantic regions in a high resolution. In addition, after upsampling the high-level semantic features with low-level high-resolution details, we refine the enhanced features by a series of convolution operations to produce more boundary details. Finally, we leverage the idea of residual networks in this subnetwork to accelerate the network convergence (See the "⊕" symbol in Figure 2).

Compared to some existing segmentation networks with refinement structure such as U-Net [Ronneberger *et al.*, 2015], the proposed PRN has the following merits: Firstly, we just need low-level features which pass through the given image with 5 convolutional operations; Secondly, only single-channel feature map is added to each confidence map ($K + 1$ in total). Thus, our PRN only imports small amount computations and GPU memories, which may benefit some future semantic works.

3.2 Boundary Regression Network

Per-pixel cross-entropy loss can train a satisfactory segmentation network with per-pixel annotations. However, for scribble annotation, only a few internal markers are drawn for each object and no boundary information is provided explicitly. In this setting, training a segmentation model with a standard segmentation network cannot yield sharp accurate object boundaries. To the end, we design a branch of BRN which can implicitly extract the important boundary information from readily available rough edges (We choose the HED method [Xie and Tu, 2015] as it has been used in other weakly-supervised semantic segmentation methods [Khoreva *et al.*, 2017; Cai *et al.*, 2018]).

Although the above idea is intuitive, designing an effective boundary regression model is still challenging in a weakly supervised fashion, since the edge labels produced by HED are not the true boundaries of objects that often introduces many noise edges actually belonging to the background (See Figure 2-Rough Edge).

In fact, directly using conventional $(K+1)$-channel structure (K is the category number of objects, "1" refers to the background) to predict edges not only cannot enable the network to obtain the boundary differentiation ability, but also damage the segmentation results. The reasons are as follows (experimental validations are also given in the section of experiments). The $(K+1)$-channel confidence map only contains segmentation results with rough object boundaries under the supervision of scribble-based weak annotations. In this case, if precise boundaries of each object are given to guide the boundary regression, the segmentation precision can be easily improved. However, only rough edge labels are available, and they may break up the feature maps into several false pieces to regress the noisy edges. Thus, to keep the semantic structures intact, we design the BRN structure below to eliminate the negative effect of noisy edges by dividing the $(K+1)$-channel feature maps into foreground/background channels. Only the resultant foreground channel is regressed to the noisy edges, thereby minimizing the chance of individual semantic channels being compromised with noisy edge labels.

The general structure of the network is illustrated at the bottom of Figure 2. Instead of directly using deeplab-v2 $(K+1)$-channel features to predict the boundary, we divide the features with predicted labels into foreground/background channels first and then regress them to the edge prediction map. As shown in Figure 3, when edge loss is back-propagated from the class-agnostic object confidence maps (Denote $M_{fore}, C:1$) to the semantic feature maps ($C:K$), at each location, only the element with the highest probability needs to calculate the gradient (see Eq. 1) and update related parameters. In other words, it means the other $(K-1)$ semantic feature maps will not be affected by the noisy edge loss.

$$d_{C_i} = \begin{cases} d_{out}, & C_i = max(C_1, C_2, ..., C_K) \\ 0, & others, \end{cases} \tag{1}$$

where d_{C_i} denotes the gradient of the i-th semantic feature map, d_{out} refers to the gradient of foreground feature map.

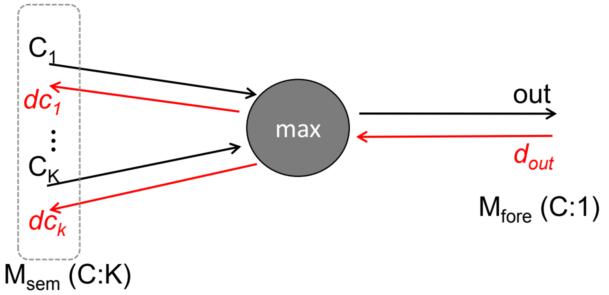


Figure 3: Forward propagation and backward propagation illustration of max pooling operation between semantic feature maps and foreground feature map: $out = max\{C_1, C_2, ..., C_k\}$.

By updating a limited number of parameters, only the boundaries around the foreground objects are learned to fit those true object boundaries in the given edge map and those noisy ones rarely have the chance to damage the K-channel semantic maps.

3.3 Loss Function

The feature decoding part of the proposed network consists of the PRN and the BRN.

The loss function to train the PRN is

$$L_{semantic} = \sum_p PCE(X_{ref\_\theta_1}(p), l_{scri}(p)), \tag{2}$$

where p indexes pixels, $X_{ref\_\theta_1}(p)$ is the predicted probability of the PRN with parameters θ_1, $l_{scri}(p)$ is the scribble mask label at pixel p, and PCE is the Partial Cross-Entropy loss [Tang *et al.*, 2018a] which only computes the loss on the labeled region.

The loss function of the BRN is

$$L_{boundary} = \sum_p MSE(X_{boundReg\_\theta_2}(p), l_{edge}(p)), \tag{3}$$

where $X_{boundReg\_\theta_2}(p)$ is the predicted probability by the BRN with parameters θ_2, $l_{edge}(p)$ is the edge label at pixel p, and MSE is the per-pixel MSE.

We train the two sub-networks end-to-end, and the total loss is

$$L_{total} = L_{semantic} + \lambda L_{boundary}, \tag{4}$$

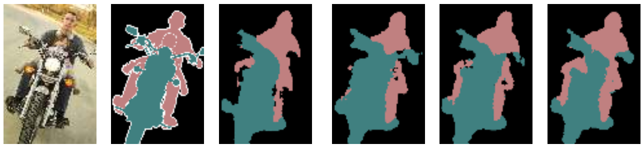
where the hyper-parameter λ balances between the two losses.

4 Experiments

We evaluate the proposed architecture on the PASCAL VOC 2012 segmentation benchmark [Everingham *et al.*, 2010], which involves 20 foreground object categories and one background class. The original dataset contains $1,461$ training images, as well as $1,449$ validation and $1,456$ testing examples, respectively. Following the evaluation protocol in the literature [Chen *et al.*, 2018a; Dai *et al.*, 2015; Lin *et al.*, 2016; Tang *et al.*, 2018a], we use the augmented dataset by the extra annotations provided by [Hariharan *et al.*, 2011], totaling $10,582$ training images. The training data are scribbles from [Lin *et al.*, 2016] for the weak supervision task.

	bkg	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	monitor	mean
deeplab-v2 [F]	93.8	87.1	39.6	89.1	60.0	80.1	94.3	86.0	90.8	37.0	86.4	47.8	87.1	87.7	84.0	85.8	65.8	83.3	46.2	87.6	73.3	75.8
deeplab-v2+crf[F]	94.0	88.0	38.6	90.1	60.0	80.5	94.4	86.1	91.9	37.2	87.2	48.0	88.4	88.8	84.7	86.4	67.9	84.0	47.2	87.4	73.7	76.4
deeplab-v2 [scri]$*$	91.1	70.9	34.2	71.7	59.3	74.4	88.4	82.5	84.4	36.2	80.7	53.2	78.0	77.1	78.7	78.3	58.7	77.5	40.7	82.7	68.7	69.9
+PRN$*$	**92.7**	72.4	34.8	74.9	60.6	74.3	**90.8**	83.8	86.4	36.9	80.7	**58.8**	81.2	79.0	79.3	79.8	59.8	77.5	**43.6**	83.7	70.1	71.4
+BRN$*$	91.9	78.0	36.6	78.0	**63.1**	**76.5**	89.5	82.4	**87.5**	35.8	**84.5**	50.9	**82.2**	80.2	80.5	**81.0**	58.0	81.6	43.1	84.3	**74.2**	72.4
BPG$*$	92.3	**80.3**	**37.6**	**79.9**	62.1	75.7	89.6	**84.0**	87.2	**37.5**	84.0	56.7	81.4	**81.4**	**81.1**	80.4	**61.5**	**84.2**	43.5	**85.0**	71.8	**73.2**
BPG+CRF	93.4	84.8	38.4	84.6	65.5	78.8	91.4	85.9	89.5	41.0	87.3	58.3	84.1	85.2	83.7	83.6	64.9	88.3	46.0	86.3	73.9	76.0

Table 1: Comparison results of different network architectures on the PASCAL VOC 2012 validation set (IoU in %). $*$ refers to scribble-based method without CRF post-processing, the best mAP for each categories among them is marked in bold.



(a) Image (b) GT (c) Deeplab-v2 (d) +PRN (e) +BRN (f) BPG

Figure 4: Weakly-Supervised semantic segmentation example: (a) the input image; (b) full mask annotation; (c) segmentation result with deeplab-v2; (d) segmentation result with [deeplab-v2 + PRN] (d) segmentation result with [deeplab-v2 + BRN]; (e) segmentation result with BPG framework [deeplab-v2 + PRN + BRN].

Methods	mAP	edge loss
wo/ BRN	71.4	n/a
w/ BRN ($C = K + 1$)	70.8	**0.026**
w/ BRN ($C = 2$)	**73.2**	0.051

Table 2: Comparison with different settings on the PASCAL VOC 2012 *val* set (mIoU in %).

4.1 Implementation Details

We re-train the ResNet101-based deeplab-v2 [He *et al.*, 2016; Chen *et al.*, 2018a] by PyTorch and take it as our baseline. The proposed weakly-supervised semantic segmentation network is simply trained on a single scale of input images. Like the setting in deeplab-v2, we employ the "poly" learning rate policy for with a mini-batch of 10 images with an initial learning rate of 0.00025. We use a momentum of 0.9 and a weight decay of 0.0005. The hyper-parameter λ in Eq. 4 is set to 1.0. We run 25 training epochs on a single NVIDIA TitanX 1080ti GPU, which takes about 10 hours on the PASCAL VOC dataset. In the testing stage, instead of using multi-scale inputs with max voting, we use the average voting over multi-scale left-right flipped inputs (i.e. $[0.5, 0.75, 1.0, 1.25]$).

Methods		wo/ CRF	w/ CRF
Supervision: Scribbles			
ScribbleSup	[Lin *et al.*, 2016]	n/a	63.1
NormalCut	[Tang *et al.*, 2018a]	72.8	74.5
KernelCut	[Tang *et al.*, 2018b]	73.0	75.0
BPG (Ours)		**73.2**	**76.0**
Supervision: Per-pixel Labels			
deeplab-v2		75.8	76.4

Table 3: Comparison with state-of-the-art methods on the PASCAL VOC 2012 *val* set (mIoU in %).

4.2 Ablation Study and Results

Ablation Study

We perform experiments on the PASCAL VOC 2012 semantic segmentation dataset with different architectures. As shown in Table 1, with deeplab-v2 architecture (ResNet101-based) alone, the model can yield 69.9% in mIoU. Adding the PRN can gain 1.5% improvement in mIoU. Combining the BRN into deeplab-v2 alone can bring 2.5% improvement. The proposed BPG framework based on the deeplab-v2 backbone and both PRN and BRN can improve the weakly-supervised segmentation performance to as high as 73.2%, which sets up the new start-of-the-art mIoU performance. From the Table 1, we also find that the segmentation performance has been improved ($1.2\% \sim 9.4\%$) on all semantic categories, which demonstrates the effectiveness of the proposed framework. Furthermore, by applying the CRF post-processing, we can achieve 76.0% in mIoU, as good as the fully supervised semantic segmentation model using per-pixel labels (76.4%) with the same CRF post-processing.

Figure 4 shows weakly-supervised segmentation examples produced by different network structures. We can see that the baseline result (deeplab-v2 only) has much coarse segmentation results, where the boundaries are not well aligned with the ground-truth counterparts. In comparison, the net-

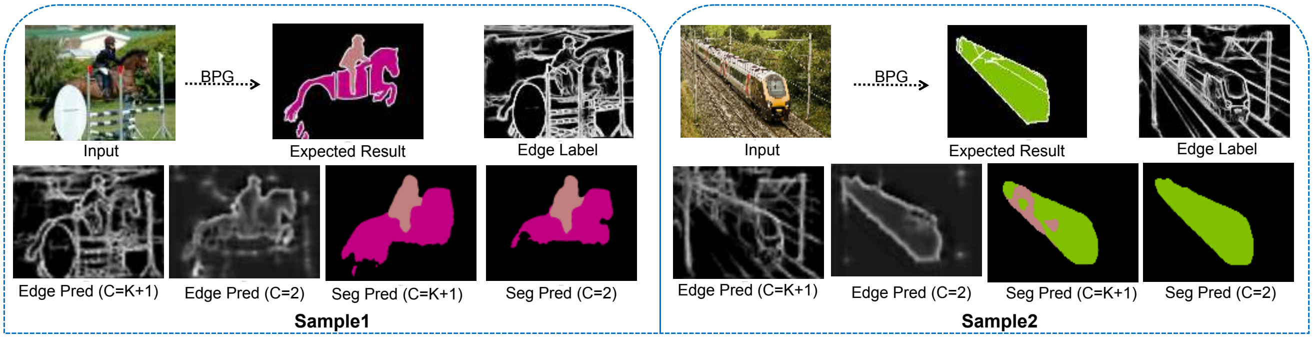


Figure 5: Segmentation predictions and edge predictions with different BRN settings: (1) Directly using K+1 feature maps following with some convolutional operations; (2) Converting K+1 feature maps to class-agnostic fg/bg maps firstly and then with the same operations.

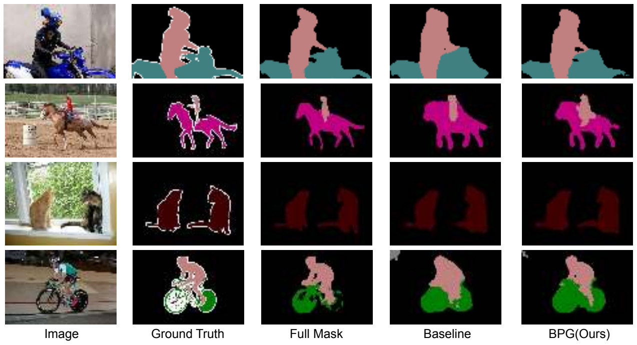


Figure 6: Visualization results on the PASCAL VOC 2012 val set: input images, ground-truths, segmentation results with deeplab-v2 (supervised/scribble-based) and segmentation results with our BPG framework before/after CRF.

work with the PRN branch gains an improved segmentation, and the segmentation result by the BRN achieves an even better result. By combining both structures, the BPG can obtain a considerable high-quality segmentation result as shown in Figure 4(f).

Boundary Regression Network Design

To evaluate the effectiveness of the proposed BRN, we conduct additional experiments to compare different settings. As shown in Table 2, when directly using $(K+1) - channel$ feature maps to regress the object boundary, the edge loss of the last iteration is only 0.026. However, the segmentation mAP drops by 0.6%, which is clearly overfitting.

In contrast, when using the proposed BRN structure (converting (K+1) semantic feature maps to fg/bg feature maps firstly), the edge loss is more than twice of the previous settings, but the segmentation precision is improved significantly (73.2% vs 71.4%). The segmentation and edge prediction results in Figure 5 give a more intuitive explanation for that. We can see from this figure, when directly using $C = K + 1$ structure, the edge predictions are close to the edge labels, but in this setting, the segmentation does not perform very well: the segmentation result of $sample1$ still has rough boundary; the segmentation result of $sample2$ contains some mistakes on semantic predictions. On the contrary, our boundary regression network can produce clear object boundaries by eliminating the negative impact of those noise edges, yielding good segmentation results with precise boundaries.

Comparisons with State-of-the-art Methods

To further analyze the segmentation performance of the proposed method, we also compare with the current state-of-the-art methods [Lin *et al.*, 2016; Tang *et al.*, 2018a]. From Table 3, we can see that the proposed BPG model is far better than the ScribbleSup baseline. NormalCut [Tang *et al.*, 2018a] and KernelCut [Tang *et al.*, 2018b] are the other two best methods at present, which also take deeplab-v2 as their network backbone. Compared with them, our model still achieves competitive segmentation result with scribble labels. ScribbleSup uses the graphical model for propagating information from scribbles. NormalCut loss and KernelCut loss are two regularized losses to train the segmentation network with scribble labels. As shown in Figure 6, the proposed BPG architecture can achieve good segmentation results with precise boundaries.

5 Conclusion

We propose the BPG model including two sub-networks to improve the segmentation network training with weakly annotations. Specifically, we first propose the branch of the PRN to refine the segmentation prediction by combining the low-level high-resolution feature map with high-level low-resolution feature maps through iterative upsampling layers. Then we introduce the branch of BRN to train the network to localize the sharp boundaries more effectively. The proposed architecture can achieve considerable better segmentation performance without per-pixel annotations. These two proposed sub-networks can be readily incorporated into any segmentation networks in the weakly-supervised task when incomplete per-pixel labels are provided.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (2017YFB1002202), and the National Natural Science Foundation of China (61572472,61871004,61525206), and 242 progject (2019A010).

References

[Bearman *et al.*, 2016] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *ECCV*, 2016.

[Cai *et al.*, 2018] Jinzheng Cai, Youbao Tang, Le Lu, Adam P Harrison, Ke Yan, Jing Xiao, Lin Yang, and Ronald M Summers. Accurate weakly-supervised deep lesion segmentation using large-scale clinical annotations: Slice-propagated 3d mask generation from 2d recist. In *MICCAI*, 2018.

[Chen *et al.*, 2015] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015.

[Chen *et al.*, 2018a] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 40(4):834–848, 2018.

[Chen *et al.*, 2018b] Long Chen, Hanwang Zhang, Jun Xiao, Wei Liu, and Shih-Fu Chang. Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In *CVPR*, 2018.

[Dai *et al.*, 2015] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, 2015.

[Everingham *et al.*, 2010] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010.

[Hariharan *et al.*, 2011] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. 2011.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[Khoreva *et al.*, 2017] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *CVPR*, 2017.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[Lin *et al.*, 2016] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, 2016.

[Liu *et al.*, 2016] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016.

[Long *et al.*, 2015] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[Pinheiro and Collobert, 2015] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015.

[Pont-Tuset *et al.*, 2017] Jordi Pont-Tuset, Pablo Arbelaez, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping for image segmentation and object proposal generation. *TPAMI*, 39(1):128–140, 2017.

[Redmon *et al.*, 2016] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[Ronneberger *et al.*, 2015] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[Saleh *et al.*, 2016] Fatemehsadat Saleh, Mohammad Sadegh Aliakbarian, Mathieu Salzmann, Lars Petersson, Stephen Gould, and Jose M Alvarez. Built-in foreground/background prior for weakly-supervised semantic segmentation. In *ECCV*, 2016.

[Simonyan and Zisserman, 2015] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[Szegedy *et al.*, 2015] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[Tang *et al.*, 2018a] Meng Tang, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers. Normalized cut loss for weakly-supervised cnn segmentation. In *CVPR*, 2018.

[Tang *et al.*, 2018b] Meng Tang, Federico Perazzi, Abdelaziz Djelouah, Ismail Ben Ayed, Christopher Schroers, and Yuri Boykov. On regularized losses for weakly-supervised cnn segmentation. In *ECCV*, 2018.

[Wei *et al.*, 2017] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Xiaohui Shen, Ming-Ming Cheng, Jiashi Feng, Yao Zhao, and Shuicheng Yan. Stc: A simple to complex framework for weakly-supervised semantic segmentation. *TPAMI*, 39(11):2314–2320, 2017.

[Xie and Tu, 2015] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *ICCV*, 2015.

[Zhang *et al.*, 2017] Rui Zhang, Sheng Tang, Yongdong Zhang, Jintao Li, and Shuicheng Yan. Scale-adaptive convolutions for scene parsing. In *ICCV*, 2017.

[Zhang *et al.*, 2019] Rui Zhang, Sheng Tang, Yongdong Zhang, Jintao Li, and Shuicheng Yan. Perspective-adaptive convolutions for scene parsing. *TPAMI*, 2019.