Weakly Supervised Semantic Segmentation: From Box to Tag and Back, Supplementary Materials

Zongliang Ji
jerryji007.github.io
Olga Veksler
https://cs.uwaterloo.ca/~oveksler/

School of Computer Science
University of Waterloo
Waterloo, Canada

1 Additional Experiments

In Table 1, we give an extended version of Table 1 of the main paper.[1] We added one new column for our method, namely column 7: **ours(all boxes)**. Now there are four different versions of our method. First we remind what the 'old' columns are. The column **ours** is training with the regularized loss in [11], column **ours(T)** is training with our version of regularized loss, with the tight box prior, column **ours(T+N)** adds all the other classes as negative samples. Finally, we test how our method performs if instead on training on 'clean' boxes, we use all boxes in the training dataset. In particular, we take all boxes even if they are small and overlap with other boxes. For consistency, we test only on clean boxes, since the method in all other columns were tested on clean boxes only. The results of training on all boxes and testing on clean boxes is in the new column **ours(all boxes)**. Note that this column is obtained with our loss function, namely the tight bounding box loss. For all four of **our** methods, we kept the $s_{min} = 0.3$ in the minimum volume loss.

Surprisingly, training on all boxes is not much worse than than training on clean boxes. The chair fares the worst by far, probably due to high confusion with the table class. Interestingly, training on all boxes gives a significantly improved performance over all other of our methods for the table class. This is most likely due to the dataset size. Class table has only 65 clean bounding boxes, and this is a very small dataset to learn the appearance from. There are 715 training boxes overall, if we do not remove overlapping boxes, thus training on all boxes for the table is advantageous, even if they have overlap with other classes. In principle, we could decide to include 'not clean' boxes in our dataset if the number of clean boxes for the class is small.

We now provide some implementation details for the results in Table 1 that we omitted from the main paper due to the lack of space. For implementing MCG, we follow the prior work [2, 5], and select the proposal that has the highest IoU measure with the bounding box. We do not use MCG ranking of proposals, since the ranking was trained with pixel-precise supervision on Pascal VOC dataset. For implementing denseCRF [3] we use the same approach as in [5]. We select $\alpha\%$ of pixels in the center of the box and label them

[1] The main paper had a shorter version due to the lack of space.

class	#boxes	MCG	denseCRF	grabcut	salient	ours (all boxes)	ours	ours(T)	ours(T+N)
aero	608	63.72	70.97	67.12	88.43	84.95	87.13	88.88	**89.12**
bike	243	58.41	69.56	69.60	80.00	84.40	84.37	85.59	**85.74**
bird	827	67.32	76.54	70.69	90.07	87.69	91.16	91.57	**91.70**
boat	444	65.73	76.68	76.58	77.57	82.89	84.91	**87.26**	87.14
bottle	376	78.61	85.19	84.97	82.50	91.52	89.28	92.45	**92.68**
bus	310	76.75	89.32	87.04	92.71	94.17	91.44	**94.73**	94.60
car	909	72.07	86.27	81.99	88.00	88.88	89.23	92.54	**92.90**
cat	901	73.86	86.49	80.44	84.62	92.55	92.69	94.71	**94.73**
chair	1003	58.73	69.41	63.48	68.28	68.72	74.71	75.99	**78.64**
cow	428	72.54	84.99	75.01	87.17	87.70	91.42	**92.56**	92.53
table	65	62.45	83.63	82.22	63.32	**85.26**	75.26	79.59	79.69
dog	953	72.58	85.95	79.40	89.25	92.68	93.98	94.36	**94.42**
horse	359	66.57	79.22	71.00	88.22	88.22	90.26	91.35	**91.53**
mbike	248	59.30	80.60	75.92	85.48	88.87	87.95	**89.61**	**89.61**
person	4029	70.02	80.01	78.29	84.60	80.56	85.83	88.02	**89.05**
plant	437	63.13	81.24	76.10	74.62	86.43	82.59	87.51	**87.69**
sheep	548	73.95	84.66	76.78	85.22	87.19	88.67	91.42	**91.68**
sofa	256	70.07	78.13	73.04	66.04	79.10	78.79	81.59	**81.99**
train	445	68.64	83.94	78.91	85.89	89.33	87.86	92.05	**92.08**
tv	455	82.09	86.96	85.00	83.52	90.16	88.18	92.20	**92.36**
mean F_β		68.83	81.00	76.68	82.28	86.56	86.79	89.20	**89.50**

Table 1: Comparison of object/background segmentation accuracy on training bounding boxes from Pascal VOC 2012 dataset using MCG [6], denseCRF [3] GrabCut [8], salient object detection [7] and four versions of our method. See text for explanation of four different versions of our method. Performance metric is F_β score (higher is better).

as preferring the object with probability p. Then we select a border of pixels outside the box to be set to the background. All other pixels are set to an unknown label. Then the unary CRF terms are set as suggested in the standard implementation package[2]. We chose the appropriate setting of α, p and denseCRF parameters on a small held-out set of training images (100 images, fully annotated).

Now we compare the performance on the final task, semantic segmentation, if we use the loss function in [10] instead of our loss function. In particular, we test the performance we get when constructing pseudo ground truth from boxes in column 8 of Table 2 instead of column 10. We use the *overlap-void* method for constructing the ground truth as described in Sec. 4.3 of the main paper and train deepLab [1] with cross-entropy for 200 epoch, with the same parameters before. The resulting *mIoU* is only 71.22, compared to 77.1 using our modified regularized loss (based on column 8 in Table 2). Such a large gap in performance is due to the accuracy of the pseudo-ground truth. We computed *mIoU* of pseudo ground truth using the results of the loss function in [10], and it is only 73.4. With our modified regularized loss, $mIoU = 79.1$. The percentage of void pixels in both cases is approximately 3.

Next we compare different training regimes for *overlap-back-void* approach, where pixels segmented as background inside any box get labeled as void, to minimize the number of object pixels that are mistakenly labeled as background. Since many pixels are labeled as void (23.53%), training benefits from including denseCRF loss from [10], in addition to the cross entropy loss. The method in [10] was specifically designed for training with the scribble form of weak supervision, so that a large portion of image pixels are labeled as void.

[2] https://github.com/lucasb-eyer/pydensecrf

method	bckg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
GCMCG(CRF) [8]	93.3	85.0	35.9	88.6	70.3	77.9	91.9	83.6	90.5	39.2	84.5	59.4	86.5	82.4	81.5	84.3	57.0	85.9	55.8	85.8	70.4	75.7
Box2Seg [8]	92.5	66.5	31.7	78.9	65.5	83.4	90.4	86.7	86.0	55.1	81.8	59.9	80.5	74.1	76.0	75.7	65.3	85.1	72.5	87.8	77.7	74.9
Box2Seg (CRF) [8]	93.3	72.4	33.0	84.2	64.9	83.5	90.9	86.7	88.7	57.2	83.6	62.5	82.6	76.8	77.0	77.8	63.3	87.2	75.1	88.3	74.1	76.4
Ours (overlap-back-void)	92.0	81.3	36.5	86.9	73.1	79.5	90.5	85.9	88.6	41.7	86.3	63.6	85.9	83.4	81.3	81.3	62.9	81.3	48.2	83.2	77.3	75.7
Ours (overlap-void)	93.1	87.8	42.1	89.3	70.9	78.0	88.6	88.0	90.4	42.1	89.4	54.6	87.9	86.0	82.6	82.6	66.1	88.7	51.7	85.9	73.4	77.1

Table 2: Per-class results on Pascal VOC 2012 validation set of our methods and prior methods that made per-class results available.

method	backbone	mIoU-Semi	mIoU-Val FullS
BCM (CRF) [9]	Resnet-101	71.6	74.5
Box2Seg (CRF)[4]	Resnet-101	>83.1	75.1
Ours (overap-void)	Resnet-101	79.8	77.8

Table 3: Semi-supervised Pascal VOC validation set results.

Adding denseCRF loss encourages a pixel with a void label to be assigned to the same label as a pixel that is similar in color and does have a ground truth label. We use the same parameter setting as in [10]. They also observe that training with just cross entropy first and then adding denseCRF loss works better. This approach also works the best for us. Training with cross-entropy alone, we get $mIoU = 71.87$. Adding denseCRF loss to cross entropy from the beginning gets $mIoU = 74.99$. Adding denseCRF loss after training with cross entropy first gives the best result, $mIoU = 75.71$. For our *overlap-void* approach, where pixels in the bounding boxes that are classified as background get labeled as background, there is no benefit in adding denseCRF loss. There are only 3.07% of pixels that are labeled as void in this case.

We also evaluated our approach in semi-supervised setting. Starting with our trained classifier in the *overlap-void* setting, we fine-tune it using pixel-precise ground truth on $1,464$ Pascal VOC training images, which is about 13% of all available training images. We train for 30 epoch using a fixed learning rate of 0.001. Table 3 summarizes our results and compares them with semi-supervised results in [9] and [4]. The last column is the result of training with full ground truth, for comparison. Note that [4] report the results of semi-supervised training with 10% and 100% of training data, therefore we report their performance for 10% of training data with pixel precise ground truth, and sign '$>$' denotes that with 13%, their performance would be better. Note that our result in semi-supervised setting is better than training with full ground truth. The results in [9] in semi-supervised setting are worse than training with full ground truth. The results in [4] are strikingly better than that of training with full ground truth. However, in [4] they do not provide any details of their supervised training (number of epochs, learning rate schedule, etc), no provide any discussion of why their training in semi-supervised setting is so much better than training with full ground truth.

Per-class result comparison with prior work that made per-class results available are in Table 2.

2 Qualitative Comparison

We first show qualitative comparison of the bounding box segmentations obtained by our approach vs. that of GrabCut and salient object detection, see Fig. 1. GrabCut tends to join

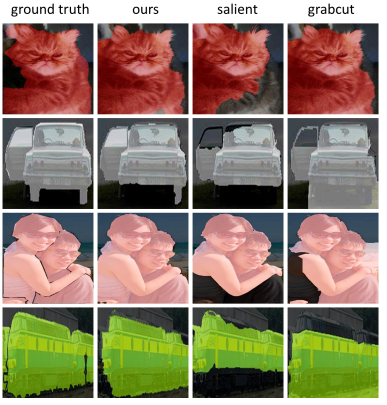

Figure 1: Examples of object/background segmentations for our method, salient object detection, and GrabCut.

pieces of the background to the object, likely due to a poor model of object appearance constructed only from one box. Salient object detection is trained on a large dataset. However, it is trained for salient object detection, and, therefore, it tends to focus on more salient object parts, such as the skin of the person or the middle part of the train. Our method learns object appearance (with weak tag supervision) for each class from a large set of class-specific samples, and therefore is able to learn a class-specific appearance model that leads to a more accurate segmentation.

We now compare our results qualitatively to the most recent prior work [2, 3]. In Fig. 2, we show segmentations on the images chosen by [3]. Our results are without any post-processing. We capture sheep legs (top row) and horse legs (middle row) better than prior work. In the last row, we have small scale human figures captured (to the right of the two main figures), while the other methods most likely smooth them out due to CRF post-processing.

In Figure 3 we show some example segmentations of our approach (*overlap-void*) on the test data.

References

[1] 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. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018.

[2] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *International Conference on Computer Vision*, pages 1635–1643, 2015.

[3] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Neural Information Processing Systems*, pages 109–117, 2011.

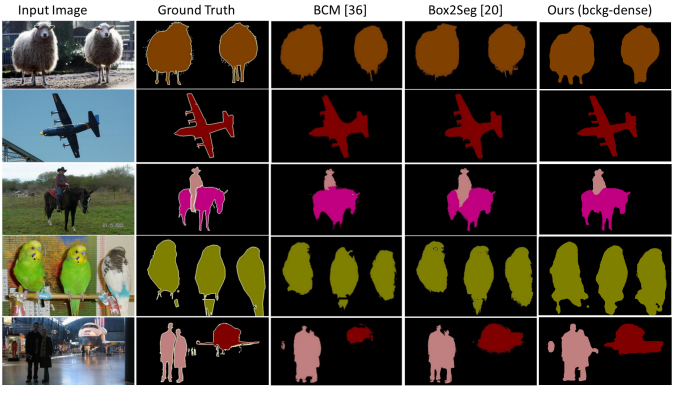

Figure 2: Comparison of our results for *overlap-void* (last column) to BCM [9] and Box2Seg [4] on Pascal VOC 2012, validation set. BCM and Box2Seg are with denseCRF post-processing. Our results are without post-processing.

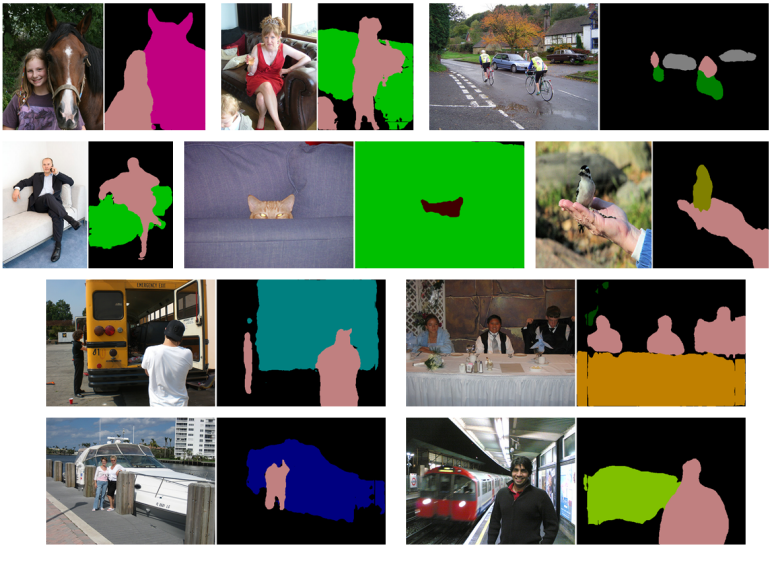

Figure 3: Examples of our segmentations for *overlap-void* on Pascal VOC 2012 test set. Each image pair shows the input image and our segmentation.

[4] Viveka Kulharia, Siddhartha Chandra, Amit Agrawal, Philip H.S. Torr, and Ambrish Tyagi. Box2seg: Attention weighted loss and discriminative feature learning for weakly supervised segmentation. *European Conference on Computer Vision*, 2020.

[5] Qizhu Li, Anurag Arnab, and Philip HS Torr. Weakly-and semi-supervised panoptic segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 102–118, 2018.

[6] George Papandreou, Liang Chieh Chen, Kevin P. Murphy, and Alan L. Yuille. Weakly and semi supervised learning of a deep convolutional network for semantic image segmentation. In *International Conference on Computer Vision*, pages 1742–1750, 2015.

[7] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7479–7489, 2019.

[8] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM TOG*, 23:309–314, 2004.

[9] Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3136–3145, 2019.

[10] Meng Tang, Federico Perazzi, Abdelaziz Djelouah, Ismail Ben Ayed, Christopher Schroers, and Yuri Boykov. On regularized losses for weakly-supervised CNN segmentation. In *European Conference on Computer Vision*, pages 524–540, 2018.

[11] Olga Veksler. Regularized loss for weakly supervised single class semantic segmentation. In *European Conference on Computer Vision*, pages 348–365. Springer, 2020.