Semi-Supervised Semantic Image Segmentation with Self-correcting Networks

Mostafa S. Ibrahim*
Simon Fraser University
msibrahi@sfu.ca

Arash Vahdat
NVIDIA
avahdat@nvidia.com

Mani Ranjbar
Sportlogiq
mani@sportlogiq.com

William G. Macready
Sanctuary AI
wgm@sanctuary.ai

Abstract

Building a large image dataset with high-quality object masks for semantic segmentation is costly and time consuming. In this paper, we introduce a principled semi-supervised framework that only uses a small set of fully supervised images (having semantic segmentation labels and box labels) and a set of images with only object bounding box labels (we call it the weak set). Our framework trains the primary segmentation model with the aid of an ancillary model that generates initial segmentation labels for the weak set and a self-correction module that improves the generated labels during training using the increasingly accurate primary model. We introduce two variants of the self-correction module using either linear or convolutional functions. Experiments on the PASCAL VOC 2012 and Cityscape datasets show that our models trained with a small fully supervised set perform similar to, or better than, models trained with a large fully supervised set while requiring ∼7x less annotation effort.

1. Introduction

Deep convolutional neural networks (CNNs) have been successful in many computer vision tasks including image classification [28, 19, 76], object detection [45, 34, 43], semantic segmentation [4, 71, 9], action recognition [14, 25, 49, 55], and facial landmark localization [53, 69, 75]. However, the common prerequisite for all these successes is the availability of large training corpora of labeled images. Of these tasks, semantic image segmentation is one of the most costly tasks in terms of data annotation. For example, drawing a segmentation annotation on an object is on average ∼8x slower than drawing a bounding box and ∼78x slower than labeling the presence of objects in images [5]. As a result, most image segmentation datasets are orders of magnitude smaller than image-classification datasets.

In this paper, we mitigate the data demands of semantic segmentation with a semi-supervised method that leverages cheap object bounding box labels in training. This approach reduces the data annotation requirements at the cost of requiring inference of the mask label for an object within a bounding box.

Current state-of-the-art semi-supervised methods typically rely on hand-crafted heuristics to infer an object mask inside a bounding box [41, 12, 26]. In contrast, we propose a principled framework that trains semantic segmentation models in a semi-supervised setting using a small set of fully supervised images (with semantic object masks and bounding boxes) and a weak set of images (with only bounding box annotations). The fully supervised set is first used to train an ancillary segmentation model that predicts object masks on the weak set. Using this augmented data a primary segmentation model is trained. This primary segmentation model is probabilistic to accommodate the uncertainty of the mask labels generated by the ancillary model. Training is formulated so that the labels supplied to the primary model are refined during training from the initial ancillary mask labels to more accurate labels obtained from the primary model itself as it improves. Hence, we call our framework a self-correcting segmentation model as it improves the weakly supervised labels based on its current probabilistic model of object masks.

We propose two approaches to the self-correction mechanism. Firstly, inspired by Vahdat [56], we use a function that linearly combines the ancillary and model predictions. We show that this simple and effective approach is the natural result of minimizing a weighted Kullback-Leibler (KL) divergence from a distribution over segmentation labels to both the ancillary and primary models. However, this approach requires defining a weight whose optimal value should change during training. With this motivation, we develop a second adaptive self-correction mechanism. We use CNNs to learn how to combine the ancillary and primary models to predict a segmentation on a weak set of images. This approach eliminates the need for a weighting schedule.

Experiments on the PASCAL VOC and Cityscapes datasets show that our models trained with a small portion of fully supervised set achieve a performance comparable to (and in some cases better than) the models trained with

*Work done while interning at D-Wave Systems

all the fully supervised images.

2. Related Work

Semantic Segmentation: Fully convolutional networks (FCNs) [37] have become indispensable models for semantic image segmentation. Many successful applications of FCNs rely on atrous convolutions [65] (to increase the receptive field of the network without down-scaling the image) and dense conditional random fields (CRFs) [27] (either as post-processing [6] or as an integral part of the segmentation model [73, 33, 48, 36]). Recent efforts have focused on encoder-decoder based models that extract long-range information using encoder networks whose output is passed to decoder networks that generate a high-resolution segmentation prediction. SegNet [4], U-Net [46] and RefineNet [32] are examples of such models that use different mechanisms for passing information from the encoder to the decoder.[1] Another approach for capturing long-range contextual information is spatial pyramid pooling [29]. ParseNet [35] adds global context features to the spatial features, DeepLabv2 [7] uses atrous spatial pyramid pooling (ASPP), and PSPNet [71] introduces spatial pyramid pooling on several scales for the segmentation problem.

While other segmentation models may be used, we employ DeepLabv3+ [9] as our segmentation model because it outperforms previous CRF-based DeepLab models using simple factorial output. DeepLabv3+ replaces Deeplabv3's [8] backbone with the Xception network [10] and stacks it with a simple two-level decoder that uses lower-resolution feature maps of the encoder.

Robust Training: Training a segmentation model from bounding box information can be formulated as a problem of robust learning from noisy labeled instances. Previous work on robust learning has focused on classification problems with a small number of output variables. In this setting, a common simplifying assumption models the noise on output labels as independent of the input [40, 39, 42, 52, 70]. However, recent work has lifted this constraint to model noise based on each instance's content (i.e., input-dependent noise). Xiao *et al*. [63] use a simple binary indicator function to represent whether each instance does or does not have a noisy label. Misra *et al*. [38] represent label noise for each class independently. Vahdat [56] proposes CRFs to represent the joint distribution of noisy and clean labels extending structural models [57, 58] to deep networks. Ren *et al*. [44] gain robustness against noisy labels by reweighting each instance during training whereas Dehghani *et al*. [13] reweight gradients based on a confidence score on labels. Among methods proposed for label correction, Veit *et al*. [59] use a neural regression model to predict clean labels given noisy labels and image features, Jiang *et al*. [24] learn curriculum, and Tanaka *et al*. [54] use the current model to predict labels. All these models have been restricted to image-classification problems and have not yet been applied to image segmentation.

Semi-Supervised Semantic Segmentation: The focus of this paper is to train deep segmentation CNNs using bounding box annotations. Papandreou *et al*. [41] propose an Expectation-Maximization-based (EM) algorithm on top of DeepLabv1 [6] to estimate segmentation labels for the weak set of images (only with box information). In each training step, segmentation labels are estimated based on the network output in an EM fashion. Dai *et al*. [12] propose an iterative training approach that alternates between generating region proposals (from a pool of fixed proposals) and fine-tuning the network. Similarly, Khoreva *et al*. [26] use an iterative algorithm but rely on GrabCut [47] and hand-crafted rules to extract the segmentation mask in each iteration. Our work differs from these previous methods in two significant aspects: i) We replace hand-crafted rules with an ancillary CNN for extracting probabilistic segmentation labels for an object within a box for the weak set. ii) We use a self-correcting model to correct for the mismatch between the output of the ancillary CNN and the primary segmentation model during training.

In addition to box annotations, segmentation models may use other forms of weak annotations such as image pixel-level [60, 62, 22, 3, 17, 61, 15], image label-level [68], scribbles [64, 31], point annotation [5], or web videos [20]. Recently, adversarial learning-based methods [23, 51] have been also proposed for this problem. Our framework is complimentary to other forms of supervision or adversarial training and can be used alongside them.

3. Methods

Our goal is to train a semantic segmentation network in a semi-supervised setting using two training sets: i) a small fully supervised set (containing images, segmentation ground-truth and object bounding boxes) and ii) a weak set (containing images and object bounding boxes only). An overview of our framework is shown in Fig. 1. There are three models: i) The **Primary segmentation model** generates a semantic segmentation of objects given an image. ii) The **Ancillary segmentation model** outputs a segmentation given an image and bounding box. The model generates an initial segmentation for the weak set, which aids training of the primary model. iii) The **Self-correction module** refines the segmentations generated by the ancillary and current primary model for the weak set. Both the ancillary and the primary models are based on DeepLabv3+ [9]. However, our framework is general and can use any existing segmentation model.

[1] SegNet [4] transfers max-pooling indices from encoder to decoder, U-Net [46] introduces skip-connections between encoder-decoder networks and RefineNet [32] proposes multipath refinement in the decoder through long-range residual blocks.

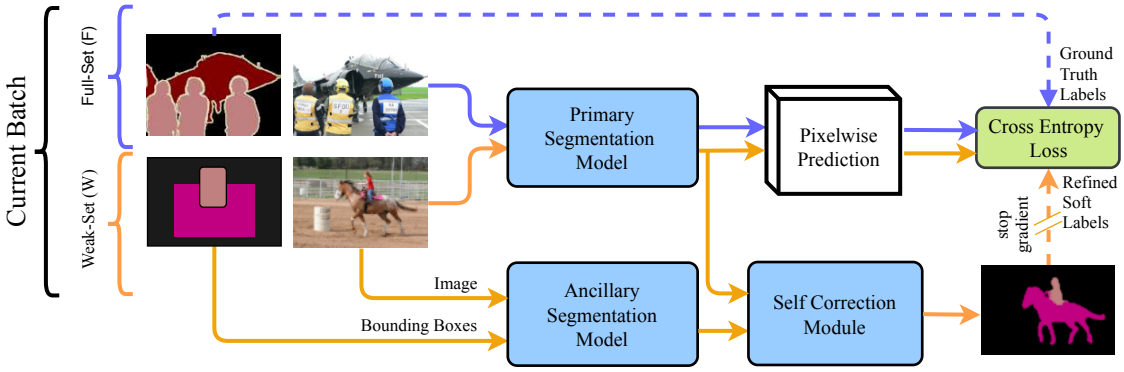

Figure 1: An overview of our segmentation framework consisting of three models: i) Primary segmentation model generates a semantic segmentation of objects given an image. This is the main model that is subject to the training and is used at test time. ii) Ancillary segmentation model outputs a segmentation given an image and bounding box. This model generates an initial segmentation for the weak set, which will aid training the primary model. iii) Self-correction module refines segmentations generated by the ancillary model and the current primary model for the weak set. The primary model is trained using the cross-entropy loss that matches its output to either ground-truth segmentation labels for the fully supervised examples or soft refined labels generated by the self-correction module for the weak set.

In Sec. 3.1, we present the ancillary model, and in Sec. 3.2, we show a simple way to use this model to train the primary model. In Sec. 3.3 and Sec. 3.4, we present two variants of self-correcting model.

Notation: $\boldsymbol{x}$ represents an image, $\boldsymbol{b}$ represents object bounding boxes in an image, and $\boldsymbol{y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_M]$ represents a segmentation label where $\boldsymbol{y}_m \in [0,1]^{C+1}$ for $m \in \{1, 2, \ldots, M\}$ is a one-hot label for the m^{th} pixel, C is the number of foreground labels augmented with the background class, and M is the total number of pixels. Each bounding box is associated with an object and has one of the foreground labels. The fully supervised dataset is indicated as $\boldsymbol{\mathcal{F}} = \{(\boldsymbol{x}^{(f)}, \boldsymbol{y}^{(f)}, \boldsymbol{b}^{(f)})\}_{f=1}^{F}$ where F is the total number of instances in $\boldsymbol{\mathcal{F}}$. Similarly, the weak set is noted by $\boldsymbol{\mathcal{W}} = \{(\boldsymbol{x}^{(w)}, \boldsymbol{b}^{(w)})\}_{w=1}^{W}$. We use $p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\phi})$ to represent the primary segmentation model and $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b};\boldsymbol{\theta})$ to represent the ancillary model. $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ are the respective parameters of each model. We occasionally drop the denotation of parameters for readability. We assume that both ancillary and primary models define a distribution of segmentation labels using a factorial distribution, i.e., $p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\phi}) = \prod_{m=1}^{M} p_m(\boldsymbol{y}_m|\boldsymbol{x};\boldsymbol{\phi})$ and $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b};\boldsymbol{\theta}) = \prod_{m=1}^{M} p_{anc,m}(\boldsymbol{y}_m|\boldsymbol{x},\boldsymbol{b};\boldsymbol{\theta})$ where each factor ($p_m(\boldsymbol{y}_m|\boldsymbol{x};\boldsymbol{\phi})$ or $p_{anc,m}(\boldsymbol{y}_m|\boldsymbol{x},\boldsymbol{b};\boldsymbol{\theta})$) is a categorical distribution (over $C+1$ categories).

3.1. Ancillary Segmentation Model

The key challenge in semi-supervised training of segmentation models with bounding box annotations is to infer the segmentation of the object inside a box. Existing approaches to this problem mainly rely on hand-crafted rule-based procedures such as GrabCut [47] or an iterative label refinement [41, 12, 26] mechanism. This latter procedure typically iterates between segmentation extraction from the image and label refinement using the bounding box information (for example, by zeroing-out the mask outside of boxes). The main issues with such procedures are i) bounding box information is not directly used to extract the segmentation mask, ii) the procedure may be suboptimal as it is hand-designed, and iii) the segmentation becomes ambiguous when multiple boxes overlap.

In this paper, we take a different approach by designing an ancillary segmentation model that forms a per-pixel label distribution given an image and bounding box annotation. This model is easily trained using the fully supervised set ($\boldsymbol{\mathcal{F}}$) and can be used as a training signal for images in $\boldsymbol{\mathcal{W}}$. At inference time, both the image and its bounding box are fed to the network to obtain $p_{anc}(\boldsymbol{y}|\boldsymbol{x}^{(w)}, \boldsymbol{b}^{(w)})$, the segmentation labels distribution.

Our key observation in designing the ancillary model is that encoder-decoder-based segmentation networks typically rely on encoders initialized from an image-classification model (e.g., ImageNet pretrained models). This usually improves the segmentation performance by transferring knowledge from large image-classification datasets. To maintain the same advantage, we augment an encoder-decoder-based segmentation model with a parallel *bounding box encoder* network that embeds bounding box information at different scales (See Fig. 2).

The input to the bounding box encoder is a 3D tensor representing a binarized mask of the bounding boxes and a 3D shape representing the target dimensions for the encoder output. The input mask tensor is resized to the target shape then passed through a 3×3 convolution layer with sigmoid

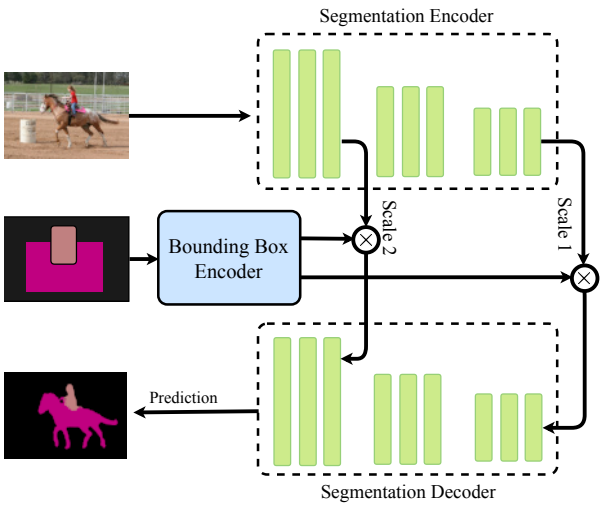

Figure 2: An overview of the ancillary segmentation model. We modify an existing encoder-decoder segmentation model by introducing a bounding box encoder that embeds the box information. The output of the bounding box encoder after passing through a sigmoid activation acts as an attention map. Feature maps at different scales from the encoder are fused (using element-wise-multiplication) with attention maps, then passed to the decoder.

activations. The resulting tensor can be interpreted as an attention map which is element-wise multiplied to the feature maps generated by the segmentation encoder. Fig. 2 shows two paths of such feature maps at two different scales, as in the DeepLabv3+ architecture. For each scale, an attention map is generated, fused with the corresponding feature map using element-wise multiplication, and fed to the decoder. For an image of size $\mathcal{W} \times \mathcal{H} \times 3$, we represent its object bounding boxes using a binary mask of size $\mathcal{W} \times \mathcal{H} \times (C+1)$ that encodes the $C+1$ binary masks. The c^{th} binary mask at a pixel has the value 1 if it is inside one of the bounding boxes of the c^{th} class. A pixel in the background mask has value 1 if it is not covered by any bounding box.

The ancillary model is trained using the cross-entropy loss on the full dataset $\boldsymbol{\mathcal{F}}$:

$$\max_{\boldsymbol{\theta}} \sum_{f \in \boldsymbol{\mathcal{F}}} \log p_{anc}(\boldsymbol{y}^{(f)} | \boldsymbol{x}^{(f)}, \boldsymbol{b}^{(f)}; \boldsymbol{\theta}), \tag{1}$$

which can be expressed analytically under the factorial distribution assumption. This model is held *fixed* for the subsequent experiments.

3.2. No Self-Correction

We empirically observe that the performance of our ancillary model is superior to segmentation models that do not have box information. This is mainly because the bounding box information guides the ancillary model to look for the object inside the box at inference time.

The simplest approach to training the primary model is to train it to predict using ground-truth labels on the fully supervised set $\boldsymbol{\mathcal{F}}$ and the labels generated by the ancillary model on the weak set $\boldsymbol{\mathcal{W}}$. For this "no-self-correction" model the **Self-correction module** in Fig. 1 merely copies the predictions made by the ancillary segmentation model. Training is guided by optimizing:

$$\max_{\boldsymbol{\phi}} \quad \sum_{f \in \boldsymbol{\mathcal{F}}} \log p(\boldsymbol{y}^{(f)} | \boldsymbol{x}^{(f)}; \boldsymbol{\phi}) + \sum_{w \in \boldsymbol{\mathcal{W}}} \sum_{\boldsymbol{y}} p_{anc}(\boldsymbol{y} | \boldsymbol{x}^{(w)}, \boldsymbol{b}^{(w)}; \boldsymbol{\theta}) \log p(\boldsymbol{y} | \boldsymbol{x}^{(w)}; \boldsymbol{\phi}), \tag{2}$$

where the first term is the cross-entropy loss with one-hot ground-truth labels as target and the second term is the cross-entropy with soft probabilistic labels generated by p_{anc} as target. Note that the ancillary model parameterized by $\boldsymbol{\theta}$ is fixed. We call this approach the *no self-correction model* as it relies directly on the ancillary model for training the primary model for examples in $\boldsymbol{\mathcal{W}}$.

3.3. Linear Self-Correction

Eq. 2 relies on the ancillary model to predict label distribution on the weak set. However, this model is trained using only instances of $\boldsymbol{\mathcal{F}}$ without benefit of the data in $\boldsymbol{\mathcal{W}}$. Several recent works [41, 12, 26, 54, 56] have incorporated the information in $\boldsymbol{\mathcal{W}}$ by using the primary model itself (as it is being trained on both $\boldsymbol{\mathcal{F}}$ and $\boldsymbol{\mathcal{W}}$) to extract more accurate label distributions on $\boldsymbol{\mathcal{W}}$.

Vahdat [56] introduced a regularized Expectation-Maximization algorithm that uses a linear combination of KL divergences to infer a distribution over missing labels for general classification problems. The main insight is that the inferred distribution $q(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b})$ over labels should be close to both the distributions generated by the ancillary model $p_{anc}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b})$ *and* the primary model $p(\boldsymbol{y}|\boldsymbol{x})$. However, since the primary model is not capable of predicting the segmentation mask accurately early in training, these two terms are reweighted using a positive scaling factor α:

$$\min_{q} \mathrm{KL}(q(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}) || p(\boldsymbol{y}|\boldsymbol{x})) + \alpha \mathrm{KL}(q(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}) || p_{anc}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b})). \tag{3}$$

The global minimizer of Eq. 3 is obtained as the weighted geometric mean of the two distributions:

$$q(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}) \propto \left(p(\boldsymbol{y}|\boldsymbol{x}) p^{\alpha}_{anc}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}) \right)^{\frac{1}{\alpha+1}}. \tag{4}$$

Since both $p_{anc}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b})$ and $p(\boldsymbol{y}|\boldsymbol{x})$ decompose into a product of probabilities over the components of $\boldsymbol{y}$, and since the distribution over each component is categorical, then $q(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}) = \prod_{m=1}^{M} q_m(\boldsymbol{y}_m|\boldsymbol{x}, \boldsymbol{b})$ is also factorial where the parameters of the categorical distribution over each component are computed by applying softmax activation to the

linear combination of logits coming from primary and ancillary models using $\sigma\Big(\big(\boldsymbol{l}_m+\alpha\,\boldsymbol{l}_m^{anc}\big)/\big(\alpha+1\big)\Big)$. Here, $\sigma(.)$ is the softmax function and, $\boldsymbol{l}_m$ and $\boldsymbol{l}_m^{anc}$ are logits generated by primary and ancillary models for the m^{th} pixel.

Having fixed $q(\boldsymbol{y}|\boldsymbol{x}^{(w)},\boldsymbol{b}^{(w)})$ on the weak set in each iteration of training the primary model, we can train the primary model using:

$$\max_{\boldsymbol{\phi}} \quad \sum_{\mathcal{F}} \log p(\boldsymbol{y}^{(f)}|\boldsymbol{x}^{(f)};\boldsymbol{\phi}) + \sum_{\mathcal{W}}\sum_{\boldsymbol{y}} q(\boldsymbol{y}|\boldsymbol{x}^{(w)},\boldsymbol{b}^{(w)}) \log p(\boldsymbol{y}|\boldsymbol{x}^{(w)};\boldsymbol{\phi}). \tag{5}$$

Note that α in Eq. 3 controls the closeness of q to $p(\boldsymbol{y}|\boldsymbol{x})$ and $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$. With $\alpha=\infty$, we have $q=p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$ and the linear self-correction in Eq. 5 collapses to Eq. 2, whereas $\alpha=0$ recovers $q=p(\boldsymbol{y}|\boldsymbol{x})$. A finite α maintains q close to both $p(\boldsymbol{y}|\boldsymbol{x})$ and $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$. At the beginning of training, $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$ cannot predict the segmentation label distribution accurately. Therefore, we define a schedule for α where α is decreased from a large value to a small value during training of the primary model.

This corrective model is called the *linear self-correction model* as it uses the solution to a linear combination of KL divergences (Eq. 3) to infer a distribution over latent segmentation labels.[2] As the primary model's parameters are optimized during training, α biases the self-correction mechanism towards the primary model.

3.4. Convolutional Self-Correction

One disadvantage of linear self-correction is the hyper-parameter search required for tuning the α schedule during training. In this section, we present an approach that overcomes this difficulty by replacing the linear function with a convolutional network that learns the self-correction mechanism. As a result, the network automatically tunes the mechanism dynamically as the primary model is trained. If the primary model predicts labels accurately, this network can shift its predictions towards the primary model.

Fig. 3 shows the architecture of the convolutional self-correcting model. This small network accepts the logits generated by $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$ and $p(\boldsymbol{y}|\boldsymbol{x})$ models and generates the factorial distribution $q_{conv}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b};\boldsymbol{\lambda})$ over segmentation labels where $\boldsymbol{\lambda}$ represents the parameters of the subnetwork. The convolutional self-correction subnetwork consists of two convolution layers. Both layers use a 3×3 kernel and ReLU activations. The first layer has 128 output feature maps and the second has feature maps based on the number of classes in the dataset.

[2] In principal, logits of $q_m(\boldsymbol{y}_m|\boldsymbol{x},\boldsymbol{b})$ can be obtained by a 1×1 convolutional layer applied to the depth-wise concatenation of $\boldsymbol{l}$ and $\boldsymbol{l}^{anc}$ with a fixed averaging kernel. This originally motivated us to develop the convolutional self-correction model in Sec. 3.4 using trainable kernels.

The challenge here is to train this subnetwork such that it predicts the segmentation labels more accurately than either $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$ or $p(\boldsymbol{y}|\boldsymbol{x})$. To this end, we introduce an additional term in the objective function, which trains the subnetwork using training examples in $\mathcal{F}$ while the primary model is being trained on the whole dataset:

$$\max_{\boldsymbol{\phi},\boldsymbol{\lambda}} \quad \sum_{\mathcal{F}} \log p(\boldsymbol{y}^{(f)}|\boldsymbol{x}^{(f)};\boldsymbol{\phi}) + \sum_{\mathcal{W}}\sum_{\boldsymbol{y}} q_{conv}(\boldsymbol{y}|\boldsymbol{x}^{(w)},\boldsymbol{b}^{(w)};\boldsymbol{\lambda}) \log p(\boldsymbol{y}|\boldsymbol{x}^{(w)};\boldsymbol{\phi}) + \sum_{\mathcal{F}} \log q_{conv}(\boldsymbol{y}^{(f)}|\boldsymbol{x}^{(f)},\boldsymbol{b}^{(f)};\boldsymbol{\lambda}), \tag{6}$$

where the first and second terms train the primary model on $\mathcal{F}$ and $\mathcal{W}$ (we do not backpropagate through q in the second term) and the last term trains the convolutional self-correcting network.

Because the q_{conv} subnetwork is initialized randomly, it is not able to accurately predict segmentation labels on $\mathcal{W}$ early on during training. To overcome this issue, we propose the following pretraining procedure:

1. Initial training of ancillary model: As with the previous self-correction models, we need to train the ancillary model. Here, half of the fully supervised set ($\mathcal{F}$) is used for this purpose.

2. Initial training of conv. self-correction network: The fully supervised data ($\mathcal{F}$) is used to train the primary model and the convolutional self-correcting network. This is done using the first and last terms in Eq. 6.

3. The main training: The whole data ($\mathcal{F}$ and $\mathcal{W}$) are used to *fine-tune* the previous model using the objective function in Eq. 6.

The rationale behind using half of $\mathcal{F}$ in stage 1 is that if we used all $\mathcal{F}$ for training the $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$ model, it would train to predict the segmentation mask almost perfectly on this set, therefore, the subsequent training of the convolutional self-correcting network would just learn to rely on $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$. To overcome this training issue, the second half of $\mathcal{F}$ is held out to help the self-correcting network to learn how to combine $p_{anc}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{b})$ and $p(\boldsymbol{y}|\boldsymbol{x})$.

4. Experiments

In this section, we evaluate our models on the PASCAL VOC 2012 and Cityscapes datasets. Both datasets contain object segmentation and bounding box annotations. We split the full dataset annotations into two parts to simulate a fully and semi-supervised setting. Similar to [9, 41], performance is measured using the mean intersection-over-union (mIOU) across the available classes.

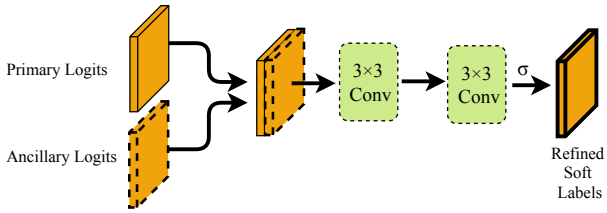

Figure 3: Convolutional self-correction model learns refining the input label distributions. The subnetwork receives logits from the primary and ancillary models, then concatenates and feeds the output to a two-layer CNN.

Training: We use the public Tensorflow [1] implementation of DeepLabv3+ [9] as the primary model. We use an initial learning rate of 0.007 and train the models for 30,000 steps from the ImageNet-pretrained *Xception-65* model [9].[3] For all other parameters we use standard settings suggested by other authors. At evaluation time, we apply flipping and multi-scale processing for images as in [9]. We use 4 GPUs, each with a batch of 4 images.

We define the following baselines in all our experiments:

1. **Ancillary Model:** This is the ancillary model, introduced in Sec. 3.1, predicts semantic segmentation labels given an image and its object bounding boxes. This model is expected to perform better than other models as it uses bounding box information.

2. **No Self-correction:** This is the primary model trained using the model introduced in Sec. 3.2.

3. **Lin. Self-correction:** This is the primary model trained with linear self-correction as in Sec. 3.3.

4. **Conv. Self-correction:** The primary model trained with the convolutional self-correction as in Sec. 3.4.

5. **EM-fixed Baseline:** Since our linear self-correction model is derived from a regularized EM model [56], we compare our model with Papandreou *et al.* [41] which is also an EM-based model. We implemented their EM-fixed baseline with DeepLabv3+ for fair comparison. This baseline achieved the best results in [41] for semi-supervised learning.

For linear self-correction, α controls the weighting in the KL-divergence bias with large α favoring the ancillary model and small α favoring the primary model. We explored different starting and ending values for α with an exponential decay in-between. We find that a starting value of $\alpha = 30$ and the final value of $\alpha = 0.5$ performs well for both datasets. This parameter setting is robust as moderate changes of these values have little effect.

[3]Note that, we do not initialize the parameters from a MS-COCO pretrained model.

4.1. PASCAL VOC Dataset

In this section, we evaluate all models on the PASCAL VOC 2012 segmentation benchmark [16]. This dataset consists of 1464 training, 1449 validation, and 1456 test images covering 20 foreground object classes and one background class for segmentation. An auxiliary dataset of 9118 training images is provided by [18]. We suspect, however, that the segmentation labels of [18] contain a small amount of noise. In this section, we refer to the union of the original PASCAL VOC training dataset and the auxiliary set as the *training* set. We evaluate the models mainly on the validation set and the best model is evaluated only once on the test set using the online evaluation server.

In Table 1, we show the performance of different variants of our model for different sizes of the fully supervised set $\mathcal{F}$. The remaining examples in the training set are used as $\mathcal{W}$. We make several observations from Table 1: i) The ancillary model that predicts segmentation labels given an image and its object bounding boxes performs well even when it is trained with a training set as small as 200 images. This shows that this model can also provide a good training signal for the weak set that lacks segmentation labels. ii) The linear self-correction model typically performs better than no self-correction model supporting our idea that combining the primary and ancillary model for inferring segmentation labels results in better training of the primary model. iii) The convolutional self-correction model performs comparably or better than the linear self-correction while eliminating the need for defining an α schedule. Fig. 4 shows the output of these models.

# images in $\mathcal{F}$	200	400	800	1464
Ancillary Model	81.57	83.56	85.36	86.71
No Self-correction	78.75	79.19	80.39	80.34
Lin. Self-correction	**79.43**	79.59	**80.69**	81.35
Conv. Self-correction	78.29	**79.63**	80.12	**82.33**

Table 1: Ablation study of models on the **PASCAL VOC 2012 validation** set using mIOU for different sizes of $\mathcal{F}$. For the last three rows, the remaining images in the training set is used as $\mathcal{W}$, i.e. $W + F = 10582$.

Table 2 compares the performance of our models against different baselines and published results. In this experiment, we use 1464 images as $\mathcal{F}$ and 9118 images originally from the auxiliary dataset as $\mathcal{W}$. Both self-correction models achieve similar results and outperform other models.

Surprisingly, our semi-supervised models outperform the fully supervised model. We hypothesize two possible explanations for this observation. Firstly, this may be due to label noise in the 9k auxiliary set [18] that negatively affects performance of Vanilla DeepLapv3+. As ev-

idence, Fig. 5 compares the output of the ancillary model with ground-truth annotations and highlights some of improperly labeled instances. Secondly, the performance gain may also be due to explicit modeling of label uncertainty and self-correction. To test this hypothesis, we train vanilla DeepLabv3+ on only 1.4K instances in the original PASCAL VOC 2012 training set[4] and obtain 68.8% mAP on the validation set. However, if we train the convolutional self-correction model on the same training set and allow the model to *refine* the ground truth labels using self-correction[5], we get mAP as high as 76.88% (the convolutional self correction on top of bounding boxes yields 75.97% mAP). This indicates that modeling noise with robust loss functions and allowing for self-correction may significantly improve the performance of segmentation models. This is consonant with self-correction approaches that have been shown to be effective for edge detection [66, 2], and is in contrast to common segmentation objectives which train models using cross-entropy with one-hot annotation masks. Very similar to our approach and reasoning, [67] uses logits to train a lightweight pose estimation model using knowledge distillation technique.

Unfortunately, the state-of-the-art models are still using the older versions of DeepLab. It was infeasible for us to either re-implement most of these approaches using DeepLabv3+ or re-implement our work using old versions. The only exception is EM-fixed baseline [41]. Our re-implementation using DeepLabv3+ achieves 79.25% on the validation set while the original paper has reported 64.6% using DeepLabv1. In the lower half of Table 2, we record previously published results (using older versions of DeepLab). A careful examination of the results show that *our work is superior* to previous work as our semi-supervised models outperform the fully supervised model while previous work normally do not.

Finally, comparing Table 1 and 2, we see that with $F = 200$ and $W = 10382$, our linear self-correction model performs similarly to DeepLabv3+ trained with the whole dataset. Using the labeling cost reported in [5], this theoretically translates to a $\sim$7x reduction in annotation cost.

4.2. Cityscapes Dataset

In this section we evaluate performance on the Cityscapes dataset [11] which contains images collected from cars driving in cities during different seasons. This dataset has good quality annotations, however some instances are over/under segmented. It consists of 2975 training, 500 validation, and 1525 test images covering 19 foreground object classes (stuff and object) for the segmentation task. However, 8 of these classes are flat or construction labels (e.g., road, sidewalk, building, and etc.), and a very few bounding boxes of such classes cover the whole scene. To create an object segmentation task similar to the PASCAL VOC dataset, we use only 11 classes (pole, traffic light, traffic sign, person, rider, car, truck, bus, train, motorcycle, and bicycle) as foreground classes and all other classes are assigned as background. Due to this modification of labels, we report the results only on the validation set, as the test set on server evaluates on all classes. We do not use the coarse annotated training data in the dataset.

[4]The auxiliary set is excluded to avoid potentially noisy labels.

[5]For this experiment 1.1K images are used as $\mathcal{F}$ and 364 images as $\mathcal{W}$. For $\mathcal{W}$, we let self-correction model to refine the original ground-truth labels.

Data Split		Method	Val	Test
F	W			
1464	9118	No Self-Corr.	80.34	81.61
1464	9118	Lin. Self-Corr.	81.35	81.97
1464	9118	Conv. Self-Corr.	**82.33**	**82.72**
1464	9118	EM-fixed Ours [41]	79.25	-
10582	-	Vanilla DeepLabv3+ [9]	81.21	-
1464	9118	BoxSup-MCG [12]	63.5	-
1464	9118	EM-fixed [41]	65.1	-
1464	9118	M $\cap$ G+ [26]	65.8	-
1464	9118	FickleNet [30]	65.8	-
1464	9118	Song *et al.* [50]	67.5	-
10582	-	Vanilla DeepLabv1 [6]	69.8	-

Table 2: Results on **PASCAL VOC 2012 validation and test** sets. The last three rows report the performance of previous semi-supervised models with the same annotation.

# images in $\mathcal{F}$	200	450	914
Ancillary Model	79.4	81.19	81.89
No Self-correction	**73.69**	75.10	75.44
Lin. Self-correction	73.56	75.24	76.22
Conv. Self-correction	69.38	**77.16**	**79.46**

Table 3: Ablation study of our models on **Cityscapes validation** set using mIOU for different sizes of $\mathcal{F}$. For the last three rows, the remaining images in the training set are used as $\mathcal{W}$, i.e., $W + F = 2975$.

Data Split		Method	mIOU
F	W		
914	2061	No Self-Corr.	75.44
914	2061	Lin. Self-Correction	76.22
914	2061	Conv. Self-Correction	**79.46**
914	2061	EM-fixed [41]	74.97
2975	-	Vanilla DeepLabv3+$_{ours}$	77.49

Table 4: Results on **Cityscapes validation** set. 30% of the training examples is used as $\mathcal{F}$, and the remaining as $\mathcal{W}$.

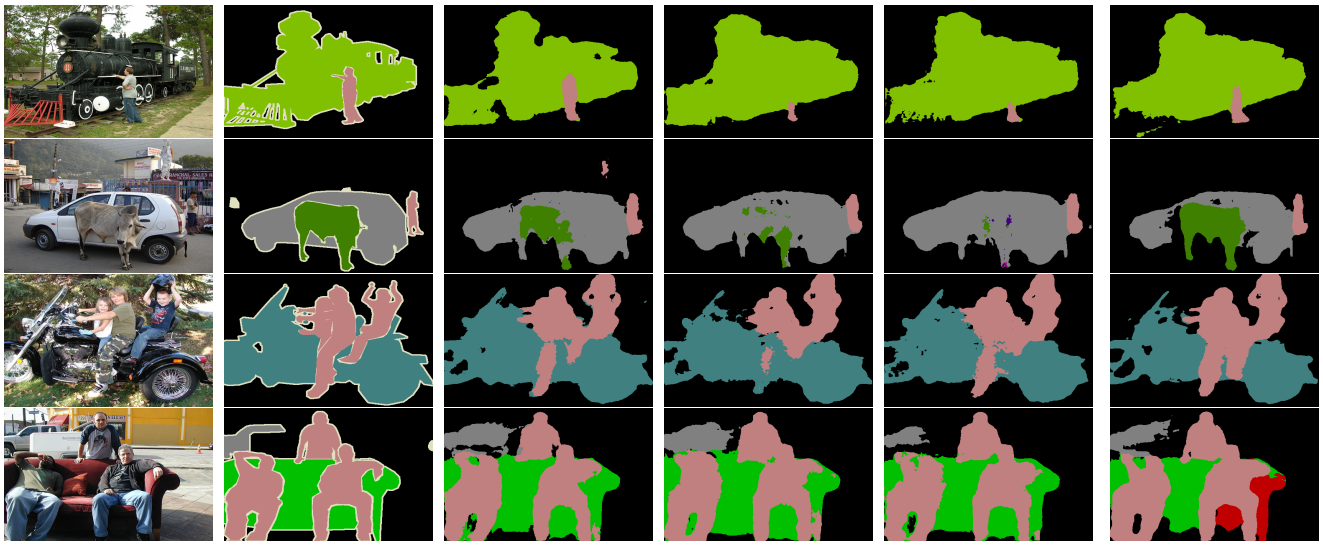

Figure 4: Qualitative results on the **PASCAL VOC 2012 validation** set. The last four columns represent the models in column 1464 of Table 1. The Conv. Self-correction model typically segments objects better than other models.

Figure 5: Qualitative results on the **PASCAL VOC 2012 auxiliary** (the weak set). The heatmap of a single class for the ancillary model is shown for several examples. The ancillary model can successfully correct the labels for *missing* or *over-segmented* objects in these images (marked by ellipses).

Table 3 reports the performance of our model for an increasing number of images as $\mathcal{F}$, and Table 4 compares our models with several baselines similar to the previous dataset. The same conclusion and insights observed on the PASCAL dataset hold for the Cityscapes dataset indicating the efficacy of our self-corrective framework.

5. Conclusion

In this paper, we have proposed a semi-supervised framework for training deep CNN segmentation models using a small set of fully labeled and a set of weakly labeled images (boxes annotations only). We introduced two mechanisms that enable the underlying primary model to correct the weak labels provided by an ancillary model. The proposed self-correction mechanisms combine the predictions made by the primary and ancillary model either using a linear function or trainable CNN. The experiments show that our proposed framework outperforms previous semi-supervised models on both the PASCAL VOC 2012 and Cityscapes datasets. Our framework can also be applied to the instance segmentation task [21, 74, 72], but we leave further study of this to future work.

References

[1] Martín Abadi, Ashish Agarwal, and et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. 2016. 6

[2] David Acuna, Amlan Kar, and Sanja Fidler. Devil is in the edges: Learning semantic boundaries from noisy annotations. In *CVPR*, 2019. 7

[3] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[4] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. 2015. 1, 2

[5] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *European Conference on Computer Vision (ECCV)*, 2016. 1, 2, 7

[6] 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 *International Conference on Learning Representations (ICLR)*, 2015. 2, 7

[7] 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 (TPAMI)*, 2017. 2

[8] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. 2017. 2

[9] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision (ECCV)*, 2018. 1, 2, 5, 6, 7

[10] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[11] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 7

[12] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 1, 2, 3, 4, 7

[13] Mostafa Dehghani, Arash Mehrjou, Stephan Gouws, Jaap Kamps, and Bernhard Schölkopf. Fidelity-weighted learning. In *International Conference on Learning Representations (ICLR)*, 2018. 2

[14] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015. 1

[15] Thibaut Durand, Taylor Mordan, Nicolas Thome, and Matthieu Cord. WILDCAT: weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[16] Mark Everingham, S. M. Ali Eslami, Luc J. Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision (IJCV)*, 2015. 6

[17] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[18] Bharath Hariharan, Pablo Arbelaez, Lubomir D. Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *IEEE International Conference on Computer Vision (ICCV)*, 2011. 6

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1

[20] Seunghoon Hong, Donghun Yeo, Suha Kwak, Honglak Lee, and Bohyung Han. Weakly supervised semantic segmentation using web-crawled videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[21] Ronghang Hu, Piotr Dollr, Kaiming He, Trevor Darrell, and Ross Girshick. Learning to segment every thing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 8

[22] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[23] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang. Adversarial learning for semi-supervised semantic segmentation. In *British Machine Vision Conference (BMVC)*, 2018. 2

[24] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Regularizing very deep neural networks on corrupted labels. In *International Conference on Machine Learning (ICML)*, 2018. 2

[25] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014. 1

[26] Anna Khoreva, Rodrigo Benenson, Jan Hendrik Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 3, 4, 7

[27] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected CRFs with Gaussian edge potentials. In *Advances in Neural Information Processing Systems (NIPS)*, 2011. 2

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012. 1

[29] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In *Conference on Computer Vision and Pattern Recognition (CPRV)*, pages 2169–2178. IEEE Computer Society, 2006. 2

[30] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 7

[31] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[32] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian D. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[33] Guosheng Lin, Chunhua Shen, Anton Van Den Hengel, and Ian Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[34] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 1

[35] Wei Liu, Andrew Rabinovich, and Alexander Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015. 2

[36] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen Change Loy, and Xiaoou Tang. Semantic image segmentation via deep parsing network. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 2

[37] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[38] Ishan Misra, C. Lawrence Zitnick, Margaret Mitchell, and Ross Girshick. Seeing through the human reporting bias: Visual classifiers from noisy human-centric labels. In *CVPR*, 2016. 2

[39] Volodymyr Mnih and Geoffrey E. Hinton. Learning to label aerial images from noisy data. In *International Conference on Machine Learning (ICML)*, pages 567–574, 2012. 2

[40] Nagarajan Natarajan, Inderjit S. Dhillon, Pradeep K. Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pages 1196–1204, 2013. 2

[41] 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 *IEEE International Conference on Computer Vision (ICCV)*, 2015. 1, 2, 3, 4, 5, 6, 7

[42] Giorgio Patrini, Alessandro Rozza, Aditya Menon, Richard Nock, and Lizhen Qu. Making neural networks robust to label noise: A loss correction approach. In *Computer Vision and Pattern Recognition*, 2017. 2

[43] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 1

[44] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *International Conference on Machine Learning (ICML)*, 2018. 2

[45] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 1

[46] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015. 2

[47] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*. ACM, 2004. 2, 3

[48] Alexander G Schwing and Raquel Urtasun. Fully connected deep structured networks. *arXiv preprint arXiv:1503.02351*, 2015. 2

[49] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014. 1

[50] 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 *Computer Vision and Pattern Recognition (CVPR)*, 2019. 7

[51] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 2

[52] Sainbayar Sukhbaatar, Joan Bruna, Manohar Paluri, Lubomir Bourdev, and Rob Fergus. Training convolutional networks with noisy labels. *arXiv preprint arXiv:1406.2080*, 2014. 2

[53] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep convolutional network cascade for facial point detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3476–3483, 2013. 1

[54] Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 4

[55] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with

3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 1

[56] Arash Vahdat. Toward robustness against label noise in training deep discriminative neural networks. In *Neural Information Processing Systems (NIPS)*, 2017. 1, 2, 4, 6

[57] Arash Vahdat and Greg Mori. Handling uncertain tags in visual recognition. In *International Conference on Computer Vision (ICCV)*, 2013. 2

[58] Arash Vahdat, Guang-Tong Zhou, and Greg Mori. Discovering video clusters from visual features and noisy tags. In *European Conference on Computer Vision (ECCV)*, 2014. 2

[59] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6575–6583. IEEE, 2017. 2

[60] Xiang Wang, Shaodi You, Xi Li, and Huimin Ma. Weakly-supervised semantic segmentation by iteratively mining common object features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[61] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[62] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[63] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[64] Jia Xu, Alexander G. Schwing, and Raquel Urtasun. Learning to segment under various forms of weak supervision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[65] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representations (ICLR)*, 2015. 2

[66] Zhiding Yu, Weiyang Liu, Yang Zou, Chen Feng, Srikumar Ramalingam, BVK Vijaya Kumar, and Jan Kautz. Simultaneous edge alignment and learning. In *European Conference on Computer Vision (ECCV)*, 2018. 7

[67] Feng Zhang, Xiatian Zhu, and Mao Ye. Fast human pose estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 7

[68] Wei Zhang, Sheng Zeng, Dequan Wang, and Xiangyang Xue. Weakly supervised semantic segmentation for social images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[69] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European Conference on Computer Vision*, pages 94–108. Springer, 2014. 1

[70] Zhilu Zhang and Mert R Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Neural Information Processing Systems (NIPS)*, 2018. 2

[71] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2

[72] Xiangyun Zhao, Shuang Liang, and Yichen Wei. Pseudo mask augmented object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 8

[73] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *International Conference on Computer Vision (ICCV)*, pages 1529–1537, 2015. 2

[74] Yanzhao Zhou, Yi Zhu, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Weakly supervised instance segmentation using class peak response. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 8

[75] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 146–155, 2016. 1

[76] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning transferable architectures for scalable image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1