PCAMs: Weakly Supervised Semantic Segmentation Using Point Supervision

R. Austin McEver B. S. Manjunath
University of California, Santa Barbara

Abstract

Current state of the art methods for generating semantic segmentation rely heavily on a large set of images that have each pixel labeled with a class of interest label or background. Coming up with such labels, especially in domains that require an expert to do annotations, comes at a heavy cost in time and money. Several methods have shown that we can learn semantic segmentation from less expensive image-level labels, but the effectiveness of point level labels, a healthy compromise between all pixels labelled and none, still remains largely unexplored. This paper presents a novel procedure for producing semantic segmentation from images given some point level annotations. This method includes point annotations in the training of a convolutional neural network (CNN) for producing improved localization and class activation maps. Then, we use another CNN for predicting semantic affinities in order to propagate rough class labels and create pseudo semantic segmentation labels. Finally, we propose training a CNN that is normally fully supervised using our pseudo labels in place of ground truth labels, which further improves performance and simplifies the inference process by requiring just one CNN during inference rather than two. Our method achieves state of the art results for point supervised semantic segmentation on the PASCAL VOC 2012 dataset [13], even outperforming state of the art methods for stronger bounding box and squiggle supervision.

1. Introduction

Today's state of the art semantic segmentation algorithms generally rely on full supervision to achieve top performance. This means that annotators must label every pixel of every image in order to train their models. While this type of information is possible to obtain and available for common classes of interests (e.g. the person, dog, and cat class in natural image settings), many scientific applications cannot feasibly gather such detailed annotations because of the cost of annotator time, which may require a field expert. However, most settings can still afford some level of annotation.

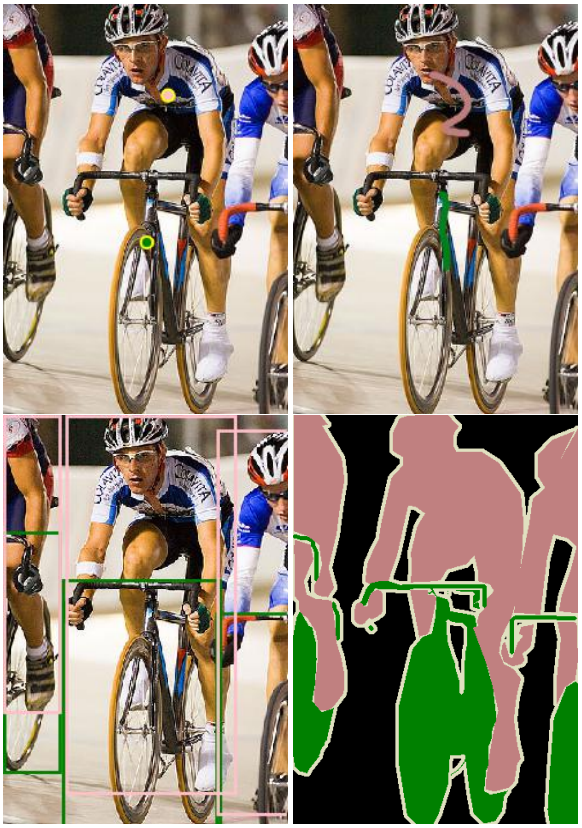

Figure 1. Top left: point annotations. Points are inflated and highlighted for visibility, but only the center pixel and its class label are collected and used. Top right: squiggle supervision. Bottom left: bounding box supervision. Bottom right: full supervision where white represents an ignore label that is not used during training and black is background. An image level label would only indicate that there are one or more instances of person and one or more instances of bicycle in this image.

This problem has motivated the exploration of weakly supervised semantic segmentation, especially the setting where image level labels are used for training [1, 2, 6, 45, 47, 14, 17, 25, 20, 46, 12, 38, 24, 28, 26, 23, 33, 35]. We make the distinction for image level labels because there has also been some attention where bounding boxes [42, 19, 50, 22, 32, 11, 37, 29], points [5], scribbles [44, 43, 27, 48] or other forms of supervision [33] have

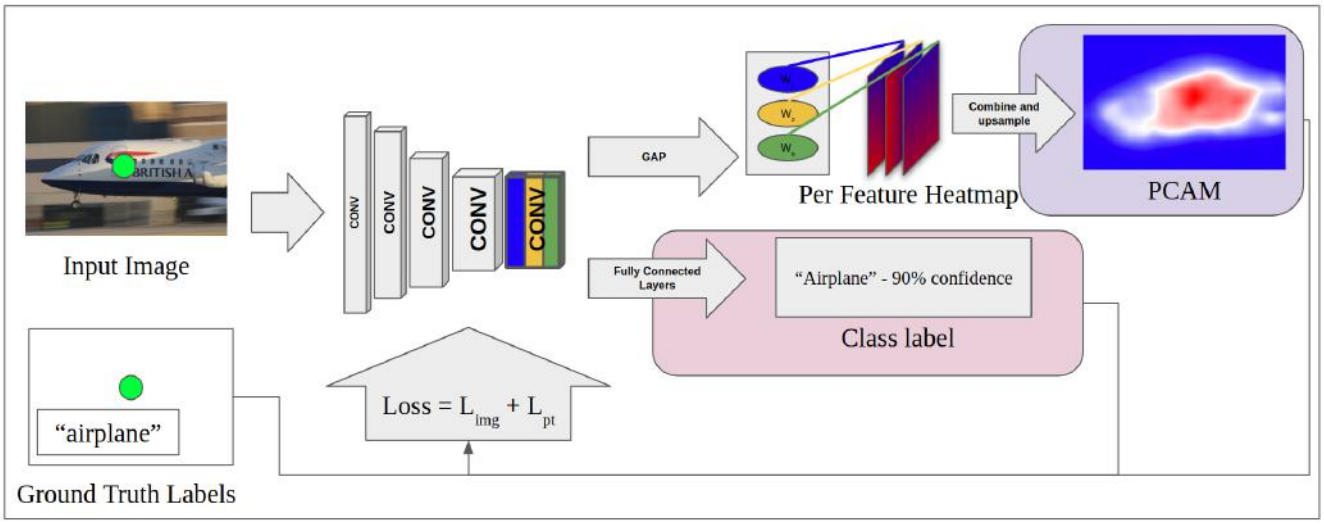

Figure 2. PCAM training overview: compute PCAM with a point supervised input image using the ResNet50 [16] backbone, produce the class labels, and compare the output with the supervised point and class labels to compute a loss used to update the PCAM network. Note that the last convolutional layer's feature maps are colored to match with their class dependent weight vector to produce per feature heatmaps, which are then combined and upsampled during training to generate the PCAM.

been used for the weakly supervised segmentation problem. Figure 1 shows visual examples of a few of these common types of weak supervision for the semantic segmentation problem.

Hence, we consider weak supervision to include any supervised learning algorithm that makes finer predictions than the annotations from which it should learn. For example, a method for solving image level supervised semantic segmentation (the most common form of weak supervision for segmentation), uses image level labels (i.e. simple labels of which classes appear in each training image) to train and then produces semantic segmentation, which is a labelling of each pixel as a particular class of interest or background. A large gap still remains in performance between algorithms using only image level labels and their fully supervised counterparts with more supervision generally providing better performance.

While focusing on image level supervision makes sense for many cases, such as those where many images of classes of interest can be scraped from the web [17, 26], not all problems are that way. Many research groups in more scientific fields heavily guard their images and annotations such that a given group might only have access to its own private dataset. Further, these datasets can be much more expensive to annotate than natural images, requiring field experts to recognize and localize classes of interest. Therefore, it is imperative that we develop methods that make optimal use of annotator time. Bearman, et al [5] showed that, in their setting, point supervision returned the best results given a fixed budget of annotator time.

In this paper, we refer to point supervision as the same setting as used presented by Bearman et al [5]. Specifically, we are interested in the setting where annotators view an image and annotate one or more instances of each class of interest in the image. This setting makes sense when users are manually annotating images as clicking on each instance takes little time beyond providing image level labels and provides valuable localization information that can be used with this paper's method and ultimately leads to significantly better segmentation results. Further, with large or dense images, this setting might even help annotators keep track of which objects have already been noted and which have not. During training, then, models would not have access to any information beyond point annotations.

Despite its effectiveness and efficiency, this type of point supervision for semantic segmentation has not been explored in several years since Bearman et al [5].

In this paper, we present a novel method for achieving state of the art point-supervised semantic segmentation. At the center of our method, we utilize point localization information during the training of class activation maps (CAMs) [51]. We then use IRNet [1] to refine these point supervised class activation maps (PCAMs) to create pseudo semantic segmentation labels, which are then used as ground truth for training a fully supervised semantic segmentation network.

The primary contributions of this paper are as follows:

- We introduce a novel method for including point supervision in the training of CAMs to produce PCAMs, discussed in Section 3.1.

- We achieve state of the art performance on the PASCAL VOC 2012 dataset [13] for point supervised se-

mantic segmentation, outperforming older fully supervised methods and current state of the art methods utilizing even more supervision such as bounding boxes and squiggles.

Our code will be made available on Github.

2. Related Work

This section reviews several methods on which our work is based. First, we review some methods for coarse localization given weak labels. Then we discuss some state of the art image-level supervised segmentation papers before finally reviewing point-level supervised segmentation works.

2.1. Weakly Supervised Object Localization

Weakly supervised localization refers to the problem of finding areas of interest in an image given weak labels. This problem has also been referred to as saliency detection. These methods [21, 18, 3] attempt to find class generic regions that are likely to contain some class of interest using varying levels of supervision. Similarly, methods such as the one presented by Cholakka et al [9] work to localize areas that are likely to contain a specific classes of interest.

Following the path of several methods which use CNNs for weakly supervised object localization [4, 31, 10, 30], CAMs as created by Zhou et al [51] have become the basis for many weakly supervised class specific localization problems including some of the methods that will be discussed in Section 2.2 [1, 2, 46].

In short, Zhou et al [51] propose a method for interpreting the activation of a CNN trained for classification that provides localization information for the classes of interest. This localization is done by adding a global average pooling (GAP) layer to a classification network and examining the activations of the final convolutional layer in the classification CNN. The CNN has class dependent weight vectors that learn which activations correspond with each class. By weighting the feature map of the last convolutional layer in the CNN and upsampling the weighted map to the image size, the authors can begin to localize which areas of an image correspond to different classes. Section 3.1 further discusses this method.

Ultimately, this process allows for coarse localization of classes of interest that can be derived from a network trained with only image level labels. Other methods have built on CAMs for creating even better class specific localization extended beyond the image domain [40, 7]. Zhang et al [49] proposed a method for iteratively improving localization performance by erasing areas of high activation from training images, forcing the CNN to learn other features associated with classes of interest thereby expanding areas of localization.

Still, this line of work is only able to achieve coarse localization, and the resultant activation maps rarely cover the full extent of classes of interest. These activation maps also correspond poorly with object boundaries.

2.2. Image-level Supervised Segmentation

Given the ease of collecting image level labels, many efforts have recently been made to create effective segmentation algorithms that use only these labels [1, 2, 6, 45, 47, 14, 17, 25, 20, 46, 12, 38, 24, 28, 26, 23, 33, 35].

Kolesnikov et al [23] propose a method for training a segmentation CNN with a loss that guides the network via its loss function to follow localization cues, expand around those cues, and adhere to object boundaries. Several methods follow a similar path including work done by Huang et al [20] which uses CAMs to generate its localization cues. Similarly, their loss function has a term for adhering to these cues, growing their regions according to some similarity criteria, and adhering to object boundaries.

The most relevant methods to this paper include recent works of Ahn et al [1, 2] which also generally rely on the method of Zhou et al [51] to create CAMs. Ahn's method generates CAMs, mines affinity labels from the CAMs, and presents a neural network that outputs information that can be used to generate a transition probability matrix. In one of their works [2], Ahn et al describe a method for carefully exploiting CAMs to generate positive and negative affinity labels for pixel pairs. Positive affinity labels should indicate a pair of pixels is of the same class while negative labels indicates differing classes. These labels can be generated automatically by giving thresholds for confidence in CAMs such that two pixels that have high confidence in the same class are assigned a positive affinity label.

Each work by Ahn et al presents different networks and methods for learning from these mined affinity labels, but they both output information that is ultimately used to generate a transition probability matrix. They then perform a random walk over CAMs using the computed transition probabilities to propagate class labels and refine CAMs into pseudo segmentation labels. Finally, these pseudo labels are used in place of ground truth to train a CNN that is designed to perform segmentation given real ground truth labels.

We largely follow this type of method with the novel introduction of point supervision during early stages that leads to a significant improvement in performance.

2.3. Point-level Supervised Segmentation

While image level supervision certainly has its place in domains where images can easily be collected for classes of interest, point supervision can significantly improve performance of segmentation algorithms and can easily be collected when annotating a new dataset. To our knowledge, no publications have tackled the problem of point supervised

segmentation algorithms since Bearman et al [5] introduced the problem, collected and provided point level annotations for the PASCAL VOC 2012 dataset [13], and proposed a method for incorporating point supervision and localization cues into training a normally fully supervised network directly. This method computes a loss over supervised points that is based on a log softmax probability.

The work by Mainis et al [29] is sometimes mentioned in literature as using point level annotations to achieve semantic segmentation; however, this paper uses extreme points rather than more random points as collected in Bearman's work where annotators are simply asked to click on objects of interest[5].

Rather than point-level annotations, then, [29] more closely resembles bounding box level supervision. Arguably, this method uses even more supervision than bounding boxes in that extreme points give bounding box information in addition to four points that are certainly on the boundary of a given object. When given only a bounding box, it is uncertain which parts of the bounding box edges actually lie on an object boundary.

3. Framework

The primary contribution of this paper is to introduce a method for training PCAMs: point supervised class activation maps as shown in Figure 2. Including point supervision in the training process in this way significantly improves the localization performance of CAMs. To achieve state of the art semantic segmentation results, we closely follow the method of [1] to train IRNet on our PCAMs and use its output to refine PCAMs to create pseudo semantic segmentation labels. Finally, we train the fully supervised segmentation network presented in [8], DeepLabv3+, on the pseudo labels and use this network for final predictions.

3.1. PCAMs

To train PCAMs we generally follow the method presented in the work by Zhou et al [51] but present an additional step that allows the inclusion of point supervision during training as shown in Figure 2. The original method for producing CAMs adds a global average pooling layer to an image classification CNN. Following the work by Ahn et al [1], we use ResNet50 [16] as the backbone classification network and drop the stride of its last downsampling layer to prevent too much reduction in resolution. Training the image classification CNN is done with a simple multilabel soft margin loss.

$$L_{img}(\hat{y}, y) = -\frac{1}{C} * \sum_{i=1}^{C} (y_i * \log((1+\exp(-\hat{y}_i))^{-1}) + (1-y_i) * \log(\frac{\exp(-\hat{y}_i)}{1+\exp(-\hat{y}_i)})) \tag{1}$$

where y is the one hot encoded vector of classes in the images, $\hat{y}$ is the predicted class vector, y_i is the label of the ith class, and C is the number of classes. Once trained, CAMs of a given class c can be generated by

$$M_c(\mathbf{x}) = \frac{\phi_c^\top f(\mathbf{x})}{max_{\mathbf{x}} \phi_c^\top f(\mathbf{x})} \tag{2}$$

Where ϕ_c represents the learned class-dependent weight vector for class c, f is the feature map from the final convolutional layer of the classification network backbone, and x is a 2D coordinate in f. As mentioned before, we use ResNet50 [16] as our backbone network with a reduced stride in the final downsampling layer. The dimensions of the CAMs are then 1/16 of the input image.

Training the classification network with point annotations, however, requires that we generate the CAMs during training. To align the CAMs with the input image, we use bilinear interpolation to upsample the CAMs to the size of the input image to create U_c, the upsampled CAM for class c. This upsampling is only done during inference in previous works, but our method introduces this upsampling during training so that we can compare the upsampled CAM with any supervised points and guide the network for better activation mapping. We then compute the mean squared error loss over each of the supervised points as follows:

$$L_{pt}^c = \frac{1}{|S|} \sum_{s \epsilon S} (U_c(s) - G_c(s))^2 \tag{3}$$

where S is the set of supervised pixel locations, $U_c(s)$ is the predicted probability that location s is of class c, and $G_c(s)$ is the binary ground truth label for class c for the pixel at location s. For the point supervised term of the loss for training a network to generate PCAMs, we average the classwise losses for each class present in the training image.

$$L_{pt} = \frac{1}{|C'|} * \sum_{c \epsilon C'} L_{pt}^c \tag{4}$$

where C' is the set of classes in the training image. This allows us to precisely use any point level annotations to guide the network to activate at specific locations. This loss term also leads to more confident activation maps that cover a greater spatial extent of objects of interest. The total loss for training the PCAM network is then

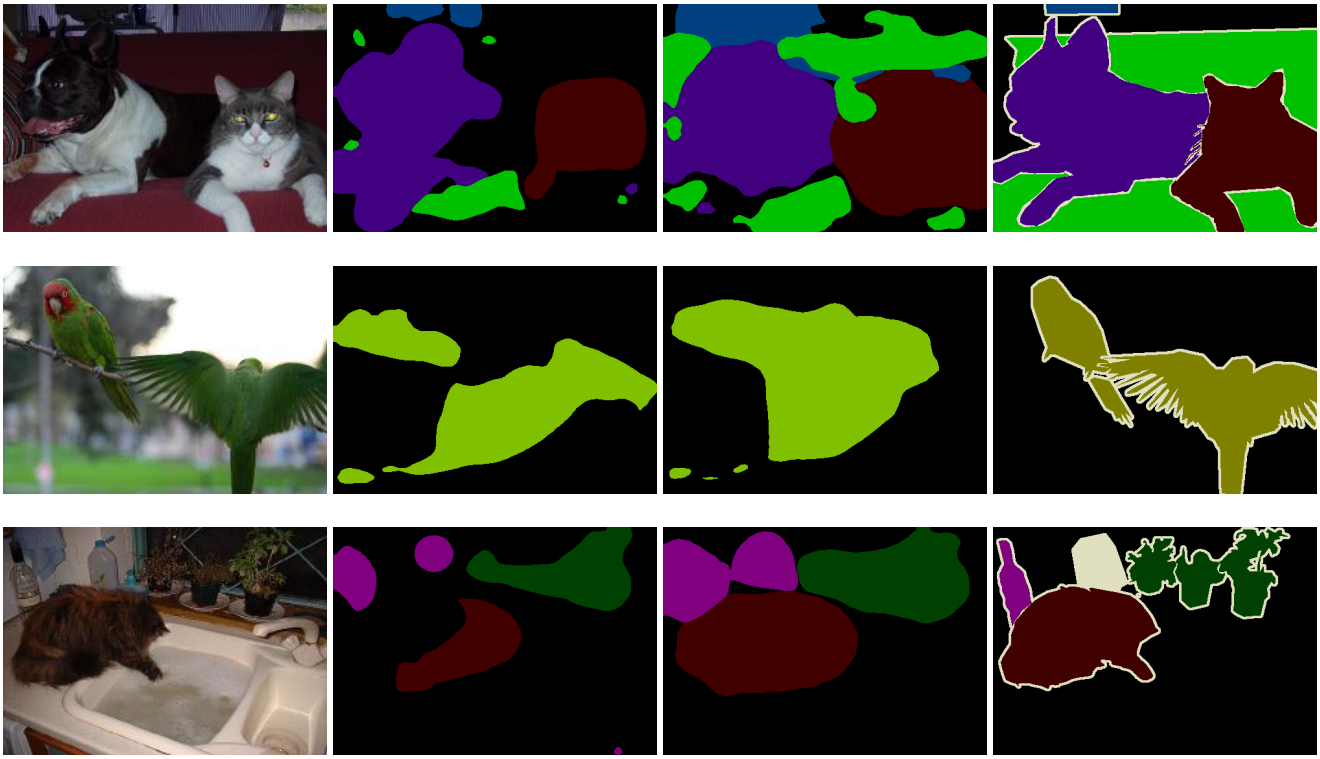

Figure 3. From left to right: original images, CAM labels, PCAM labels, and ground truth segmentation for each image

$$L = L_{img} + \alpha L_{pt} \tag{5}$$

where α is a weighting term.

3.2. IRNet

We follow the work of Ahn et al [1] in training and using IRNet. We mine semantic affinity labels from our improved PCAMs, which use point level supervision, rather than mining labels from CAMs which use image level supervision, and use them for training IRNet. In order to mine semantic affinity labels, we threshold each PCAM such that low values are considered background and high values are considered as the corresponding class. We ignore all pixels that have middling confidence values as their labels are uncertain and affinity labels must be mined reliably to optimize the performance of IRNet.

We then examine all pixels within a small radius of confident pixels. If a pair of pixels are both confident or background and have the same label, the pair is assigned a positive affinity label, and if it has a different label it is assigned a negative affinity label.

IRNet uses training images and these mined affinity labels to learn to predict a displacement vector field and class boundary map. The displacement field should indicate centroids of class boundaries, which aims to segment class instance but also aids in separating instances of adjacent differing classes. The class boundary map aims to indicate boundaries of classes.

The class boundary map can be synthesized to create an affinity matrix. If a pair of pixels have a positive class boundary pixel on the line between them, they have low semantic affinity. Otherwise, they are likely to be of the same class and have high semantic affinity. Next, the semantic affinities are used to compute a transition probability matrix for a random walk which is performed over instance-wise PCAMs. Guided by this transition probability matrix, a random walk over PCAMs expands and refines activation areas. These refined PCAMs are then combined to create pseudo semantic segmentation labels.

3.3. DeepLabv3+

The final step in our method takes the pseudo semantic segmentation labels generated via random walk propagation over PCAMs and uses them as ground truth for training DeepLabv3+ [8]. We find that, despite training on imperfect labels, the network is still able to generate slightly better segmentation results on unseen images when compared to refined PCAMs.

Further, using a trained fully supervised network makes inference simpler. To infer a new image otherwise, we would need to first run inference using the trained PCAM

network and run inference using IRNet before we could compute the transition probability network from IRNet's output. Finally, we would need to run the random walk algorithm on the PCAM using the generated transition probability matrix to refine the PCAM.

After we train DeepLabv3+ on our pseudo semantic segmentation labels, inference on an unseen image can be done simply by running inference with our trained DeepLabv3+ model.

4. Experiments and Results

The following section describes our experiments on the PASCAL VOC 2012 dataset [13] on which we achieve state of the art performance for point-supervised semantic segmentation.

4.1. Dataset

All of our experimental results are reported on the PASCAL VOC 2012 dataset [13] and trained using the PASCAL VOC 2012 training images supplemented with the images from the SBD dataset [15], following common practice [5, 1, 2]. The PASCAL VOC 2012 dataset includes 1,464 training images and 1,449 validatation images. The SBD dataset contains annotations for 11,355 images from the PASCAL VOC 2012 dataset. In total, we train with 10,582 training images and test with 1,449 validation images.

For training PCAMs, we use the point level labels provided by [5]. These labels include one ore more annotated points per class in each training image. Overall, we use an average of approximately 2.4 points per image for training.

4.2. Hyperparameters

4.2.1 PCAM Network

The PCAM network uses ResNet50 [16] as its backbone network. The learning rate is initially set to 0.001 for the backbone parameters and 0.01 for the classification layers. The loss weighting term α is set to 0.1.

4.2.2 IRNet

We generally use IRNet as presented by Ahn et al [1]. We set the CAM evaluation threshold for producing the labels for IRNet to 0.3. Similarly, we set the threshold for semantic segmentation at 0.3. These adjustments compensate for PCAMs being somewhat more confident than CAMs. For our setting, we had slightly better results setting IRNet's β parameter to 12. This parameter affects the generation of the transition matrix that is used for the random walk propogation of attention scores and is detailed in the paper by Ahn et al[1].

Activation Map	Sup.	*train*	*val*
CAM	*I*	48.3	46.0
PCAM	*P*	56.5	54.4
PCAM	*F*	64.0	55.5

Table 1. mIOU comparison of CAM, PCAM, and PCAM trained with all points on the PASCAL VOC 2012 *train* and *val* sets

Refined Activation Map	*train*	*val*
CAM	66.5	57.4
PCAM	70.8	68.5

Table 2. mIOU comparison of performance on PASCAL VOC 2012 *train* and *val* sets of CAMs and PCAMs after being refined by random walk via IRNet's transition matrix

Method	Sup.	mIOU
WhatsPoint [5]	*P*	46.1
MIL-FCN [34]	*I*	25.7
CCNN [33]	*I*	35.3
EM-Adapt [32]	*I*	38.2
MIL+seg [35]	*I*	42.0
DCSM [41]	*I*	44.1
BFBP [39]	*I*	46.6
SEC [23]	*I*	50.7
AF-SS [36]	*I*	52.6
Combining Cues [38]	*I*	52.8
AE-PSL [46]	*I*	55.0
DSRG [20]	*I*	61.4
AffinityNet [2]	*I*	61.7
IRNet [1]	*I*	63.5
FickleNet [25]	*I*	64.9
ScribbleSup [27]	*S*	63.1
NormCut [43]	*S*	65.1
BBox-seg [32]	*B*	60.6
SDI [22]	*B*	65.7
BCM [42]	*B*	66.8
Ours - refined PCAM	*P*	68.5
Ours - final	*P*	70.5
DeepLabv3+	*F*	78.9

Table 3. mIOU comparison of recent or related weakly supervised semantic segmentation methods on the PASCAL VOC 2012 *val* set. Sup. shows the level of supervision of each method where *P* is point level, *I* is image level, *B* is bounding box level, *S* is scribble level, and *F* is fully supervised. Ours - final uses DeepLabv3+ trained on refined PCAMs for inference.

4.2.3 DeepLabv3+

We train DeepLabv3+ [8] using the labels generated from refining PCAMs with IRNet. We use a ResNet backbone, a learning rate of 0.007, weight decay of 4e-5, and a Nesterov momentum optimizer with a momentum of 0.9. We use an output stride of 16 during training and evaluation, and

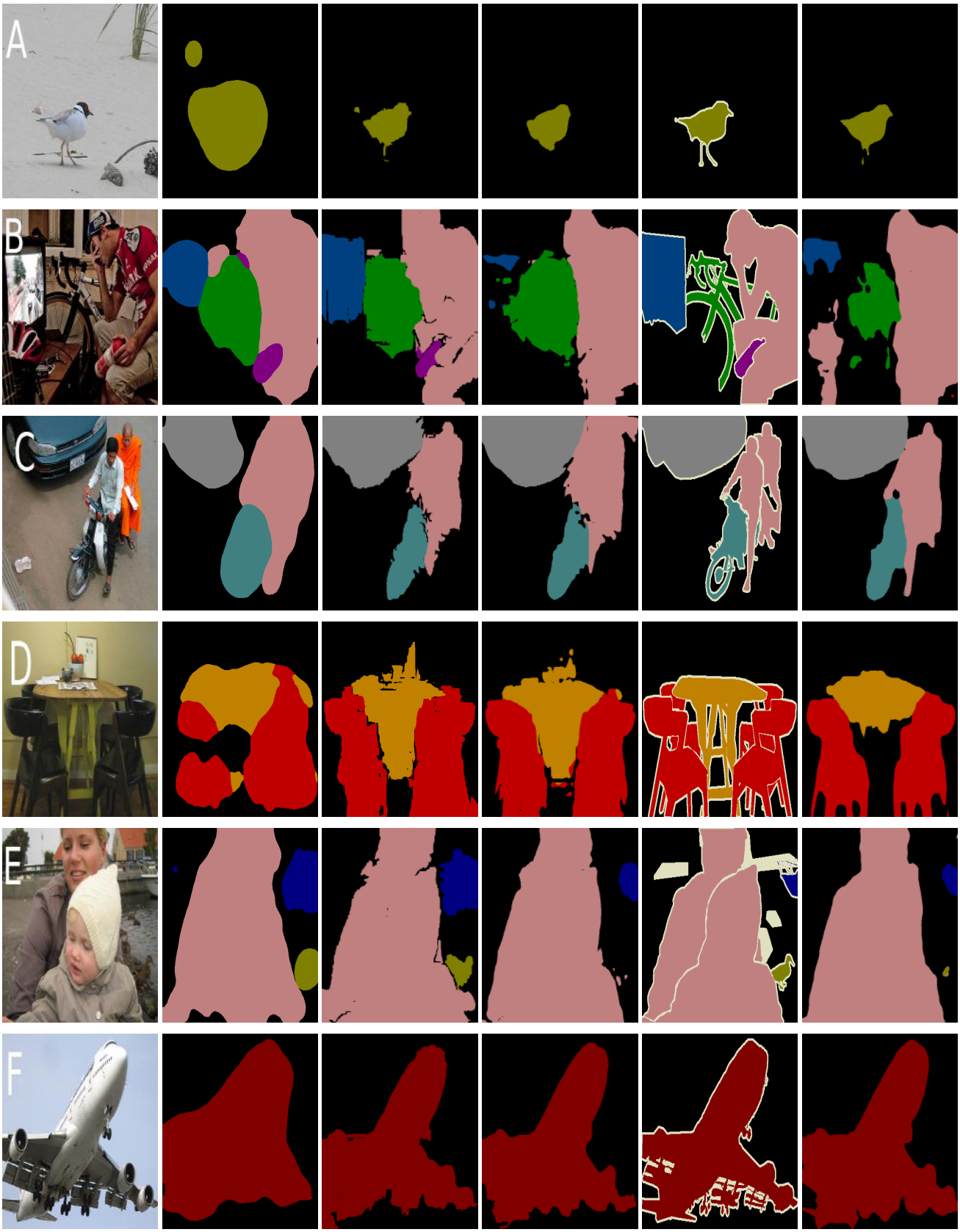

Figure 4. From left to right: original images, PCAM labels, refined PCAM labels, predictions from PCAM-supervised DeepLabv3+, ground truth segmentation, and predictions from fully supervised DeepLabv3+ for each image

Method	Sup.	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mkb	person	plant	sheep	sofa	train	tv	mean
EM-Adapt [32]	*I*	67.2	29.2	17.6	28.6	22.2	29.6	47.0	44.0	44.2	14.6	35.1	24.9	41.0	34.8	41.6	32.1	24.8	37.4	24.0	38.1	31.6	33.8
CCNN [33]	*I*	68.5	25.5	18.0	25.4	20.2	36.3	46.8	47.1	48.0	15.8	37.9	21.0	44.5	34.5	46.2	40.7	30.4	36.3	22.2	38.8	36.9	35.3
MIL+seg [35]	*I*	79.6	50.2	21.6	40.9	34.9	40.5	45.9	51.5	60.6	12.6	51.2	11.6	56.8	52.9	44.8	42.7	31.2	55.4	21.5	38.8	36.9	42.0
SEC [23]	*I*	82.4	62.9	26.4	61.6	27.6	38.1	66.6	62.7	75.2	22.1	53.5	28.3	65.8	57.8	62.3	52.5	32.5	62.6	32.1	45.4	45.3	50.7
AE-PSL [46]	*I*	83.4	71.1	30.5	72.9	41.6	55.9	63.1	60.2	74.0	18.0	66.5	32.4	71.7	56.3	64.8	52.4	37.4	69.1	31.4	58.9	43.9	55.0
What's Point [5]	*P*	80	49	23	39	41	46	60	61	56	18	38	41	54	42	55	57	32	51	26	55	45	46.0
AffinityNet [2]	*I*	88.2	68.2	**30.6**	81.1	49.6	61.0	77.8	66.1	75.1	29.0	66.0	40.2	80.4	62.0	70.4	**73.7**	42.5	70.7	42.6	68.1	51.6	61.7
BCM [42]	*B*	89.8	**68.3**	27.1	73.7	56.4	72.6	84.2	75.6	79.9	**35.2**	78.3	53.2	77.6	66.4	68.1	73.1	**56.8**	80.1	45.1	**74.7**	54.6	66.8
Ours - final	*P*	**89.9**	66.1	30.1	**85.2**	**62.5**	**75.8**	**87.1**	**80.4**	**87.1**	34.0	**85.1**	**60.0**	**84.4**	**82.4**	**77.4**	68.4	56.1	**84.0**	**46.1**	72.6	**64.2**	**70.5**

Table 4. per class mIOU comparsion of recent or related methods on the PASCAL VOC 2012 val set

we do not implement multi-scale or flipped inputs during training. This is the same training setting as DeepLabv3+ shown in Table 3; however, the performance shown there for "DeepLabv3+" is for the fully supervised setting trained on actual ground truth as opposed to "Ours - final" which is DeepLabv3+ trained on refined PCAMs.

4.3. PCAM Performance

Figure 3 shows some localization performance of PCAMs compared to CAMs generated from the same network without point supervision. This figure shows that the point supervision generally leads to better localization. Further, PCAMs do a much better job of covering more of a given object's extent.

Table 1 shows the quantitative performance difference of each activation map. In addition to the image level CAM, we also experimented with training the network with the PCAM loss given every ground truth point, i.e. full supervision, rather than the much smaller one point per class per image. Point supervision strongly increases the localization performance of activation maps, though using all points in an image with our method seems to have relatively little influence on performance on unseen images.

4.4. IRNet Label Refinement

Figure 4 shows several examples of our method's performance as well as the performance of the fully supervised DeepLabv3+. We can see evidence that the label propagation via IRNet's transition matrix usually helps refine boundaries more precisely, though we tend to see choppy edges as a result of the random walk propagation.

Table 2 shows the performance difference of each activation map after having been refined by IRNet. Unsurprisingly, the already better PCAMs have significantly better localization performance after being refined than do CAMs. The performance increase between PCAMs and CAMs only becomes amplified after refinement. Before refinement, PCAMs outperform CAMs by 8.4%, and after refinement, PCAMs surpasses CAMs by 11.1%.

4.5. Semantic Segmentation Results

Figure 4 shows the performance of DeepLabv3+ after being trained with refined PCAMs as well as its predictions after being trained with ground truth. The results are quite similar, with the PCAM-trained network generally making fewer predictions, which is helpful in the case of the image B where there is no false prediction in the bottom left corner. However, fewer predictions are not helpful in cases such as in the image E where the duck in the bottom right corner is not predicted at all.

While the PCAM-trained CNN makes fewer predictions and seems to sometimes miss smaller objects, it adheres to boundaries better and performs slightly better quantitatively than the refined PCAMs themselves.

Table 3 shows the performance of several recent methods on semantic segmentation of the VOC *val* set. Our method achieves state of the art performance in point supervision. Further, it surpasses the method from Tang et al [43], which uses stronger scribble level supervision, and it even outperforms the very recent method by Song et al [42], which uses stronger bounding box supervision for this task. Our method recovers an impressive 89% of the performance of its fully supervised counterpart using only point supervision.

Table 4 shows a number of recent methods, their levels of supervision, and their class-wise IOU performance. Unsurprisingly, the better supervised method from Song et al [42] performs better on the some of the challenging classes like chair; however, our method performs better on nearly every class and overall.

5. Conclusion

While numerous methods for using image-level supervision for weakly supervised semantic segmentation have been introduced recently, using point supervision has been largely unexplored. We propose a new method for training a class activation network to include point supervision. We demonstrate that this approach greatly enhances class activation maps, and we achieve state of the art performance for point supervised semantic segmentation that is even better than the state of the art methods using the stronger bounding box supervision. We achieve this performance using today's state of the art methods for label propagation over activation maps, but our method and PCAMs can be used in other pipelines as well.

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2209–2218, 2019. 1, 2, 3, 4, 5, 6

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4981–4990, 2018. 1, 3, 6, 8

[3] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2189–2202, 2012. 3

[4] Loris Bazzani, Alessandra Bergamo, Dragomir Anguelov, and Lorenzo Torresani. Self-taught object localization with deep networks. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016. 3

[5] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. Whats the point: Semantic segmentation with point supervision. In *European conference on computer vision*, pages 549–565. Springer, 2016. 1, 2, 4, 6, 8

[6] Rania Briq, Michael Moeller, and Jürgen Gall. Convolutional simplex projection network for weakly supervised semantic segmentation. In *BMVC*, page 263, 2018. 1, 3

[7] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847. IEEE, 2018. 3

[8] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 4, 5, 6

[9] Hisham Cholakkal, Jubin Johnson, and Deepu Rajan. Backtracking scspm image classifier for weakly supervised top-down saliency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5278–5287, 2016. 3

[10] Ramazan Gokberk Cinbis, Jakob Verbeek, and Cordelia Schmid. Weakly supervised object localization with multi-fold multiple instance learning. *IEEE transactions on pattern analysis and machine intelligence*, 39(1):189–203, 2016. 3

[11] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1635–1643, 2015. 1

[12] Thibaut Durand, Taylor Mordan, Nicolas Thome, and Matthieu Cord. Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 642–651, 2017. 1, 3

[13] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 1, 2, 4, 6

[14] Weifeng Ge, Sibei Yang, and Yizhou Yu. Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1277–1286, 2018. 1, 3

[15] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 International Conference on Computer Vision*, pages 991–998. IEEE, 2011. 6

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 4, 6

[17] Seunghoon Hong, Donghun Yeo, Suha Kwak, Honglak Lee, and Bohyung Han. Weakly supervised semantic segmentation using web-crawled videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7322–7330, 2017. 1, 2, 3

[18] Xiaodi Hou and Liqing Zhang. Saliency detection: A spectral residual approach. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. Ieee, 2007. 3

[19] Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick. Learning to segment every thing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4233–4241, 2018. 1

[20] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7014–7023, 2018. 1, 3, 6

[21] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (11):1254–1259, 1998. 3

[22] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 876–885, 2017. 1, 6

[23] Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *European Conference on Computer Vision*, pages 695–711. Springer, 2016. 1, 3, 6, 8

[24] Suha Kwak, Seunghoon Hong, and Bohyung Han. Weakly supervised semantic segmentation using superpixel pooling network. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017. 1, 3

[25] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5267–5276, 2019. 1, 3, 6

[26] Xiaodan Liang, Yunchao Wei, Liang Lin, Yunpeng Chen, Xiaohui Shen, Jianchao Yang, and Shuicheng Yan. Learning to segment human by watching youtube. *IEEE transactions on pattern analysis and machine intelligence*, 39(7):1462–1468, 2016. 1, 2, 3

[27] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3159–3167, 2016. 1, 6

[28] Zhiwu Lu, Zhenyong Fu, Tao Xiang, Peng Han, Liwei Wang, and Xin Gao. Learning from weak and noisy labels for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(3):486–500, 2016. 1, 3

[29] Kevis-Kokitsi Maninis, Sergi Caelles, Jordi Pont-Tuset, and Luc Van Gool. Deep extreme cut: From extreme points to object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 616–625, 2018. 1, 4

[30] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1717–1724, 2014. 3

[31] Maxime Oquab, Léon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 685–694, 2015. 3

[32] 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 *Proceedings of the IEEE international conference on computer vision*, pages 1742–1750, 2015. 1, 6, 8

[33] Deepak Pathak, Philipp Krahenbuhl, and Trevor Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1796–1804, 2015. 1, 3, 6, 8

[34] Deepak Pathak, Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional multi-class multiple instance learning. *arXiv preprint arXiv:1412.7144*, 2014. 6

[35] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1713–1721, 2015. 1, 3, 6, 8

[36] Xiaojuan Qi, Zhengzhe Liu, Jianping Shi, Hengshuang Zhao, and Jiaya Jia. Augmented feedback in semantic segmentation under image level supervision. In *European Conference on Computer Vision*, pages 90–105. Springer, 2016. 6

[37] Martin Rajchl, Matthew CH Lee, Ozan Oktay, Konstantinos Kamnitsas, Jonathan Passerat-Palmbach, Wenjia Bai, Mellisa Damodaram, Mary A Rutherford, Joseph V Hajnal, Bernhard Kainz, et al. Deepcut: Object segmentation from bounding box annotations using convolutional neural networks. *IEEE transactions on medical imaging*, 36(2):674–683, 2016. 1

[38] Anirban Roy and Sinisa Todorovic. Combining bottom-up, top-down, and smoothness cues for weakly supervised image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3529–3538, 2017. 1, 3, 6

[39] 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 *European Conference on Computer Vision*, pages 413–432. Springer, 2016. 6

[40] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017. 3

[41] Wataru Shimoda and Keiji Yanai. Distinct class-specific saliency maps for weakly supervised semantic segmentation. In *European Conference on Computer Vision*, pages 218–234. Springer, 2016. 6

[42] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3136–3145, 2019. 1, 6, 8

[43] Meng Tang, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers. Normalized cut loss for weakly-supervised cnn segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1818–1827, 2018. 1, 6, 8

[44] Paul Vernaza and Manmohan Chandraker. Learning random-walk label propagation for weakly-supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7158–7166, 2017. 1

[45] Xiang Wang, Shaodi You, Xi Li, and Huimin Ma. Weakly-supervised semantic segmentation by iteratively mining common object features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1354–1362, 2018. 1, 3

[46] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1568–1576, 2017. 1, 3, 6, 8

[47] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7268–7277, 2018. 1, 3

[48] Jia Xu, Alexander G Schwing, and Raquel Urtasun. Learning to segment under various forms of weak supervision. In

Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3781–3790, 2015. 1

[49] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas S Huang. Adversarial complementary learning for weakly supervised object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1325–1334, 2018. 3

[50] Xiangyun Zhao, Shuang Liang, and Yichen Wei. Pseudo mask augmented object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4061–4070, 2018. 1

[51] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 2, 3, 4