Simpler Does It: Generating Semantic Labels with Objectness Guidance

Md Amirul Islam[1,4]
cs.ryerson.ca/~amirul
Matthew Kowal[2,4]
mkowal2.github.io
Sen Jia[5]
github.com/SenJia
Konstantinos G. Derpanis[2,4,6]
www.eecs.yorku.ca/~kosta
Neil D. B. Bruce[3,4]
socs.uoguelph.ca/~brucen

[1] Ryerson University, Canada
[2] York University, Canada
[3] University of Guelph, Canada
[4] Vector Institute for AI, Canada
[5] Toronto AI Lab, LG
[6] Samsung AI Centre Toronto

Abstract

Existing weakly or semi-supervised semantic segmentation methods utilize image or box-level supervision to generate pseudo-labels for weakly labeled images. However, due to the lack of strong supervision, the generated pseudo-labels are often noisy near the object boundaries, which severely impacts the network's ability to learn strong representations. To address this problem, we present a novel framework that generates pseudo-labels for training images, which are then used to train a segmentation model. To generate pseudo-labels, we combine information from: (i) a class agnostic 'objectness' network that learns to recognize object-like regions, and (ii) either image-level or bounding box annotations. We show the efficacy of our approach by demonstrating how the objectness network can naturally be leveraged to generate object-like regions for *unseen* categories. We then propose an end-to-end multi-task learning strategy, that jointly learns to segment semantics and objectness using the generated pseudo-labels. Extensive experiments demonstrate the high quality of our generated pseudo-labels and effectiveness of the proposed framework in a variety of domains. Our approach achieves better or competitive performance compared to existing weakly-supervised and semi-supervised methods.

1 Introduction

State-of-the-art methods for semantic segmentation [4, 9, 10, 11, 19, 19, 27, 28, 31, 32, 36, 37, 47, 50, 51, 60, 68, 70] are founded on fully convolutional networks (FCN) [50] to segment semantic objects in an end-to-end manner. A caveat of such training is that it requires supervision with an extensive amount of pixel-level annotations. Since the expense for generating semantic segmentation annotations is large, a natural solution is to address the problem of semantic segmentation with one of two common supervision settings, weakly or semi-supervised.

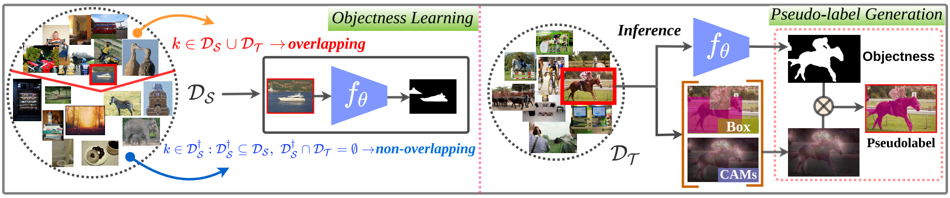


Figure 1: **Left:** An illustration of our process for generating high-quality semantic segmentation pseudo-labels for a target dataset, $\mathcal{D}_\mathcal{T}$. We first train a objectness network, f_θ, on a source dataset under one of two data settings, (*overlapping* ($\mathcal{D}_\mathcal{S}$) or *non-overlapping* ($\mathcal{D}_\mathcal{S}^\dagger$) categories ($k$) with $\mathcal{D}_\mathcal{T}$), that learns to generate a class-agnostic objectness prior. **Right:** Then, we use either Class Activation Maps (CAMs) [71] or bounding box proposals combined with a class agnostic objectness prior to generate a pseudo-label.

In the weakly supervised semantic segmentation (WSSS) setting, labels used during training contain only partial information. Recently proposed WSSS methods utilize image-level labels [2, 3, 7, 12, 17, 18, 23, 25, 35, 43, 44, 63, 69], scribbles [46], or bounding box [15, 38, 58] supervision to learn semantic masks. Most of these methods rely on incorporating additional guidance to obtain the location and shape information. A common way to obtain location cues from class labels is by using *Class Activation Maps* (CAMs) [71] as it roughly localizes semantic regions of each class. However, utilizing CAMs directly as supervision can be problematic as they roughly localize objects and cannot capture detailed object boundaries between different semantic regions. Recent works have addressed this issue in a variety of ways [2, 3, 42, 54], one of the most effective being the use of object guidance via the use of class agnostic saliency [25, 35, 43, 69]. Similarly, bounding box based methods [25, 38, 40, 45, 58] typically rely on generating rough pseudo-labels by applying the unsupervised CRF [39], MCG [55], or GrabCut [57] methods to remove irrelevant regions from the semantic region proposal in an iterative way to obtain stronger pseudo-labels at each iteration. However, the quality gap between the pseudo-labels and groundtruth is typically *large* for the CAM-based and bounding box-based approaches. Furthermore, iterative procedures and complex pipelines can make the data generation process for these methods computationally expensive and time consuming.

In the semi-supervised semantic segmentation (SSSS) setting, the groundtruth annotations are used but only for a fraction of the total number of training examples, e.g., 10% of the labels [59]. Similar to the techniques used in WSSS methods, pseudo-labels are then generated for the unlabelled data (e.g., by using additional image-level annotations [25, 38, 43, 63, 64]). Recent work [24] introduced a partially supervised training paradigm which learns to segment everything using a portion of box and mask annotations. However, these methods still require labour-intensive pixel-level *semantic* annotations and the performance heavily depends on the *quantity* of the labeled data and the *quality* of the generated pseudo-labels.

In the light of the highlighted issues that arise in WSSS and SSSS methods, we propose a novel simple yet effective pipeline which transfers 'objectness' knowledge to weakly labeled images for learning semantic segmentation. The intuition behind using the objectness guidance instead of widely used saliency-based approaches [52, 63, 67, 69] is that groundtruth saliency masks inherently ignore objects near the border of the image due to the well-known centre bias [1, 5]. Recent works [62, 64] also utilize the objectness prior to refine the semantic proposals. There are two key differences between our work and [64]. First is the use of a source dataset. [64] obtains the objectness prior strictly from the target data distribution, which is arguably an easier problem to solve. However in our work, we strictly prohibit the use of per-pixel labels from the *target* dataset and only use a *source* dataset (i.e., COCOStuff) for the objectness prior. We argue that using COCOStuff as the *source* data (instead of VOC like in [64]) will allow our objectness network to generate better pseudo-labels for a more diverse set of categories and can be generalized well to different target datasets. Second, during the segmentation network training, [64] uses the semantic segmentation labels for the *strong* categories (i.e., the classes used to train their objectness network), while in our settings we only use pseudo-labels when training the semantic segmentation network.

The key component of our pipeline is the pseudo-label generation approach (see Fig. 1), where we first train an objectness network on a source dataset which generates a class agnostic objectness prior. We then combine this prior with weak semantic proposals (e.g., image or box-level) to generate semantic segmentation labels for a target dataset. We further show that the objectness prior is robust enough to generalize the objectness knowledge onto categories that have *never been seen* by the objectness network; when the source dataset has no class *overlap* with the target dataset (i.e., the *non-overlapping case*). We view the *non-overlapping* setting as comparable with *weak-supervision*, as the objectness model has no direct understanding of the shape of the target domain classes (unlike previous methods [52, 63, 67, 69] which use overlapping groundtruth saliency annotations). In contrast, the *overlapping* setting (i.e., the class agnostic source dataset contains objects found in the target dataset) is comparable (but has less supervision) to *semi-supervision* as class-agnostic (i.e., binary) segmentation annotations are used. Finally, for segmentation learning, we adopt a multi-task joint-learning [13, 21, 29, 30, 69] based **S**emantic **O**bjectness **Net**work (denoted as SONet), with the addition of an 'objectness branch', that explicitly models the relationship between *semantics* and *objectness*. We summarize our main contributions as follows:

1. We introduce a simple yet effective pseudo-label generation technique that combines a class agnostic 'objectness' prior with semantic region proposals. The flexibility of our technique is demonstrated by its ability to incorporate either image *or* box-level labels into the pseudo-label generation pipeline.

2. We propose a joint learning based Semantic Objectness Network, SONet, that improves the semantic segmentation quality through objectness guidance.

3. We present an extensive set of experimental results which demonstrates the effectiveness of our proposed method in both the simplicity of the pseudo-label generation process as well as the quality of the pseudo-labels. Our proposed approach achieves competitive performance compared to existing WSSS methods and outperforms SSSS methods without ever using groundtruth semantic segmentation supervision.

2 Proposed Framework

Our pipeline consists of two key components. First, we generate pseudo-labels for training images by combining our generated objectness prior with weak semantic proposals, which are produced from either image labels or box annotations (Sec. 2.1). Second, we introduce our multi-task model, SONet, that jointly learns to segment both semantic categories and a binary 'objectness' mask, which enforces richer boundary detail and semantic information (Sec. 2.2).

2.1 Semantic Pseudo-Label Generation

Our pseudo-label generation process consists of two separate components. We first describe the procedure behind training the 'objectness' network which is designed to obtain detailed boundary information for any object-like region. Next, we describe two different techniques for generating semantic pseudo-labels by combining the output of the objectness network with semantic region proposals, which are obtained from either image-level class labels or bounding box annotations.

Training an Objectness Network. Pixel objectness [66] quantifies how likely a pixel belongs to an object of any class (i.e., other than "stuff" classes like background, grass, sky, sidewalks, etc.), and should be high even for objects unseen during training. We use DeepLabv3 network [10], $\phi_{\mathcal{P}}$, on a source dataset, $\mathcal{D}_{\mathcal{S}}$, to learn an objectness prior from the 'things' label. We use a weak form of the COCOStuff dataset, denoted as *COCO-Binary* and consider it as the source dataset, $\mathcal{D}_{\mathcal{S}}$. More specifically, we generate *COCO-Binary* by *removing all semantic labels from the COCOStuff dataset* so what remains is binary maps where all the *things* categories are assigned to the label one, and the *stuff* categories to zero. We then train the objectness network, $\phi_{\mathcal{P}}$, on the source dataset, $\mathcal{D}_{\mathcal{S}}$, under two different settings which outputs a pixel-wise 'objectness score' (similar to the saliency detection models). In the first setting, we include all the images from the source data, $\mathcal{D}_{\mathcal{S}}$, regardless of whether the objects found in $\mathcal{D}_{\mathcal{S}}$ images overlap with target data, $\mathcal{D}_{\mathcal{T}}$. In the second setting, we create a subset of $\mathcal{D}_{\mathcal{S}}$ by excluding those images containing any categories which overlap with $\mathcal{D}_{\mathcal{T}}$ categories. We can formalize the *overlapping* and *non-overlapping* settings as follows:

$$k \in \begin{cases} \mathcal{D}_{\mathcal{S}} & \text{overlapping} \\ \mathcal{D}_{\mathcal{S}}^{\dagger} : \mathcal{D}_{\mathcal{S}}^{\dagger} \subseteq \mathcal{D}_{\mathcal{S}}, \;\; \mathcal{D}_{\mathcal{S}}^{\dagger} \cap \mathcal{D}_{\mathcal{T}} = \emptyset & \text{non-overlapping,} \end{cases} \tag{1}$$

where k denotes the set of object classes contained in *COCO-Binary* used to train the objectness model, $\phi_{\mathcal{P}}$. $\mathcal{D}_{\mathcal{S}}^{\dagger}$ represents the *non-overlapping* subset where there is no semantic category overlap between $\mathcal{D}_{\mathcal{S}}^{\dagger}$ and $\mathcal{D}_{\mathcal{T}}$. Note that the semantic annotations are solely used to generate the subset of non-overlapping data, $\mathcal{D}_{\mathcal{S}}^{\dagger}$, and is not required for training the objectness model, $\phi_{\mathcal{P}}$. We believe the *non-overlapping* setting is more challenging than saliency-based WSSS [52, 63, 67, 69], because those methods contain semantic overlap within the source and target data. In both settings, we train the objectness classifier using the class-agnostic segmentation groundtruth and use the binary cross entropy loss function. The main goal of the objectness classifier is to learn a strong objectness representation [33] that contributes towards creating pseudo-labels for semantic supervision.

Class-Driven Pseudo-labels. CAM [71] is widely used as a weak source of supervision as it roughly localizes semantic object areas. Following previous works [2, 3], we first generate CAMs for training images by adopting the method of [71] using a multi-label image classification network. For a fair comparison, we use a ResNet-50 [20] model as the classification network, as used in other CAM-based methods [2, 3, 25, 67]. We directly utilize the raw CAMs to generate pseudo-labels by thresholding their confidence scores for each class label at every pixel predicted to be an object by the class agnostic objectness network (see Fig. 1(B)). We can formalize this procedure as follows:

$$\mathcal{G}_{i,j}^{\mathcal{C}_m} = \begin{cases} \underset{k \in K}{\arg\max}(\mathcal{C}_m(i,j,k)) & \text{if } \mathcal{O}_{i,j} > 0 \;\; \mathcal{C}_m(i,j) > \delta \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

where $\mathcal{G}_{i,j}^{\mathcal{C}_m}$ denotes the pseudo-label value at pixel (i,j), K is the set of class indices, $\mathcal{O}_{i,j}$ is the objectness score, $\mathcal{C}_m$ is the non-thresholded CAM proposals, and δ is a threshold (we use $\delta = 0.01$ in all experiments).

Box-Driven Pseudo-labels. The simplest box-driven pseudo-labels can be obtained by filling the bounding box annotations with the corresponding class label. Some methods [38, 58] use semi-automatic segmentation techniques (e.g., CRF [53], GrabCut [57]) to further refine the box annotations, as rectangular regions contain a significant number of incorrectly labeled background pixels. However, these techniques are time consuming and the quality of the pseudo-label is lacking. To address this challenge, we propose an approach to generate pseudo-labels using the class agnostic objectness masks, $\mathcal{O}$, and the box annotations, $\mathcal{B}$.

Following common practice [26, 38, 58], if any two bounding boxes overlap, we assume the box with smaller area appears in front. Additionally, if the overlap between any box and the largest box in the image is greater than some threshold, we only keep the inner 60% of the box and fill the rest of the box as 255 (which is ignored during training). The intuition behind the *ignoring strategy* is simply trading-off lower recall (ignore more pixels where high-degree of overlap occurs) for higher precision (more pixels are correctly labelled). We then mask the resulting box proposal, $\mathcal{B}$, with the objectness map, $\mathcal{O}$, to filter out the irrelevant regions from $\mathcal{B}$ and $\mathcal{O}$, and only keep the regions overlapping the object of interest. We set any pixel to the background class if it does not overlap any boxes. Formally, for each bounding box, $\mathcal{B}_k$, $k \in \{1, \ldots, n\}$, in an image:

$$\mathcal{G}_{\text{ign}}^{\mathcal{B}}(i,j) = \begin{cases} \mathcal{B}_k^{\text{cls}} & \text{if } \mathcal{O}_{i,j} > 0,\ \mathcal{B}_0 \cap \mathcal{B}_k < \alpha,\ (i,j) \in \mathcal{B}_k \\ \mathcal{B}_k^{\text{cls}} & \text{if } \mathcal{O}_{i,j} > 0,\ \mathcal{B}_0 \cap \mathcal{B}_k > \alpha,\ (i,j) \in \mathcal{B}_k^{\text{in}} \\ 255 & \text{if } \mathcal{O}_{i,j} > 0,\ \mathcal{B}_0 \cap \mathcal{B}_k > \alpha\ (i,j) \in \mathcal{B}_k^{\text{out}} \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

where $\mathcal{B}_0$ denotes the largest box, n is the number of boxes in each image, $\mathcal{B}_k^{\text{out}}$ is the outer 40% of the bounding box's area, $\mathcal{B}_k^{\text{in}}$ is the inner 60% of the bounding box, '$\cap$' calculates the area of intersection between two bounding boxes, and α is a threshold (we set $\alpha = 0.3$).

2.2 Semantic Objectness Network: SONet

The Semantic Objectness Network (SONet) consists of a segmentation network and an objectness module. The objectness module receives the output of the segmentation network as input, and predicts a binary mask (see Fig. 2).

Network Architecture. We use DeepLabv3 [10] as our segmentation network, $\phi_{\mathcal{S}}$, which outputs feature maps of 1/16 of the input image size. Given an input image, $\mathcal{I}_t$, $\phi_{\mathcal{S}}$ generates a semantic segmentation map, $\mathcal{S}$, using the pseudo-label as supervision. The objectness module, $\phi_{\mathcal{O}}$, takes $\mathcal{S}$ as input and consists of a stack of five convolutional layers that includes batch normalization and ReLU layers (see Table S1 in the supplementary for architectural details). We use a 3×3 kernel in the first four

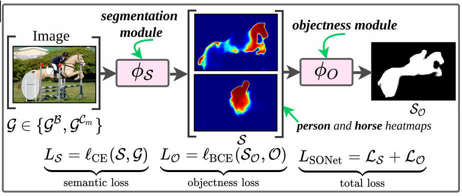


Figure 2: Overview of our proposed SONet. With the generated pseudo-labels, we jointly train SONet on the target dataset, $\mathcal{D}_{\mathcal{T}}$, to predict semantics, $\mathcal{S}$, and objectness, $\mathcal{S}_{\mathcal{O}}$.

convolution layers and use a 1×1 kernel in the last layer which outputs the objectness map, $\mathcal{S}_{\mathcal{O}}$. The procedure for obtaining the semantic and objectness maps can be described as:

$$\mathcal{S} = \phi_{S}(\mathcal{I}_t; \mathbf{W}_T), \quad \mathcal{S}_{\mathcal{O}} = \phi_O(\mathcal{S};\, \mathbf{W}_O), \tag{4}$$

where $\mathbf{W}_T$ and $\mathbf{W}_O$ refer to trainable weights for the $\phi_{\mathcal{S}}$ and ϕ_O modules, respectively.

Joint Learning of Semantics and Objectness. We train our proposed SONet method using the generated pixel-level semantic and objectness pseudo-labels in an end-to-end manner (see Fig. 2). Let $\mathcal{D}_{\mathcal{T}} = (\mathcal{I}_t, \mathcal{G}, \mathcal{O})$ denote the target semantic segmentation dataset with images $\mathcal{I}_t$, pseudo-labels $\mathcal{G} \in \{\mathcal{G}^{\mathcal{C}_m}, \mathcal{G}^{\mathcal{B}}\}$, and $\mathcal{O} \in \{0,1\}$ is the objectness prior. More specifically, let $\mathcal{I}_t \in \mathbb{R}^{h\times w\times 3}$ be a training image from $\mathcal{D}_{\mathcal{T}}$ with semantic segmentation pseudo-label $\mathcal{G} \in \mathbb{R}^{h\times w}$ and the objectness prior $\mathcal{O} \in \mathbb{R}^{h\times w}$. We denote the pixel-wise cross entropy loss function $L_{\mathcal{S}}$ and $L_{\mathcal{O}}$ between $(\mathcal{S},\mathcal{G})$ and $(\mathcal{S}_{\mathcal{O}},\mathcal{O})$, respectively. The final loss function of the network is the sum of the segmentation and objectness losses as follows:

$$L_{\mathcal{S}} = \ell_{\text{CE}}(\mathcal{S},\mathcal{G}),\;\; L_{\mathcal{O}} = \ell_{\text{BCE}}(\mathcal{S}_{\mathcal{O}},\mathcal{O}),\;\; L_{\text{SONet}} = L_{\mathcal{S}} + L_{\mathcal{O}}, \tag{5}$$

where ℓ_{CE} and ℓ_{BCE} denote the multi-class and binary cross entropy loss function, respectively. The joint training allows our network to propagate objectness information together with semantics, and suppress erroneous decisions which allows for more accurate final predictions for both outputs. During inference, we simply take the segmentation map to measure the overall performance of our proposed approach.

3 Experiments

We evaluate our proposed framework on the PASCAL VOC 2012 [16] and Cityscapes [14] semantic segmentation benchmarks. We generate objectness masks for the VOC12 target dataset from two different objectness trained models on COCO-Stuff: *overlapping* (all images) and *non-overlapping* (images with no overlapping objects with *target* data). We also report experiments under *domain adaptation* settings by training on a different target dataset, OpenV5 [41], and evaluating on VOC12. OpenV5 [41] is a recently released dataset consisting of image-level, bounding box, and semantic segmentation annotations for over 600 classes. For these experiments, we randomly select 42,621 images from the same 21 classes as VOC12 and generate pseudo-labels using our box-driven approach. We train SONet with the generated pseudo-labels from OpenV5 and then evaluate on VOC12 (denoted as O $\rightarrow$ V). We also finetune SONet on VOC12 before evaluation (denoted as O + V). We report experimental results with different backbone networks for a fair comparison.

3.1 Analysis of Generated Pseudo-labels

We first evaluate the quality of our generated pseudo-labels to explore the upper bound for different types of weak supervision and report the results in Table 1(a). We consider the generated pseudo-labels as predictions and obtain the upper bound for each supervision type by calculating the mIoU between the pseudo-label and the groundtruth. To generate CAMs pseudo-labels, $\mathcal{C}_m$, we simply threshold the scores of raw CAMs. When we apply our objectness mask, $\mathcal{O}$, to improve the boundary of CAMs ($\mathcal{G}^{\mathcal{C}_m}$), we obtain 70.6% mIoU. Further, using the bounding boxes and objectness map ($\mathcal{G}^{\mathcal{B}}$) achieves 76.6% mIoU that further improves the upper bound mIoU by 6%. In addition, applying the non-overlapping objectness mask, $\mathcal{O}_{\mathcal{N}}$, substantially improves the CAMs ($\mathcal{G}^{\mathcal{C}_m}_{\dagger}$) or bounding box ($\mathcal{G}^{\mathcal{B}}_{\dagger}$) proposals. As shown in Table 1(a), exploiting an objectness map with CAMs or bounding boxes significantly improves the quality of pseudo-labels as it removes incorrectly labeled pixels from the CAM and bounding box proposals. Next, we evaluate the performance of our proposed SONet (Table 1(b)) with CAMs, box annotations, and the generated pseudo-labels. SONet trained with box proposals achieves 54.6% mIoU which outperforms the same model trained using CAM proposals (50.2%). When we use our generated pseudo-labels during training, SONet achieves 70.5% mIoU ($\mathcal{G}^{\mathcal{C}_m}$) and 73.8% ($\mathcal{G}^{\mathcal{B}}$) on the VOC12 val set. In the domain adaptation settings (see Table 1(c)), SONet trained only with OpenV5 groundtruth boxes achieves 51.5% mIoU accuracy on the VOC12 val set. When SONet is trained on OpenV5 with $\mathcal{G}^{\mathcal{B}}$ as supervision, it drastically improves the overall mIoU to 71.0% (note that in this setting we *only* use OpenV5 images to train SONet). Additionally, fine-tuning SONet on the VOC12 training set with $\mathcal{G}^{\mathcal{B}}$ supervision increases the mIoU to 76.9%. These experiments indicate that our pseudo-label generation technique achieves good upper bound performance compared to the groundtruth.

Sup.	Train
$\mathcal{C}_m$	48.3
$\mathcal{G}_{\dagger}^{\mathcal{C}_m}$	63.5
$\mathcal{G}^{\mathcal{C}_m}$	70.6
$\mathcal{B}$	60.2
$\mathcal{G}_{\dagger}^{\mathcal{B}}$	68.6
$\mathcal{G}^{\mathcal{B}}$	76.6

(a)

Method	Sup.	Val.
SONet	$\mathcal{C}_m$	50.2
	$\mathcal{G}_{\dagger}^{\mathcal{C}_m}$	65.3
	$\mathcal{G}^{\mathcal{C}_m}$	**70.5**
	$\mathcal{B}$	54.6
	$\mathcal{G}_{\dagger}^{\mathcal{B}}$	67.9
	$\mathcal{G}^{\mathcal{B}}$	**73.8**

(b)

Method	Sup.	mIoU
O → V		
SONet	$\mathcal{B}$	51.5
	$\mathcal{G}^{\mathcal{B}}$	71.0
O + V		
SONet	$\mathcal{G}^{\mathcal{B}}$	75.9
	$\mathcal{G}^{\mathcal{B}}_{\text{ign}}$	76.9

(c)

Table 1: (a) Upper-bound analysis for different pseudo-label types on the VOC12 train set. (b) Quantitative results on VOC12 val set for the baseline and our approach. (c) SONet's performance under domain adaptation settings, trained on OpenV5 and then either directly evaluated (O → V) or fine-tuned (O + V) on VOC12.

3.2 Image Segmentation Results

In this section, we compare our proposed SONet method with previous state-of-the-art WSSS and SSSS methods [17, 18, 35, 38, 43, 53, 58]. Table 2 presents a comparison with recent weakly and semi supervised methods using image and bounding box-level supervision. For fair comparison in WSSS setting, we compare with other methods that use ResNet-101 as the backbone and additional guidance (e.g., saliency maps and optical flow) as supervision. SONet* outperforms the current state-of-the-art image-level + extra guidance based methods by a reasonable margin, achieving 68.1% mIoU on the VOC12 val set. Interestingly, SONet-O*, which is trained on the pseudo-labels generated under the non-overlapping settings, also achieves comparable performance with the baseline WSSS methods. When compared to methods that use bounding box-level supervision with extra guidance,

Method	Backbone	Guidance	mIoU	
			val	test
Weakly-Supervised Approaches				
Image-Level Supervision (CAM)				
FickleNet [43]	Res-101	Saliency	64.9	65.3
OAA* [35]	Res-101	Saliency	65.2	66.4
ME [18]	Res101	Saliency	67.2	66.7
ICD [17]	Res101	Saliency	67.8	68.0
SGAN*	Res-101	Saliency	67.1	67.2
SONet-O*	Res-101	Objectness	64.5	65.8
SONet*	Res-101	Objectness	**68.1**	**69.7**
SONet	Res-101	Objectness	**70.5**	**71.5**
Box-Level Supervision				
SDI* [38]	Res-101	BSDS	69.4	-
BCM* [58]	Res-101	CRF	70.2	-
SONet*	Res-101	Objectness	**72.2**	73.7
SONet	Res-101	Objectness	**74.8**	76.0
Semi-Supervised Approaches				
WSSL [53]	VGG-16	1.4k GT	64.6	-
SDI [38]	VGG-16	1.4k GT	65.8	66.9
FickleNet [43]	VGG-16	1.4k GT	65.8	-
SONet	VGG-16	-	**66.1**	**67**

Table 2: Quantitative comparison with weakly and semi supervised methods on the PASCAL VOC12 validation and test sets. SONet-O denotes the method used non-overlapping objectness prior. Methods marked by ∗ used DeepLabv2-Res101.

Method	Sup.	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	**mIoU**
SONet	$\mathcal{G}^{\mathcal{C}_m}$	84.4	37.6	83.7	63.9	51.6	88.4	84.7	68.3	30.8	81.4	57.9	68.0	79.6	83.6	74.2	58.5	84.6	53.4	81.5	57.1	69.7
SONet(O→V)	$\mathcal{G}^{\mathcal{B}}$	91.3	39.9	89.7	68.5	68.6	89.8	77.0	80.8	21.4	71.9	34.2	81.3	83.4	82.4	71.4	58.6	82.9	53.2	**86.4**	66.5	71.1
SONet	$\mathcal{G}^{\mathcal{B}}$	91.4	37.3	88.7	68.0	66.0	94.1	88.0	79.9	32.3	83.4	64.3	77.5	86.3	78.0	74.4	59.4	86.3	57.3	84.8	66.2	74.1
SONet(O+V)	$\mathcal{G}^{\mathcal{B}}$	90.7	40.0	90.2	69.7	72.7	94.1	87.4	79.2	32.7	**86.7**	62.6	80.1	**88.5**	81.3	74.2	62.6	**91.9**	**58.5**	**89.2**	69.5	76.0
SONet	$\mathcal{G}^{\mathcal{B}}_{\text{ign}}$	92.1	40.9	90.6	68.5	**74.0**	94.1	87.1	83.2	31.3	**86.4**	**67.2**	78.2	84.6	84.0	77.1	61.6	**90.6**	55.3	**85.4**	69.2	76.0
SONet (O+V)	$\mathcal{G}^{\mathcal{B}}_{\text{ign}}$	91.8	39.9	89.9	**71.3**	**74.8**	**94.6**	**88.2**	80.9	33.0	**89.5**	62.8	82.5	**89.7**	83.8	76.9	62.8	**90.3**	**59.9**	**89.3**	70.0	77.0
DeepLabV3	$\mathcal{F}$	**92.9**	**60.3**	**93.0**	70.5	73.3	94.1	88.1	**90.9**	**35.3**	83.4	65.7	**86.3**	87.5	**85.2**	**86.5**	**63.8**	88.1	57.6	85.0	**72.3**	**78.8**

Table 3: Class-wise IoU on the VOC12 test set. (O $\rightarrow$ V) refers to training on OpenV5 and test on VOC. (O + V) denotes training on OpenV5 and fine-tuning on VOC12. $\mathcal{F}$ denotes full supervision. Bolded values denote the results that surpass the fully supervised method.

SONet* also improves upon the state-of-the-art [38, 58] by 2.0%. Note that both BCM [58] and SDI [38] take much longer to produce pseudo-labels than our approach due to their iterative procedures and use complex training protocols. We do not include results for a recent box based method, Box2Seg [40], as they use a higher capacity network architecture [65] for segmentation learning without publicly available code. Our SONet method achieves 74.8% mIoU on the VOC12 val set which is very close (2.4% lower) to the fully supervised trained baseline [10] model (77.2%). In the SSSS setting, we use a similar backbone network as existing methods to ensure a fair comparison. Note, existing SSSS methods use a portion of the target semantic segmentation groundtruth while we *solely* use our generated pseudo-labels to train the network. Surprisingly, SONet (VGG-16 backbone) marginally outperforms the existing SSSS methods (66.1% vs. 65.8% mIoU). These results demonstrate that our pseudo-label generation procedure is flexible and achieves substantial improvements or competitive performance compared to existing methods in WSSS and SSSS. Table 3 presents a class-wise IoU comparison of SONet with different training strategies as well as with the fully supervised baseline DeepLabv3 model on the VOC12 test set. Notably, although the fully supervised model achieves the highest mIoU, SONet (O+V) trained using $\mathcal{G}^{\mathcal{B}}_{\text{ign}}$ outperforms the fully supervised model on *half* of the categories, and is competitive in many others. In general, when trained using bounding box-based pseudo-labels, SONet performs well on rectangular shaped classes, e.g., bus, car, tv, cow, bottle, bus, and train. However, it is still difficult for any training protocol combined with SONet to achieve comparable performance with classes like bike, motorbike, cat, dog or person, where the objects have complex boundary information or are occluded with other classes, e.g., person on a horse or bike. Furthermore, using the ignore strategy ($\mathcal{G}^{\mathcal{B}}_{\text{ign}}$ vs. $\mathcal{G}^{\mathcal{B}}$) improves the performance notably for both the normal and domain adaptation settings. The quantitative results indicate that our SONet model can achieve competitive performance with the fully supervised model, showing the effectiveness of the proposed pseudo-label generation and the joint learning techniques.

Method	Sup.	mIoU
DLabv3 (Things)	Full	81.5
SONet (Things)	$\mathcal{G}^{\mathcal{B}}$	76.6

Table 4: Quantitative results on the Cityscapes [14] val set.

We further use Cityscapes [14] as our target dataset and report results in Table 4. Cityscapes consists of eight *things* classes and 11 stuff classes. Similar to the VOC12, we first generate class agnostic objectness masks for the Cityscapes train set and combine it with the bounding box annotations to generate semantic pseudo-labels. Since our objectness network is not trained for generating masks for *stuff* classes, we only consider the *things* classes from Cityscapes during pseudo-label generation, training, and evaluation. Next, we train DeepLabv3-ResNet50 [10] with full supervision as a baseline and SONet (DeepLabv3-ResNet50 as backbone) using the generated pseudo-labels ($\mathcal{G}^{B}$). Table 4 shows that, our SONet

Sup.	VOC12		O $\rightarrow$ V	
	[11]	SONet	[11]	SONet
$\mathcal{B}$	52.1	**54.6**	50.5	**51.5**
$\mathcal{G}^{\mathcal{B}}$	73.1	**73.8**	70.4	**71.0**

(a)

Sup.	Val.	Test
$[38]_{\text{ign}=0.1}$	67.9	-
$[38]_{\text{ign}=0.2}$	67.6	-
$\mathcal{G}^{\mathcal{B}}$	73.8	74.1
$\mathcal{G}^{\mathcal{B}}_{\text{ign}}$	74.8	76.0

(b)

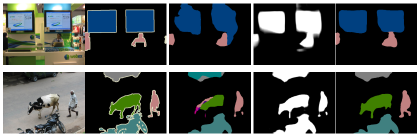


Image GT DeepLabv3 Mask SONet

Table 5: (a) Comparison between SONet and DeepLabv3 [11] on the VOC12 val set. (b) Comparison of our ignore strategy with SDI [38]. (Right) Visualization of the effect on segmentation results of the model trained with or w/o objectness branch.

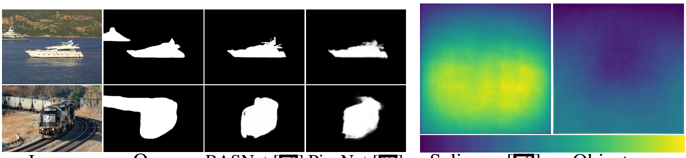
performs well (76.6% mIoU) and obtains 94% relative to the baseline (similar to our results on VOC12). This result further confirms the generalizability of our pseudo-label generation technique despite the significant distribution gap between the target (Cityscapes [14]) and the source (COCOStuff [6]) dataset.

3.3 Ablation Studies

Effectiveness of Objectness Branch. We first validate the effect of the objectness branch by comparing the results of SONet trained in both multi- and single-task settings. We train SONet with and without the objectness branch. Note that SONet without the objectness branch is equivalent to DeepLabv3 [11]. The result of these comparisons are summarized in Table 5 (a). It is clear that the models trained with the objectness branch achieve superior performance compared to the models trained only for the task of semantic segmentation. Interestingly, the objectness branch improves the boundary details to bring more smoothness (see Fig. in Table 5 (top row)) as expected, as well as the semantic information (see the figure in Table 5 (bottom row)). SONet's multi-task objective not only provides it with the ability to robustly predict both binary and semantic segmentation, but the objectness-based learning naturally provides the segmentation network a significant performance boost.

Effectiveness of Ignoring Strategy. In Table 5(b), we compare our ignore strategy to the strategy in SDI [38] when trained using SONet. We show that our ignoring strategy outperforms both SDI [38] and SONet trained without an ignore strategy.

Improving Semantic Proposals: Objectness or Saliency Guidance? It is common to utilize pixel-level *saliency* information as additional guidance to be combined with the CAM proposals [52, 63, 67, 69]. Specifically, DHSNet [48] and DSS [22] have been used in [8, 23, 63] to generate a saliency mask for each training sample. This guidance of saliency can deliver non-semantic pixel-level supervision for a better boundary segmentation. However, the saliency information used in the previous studies only focus on the *most* salient object due to the problem of centre bias [1, 5]. For instance, as shown in Fig. 3(a), the masks generated by saliency models can only detect the objects near the centre of an image, the "ship" near the corner will be incorrectly labelled as background (top row). Furthermore, the region of the "train" can only be partially labelled because the back of the train is not salient. This problem introduces outliers (incorrectly labelled regions) when training a segmentation model. In contrast, our proposed objectness model learns to recognize objects in all image locations, even if they are not salient or near the image boundary, see Fig. 3(a). Figure 3(b) further illustrates that the objectness network is equally likely to make errors in all image locations, while the saliency detection network is biased towards making erroneous predictions near the image border. To validate this claim quantitatively, we conduct experiments (see Table in



(a) (b)

*	$\mathcal{G}^{\mathcal{C}_m}$	$\mathcal{G}^{\mathcal{B}}$
PicaNet	53.7	64.7
BASNet	58.5	64.0
Ours	**70.6**	**76.6**

(c)

Figure 3: (a) Visual comparison between our class agnostic objectness model and top performing saliency detectors. (b) Saliency and objectness error distribution across the predictions on VOC12 training set. (c) Pseudo-label quality comparison in terms of mIoU on the VOC12 train set, between the objectness model and saliency methods.

Fig. 3(c)) by replacing the objectness mask with a saliency mask for creating semantic pseudo-labels. We use two recent saliency detectors, PiCaNet [49] and BASNet [56] to generate the saliency mask for VOC12 training images. Combining saliency masks with $\mathcal{G}^{\mathcal{C}_m}$ and $\mathcal{G}^{\mathcal{B}}$ achieves performance which is significantly lower than the quality of our pseudo-labels generated using the objectness guidance.

4 Discussion and Conclusion

Existing saliency-based WSSS [17, 18, 25, 35, 43, 53, 69] and SSSS [43] methods utilize both saliency detectors (trained on the DUT-S [61] or MSRA-B [34] datasets which have pixel-level binary segmentation ground-truth for a large number of overlapping instances in the VOC12 dataset) and a portion of semantic segmentation GT, respectively. Following this line of work, we choose the objectness-based dataset to introduce a better proposal model which addresses the severe center bias issue of saliency detectors (see Fig. 3 (a, b)) for WSSS (e.g., saliency inherently ignores objects near the border). We compare with both WSSS and SSSS techniques since we do not fall neatly within either category of supervision (i.e., comparing against methods which use only CAM is unfair but we also do not use any semantic segmentation GT). Moreover, in contrast to the previous methods [2, 3, 38, 64], our framework does not require multiple stages of label inference and training for pseudo-label generation, but instead operates in a single stage. Additionally, the objectness branch improves the performance of the segmentation network by propagating boundary and semantic information back through the network. We believe the objectness branch helps with semantics because it forces the model to treat objects more uniformly (since the objectness label is binary). This can guide the segmentation model to treat nearby pixels as the same semantic object class and promote more spatially uniform predictions, which is correct in many cases.

In summary, we have presented a pseudo-label generation and joint learning strategy for the tasks of both WSSS and SSSS. We first introduced a novel technique to generate high quality pseudo-labels that combines class agnostic objectness priors with either image-level labels or bounding box annotations. Next, we proposed a model that jointly learns semantics and objectness to guide the network to encode more accurate boundary information and better semantic representations. We conducted an extensive set of experiments under different settings and supervision strategies to validate the effectiveness of the proposed methods. The ablation studies isolated the improvements due to the proposed objectness branch, and validated the efficacy of our ignoring strategy. Furthermore, the pseudo-label generation pipeline is simple, efficient, and can be used for large-scale data annotation.

References

[1] Alper Açık, Andreas Bartel, and Peter Koenig. Real and implied motion at the center of gaze. *Journal of Vision*, 2014.

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018.

[3] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *CVPR*, 2019.

[4] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for scene segmentation. *TPAMI*, 2017.

[5] Neil DB Bruce, Calden Wloka, Nick Frosst, Shafin Rahman, and John K Tsotsos. On computational modeling of visual saliency: Examining what's right, and what's left. *Vision research*, 2015.

[6] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and stuff classes in context. In *CVPR*, 2018.

[7] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *CVPR*, 2020.

[8] Arslan Chaudhry, Puneet K Dokania, and Philip HS Torr. Discovering class-specific pixels for weakly-supervised semantic segmentation. *arXiv preprint arXiv:1707.05821*, 2017.

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

[10] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv:1706.05587*, 2017.

[11] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2018.

[12] Liyi Chen, Weiwei Wu, Chenchen Fu, Xiao Han, and Yuntao Zhang. Weakly supervised semantic segmentation with boundary exploration. In *ECCV*, 2020.

[13] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. SegFlow: Joint learning for video object segmentation and optical flow. In *ICCV*, 2017.

[14] 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 *CVPR*, 2016.

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

[16] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The Pascal visual object classes challenge: A retrospective. *IJCV*, 2015.

[17] Junsong Fan, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan. Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation. In *CVPR*, 2020.

[18] Junsong Fan, Zhaoxiang Zhang, and Tieniu Tan. Employing multi-estimations for weakly-supervised semantic segmentation. In *ECCV*, 2020.

[19] Golnaz Ghiasi and Charless C Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *ECCV*, 2016.

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

[21] Lei He, Jiwen Lu, Guanghui Wang, Shiyu Song, and Jie Zhou. Sosd-net: Joint semantic object segmentation and depth estimation from monocular images. *Neurocomputing*, 2021.

[22] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr. Deeply supervised salient object detection with short connections. In *CVPR*, 2017.

[23] Qibin Hou, PengTao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. In *NIPS*, 2018.

[24] Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick. Learning to segment every thing. In *CVPR*, 2018.

[25] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *CVPR*, 2018.

[26] Mostafa S Ibrahim, Arash Vahdat, and William G Macready. Weakly supervised semantic image segmentation with self-correcting networks. *arXiv preprint arXiv:1811.07073*, 2018.

[27] Md Amirul Islam, Shujon Naha, Mrigank Rochan, Neil Bruce, and Yang Wang. Label refinement network for coarse-to-fine semantic segmentation. *arXiv:1703.00551*, 2017.

[28] Md Amirul Islam, Mrigank Rochan, Neil D. B. Bruce, and Yang Wang. Gated feedback refinement network for dense image labeling. In *CVPR*, 2017.

[29] Md Amirul Islam, Mahmoud Kalash, and Neil D.B. Bruce. Revisiting salient object detection: Simultaneous detection, ranking, and subitizing of multiple salient objects. In *CVPR*, 2018.

[30] Md Amirul Islam, Mahmoud Kalash, and Neil DB Bruce. Semantics meet saliency: Exploring domain affinity and models for dual-task prediction. In *BMVC*, 2018.

[31] Md Amirul Islam, Matthew Kowal, Konstantinos G Derpanis, and Neil DB Bruce. Feature binding with category-dependant mixup for semantic segmentation and adversarial robustness. In *BMVC*, 2020.

[32] Md Amirul Islam, Matthew Kowal, Konstantinos G Derpanis, and Neil DB Bruce. Segmix: Co-occurrence driven mixup for semantic segmentation and adversarial robustness. *arXiv preprint arXiv:2108.09929*, 2021.

[33] Md Amirul Islam, Matthew Kowal, Patrick Esser, Sen Jia, Björn Ommer, Konstantinos G. Derpanis, and Neil Bruce. Shape or texture: Understanding discriminative features in CNNs. In *ICLR*, 2021.

[34] Huaizu Jiang, Jingdong Wang, Zejian Yuan, Yang Wu, Nanning Zheng, and Shipeng Li. Salient object detection: A discriminative regional feature integration approach. In *CVPR*, 2013.

[35] Peng-Tao Jiang, Qibin Hou, Yang Cao, Ming-Ming Cheng, Yunchao Wei, and Hong-Kai Xiong. Integral object mining via online attention accumulation. In *ICCV*, 2019.

[36] Rezaul Karim, Md Amirul Islam, and Neil DB Bruce. Recurrent iterative gating networks for semantic segmentation. In *WACV*, 2019.

[37] Rezaul Karim, Md Amirul Islam, and Neil DB Bruce. Distributed iterative gating networks for semantic segmentation. In *WACV*, 2020.

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

[39] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, 2011.

[40] Viveka Kulharia, Siddhartha Chandra, Amit Agrawal, Philip Torr, and Ambrish Tyagi. Box2Seg: Attention weighted loss and discriminative feature learning for weakly supervised segmentation. In *ECCV*, 2020.

[41] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, and Tom Duerig. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982*, 2018.

[42] Suha Kwak, Seunghoon Hong, and Bohyung Han. Weakly supervised semantic segmentation using superpixel pooling network. In *AAAI*, 2017.

[43] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. FickleNet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *CVPR*, 2019.

[44] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Frame-to-frame aggregation of active regions in web videos for weakly supervised semantic segmentation. In *ICCV*, 2019.

[45] Qizhu Li, Anurag Arnab, and Philip HS Torr. Weakly-and semi-supervised panoptic segmentation. In *ECCV*, 2018.

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

[47] G. Lin, A. Milan, C. Shen, and I. Reid. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, 2017.

[48] Nian Liu and Junwei Han. Dhsnet: Deep hierarchical saliency network for salient object detection. In *CVPR*, 2016.

[49] Nian Liu, Junwei Han, and Ming-Hsuan Yang. PicaNet: Learning pixel-wise contextual attention for saliency detection. In *CVPR*, 2018.

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

[51] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015.

[52] Seong Joon Oh, Rodrigo Benenson, Anna Khoreva, Zeynep Akata, Mario Fritz, and Bernt Schiele. Exploiting saliency for object segmentation from image level labels. In *CVPR*, 2017.

[53] 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 *ICCV*, 2015.

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

[55] Jordi Pont-Tuset, Pablo Arbelaez, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping for image segmentation and object proposal generation. *TPAMI*, 2016.

[56] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *CVPR*, 2019.

[57] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *TOG*, 2004.

[58] 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 *CVPR*, 2019.

[59] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In *ICCV*, 2017.

[60] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-SCNN: Gated shape cnns for semantic segmentation. In *ICCV*, 2019.

[61] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *CVPR*, 2017.

[62] Yuhang Wang, Jing Liu, Yong Li, Junjie Yan, and Hanqing Lu. Objectness-aware semantic segmentation. In *ICM*, 2016.

[63] 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 *CVPR*, 2018.

[64] Huaxin Xiao, Yunchao Wei, Yu Liu, Maojun Zhang, and Jiashi Feng. Transferable semi-supervised semantic segmentation. In *AAAI*, 2018.

[65] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018.

[66] Bo Xiong, Suyog Dutt Jain, and Kristen Grauman. Pixel objectness: learning to segment generic objects automatically in images and videos. *TPAMI*, 2018.

[67] Qi Yao and Xiaojin Gong. Saliency guided self-attention network for weakly and semi-supervised semantic segmentation. *IEEE Access*, 2020.

[68] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016.

[69] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, and Lihe Zhang. Joint learning of saliency detection and weakly supervised semantic segmentation. In *CVPR*, 2019.

[70] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017.

[71] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016.