The Devil is in the Points: Weakly Semi-Supervised Instance Segmentation via Point-Guided Mask Representation

Beomyoung Kim[1,2]   Joonhyun Jeong[1,2]   Dongyoon Han[3]   Sung Ju Hwang[2]

NAVER Cloud, ImageVision[1]   KAIST[2]   NAVER AI Lab[3]

Abstract

In this paper, we introduce a novel learning scheme named weakly semi-supervised instance segmentation (WS-SIS) with point labels for budget-efficient and high-performance instance segmentation. Namely, we consider a dataset setting consisting of a few fully-labeled images and a lot of point-labeled images. Motivated by the main challenge of semi-supervised approaches mainly derives from the trade-off between false-negative and false-positive instance proposals, we propose a method for WSSIS that can effectively leverage the budget-friendly point labels as a powerful weak supervision source to resolve the challenge. Furthermore, to deal with the hard case where the amount of fully-labeled data is extremely limited, we propose a MaskRefineNet that refines noise in rough masks. We conduct extensive experiments on COCO and BDD100K datasets, and the proposed method achieves promising results comparable to those of the fully-supervised model, even with 50% of the fully labeled COCO data (38.8% vs. 39.7%). Moreover, when using as little as 5% of fully labeled COCO data, our method shows significantly superior performance over the state-of-the-art semi-supervised learning method (33.7% vs. 24.9%). The code is available at https://github.com/clovaai/PointWSSIS.
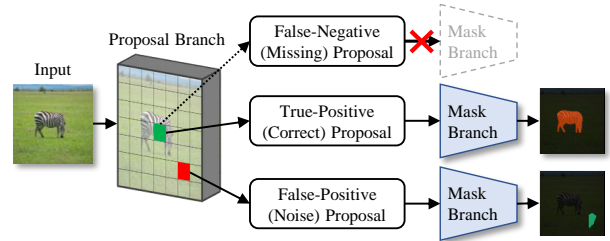
1. Introduction

Recently proposed instance segmentation methods [5, 8, 9, 13, 15, 16, 24, 33, 43, 47] have achieved remarkable performance owing to the availability of abundant of segmentation labels for training. However, compared to other label types (*e.g.*, bounding box or point), segmentation labels necessitate delicate pixel-level annotations, demanding much more monetary cost and human effort. Consequently, weakly-supervised instance segmentation (WSIS) and semi-supervised instance segmentation (SSIS) approaches have gained attention to reduce anno-

Figure 1. **Proposals and instance masks**. The absence of a proposal leads to the missing mask, even though the mask could be generated if given the correct proposal (zebra). Also, noise proposal often leads to noisy masks. Our motivation stems from the bottleneck in the proposal branch, and this paper shows economic point labels can be leveraged to resolve it.

tation costs. WSIS approaches alternatively utilize inexpensive weak labels such as image-level labels [1, 27, 55], point labels [11, 27, 28] or bounding box labels [19, 32, 44]. Besides, SSIS approaches [49, 54] employ a small amount of pixel-level (fully) labeled data and a massive amount of unlabeled data. Although they have shown potential in budget-efficient instance segmentation, there still exists a large performance gap between theirs and the results of fully-supervised learning methods.

Specifically, SSIS approaches often adopt the following training pipeline: (1) train a base network with fully labeled data, (2) generate pseudo instance masks for unlabeled images using the base network, and (3) train a target network using both full and pseudo labels. The major challenge of SSIS approaches comes from the trade-off between the number of missing (*i.e.*, false-negative) and noise (*i.e.*, false-positive) samples in the pseudo labels. Namely, some strategies for reducing false-negatives, which is equivalent to increasing true-positives, often end up increasing false-positives accordingly; an abundance of false-negatives or false-positives in pseudo labels impedes stable convergence of the target network. However, optimally reducing false-negatives/positives while increasing true-positives is quite challenging and remains a significant challenge for SSIS.

To address this challenge, we first revisit the fundamental behavior of the instance segmentation framework. Most existing instance segmentation methods adopt a two-step inference process as shown in Figure 1: (1) generate instance proposals where an instance is represented as a box [9,16,22,33] or point [43,45,47,48] in proposal branch, and (2) produce instance masks for each instance proposal in mask branch. As shown in Figure 1, if the network fails to obtain an instance proposal (*i.e.*, false-negative proposal), it cannot produce the corresponding instance mask. Although the network could represent the instance mask in the mask branch, the absence of the proposal becomes the bottleneck for producing the instance mask. From the behavior of the network, we suppose that addressing the bottleneck in the proposals is a shortcut to the success of the SSIS.

Motivated by the above observations, we rethink the potential of using point labels as weak supervision. The point label contains only a one-pixel categorical instance cue but is budget-friendly as it is as easy as providing image-level labels by human annotators [3]. We note that the point label can be leveraged as an effective source to (i) resolve the performance bottleneck of the instance segmentation network and (ii) optimally balance the trade-off between false-negative and false-positive proposals. Thus, we formulate a new practical training scheme, **Weakly Semi-Supervised Instance Segmentation (WSSIS) with point labels**. In the WSSIS task, we utilize a small amount of fully labeled data and a massive amount of point labeled data for budget-efficient and high-performance instance segmentation.

Under the WSSIS setting, we filter out the proposals to keep only true-positive proposals using the point labels. Then, given the true-positive proposals, we exploit the mask representation of the network learned by fully labeled data to produce high-quality pseudo instance masks. For properly leveraging point labels, we consider the characteristics of the feature pyramid network (FPN) [35], which consists of multi-level feature maps for multi-scale instance recognition. Each pyramid level is trained to recognize instances of particular sizes, and extracting instance masks from unfit levels often causes inaccurate predictions, as shown in Figure 4. However, since point labels do not have instance size information, we handle this using an effective strategy named Adaptive Pyramid-Level Selection. We estimate which level is the best fit based on the reliability of the network (*i.e.*, confidence score) and then adaptively produce an instance mask at the selected level.

Meanwhile, on an extremely limited amount of fully labeled data, the network often fails to sufficiently represent the instance mask in the mask branch, resulting in rough and noisy mask outputs. In other words, the true-positive proposal does not always lead to a true-positive instance mask in this case. To cope with this limitation, we propose a MaskRefineNet to refine the rough instance mask.

The MaskRefineNet takes three input sources, *i.e.*, image, rough mask, and point; the image provides visual information about the target instance, the rough mask is used as the prior knowledge to be refined, and the point information explicitly guides the target instance. Using the richer instructive input sources, MaskRefineNet can be stably trained even with a limited amount of fully labeled data.

To demonstrate the effectiveness of our method, we conduct extensive experiments on the COCO [36] and BDD100K [51] datasets. When training with half of the fully labeled images and the rest of the point labeled images on the COCO dataset (*i.e.*, 50% COCO), we achieve a competitive performance with the fully-supervised performance (38.8% vs. 39.7%). In addition, when using a small amount of fully labeled data, *e.g.*, 5% of COCO data, the proposed method shows much superior performance than the state-of-the-art SSIS method [49] (33.7% vs. 24.9%).

In summary, the contributions of our paper are

- We show that point labels can be leveraged as effective weak supervisions for budget-efficient and high-performance instance segmentation. Further, based on this observation, we establish a new training protocol named Weakly Semi-Supervised Instance Segmentation (WSSIS) with point labels.

- To further boost the quality of the pseudo instance masks when the amount of fully labeled data is extremely limited, we propose the MaskRefineNet, which refines noisy parts of the rough instance masks.

- Extensive experimental results show that the proposed method can achieve competitive performance to those of the fully-supervised models while significantly outperforming the semi-supervised methods.

2. Related Work

2.1. Instance Segmentation

Mask R-CNN [16] is the most widely used method for instance segmentation. They represent an instance as a bounding box and produce the instance mask after pooling each box region. These box-based approaches have many variants, such as [9,22,33,41] and have shown state-of-the-art results. Meanwhile, there is a different type of approach, named point-based approaches [43,45,47,48]. They represent an instance as a point and generate the instance mask using the point-encoded mask representation. For example, SOLOv2 [47] extracts point-encoded kernel parameters and generates instance masks with a dynamic convolution scheme. We note that the inference pipeline of these two approaches is the same as shown in Figure 1; they generate proposals in the form of either bounding boxes or points and then produce an instance mask for each proposal. Here, the proposal is indispensable for producing the instance mask.

| (a) Confidence Threshold=0.1 | (b) Confidence Threshold=0.5 | (c) With Point Labels |

Figure 2. **The qualitative results of pseudo instance masks**. (a) and (b): the quality of pseudo masks is largely affected by the confidence score of the proposal due to the trade-off between false-negative and false-positive instance proposals. (c): our point-driven method can filter the proposals to keep only true-positive proposals, resulting in clearer quality of pseudo instance masks.

2.2. Budget-Efficient Instance Segmentation

Instance segmentation requires a huge amount of instance-level segmentation labels. However, the annotation cost of segmentation labels is much higher than other labels. According to seminar works [3, 4], the annotations time is measured on VOC dataset [14] as follows: image-level (20.0 *s/img*), point (23.3 *s/img*), bounding box (38.1 *s/img*), full mask (239.7 *s/img*). To reduce the annotation cost, weakly-supervised instance segmentation (WSIS) and semi-supervised instance segmentation (SSIS) have been actively studied. The WSIS methods exploit the activation maps generated by self-attention of the network trained with only cost-efficient labels such as image-level [1, 27, 55], point [11, 27, 28], and bounding box [19, 32, 44] labels. Meanwhile, the SSIS methods [49, 54] use a small amount of fully labeled data and an abundant amount of unlabeled data. Utilizing the knowledge of the segmentations learned with the fully labeled data, they generate pseudo instance masks for the unlabeled data. Although they can reduce the annotation cost, their performance is still far behind those of the fully-supervised models.

2.3. Weakly Semi-Supervised Object Detection

There exist some previous attempts to tackle the weakly semi-supervised object detection problem using point labels (WSSOD) [10,52]. Namely, they use a few box-labeled data and a lot of point-labeled data. Leveraging the point labels, they show improved detection performances compared to the semi-supervised setting. Object detection and instance segmentation tasks share a similar goal: both are object-level recognition tasks. However, we point out that the motivation for leveraging point labels is different. We focus on the fundamental drawback of the instance segmentation network to handle the trade-off between false-negative and false-positive proposals. In contrast, PointDETR [10] leverages the point labels as input queries for single-level feature map inference of DETR [6] architecture, and Group R-CNN [52] employs the point labels to filter and augment proposals with improved positive sample assignments. In addition, we propose the MaskRefineNet for high-fidelity mask refinement to handle the distinct challenge of instance segmentation, which is a pixel-level recognition task.

3. Proposed Method

3.1. Motivation

Existing instance segmentation methods typically adopt a two-step inference process: (1) generate proposals where each instance is represented as bounding box [9, 16, 22, 33] or point [43, 45, 47, 48] in proposal branch, and (2) produce a mask for each instance in mask branch. Figure 1 provides an intuition that the performance of the instance segmentation network critically depends on the correctness of proposals at the proposal branch. Thus, improving the proposal branch may lead to a significant performance improvement in semi-supervised instance segmentation (SSIS).

To delve deeper into the problem, we adjust a confidence threshold in the proposal branch to verify the influence of the proposal on the output instance mask as shown in Figure 2. At a low threshold of 0.1 with a larger number of proposals, we obtain more true-positive masks but much more false-positive masks as well (see Figure 2a). The reason is that false-positive proposals (*e.g.*, misclassified or erroneously localized proposals) often lead to noisy mask predictions. Conversely, when we increase the threshold to 0.5, we lose several true-positive masks that were detected at lower thresholds (see Figure 2b). In other words, although the mask branch could represent the instance mask, the absence of the thresholded proposal results in missing instance masks. However, finding an optimal threshold per instance is impractical, and balancing between true-positive and false-positive proposals still remains a challenging problem in SSIS.
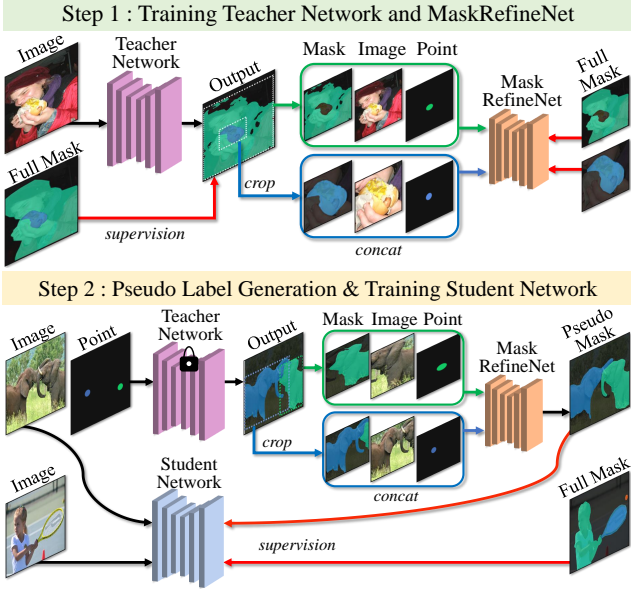
Figure 3. **Overview of the proposed method**. Top: (step 1) training the teacher network and MaskRefineNet with fully labeled data. Bottom: (step 2) under the point label guidance, pseudo labels are generated through the teacher network and further refined using MaskRefineNet. Then, the student network is trained on both the pseudo-labeled data and the fully labeled data.

3.2. Weakly Semi-Supervised Instance Segmentation using Point Labels

From the above observations, we can expect that obtaining correct instance proposals will yield accurate mask representations to improve an SSIS network. To this end, we revisit the point label, which is a one-pixel categorical instance representation cue. The annotation budget for point labels is known as costly-efficient by the literature [3, 4].

Task definition. We propose a new training protocol named Weakly Semi-Supervised Instance Segmentation (WSSIS) using point labels and verify that the budget-friendly point labels can provide effective guidance. The training protocol employs point-labeled data with a small amount of fully labeled data, which yields reduced annotation costs.

Training basline. Figure 3 shows our proposed baseline of a two-step learning procedure for WSSIS: (1) train a teacher network using only the full labels; (2) train a student network using both the full and pseudo labels generated by the teacher network along with the point labels. Generating high-quality pseudo labels is crucial for WSSIS, so we employ point labels as guidance for filtering the proposals to remain only true-positive proposals. Then, given the filtered proposals, we generate instance masks by exploiting the mask representation of the teacher network. Note that the proposed architecture is a baseline for the proposed task so that one can explore a more advanced training scheme.
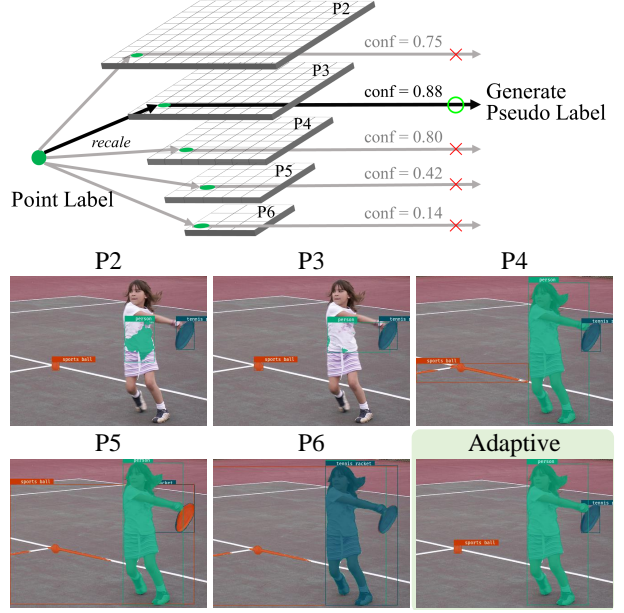

Figure 4. **Adaptive strategy with FPN and qualitative results**. Top: illustration of Adaptive Pyramid-Level Selection. Bottom: the qualitative results from each pyramid level.

FPN head. Most existing instance segmentation approaches [9, 16, 43, 47] adopt Feature Pyramid Network (FPN) [35] architecture for multi-scale instance prediction. Namely, SOLOv2 [47] employs a 5-level feature pyramid (P2∼P6), and each pyramid level recognizes instances of particular sizes. When combined with using point labels for sampling proposals, a careful approach to which level to extract proposals based on the size of the instance is demanding. Otherwise, generated instance masks are often noisy as shown in Figure 4 below.

Strategy of using pyramid-level adaptively. Since points do not contain instance size information, we estimate which pyramid level is proper for each point. To this end, we propose a strategy named Adaptive Pyramid-Level Selection, which adaptively selects a pyramid level that is expected to produce the most appropriate instance mask based on the reliability of the network. Namely, we rescale the coordinate of point labels according to the resolution of each level and extract confidence scores for all levels. Then, we generate an instance mask only from the pyramid level with the maximum confidence score, as illustrated in Figure 4. Formally, there is N proposal branches $\{\mathbf{F}_i^p\}_{i=1}^N$, and we follows the configuration of FPN [35] with $N{=}5$. For each point label (x, y, c), where c denotes category id, we extract an instance proposal and confidence score $(\mathbf{P}_i, \mathbf{s}_i) = \mathbf{F}_i^p(x, y, c)$. Regarding the confidence score as the reliability of the prediction, we adaptively select a pyramid level k with the maximum score, $k = \text{argmax}_{k \in \{1,2,...,N\}} \mathbf{s}_k$. Finally, at the mask branch $\mathbf{F}^m$, we generate a pseudo instance mask $M = \sigma(\mathbf{F}^m(\mathbf{P}_k))$, where σ is sigmoid function.
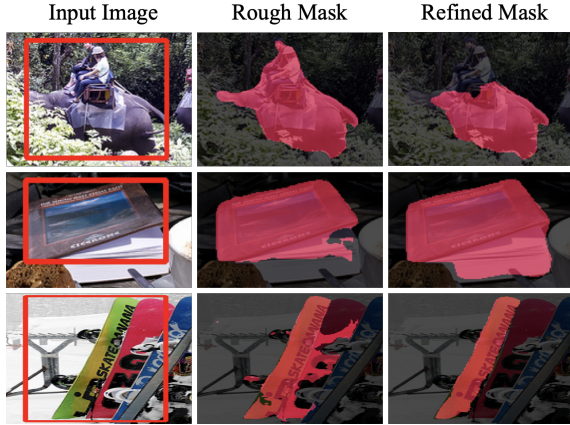
| Input Image | Rough Mask | Refined Mask |

Figure 5. **Effect of MaskRefineNet**. The qualitative results under 10% of COCO fully labeled data condition. When the teacher network fails to disentangle objects in a rough mask, MaskRefineNet can separate each representation owing to the given point label (1st row). Our MaskRefineNet further enriches the resultant mask representation (2nd row) and removes noisy parts (3rd row).

3.3. Mask Refinement Network

With a sufficient amount of fully labeled data (*e.g.*, using 50% images), the teacher network can afford to generate reasonable pseudo-instance masks given true-positive proposals. However, when the amount of fully labeled data is extremely small (*e.g.*, using only 1% images), the mask representation by the network would produce rough instance masks; it means the true-positive proposal could not ensure that the instance mask is a true-positive.

To handle such a challenging case, we propose a simple yet effective post-hoc mask refinement method named MaskRefineNet. Figure 3 shows that MaskRefineNet can refine the rough mask output from the teacher network based on three input sources, including input image, rough mask, and point information. Specifically, we loosely crop each instance region in the input image, rough mask, and point information, and resize them to 256×256, then concatenate them together into an input tensor. For the point information, we transform the point label to the form of a heatmap where each point is encoded into a 2D gaussian kernel with a sigma of 6. The effectiveness of the MaskRefineNet can be attributed to two reasons; (1) it leverages the prior knowledge of the teacher network; since MaskRefineNet takes the rough mask predictions from the teacher network as the input, it learns how to calibrate common errors of predictions from the teacher network; (2) it takes guidance from the input point that is likely to provide an accurate target instance cue for recognizing overlapping instances and falsely predicted pixels. Consequently, MaskRefineNet refines the missing & noisy parts and disentangles the crowded target instances in the rough mask as shown in Figure 5 with the help of the point guidance.

4. Experiments

4.1. Datasets

We evaluate our method on the COCO 2017 dataset [36] that contains 118,287 training samples and 5,000 validation samples for 80 common object categories. To validate our method under the WSSIS regime, we randomly sample subsets containing 1%, 2%, 5%, 10%, 20%∼50% of the COCO training dataset. COCO 10% means using 10% of the fully labeled data and the rest of 90% of the point labeled data. We use a centroid point of an instance mask label as a point label. In addition, we conduct experiments on BDD100K dataset [51], which is a large-scale driving scene dataset with diverse scene types and 8 classes. The BDD100K dataset contains 7k mask-labeled images and 67k box-labeled images, and we use the center of the box as the point label for this dataset.

4.2. Implementation Details

We adopt SOLOv2 [47] as the baseline instance segmentation network since it is a point-based and box-free straightforward method. For both teacher and student networks, we use the same ResNet-101 [17] backbone network and follow the default training recipe and network setting as in [47]. For the MaskRefineNet, we adopt the ResNet-101 FPN [35] architecture and produce the output only from the highest resolution pyramid level, P2. We set the batch size of 16, the learning rate of 1e-4 with cosine decay scheduling, dice loss [38], and input size of 256×256 for training the MaskRefineNet. After training the teacher network, the MaskRefineNet is trained by taking the rough mask outputs from the teacher network. We implement the proposed method using Pytorch [40] and train on 8 V100 GPUs.

Following the labeling budget calculation in [3,4], we estimate the labeling budget for the COCO trainset as follows: Full mask ($645.9s/img$), Bounding box ($127.5s/img$), Point ($87.9s/img$), Image-level ($80s/img$). Detailed calculation method is described in our supplementary material.

4.3. Experimental Results

We compare the proposed method against two baselines with the same network architecture and optimization strategy. The first is training with only fully labeled data, and the second is training with fully labeled and unlabeled data, which is a semi-supervised setting. For the second baseline, we generate pseudo instance masks for the unlabeled data from the teacher network without any weak labels. As shown in Figure 6, our method achieves remarkable performances on all COCO subsets. Especially, the performance gap between ours and the baselines is notably larger when we use smaller subsets with fully labeled images, *e.g.*, COCO 1% or 5%. Compared to the fully-supervised setting (COCO 100%), our method with COCO 50% shows
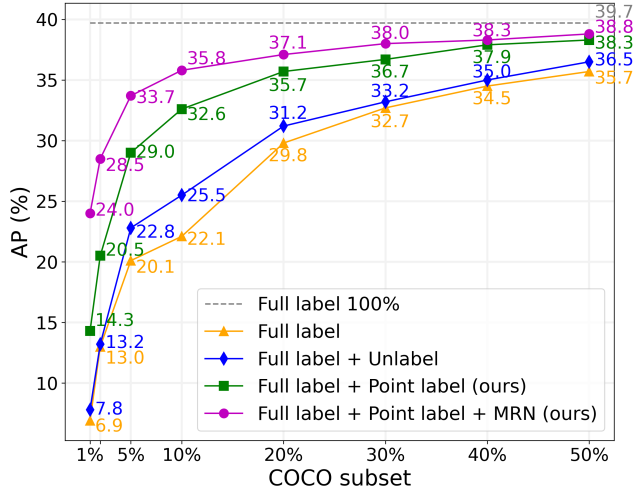
Figure 6. **Performance trend comparison under supervisions**. We visualize the AP scores when varying numbers of fully labeled data in the COCO test-dev. MRN means applying our point-guided MaskRefineNet.

Method	Label Types	Budget (days) $\downarrow$	AP (%) $\uparrow$
Weakly-supervised Models $\downarrow$			
BESTIE [27]	$\mathcal{I}$ 100%	109.5	14.3
BESTIE [27]	$\mathcal{P}$ 100%	120.3	17.7
BBAM [32]	$\mathcal{B}$ 100%	174.5	26.0
BoxInst [44]	$\mathcal{B}$ 100%	174.5	33.2
Semi-supervised Models $\downarrow$			
NB [49]	$\mathcal{F}$ 5% + $\mathcal{U}$ 95%	44.2	25.6
NB [49]	$\mathcal{F}$ 10% + $\mathcal{U}$ 90%	88.4	30.3
NB [49]	$\mathcal{F}$ 30% + $\mathcal{U}$ 70%	265.2	35.5
NB [49]	$\mathcal{F}$ 50% + $\mathcal{U}$ 50%	442.1	36.8
Weakly Semi-supervised Models $\downarrow$			
Ours	$\mathcal{F}$ 5% + $\mathcal{P}$ 95%	158.5	33.7
Ours	$\mathcal{F}$ 10% + $\mathcal{P}$ 90%	196.7	35.8
Ours	$\mathcal{F}$ 30% + $\mathcal{P}$ 70%	349.5	38.0
Ours	$\mathcal{F}$ 50% + $\mathcal{P}$ 50%	502.3	38.8
Fully Supervised Models $\downarrow$			
MRCNN [16]	$\mathcal{F}$ 100%	884.2	38.8
SOLOv2 [47]	$\mathcal{F}$ 100%	884.2	39.6

Table 1. **Performance trade-off of annotation budgets and AP**. We compare the methods on the COCO test-dev under various supervisions; $\mathcal{U}$ (unlabeled data), $\mathcal{I}$ (image-level label), $\mathcal{P}$ (point label), $\mathcal{B}$ (box label), $\mathcal{F}$ (full label). All methods use the same backbone network of R-101 [17]

a highly competitive result (38.8% vs. 39.7%). Moreover, the qualitative results in Figure 7 show that ours with COCO 5% can properly segment instances of various sizes. These results demonstrate that the cost-efficient point labels can be leveraged as an effective source for instance segmentation.

We also compare against other methods that use various weak labels based on the labeling budget in Table 1. According to the type and the number of labels, we calculate the labeling budget following the aforementioned cost. All methods use the same amount of total training images, so the labeling time of unlabeled data is treated as zero. Compared to the state-of-the-art semi-supervised method, NB [49], ours show superior performances when using the same amount of fully labeled data, especially when using 5% of fully labeled data (33.7% vs. 25.6%). We also achieve higher performance with a lower labeling budget (35.8% with a budget of 196.7 vs. 35.5% with a budget of 265.2). In addition, compared to the state-of-the-art box-supervised method, BoxInst [44], our efficiency is better (33.7% with a budget of 158.5 vs. 33.2% with a budget of 174.5). This result demonstrates the effectiveness of budget-friendly point labels with the proposed method. Furthermore, we emphasize the potential of our method for performance improvement when using more labeling budget.

Also, we conduct experiments on BDD100K dataset. As we increase the amount of point labeled data with a fixed amount of 7k fully labeled data, the performance is gradually improved, as in Table 6. Especially, when leveraging all available point labels (67k), ours can achieve significant performance improvements compared to using only 7k fully labeled data (22.1%→27.9%).

4.4. Ablation Study

We conduct an ablation study of our method on the COCO 10% setting. Unless otherwise specified, we measure the quality of pseudo labels generated by the teacher network using randomly sampled 5,000 images in the rest 90% of COCO data, we name it COCO *train5K*.

Effect of Point Labels. In Table 5, we verify the effectiveness of each weak label candidate (*i.e.*, unlabeled, image-level, and point label) in instance segmentation. For this analysis, we measure the quality of pseudo labels and the performance of the student network on the COCO 2017 validation set. When the unlabeled data is leveraged as a weak label, we should carefully tune the confidence threshold to balance between false-negative and false-positive proposals; the average recall (AR_{100}) and precision (AP) largely vary according to the confidence threshold. It implies that human effort for tuning the threshold is required for target datasets, and this global threshold may not be optimal for every instance. Leveraging the image-level label as a weak label can eliminate the misclassified proposals, boosting the performance from 25.9% to 29.5%. However, the performance gap with the fully-supervised setting is still significant (29.5% vs. 39.0%). When we leverage the point label as a weak label, we filter out the proposals to keep only true-positive proposals, deprecating the requirement of the confidence threshold. It makes a more straightforward and effec-

FPN	Proposal	$AP\uparrow$	$AP_{50}\uparrow$	$AR_{100}\uparrow$
P2	P2	23.4	48.3	37.6
P2$\sim$P6	P2$\sim$P6	10.3	20.8	44.7
P2$\sim$P6	$\mathrm{argmax}_{k\in\{2,3,\dots,6\}}\mathbf{s}_k$	28.6	56.7	42.6
P2$\sim$P6	w/ ground-truth size	30.9	62.0	44.9

Table 2. **Impact of choosing features adaptively**. Pn denotes n-th feature pyramid. $\mathbf{s}_i$ is the confidence score of the proposal in i-th pyramid level.

Rough mask	Point	$AP\uparrow$	$AP_{50}\uparrow$
w/o MaskRefineNet		28.6	56.7
✓		14.8	30.2
✓		29.7	54.4
	✓	30.9	52.9
✓	✓	39.1	65.3

Table 3. **Impat of the input sources for MaskRefineNet**.

Point	$AP\uparrow$	$AP_{50}\uparrow$
Center	28.6	56.7
Random	28.8	57.0

Table 4. **Robustness to point sources**. We compare APs trained with different locations – center and random points.

Label Types	COCO *train5K*			COCO *val*	
	$AP\uparrow$	$AP_{50}\uparrow$	$AR_{100}\uparrow$	$AP\uparrow$	$AP_{50}\uparrow$
$\mathcal{U}$ (τ=0.1)	6.0	11.3	33.8	20.8	33.5
$\mathcal{U}$ (τ=0.3)	13.1	22.9	23.4	25.9	41.5
$\mathcal{U}$ (τ=0.5)	12.2	19.3	15.6	24.3	38.2
$\mathcal{I}$ (τ=0.3)	19.5	33.1	24.1	29.5	48.9
$\mathcal{P}$	28.6	56.7	42.6	32.2	52.3
$\mathcal{P}^{\dagger}$	39.1	65.3	52.0	35.5	56.0

Table 5. **Impact of using point labels**. We have the notations: $\mathcal{U}$ (unlabeled data), $\mathcal{I}$ (image-level label), $\mathcal{P}$ (point label), and $\mathcal{F}$ (full label). We use COCO *train5K* to measure the quality of pseudo labels and COCO *val* to evaluate the baseline network trained with the pseudo labels. τ is a confidence threshold in the proposal branch. $\dagger$ means applying our point-guided MaskRefineNet.

Label Types	$AP\uparrow$	$AP_{50}\uparrow$	$AP_{75}\uparrow$
$\mathcal{F}$ 7k	22.1	40.2	21.2
$\mathcal{F}$ 7k + $\mathcal{P}$ 20k	26.7	44.4	27.8
$\mathcal{F}$ 7k + $\mathcal{P}$ 40k	27.3	44.5	28.9
$\mathcal{F}$ 7k + $\mathcal{P}$ 67k	27.9	44.8	29.2

Table 6. **Quantitative results on BDD100K validation set**. We report the AP scores with different training regimes concerning the number of point labels.

tive pipeline, resulting in 32.2%. Compared to the annotation cost of the image-level label (80 s/img), the point label is still budget-friendly (87.9 s/img) and gives a noticeable performance improvement (32.2% vs. 29.5%). Moreover, our point-guided MaskRefineNet further reduces the performance gap with the fully-supervised setting (35.5% vs. 39.0%). This result demonstrates that our method can effectively leverage the point label for cost-efficient and high-performance instance segmentation.

Furthermore, we test the robustness of our method to the position of the point label. We originally used the centroid point of each instance as our point label. For the analysis, we randomly choose one pixel in an instance mask as a point label five times and measure the average quality of the pseudo labels. As shown in Table 4, the performance gap between the center point and the random point is marginal. The reason is that all pixels included in the instance region within the proposal branch are trained to generate instance proposals, as in [47]. This result demonstrates the robustness of our method to the position of the point labels, which gives us more opportunity to reduce the annotation effort.

Effect of Adaptive Pyramid-Level Selection. We quantitatively analyze the behavior of the FPN in Table 2. When we produce pseudo instance masks from a single layer feature map, *i.e.*, without FPN, we achieve an unsatisfactory

pseudo label quality of 23.4% as shown in the first row in Table 2. When we generate pseudo masks from all pyramid feature maps (P2$\sim$P6), we achieve an inferior quality of 10.3% because the outputs from unfit pyramid levels are pretty noisy, as shown in Figure 4. Using our Adaptive Pyramid-Level Selection strategy, we choose one appropriate pyramid level based on the reliability of the network, achieving the improved quality of 28.6%. The result demonstrates that the proposed strategy is highly effective in leveraging the behavior of the FPN structure for generating high-quality pseudo labels. Also, the result of 30.9% when using ground-truth instance size information leaves us room for improvement of our method.

Effect of MaskRefineNet. In Figure 6, we conduct experiments on various subsets without the MaskRefineNet. When the teacher network has enough mask representation ability as in the COCO 50% setting, the improvement of the MaskRefineNet is marginal (38.3% vs. 38.8%). However, the MaskRefineNet yields a considerable performance improvement, especially in the limited number of fully labeled data settings, *e.g.*, COCO 1% (14.3%→24.0%) and COCO 5% (29.0%→33.7%) settings. The result demonstrates that MaskRefineNet is a remarkably effective method to improve the quality of pseudo labels in the limited quantity of fully labeled data conditions.

In addition, we analyze the effect of input sources of the MaskRefineNet in Table 3. Before applying the MaskRefineNet, the quality of pseudo labels is measured as 28.6%. When the MaskrefineNet only takes an image as input, the accuracy of the pseudo labels drastically reduces to 14.8% since the network fails to converge due to the ab-
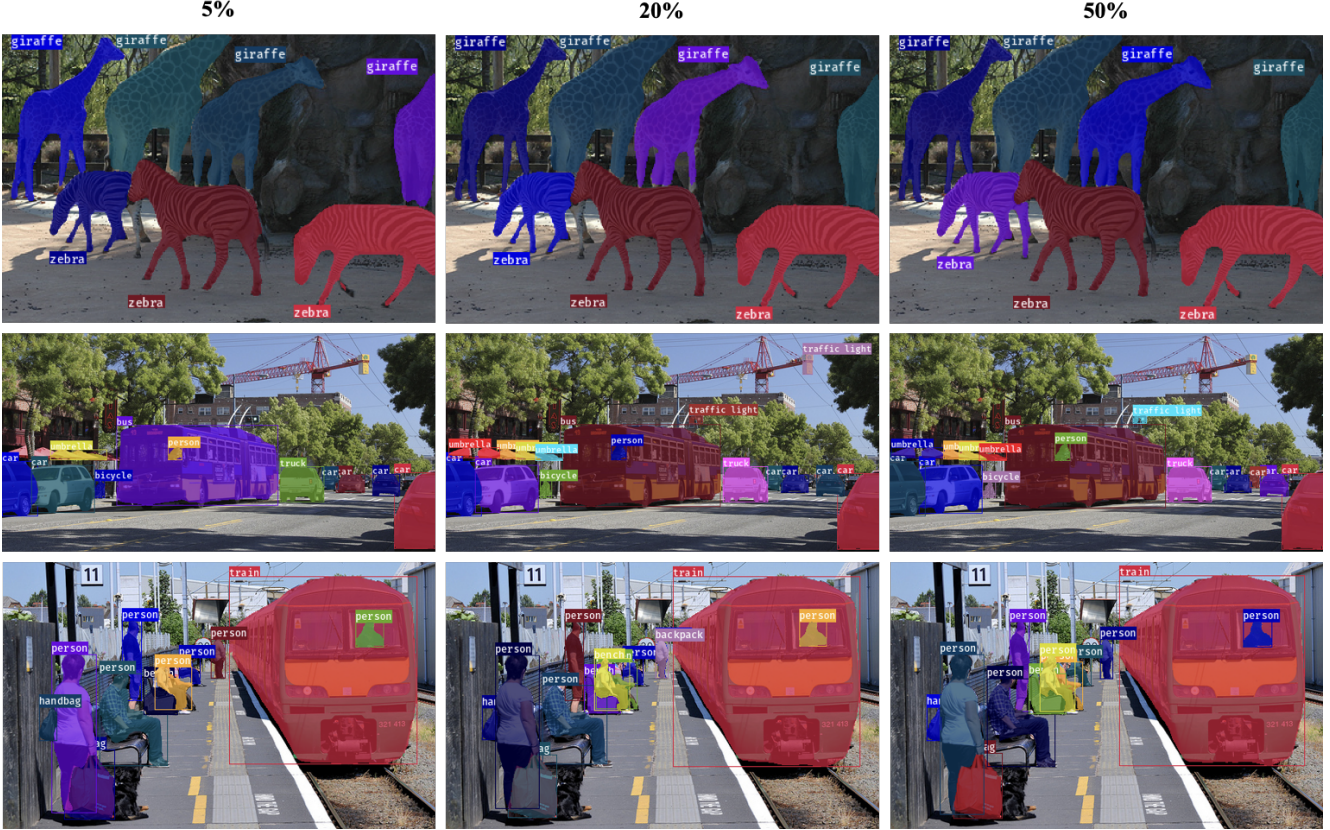
Figure 7. **Qualitative results according to the various subsets in COCO data**. We observe that training with 5% of full labeled data appropriately localizes all the object instance masks owing to our proposed point guidance method along with MaskRefineNet.

sence of prior knowledge. When taking the rough mask as an input source, the quality of the pseudo labels improves from 14.8% to 29.7% because the MaskRefineNet takes the knowledge of the teacher network for fast and stable convergence. However, the improvement is still minor compared to the model without MaskRefineNet. When additionally taking the point information as an input source, the quality of pseudo labels dramatically improves to 39.1%. The reason is that the point information is used as a guidance seed for the target instance, helping a more accurate segment of occluded instances and refining the missing predictions in the rough mask, as shown in Figure 5.

5. Conclusion and Limitation

In this paper, we proposed a novel and practical weakly semi-supervised instance segmentation scheme leveraging point labels as weak supervision for cost-efficient and high-performance instance segmentation. We motivated that the main performance bottleneck of modern instance segmentation frameworks arises from the instance proposal extraction. To this end, we proposed a method that can effectively exploit the budget-friendly point labels as weak su-

pervision to resolve the bottleneck. Moreover, we presented the MaskRefineNet to deal with hard learning scenarios where the amount of fully labeled data is extremely limited. Owing to the effectiveness of the proposed method, we can generate high-quality pseudo instance masks, achieving promising instance segmentation results. Despite our remarkable results driven by cost-efficient point labels, we have a limit to straightforwardly exploiting the tremendous amount of unlabeled image pools, such as web-crawling images without any annotations. Our future direction may involve incorporating unlabeled images in our framework to extend its application to semi-supervised learning scenarios.

Appendix: Additional Experimental Details

Labeling Budget Calculation. Seminar works [3, 4] offered the annotation time of various labeling sources (*e.g.*, full mask, bounding box, point, image-level labels) on Pascal VOC dataset [14]. Since the COCO dataset [36] we used has more categories and instances per image than the VOC dataset, we estimate the labeling budget for the COCO dataset following their budget calculation method. The COCO 2017 trainset has a total of 80 categories and contains 118,287 images and 860,001 instances. Also, it has an average of 7.2 instances and 2.9 categories per image. By considering this statistic of COCO dataset, we calculate the labeling budget as follows:

- **Full mask**: $77.1 classes/img \times 1s/class + 7.2 inst/img \times 79s/mask =$ **645.9s/img**.

- **Bounding box**: $77.1 classes/img \times 1s/class + 7.2 inst/img \times 7s/bbox =$ **127.5s/img**.

- **Point**: $77.1 classes/img \times 1s/class + 2.9 classes/img \times 2.4s/point + (7.2 inst/img - 2.9 classes/img) \times 0.9s/point =$ **87.9s/img**.

- **Image-level**: $80 classes/img \times 1s/class =$ **80s/img**.

Input of MaskRefineNet. In this section, we further provide the details about the input sources for MaskRefineNet. After training the teacher network using the fully labeled data, we generate instance mask outputs for the point-guided filtered proposals (*i.e.*, true-positive proposals) using the trained teacher network. We treat the mask outputs as rough masks to be used as the input source of the MaskRefineNet. For each rough mask, we loosely crop each instance region in the input image, rough mask, and point heatmap. Specifically, after obtaining the bounding box from the rough mask using the min-max operations, we re-scale the size of the box to double, and then we use this box region as the cropping region. In addition, for the point heatmap, we encode each point to a 2-dimensional gaussian kernel with a sigma of 6, as done in [48, 53]. We concatenate the three input sources (*i.e.*, cropped input image $\mathcal{R}^{H \times W \times 3}$, cropped rough mask $\mathcal{R}^{H \times W \times 1}$, and cropped point heatmap $\mathcal{R}^{H \times W \times C}$) to be the input tensor $\mathcal{R}^{H \times W \times (3+1+C)}$ of the MaskRefineNet, where C is the number of classes.

Appendix: Additional Analysis

Effect of the input size of MaskRefineNet. We originally set the input size of MaskRefineNet to 256×256. Here, we change the input size to verify its effect on the WSSIS result in table 7. For this, we train the MaskRefineNet using the input size of 128×128 or 384×384. We

Input Size	AP	AP_{50}	AP_{75}
128×128	34.1	53.4	36.1
256×256	35.5	56.0	37.8
384×384	35.5	55.9	37.7

Table 7. **Effect of the input size of MaskRefineNet**. The APs are measured on COCO 2017 validation set.

Iterative	1%	2%	5%	10%	30%	50%	100%
	23.9	25.1	33.4	35.5	37.4	38.3	39.0
✓	25.6	26.0	34.5	35.9	37.6	38.3	39.0

Table 8. **Effect of iterative training strategy.** The APs are measured on COCO 2017 validation set according to COCO subsets.

Method	Label Types	Budget (days) ↓	AP (%) ↑
Weakly-supervised Models			
BBTP [20]	$\mathcal{B}$ 100%	174.5	21.1
BBAM [32]	$\mathcal{B}$ 100%	174.5	25.7
BoxInst [44]	$\mathcal{B}$ 100%	174.5	33.2
BoxLevelSet [34]	$\mathcal{B}$ 100%	174.5	33.4
BoxTeacher [12]	$\mathcal{B}$ 100%	174.5	35.4
Point-sup [11]	$\mathcal{P}_{10}$ 100%	263.2	37.7
Weakly Semi-supervised Models			
Ours	$\mathcal{F}$ 5% + $\mathcal{P}$ 95%	158.5	33.7
Ours	$\mathcal{F}$ 10% + $\mathcal{P}$ 90%	196.7	35.8
Ours	$\mathcal{F}$ 20% + $\mathcal{P}$ 80%	273.1	37.1
Ours	$\mathcal{F}$ 30% + $\mathcal{P}$ 70%	349.5	38.0
Ours	$\mathcal{F}$ 50% + $\mathcal{P}$ 50%	502.3	38.8
Fully Supervised Models			
MRCNN [16]	$\mathcal{F}$ 100%	884.2	38.8
CondInst [43]	$\mathcal{F}$ 100%	884.2	39.1
SOLOv2 [47]	$\mathcal{F}$ 100%	884.2	39.7

Table 9. **Additional comparisons with weakly-supervised methods in terms of labeling budget and accuracy**. We compare the methods on the COCO *test-dev* under various supervisions; $\mathcal{B}$ (box label), $\mathcal{P}_{10}$ (10-points label), $\mathcal{P}$ (single-point label), $\mathcal{F}$ (full mask label). All methods use the same backbone network of ResNet-101 [17].

measure the AP result of the student network trained with the pseudo and full labels on the COCO 2017 validation set. Consequently, the 256×256 size yields the best AP score of 35.5% but its performance gap with the 384×384 size is marginal (35.5% vs 35.4%).

Effect of iterative training strategy. Some weakly-supervised methods [2,25,46] utilize iterative training strategy; after training the target network, they generate pseudo labels using the target network, and then they newly train the target network using the pseudo labels. This strategy could give additional performance improvement but de-

Method	5%	10%	20%	30%	40%	50%
Point DETR [10]	26.2	30.3	33.3	34.8	35.4	35.8
Group R-CNN [52]	30.1	32.6	34.4	35.4	35.9	36.1
ours	32.4	34.3	35.6	36.9	37.0	37.6

Table 10. **Qualitative comparisons on COCO *test-dev* object detection benchmark**. All methods used the ResNet-50 backbone.

mands a more complex training pipeline. In this work, we suffer from the insufficient mask representation of the network when the amount of fully labeled data is extremely limited (*e.g.*, COCO 1%). Although we can alleviate the problem with the proposed MaskRefineNet, we additionally try to adopt this strategy since we assume that the trained student network may have stronger mask representation ability than the teacher network. For this, after training the student network, we newly generate pseudo instance masks for point labeled images. Using both full labels and new pseudo labels, we train a new student network. As the results in table 8, the iterative training strategy yields meaningful improvements on tiny fully labeled data conditions (COCO 1%: 23.9%→25.6%). However, there is no significant performance improvement for subsets above COCO 30%. This result demonstrates that (1) the iterative training strategy is helpful only when the amount of fully labeled data is extremely limited, (2) in more generous conditions such as COCO 30% and 50%, our MaskRefineNet is enough to replenish the mask representation of the network.

Additional Comparison with weakly-supervised method: Point-sup [11] introduced a new type of weak supervision source, multiple (10) points. They achieved remarkable instance segmentation results with a highly reduced annotation cost. To compare with them, we estimate the annotation time for 10-points according to the literature; they labeled 10-points in the bounding box region.

- **10 Points**: $77.1 classes/img \times 1s/class + 7.2 inst/img \times (7s/bbox + 10 points \times 0.9s/point) = $ **192.3s/img**.

In table 9, we provide the results for weakly-supervised methods and ours on COCO *test-dev* in terms of accuracy and labeling budget. Although Point-sup shows a slightly better efficiency than ours (37.7% with a budget of 263.2 days vs. 37.1% with a budget of 273.1 days), we argue that our training setting is more applicable for the current dataset conditions than them because they require newly annotating of 10-points. Also, we show the possibility for more performance improvement up to 38.8%, which is highly close to the result of the fully-supervised setting. Furthermore, they give us a new future direction; incorporating 10-points and single-point without any mask labels.

Comparison with weakly semi-supervised object detection methods: In our main paper, we discussed the weakly semi-supervised object detection (WSSOD) methods [10, 52], which used the box labels as strong labels and the point labels as weak labels. Since the instance segmentation covers object detection, we measure our performance on the COCO *test-dev* object detection benchmark. For this, we use the min-max points from the instance mask output as our bounding box output. Even though our strong label is different from theirs (full mask vs. bounding box), the results in table 10 show that ours can surpass the state-of-the-art WSSOD performance. We note that all methods use the same ResNet-50 [17] backbone network and the same amount of total strong and weak labels.

Qualitative analysis for the effect of input sources of MaskRefineNet: In Table 2 of our main paper, we provided the quantitative analysis of the effect of input sources of MaskRefineNet. Here, we supplement our analysis with the qualitative results according to the input sources of the MaskRefineNet in Figure 8. When given all three informative input sources, the MaskRefineNet can produce high-quality refined masks by separating overlapping instances and removing noisy pixels.

Qualitative comparison of baselines and our WSSIS method. In Figure 6 of our main paper, we provided the AP evolution of two baselines and our WSSIS method according to the COCO subsets. In Figure 9, we provide the qualitative results of two baselines and our method under the COCO 10% setting. There are four types of methods: (a) training with fully labeled data only, (b) training with fully labeled data and unlabeled data, (c) training with fully labeled data and point labeled data, and (d) training with fully labeled data and point labeled data along with our point-guided MaskRefineNet. The results demonstrate that the network trained with our method can be guided with higher-quality pseudo labels, resulting less false-positive and false-negative outputs.

Additional qualitative results on COCO dataset. In Figure 10, we provide additional qualitative results of ours trained with 5%, 20%, and 50% COCO subsets.

Qualitative results on BDD100K dataset. We qualitatively analyze the effect of leveraging point labels for the instance segmentation model using the BDD100K dataset [51]. There are two types of networks: the first is the network trained with only 7K fully labeled data, and the second is the network trained with 7K fully labeled data and 67K point labeled data. As shown in Figure 11, due to our effective leveraging of the point labels, the second network is much more robust to large and small instances and occluded instances.
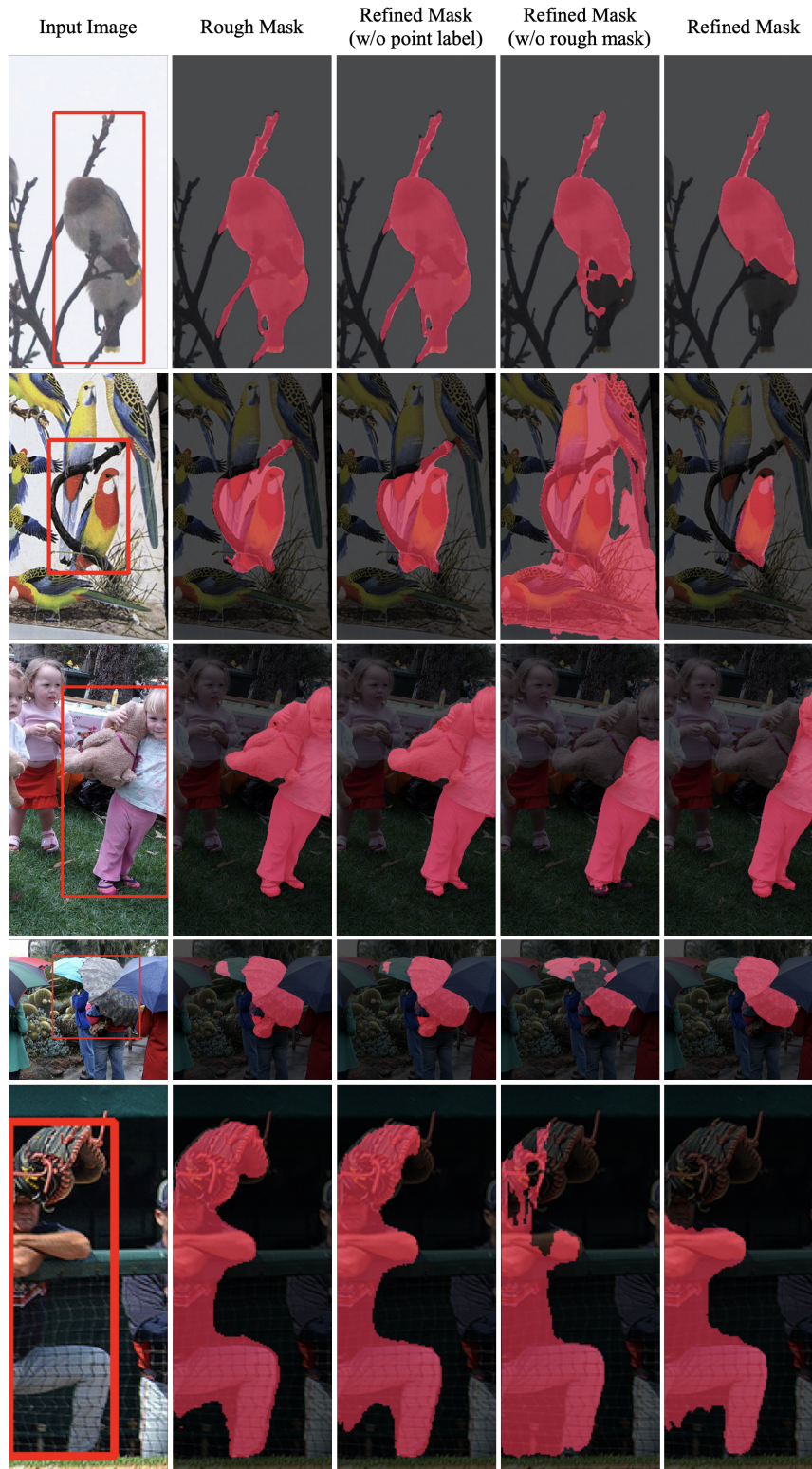
Figure 8. **Qualitative analysis of the effect of input sources of MaskRefineNet**. Object instances can not be distinguished when the point label is not given for MaskRefineNet (3rd col). Meanwhile, mask representations are inaccurate due to the absence of prior rough masks (4th col). Based on these low-cost priors, we can obtain sophisticated masks per object instance (5th col).
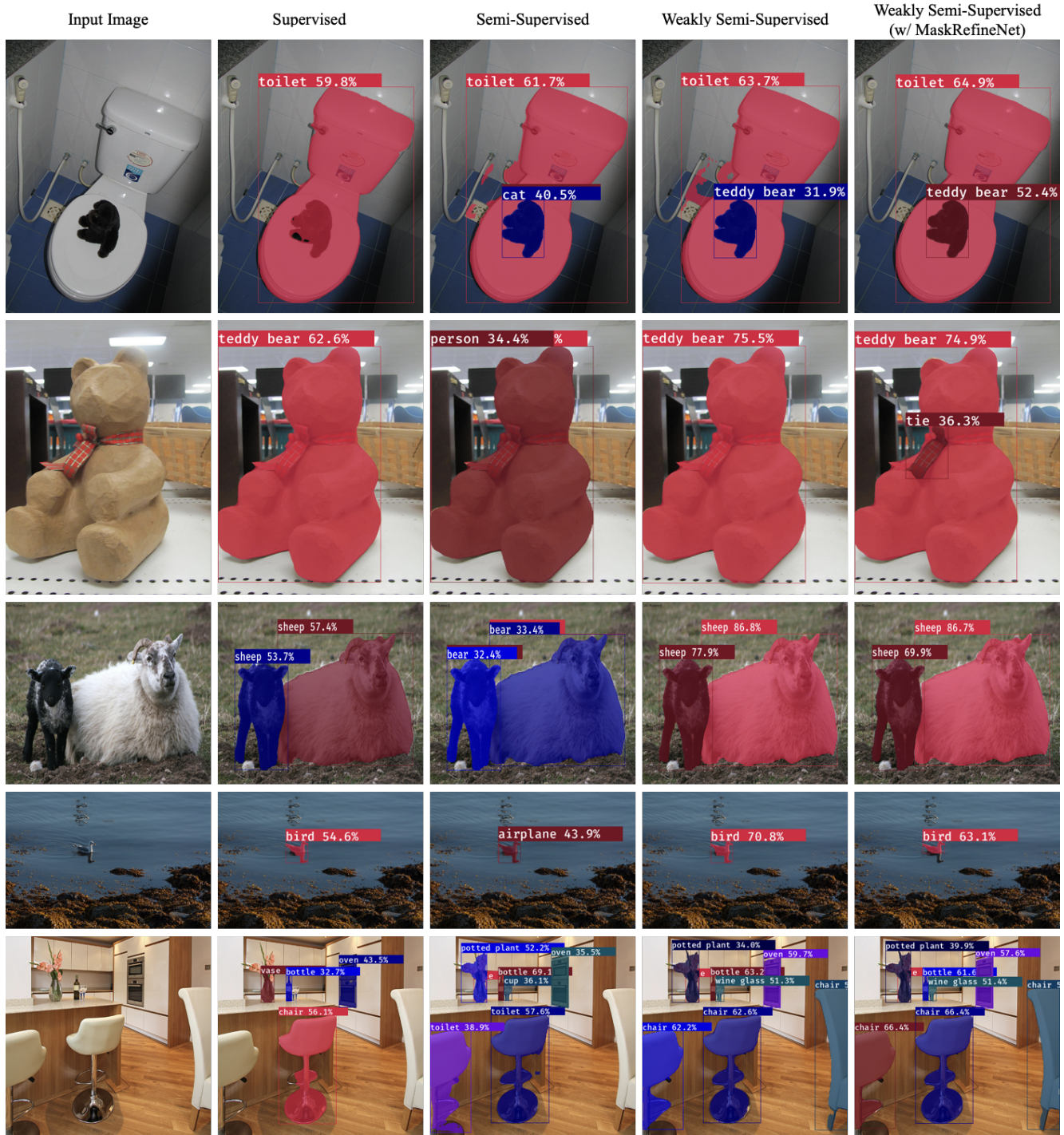
Figure 9. **Qualitative comparison of models trained with different types of supervision on COCO 10% setting**. The result of the semi-supervised setting can detect instance masks for all objects but is vulnerable to misclassification (e.g. cat, person, bear, airplane, toilet). Meanwhile, our point-guided model presents accurate class predictions. Our MaskRefineNet further elaborates the mask representation.
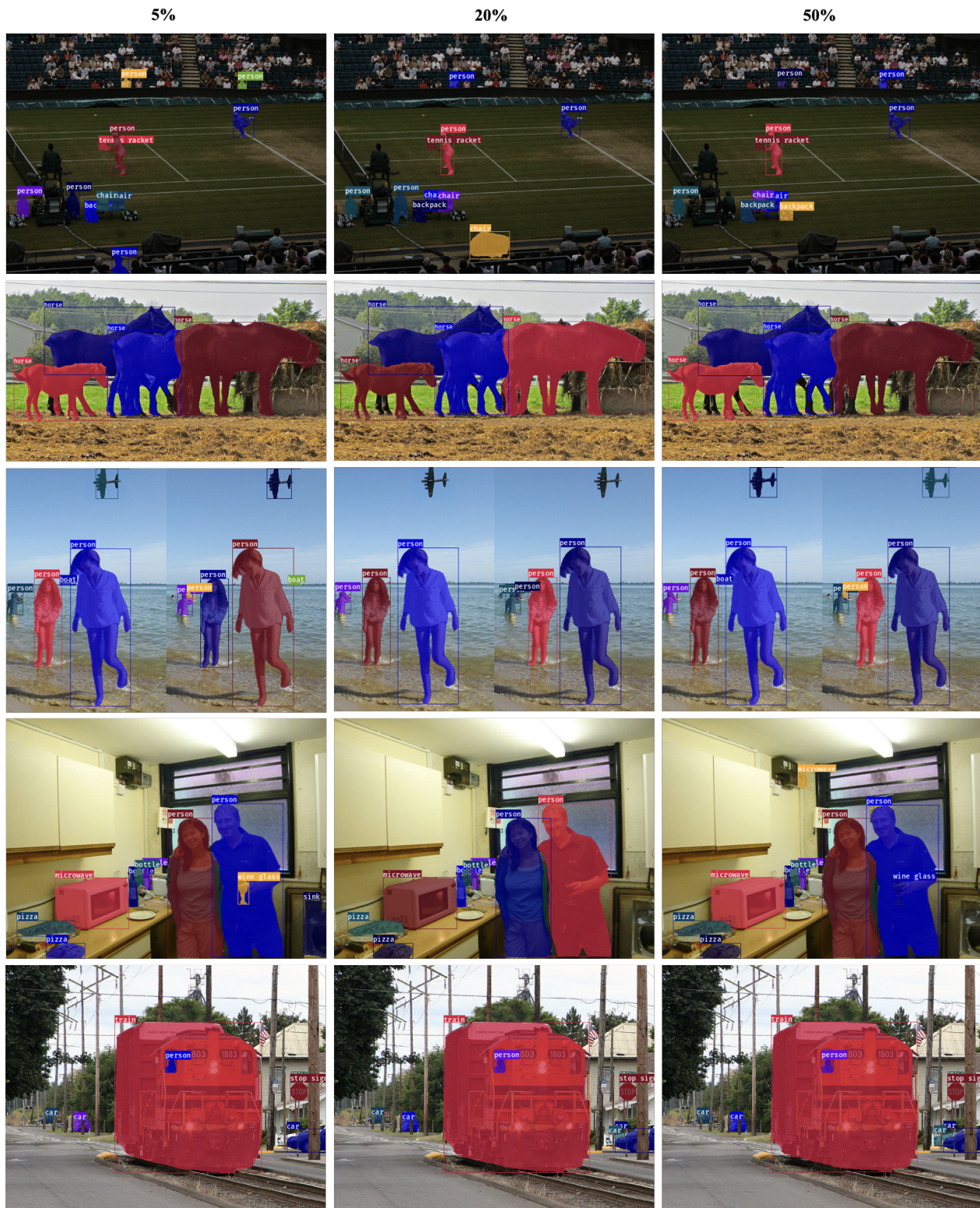
Figure 10. **Additional qualitative results according to the various subsets in COCO data**. Owing to our point guidance along with MaskRefineNet, leveraging only 5% of full labeled data sufficiently localizes all the instances with elaborative mask representations.
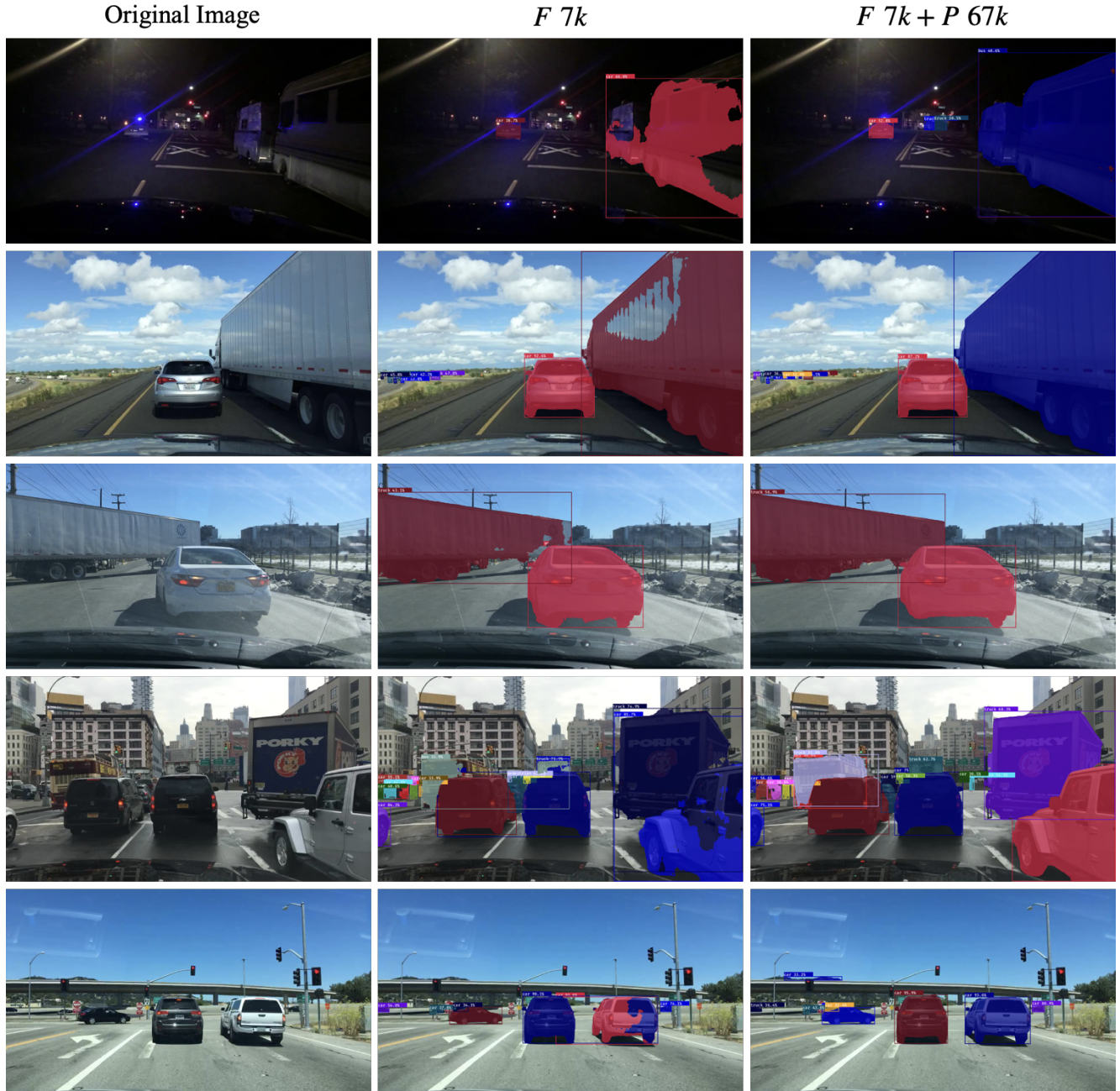
Figure 11. **Qualitative comparison of leveraging point labels on BDD100K**. Training with point labels clearly enriches the mask representation and removes the noise incurred by visually hard samples (*e.g.*, dark light condition in the first row).

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2209–2218, 2019. 1, 3

[2] Aditya Arun, CV Jawahar, and M Pawan Kumar. Weakly supervised instance segmentation by learning annotation con-

sistent instances. In *European Conference on Computer Vision*, pages 254–270. Springer, 2020. 9, 10

[3] 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*, pages 549–565. Springer, 2016. 2, 3, 4, 5, 9

[4] Míriam Bellver Bueno, Amaia Salvador Aguilera, Jordi Torres Viñals, and Xavier Giró Nieto. Budget-aware semi-

supervised semantic and instance segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2019*, pages 93–102, 2019. 3, 4, 5, 9

[5] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9157–9166, 2019. 1

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 3

[7] Sungmin Cha, Beomyoung Kim, YoungJoon Yoo, and Taesup Moon. Ssul: Semantic segmentation with unknown label for exemplar-based class-incremental learning. *Advances in neural information processing systems*, 34:10919–10930, 2021. 10

[8] Hao Chen, Kunyang Sun, Zhi Tian, Chunhua Shen, Yongming Huang, and Youliang Yan. Blendmask: Top-down meets bottom-up for instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8573–8581, 2020. 1

[9] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4974–4983, 2019. 1, 2, 3, 4

[10] Liangyu Chen, Tong Yang, Xiangyu Zhang, Wei Zhang, and Jian Sun. Points as queries: Weakly semi-supervised object detection by points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8823–8832, 2021. 3, 10

[11] Bowen Cheng, Omkar Parkhi, and Alexander Kirillov. Pointly-supervised instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2617–2626, 2022. 1, 3, 9, 10

[12] Tianheng Cheng, Xinggang Wang, Shaoyu Chen, Qian Zhang, and Wenyu Liu. Boxteacher: Exploring high-quality pseudo labels for weakly supervised instance segmentation. *arXiv preprint arXiv:2210.05174*, 2022. 9, 10

[13] Bin Dong, Fangao Zeng, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Solq: Segmenting objects by learning queries. *Advances in Neural Information Processing Systems*, 34:21898–21909, 2021. 1

[14] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 3, 9

[15] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6910–6919, 2021. 1

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1, 2, 3, 4, 6, 9

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5, 6, 9, 10

[18] Ruifei He, Jihan Yang, and Xiaojuan Qi. Re-distributing biased pseudo labels for semi-supervised semantic segmentation: A baseline investigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6930–6940, 2021. 10

[19] Cheng-Chun Hsu, Kuang-Jui Hsu, Chung-Chi Tsai, Yen-Yu Lin, and Yung-Yu Chuang. Weakly supervised instance segmentation using the bounding box tightness prior. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 3

[20] Cheng-Chun Hsu, Kuang-Jui Hsu, Chung-Chi Tsai, Yen-Yu Lin, and Yung-Yu Chuang. Weakly supervised instance segmentation using the bounding box tightness prior. *Advances in Neural Information Processing Systems*, 32, 2019. 9

[21] Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick. Learning to segment every thing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4233–4241, 2018. 10

[22] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6409–6418, 2019. 2, 3

[23] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. *Advances in neural information processing systems*, 32, 2019. 10

[24] Lei Ke, Martin Danelljan, Xia Li, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Mask transfiner for high-quality instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4412–4421, 2022. 1

[25] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 876–885, 2017. 9, 10

[26] Beomyoung Kim, Sangeun Han, and Junmo Kim. Discriminative region suppression for weakly-supervised semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1754–1761, 2021. 10

[27] Beomyoung Kim, Youngjoon Yoo, Chae Eun Rhee, and Junmo Kim. Beyond semantic to instance segmentation: Weakly-supervised instance segmentation via semantic knowledge transfer and self-refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4278–4287, 2022. 1, 3, 6

[28] Issam H Laradji, Negar Rostamzadeh, Pedro O Pinheiro, David Vazquez, and Mark Schmidt. Proposal-based instance segmentation with point supervision. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 2126–2130. IEEE, 2020. 1, 3

[29] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural net-

15

works. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013. 10

[30] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5267–5276, 2019. 10

[31] Jungbeom Lee, Eunji Kim, and Sungroh Yoon. Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4071–4080, 2021. 10

[32] Jungbeom Lee, Jihun Yi, Chaehun Shin, and Sungroh Yoon. Bbam: Bounding box attribution map for weakly supervised semantic and instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2643–2652, 2021. 1, 3, 6, 9

[33] Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13906–13915, 2020. 1, 2, 3

[34] Wentong Li, Wenyu Liu, Jianke Zhu, Miaomiao Cui, Xian-Sheng Hua, and Lei Zhang. Box-supervised instance segmentation with level set evolution. In *European Conference on Computer Vision*, pages 1–18. Springer, 2022. 9, 10

[35] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 2, 4, 5

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2, 5, 9

[37] Yun Liu, Yu-Huan Wu, Pei-Song Wen, Yu-Jun Shi, Yu Qiu, and Ming-Ming Cheng. Leveraging instance-, image- and dataset-level information for weakly supervised instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 10

[38] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016. 5

[39] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12674–12684, 2020. 10

[40] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5

[41] Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. Detectors: Detecting objects with recursive feature pyramid and switch-able atrous convolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10213–10224, 2021. 2

[42] Yunhang Shen, Rongrong Ji, Yan Wang, Yongjian Wu, and Liujuan Cao. Cyclic guidance for weakly supervised joint detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 697–707, 2019. 10

[43] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *European conference on computer vision*, pages 282–298. Springer, 2020. 1, 2, 3, 4, 9

[44] Zhi Tian, Chunhua Shen, Xinlong Wang, and Hao Chen. Boxinst: High-performance instance segmentation with box annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5443–5452, 2021. 1, 3, 6, 9

[45] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. In *European Conference on Computer Vision*, pages 649–665. Springer, 2020. 2, 3

[46] Xiang Wang, Shaodi You, Xi Li, and Huimin Ma. Weakly-supervised semantic segmentation by iteratively mining common object features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1354–1362, 2018. 9

[47] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. *Advances in Neural information processing systems*, 33:17721–17732, 2020. 1, 2, 3, 4, 5, 6, 7, 9

[48] Yuqing Wang, Zhaoliang Xu, Hao Shen, Baoshan Cheng, and Lirong Yang. Centermask: single shot instance segmentation with point representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9313–9321, 2020. 2, 3, 9

[49] Zhenyu Wang, Yali Li, and Shengjin Wang. Noisy boundaries: Lemon or lemonade for semi-supervised instance segmentation? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16826–16835, 2022. 1, 2, 3, 6

[50] Ziang Yan, Jian Liang, Weishen Pan, Jin Li, and Changshui Zhang. Weakly-and semi-supervised object detection with expectation-maximization algorithm. *arXiv preprint arXiv:1702.08740*, 2017. 10

[51] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020. 2, 5, 10

[52] Shilong Zhang, Zhuoran Yu, Liyang Liu, Xinjiang Wang, Aojun Zhou, and Kai Chen. Group r-cnn for weakly semi-supervised object detection with points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9417–9426, 2022. 3, 10

[53] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 9

[54] Yanzhao Zhou, Xin Wang, Jianbin Jiao, Trevor Darrell, and Fisher Yu. Learning saliency propagation for semi-supervised instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10307–10316, 2020. 1, 3

[55] Yanzhao Zhou, Yi Zhu, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Weakly supervised instance segmentation using class peak response. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3791–3800, 2018. 1, 3, 10