Instance Segmentation with Point Supervision

Issam H. Laradji[1,2], Negar Rostamzadeh[1], Pedro O. Pinheiro[1], David Vazquez[1], Mark Schmidt[2,1]
[1]Element AI, Montreal, Canada [2]University of British Columbia, Vancouver, Canada
{issamou,schmidtm}@cs.ubc.ca {negar,pedro,dvazquez}@elementai.com

Abstract

Instance segmentation methods often require costly per-pixel labels. We propose a method that only requires point-level annotations. During training, the model only has access to a single pixel label per object, yet the task is to output full segmentation masks. To address this challenge, we construct a network with two branches: (1) a localization network (L-Net) that predicts the location of each object; and (2) an embedding network (E-Net) that learns an embedding space where pixels of the same object are close. The segmentation masks for the located objects are obtained by grouping pixels with similar embeddings. At training time, while L-Net only requires point-level annotations, E-Net uses pseudo-labels generated by a class-agnostic object proposal method. We evaluate our approach on PASCAL VOC, COCO, KITTI and CityScapes datasets. The experiments show that our method (1) obtains competitive results compared to fully-supervised methods in certain scenarios; (2) outperforms fully- and weakly- supervised methods with a fixed annotation budget; and (3) is a first strong baseline for instance segmentation with point-level supervision.

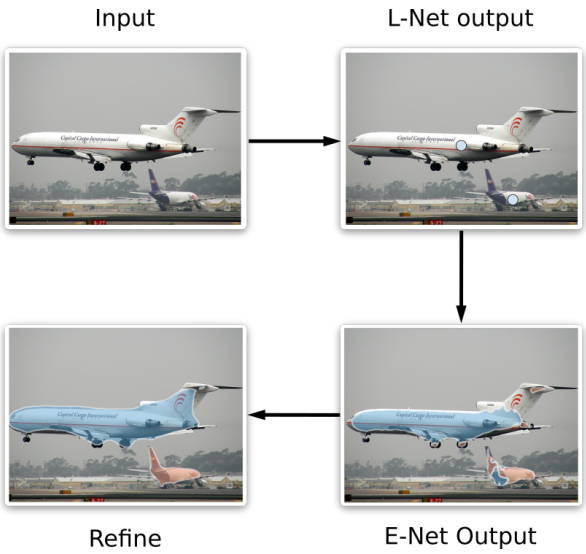

Figure 1. **WISE network.** Our method, WISE, is trained using point-level annotations only. At test time, WISE first uses L-Net to locate the objects in the image, and then uses E-Net to predict the masks of the located objects. Finally, the predicted masks are refined with the help of an object proposal method.

1. Introduction

Instance segmentation is the task of classifying every object pixel into a category and discriminating between individual object instances. It has a wide variety of applications such as autonomous driving [9], scene understanding [31, 12], and medical imaging [40].

Most instance segmentation methods, such as Mask-RCNN [17] and MaskLab [6], rely on per-pixel labels which requires huge human effort. For instance, obtaining labels for PASCAL VOC [12] requires an average time of 239.7 seconds per image [4]. Other datasets with more objects to annotate such as CityScapes [9] can take up to 1.5 hours per image.

Indeed, having a method that can train with weaker supervision can vastly reduce the required annotation cost. According to Bearman *et al.* [4], manually collecting image-level and point-level labels for the PASCAL VOC dataset took only 20.0 and 22.1 seconds per image, respectively. These annotation methods are an order of magnitude faster than acquiring full segmentation labels (see Figure 2 for a comparison between the point-level and per-pixel annotation methods).

For semantic segmentation, other forms of weaker labels were explored such as bounding boxes [20], scribbles [30], and image-level annotation [56]. For instance segmentation, few works exist that use weak supervision [56, 8]. In this paper, we propose a Weakly-supervised Instance SEgmentation (WISE) network, the first to address this task with point-level annotations.

WISE has two branches: (1) a localization network (L-Net) that predicts the location of each object; and (2) an embedding network (E-Net) that learns an embedding space where pixels of the same object are closer. L-Net is trained

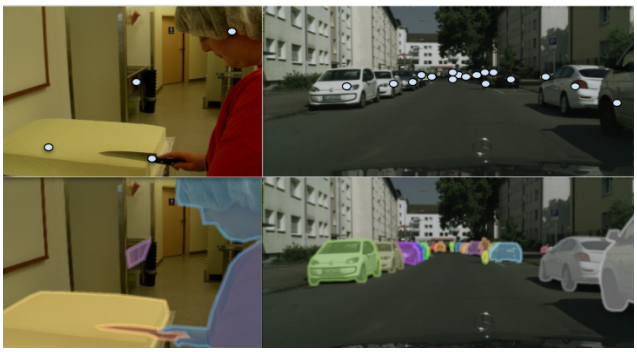

Figure 2. **Image annotation.** Point-level (top) and per-pixel (bottom) labels for COCO and the CityScapes datasets.

using a loss function that forces the network to output a single point per object instance. E-Net is trained using a similarity-based objective function to force the pixel embeddings to be similar within the same object mask. Since we do not have access to the ground-truth object masks, we instead use pseudo-masks generated by an object proposal method. These pseudo-masks belong to arbitrary objects and have no class labels and therefore cannot be directly applied for instance segmentation. At test time, L-Net first predicts the object locations. Second, E-Net outputs the embedding value for each pixel. Then the pixels with the most similar embeddings to an object's predicted pixel location become part of that object's mask (Figure 1).

We summarize our contributions as follows: (1) we provide a first strong baseline for instance segmentation with point-level supervision; (2) we evaluate our method on a wide variety of datasets, including, PASCAL VOC [12], COCO [31], CityScapes [9], and KITTI [15] datasets; (3) we obtain competitive results compared to fully-supervised methods; and (4) our method outperforms fully- and weakly- supervised methods when the annotation budget is limited.

2. Related Work

Our approach lies at the intersection of object localization, metric learning, object proposal methods, and instance segmentation. These topics have been studied extensively and we review the literature below. The novelty of our method is the combination of these techniques into a new setup, namely, instance segmentation with point-level supervision.

Instance segmentation. Instance segmentation is an important computer vision task that can be applied in many real-life applications [43, 45]. This task consists of classifying every object pixel into categories and distinguishing between object instances. Most methods follow a two step procedure [17, 6, 14], where they first detect objects and then segment them. For instance, Mask-RCNN [17] uses Faster-RCNN [44] for detection and an FCN network [33] for segmentation. However, these methods require dense labels which leads to a high annotation time for new applications.

Embedding-based instance segmentation. Another class of instance segmentation methods obtain the object masks by grouping pixels based on a similarity measure. Notable works in this category include methods based on watershed [3], template matching [52] and associative embedding [36]. Fathi *et al.* [13] propose a grouping-based method that first learns the object locations and then learns the pixel embeddings in order to distinguish between object instances. These methods also require per-pixel labels which are costly to acquire for new applications. However, our method follows a similar procedure for obtaining the segmentation masks while requiring weaker supervision.

Weakly supervised instance segmentation. Per-pixel labels used by fully supervised instance segmentation methods require high annotation cost [12, 9]. Therefore many weakly supervised methods have been explored for object detection [51, 5], semantic segmentation [37, 23, 1, 47] and instance segmentation [20, 56, 8]. Point-level annotation is one of the fastest ways to annotate object instances, albeit one of the least informative forms of weak supervision. However, they were shown to be effective for semantic segmentation [4]. Inspired by their cost-effectiveness, we explore the novel problem setup of instance segmentation with point-supervision in this work.

Object localization with point supervision. An important step in instance segmentation is to locate objects of interest before segmenting them. One way to perform object localization is to use object detection methods [44, 41]. However, these methods require bounding-box labels. In contrast, several methods exist that use weaker supervision to identify object locations [49, 50, 26, 27]. Close to our work is LCFCN [25] which uses point-level annotations in order to obtain the locations and counts of the objects of interest. While this method gives accurate counts and identifies a partial mask for each instance, it does not produce accurate segmentation of the instances. We extend this method by using an embedding network that groups pixels that are most similar to the predicted object locations in order to obtain their masks.

Object proposals. Weakly supervised methods often rely on object proposals [19] to ease the task of detection [51, 5], and segmentation [37, 4, 56, 23]. Object proposals are class-agnostic methods that can output thousands of object candidates per image and have received great progress over the last decade [53, 58, 2, 34, 38, 39]. SharpMask [39] is a popular deep-learning based object proposal method

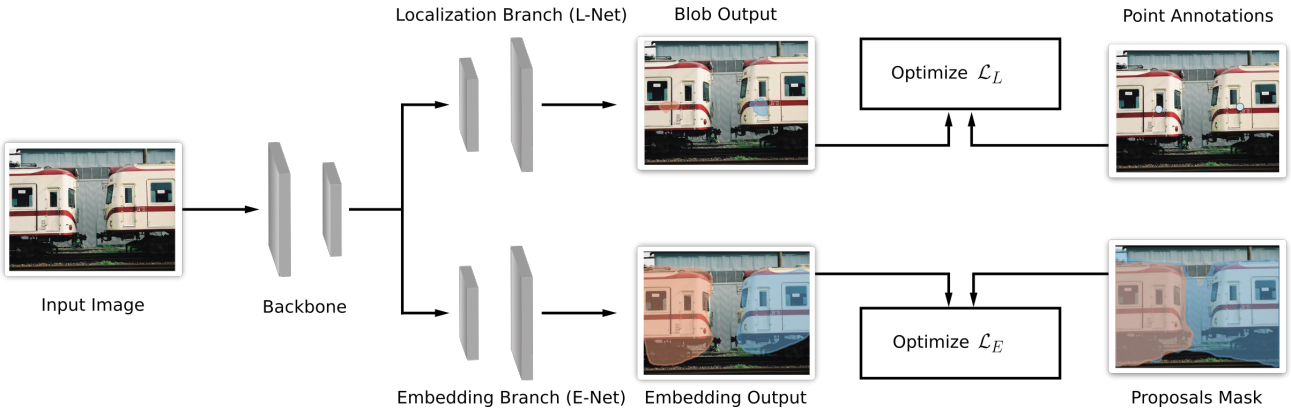

Figure 3. **Training WISE.** Our method consists of a localization branch (L-Net) and an embedding branch (E-Net). During training, L-Net optimizes Eq. 1 in order to output a single point per object instance. E-Net optimizes Eq. 3 in order to group pixels that belong to the same object instance.

that has been successfully applied to many weakly supervised computer vision problems. However, their output object masks cannot be directly used for instance segmentation as they belong to arbitrary objects and have no class labels. Our framework uses pseudo-masks generated by SharpMask.

3. Proposed Method

We address the problem of weakly-supervised instance segmentation, where each labeled object has a single point annotation. Our method, WISE network, has two output branches that share a common feature extraction backbone (Figure 3): (1) a localization branch (L-Net) that is trained for locating objects in the image, and (2) an embedding branch (E-Net) that outputs an embedding vector for each pixel. L-Net is trained using point-level annotations in order to output a single pixel for each object to represent its location and category in the image. On the other hand, E-Net is trained using pseudo-masks obtained by a pretrained proposal method. This allows E-Net to output an embedding vector for each pixel such that similar ones belong to the same object's pseudo-mask. Note that proposal methods have been widely used for different weakly-supervised problem setups [56, 8, 5, 37, 4]

WISE obtains the mask of an object as follows. First, L-Net outputs a pixel label per object to identify its location, category, and instance. Then, the embedding of every pixel in the image is compared to the embedding of the pixels predicted by L-Net to identify which object instance they belong to. Finally, the pixels are grouped to form the object masks in the image.

3.1. Localization Branch (L-Net)

The goal of L-Net is to obtain the locations and categories of the objects in the image. L-Net is based on LC-FCN [25] which trains with point level annotations to produce a single blob per object. While this was originally designed for counting, it is able to locate objects effectively. LC-FCN is based on a semantic segmentation architecture that is similar to FCN [33]. Indeed, semantic segmentation methods are not suitable for instance segmentation as they often predict large blobs that merge several object instances together. LC-FCN addresses this issue by optimizing a loss function that ensures that only a single small blob is predicted around the center of each object.

The location loss term $\mathcal{L}_L$ is described as:

$$\mathcal{L}_L = \underbrace{\mathcal{L}_I(S,T)}_{\text{Image-level loss}} + \underbrace{\mathcal{L}_P(S,T)}_{\text{Point-level loss}} + \underbrace{\mathcal{L}_S(S,T)}_{\text{Split-level loss}} + \underbrace{\mathcal{L}_F(S,T)}_{\text{False positive loss}}, \quad (1)$$

where T represent the point annotation ground-truth, and S is LC-FCN's output mask. $\mathcal{L}_L$ consists of four terms: an image-level loss ($\mathcal{L}_I$) that trains the model to predict whether there is an object in the image; a point-level loss ($\mathcal{L}_P$) that encourages the model to predict a pixel for each object instance; a split-level ($\mathcal{L}_S$) and a false-positive ($\mathcal{L}_F$) loss that enforce the model to predict a single blob per instance (see [25] for details for each of the loss components). Since LC-FCN's predicted blobs are too small to be considered as useful segmentation masks, we instead leverage the location of each blob by identifying the pixel with the highest probability of being foreground (Figure 4).

3.2. Embedding Branch (E-Net)

The goal of E-Net is to produce object masks by grouping pixels with similar embeddings together. E-Net's architecture is based on FCN8 [33], which can output an embedding vector per image pixel. Using a similarity loss, E-Net

Figure 4. **Localization branch (L-Net).** L-Net's raw output is a small blob per predicted object (top). L-Net's final output is the set of pixels with the largest activation within their respective blobs (bottom). These pixels are used as input to E-Net at test time.

learns to output similar embeddings for pixels that belong to the same object and dissimilar otherwise. This loss requires several points per object (including the background) in order to distinguish between different objects. While we do not have access to the ground-truth masks, we instead use pseudo-masks generated by an object proposal method to assign a mask for each object.

E-Net learns a mapping from an input image to a set of embedding vectors of size d for each pixel. Let E_i and E_j be the embeddings for pixel i and pixel j, respectively. We measure the similarity between a pair of pixels using a squared exponential kernel function, similar to that of Fathi *et al.* [13]:

$$S(i,j) = \exp\left(-\frac{||E_i - E_j||_2^2}{2d}\right), \tag{2}$$

where $S(E_i, E_j)$ tends to 1 as E_i and E_j get closer, and tends to 0 as they get farther in the embedding space. Note that our method can use other similarity functions as in [36, 13, 24].

Our goal is to train E-Net such that embeddings of pixel pairs belonging to the same object instance (*i.e.* $y_i = y_j$) have the same embedding (*i.e.* $S(i,j) = 1$) and to different object instances (*i.e.* $y_i \neq y_j$) have different embeddings (*i.e.* $S(i,j) = 0$). Therefore, E-Net minimizes the following loss function[1]:

$$\mathcal{L}_E = -\sum_{(i,j)\in P} \Big[\mathbb{1}_{\{y_i=y_j\}} \log S(E_i,E_j) \; + \; \mathbb{1}_{\{y_i\neq y_j\}} \log\left(1 - S(E_i,E_j)\right)\Big], \tag{3}$$

where P is a set of pixel pairs.

[1]Note that the log and exp cancel out in the first term of the equation but not the second term.

Figure 5. **Pseudo-mask labels**. (Left) ground-truth point-level annotations; (Center) a set of generated object proposals that intersect with the point annotations; (Right) proposals with best "objectness".

Since we require more than one point label per object to optimize Equation 3, we use extra points from pseudo-masks generated by an object proposal method (see Figure 5). At each training iteration, the pseudo-mask of an object is randomly selected from the set of proposals (obtained by the proposal method) that intersect with the object's point annotation. Further, we define the background as the region that does not contain any proposal mask.

We obtain the set of pixel pairs P for Eq. 3 as follows. We pair each pixel represented by the point-level annotation with k random pixels[2] from each object's pseudo-mask including the background region. This randomness allows the model to learn the important pixels that correspond to the objects of interest. The final objective function of WISE is defined as:

$$\mathcal{L}_W = \lambda \cdot \mathcal{L}_L + (1-\lambda) \cdot \mathcal{L}_E \,, \tag{4}$$

where λ is the weight that balances between L-Net's and E-Net's loss terms.

3.3. Prediction at Test Time

WISE predicts masks of objects using the following steps. First, L-Net outputs a pixel coordinate for each object representing its location and category. Second, E-Net outputs the embedding vectors for all pixels in the image. Third, we compute the similarity (Equation 2) between each pixel in the image and two sets of pixels: (1) L-Net's predicted pixel coordinates, and (2) several selected background pixels. Next, we assign each pixel to the most similar object, resulting in a mask for each object including the background region. Finally, the object masks are refined by replacing them with the pseudo-mask (generated from a proposal method) with the largest Jaccard similarity (see Figure 1).

For selecting the background pixels deterministically, we first define the background regions as the pixels that do not correspond to any of the generated proposal masks. We use the k-means algorithm for clustering the pixels embeddings into k groups. Then, for each cluster we select the closest pixel to the mean of that cluster, giving us k representative pixels from the background.

[2]We chose k as the number of objects in the image.

Method	AP_{25}	AP_{50}	AP_{75}
L-Net + Blobs	08.4	01.2	00.1
L-Net + Best proposal	42.9	33.4	19.1
L-Net + Oracle proposal	57.3	45.1	37.2
L-Net + GT-Mask	61.2	61.2	61.2
PRM + E-Net	43.0	32.0	19.0
GT-points + E-Net	63.1	47.0	26.3
WISE (L-Net + E-Net)	53.5	43.0	25.9

Table 1. **Ablation Studies**. A benchmark illustrating the contribution of each WISE's component on PASCAL VOC 2012.

Method	Annotation	AP_{25}	AP_{50}	AP_{75}
Mask R-CNN [57]	per-pixel	17.1	11.2	03.4
SPN [57]	image-level	26.0	13.0	04.0
PRM [56]	image-level	44.0	27.0	09.0
Cholakkal *et al.* [8]	image-level	**48.5**	30.2	14.4
PRM + E-Net (Ours)	image-level	43.0	32.0	19.0
WISE (Ours)	point-level	47.5	**38.1**	**23.5**

Table 2. **PASCAL VOC 2012 with a fixed annotation budget.** Comparison across methods with the same annotation budget.

4. Experiments

We evaluate the WISE network on a wide variety of datasets: PASCAL VOC [12], COCO [31], CityScapes [9], and KITTI [15] datasets. We compare our results against fully-supervised, and weakly-supervised methods. We compare WISE against several baselines to showcase the efficacy of each of its components. We also fix the annotation budget for acquiring per-pixel, point-level, and image-level labels and compare several models based on the type of label they require. Unless otherwise specified, the performance is measured using average precision (AP) as in [18], computed with Intersection-over-Union (IoU) thresholds of 0.25, 0.5, and 0.75.

4.1. Methods and Baselines

We include the following methods in our benchmarks:

L-Net + Blobs: use the raw output of L-Net (see Figure 4) (which is a predicted blob per object in the scene) as mask prediction.

L-Net + Best proposal: replace each object location predicted by L-Net with the SharpMask's proposal that has the highest "objectness" score.

L-Net + Oracle proposal: replace each object location predicted by L-Net with the SharpMask's proposal that achieves the highest evaluation score (*e.g.* mAP).

L-Net + GT-Mask: replace each object location predicted by L-Net with the ground-truth mask.

PRM + E-Net: use the object locations predicted by PRM (as described in [56]) as input to E-Net to obtain the object masks. Note that PRM only requires image-level labels.

GT-points + E-Net: use the ground-truth object locations (point-level annotations) as input to E-Net to obtain the object masks.

WISE (L-Net + E-Net): use L-Net's predicted object locations as input to E-Net to obtain the object masks.

4.2. Implementation Details

L-Net and E-Net share the same backbone, a ResNet-50 [18] pretrained on ImageNet [10]. They also have independent upsampling paths with similar architecture as FCN8 [33]. The number of output channels for L-Net is the number of classes, and for E-Net is $d = 64$, the size of a pixel's embedding vector. We observed minor differences in the results between different embedding dimensions. For each image, we use 1000 pretrained SharpMask [39] proposals (note that we do not finetune the proposal on any dataset). During training, for each point-annotation we sample a proposal non-uniformly based on its "objectness" score to represent its pseudo-mask. We set k as the number of predicted objects (by L-Net) for selecting the background pixels at test time. The model is trained using Adam [21] optimizer with a learning rate of 10^{-5} and a weight decay of 0.0005 for $200k$ iterations with a batch size of 1. We choose $\lambda = 0.1$ in Equation 4 in order to make the scale between its two loss terms similar.

4.3. Experiments on PASCAL VOC 2012

PASCAL VOC 2012 [12] contains $1,464$ and $1,449$ images for training and validation respectively, where objects come from 20 categories. We use the point-level annotations provided by Bearman *et al.* [4] as ground-truth for training our methods. We report the AP across several thresholds on the validation set, as described in the dataset's instance segmentation setup [12].

4.3.1 Comparison to methods and baselines.

In this section, we discuss the results shown in Table 1. A straightforward method to obtain object masks is to use L-Net's raw output (which we refer to as "L-Net + Blobs"). However, it performs poorly as the predicted blobs are often small around the center of the object.

A natural extension is to replace L-Net's predicted blobs by a segment proposal obtained from an object proposal method. Therefore, we discovered a reasonable strategy which is to replace each of L-Net's predicted blobs by the proposal of highest "objectness" score ("L-Net + Best-proposal"). However, "L-Net + Oracle" shows that a perfect proposal selection strategy can vastly improve on the segmentation results.

Accordingly, we propose WISE which improves on "L-Net + Best-proposal" by having E-Net that learns rough seg-

Method	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	Avg.
SDS [16]	58.8	0.5	60.1	34.4	29.5	60.6	40.0	73.6	6.5	52.4	31.7	62.0	49.1	45.6	47.9	22.6	43.5	26.9	66.2	66.1	43.8
Chen *et al.* [7]	63.6	0.3	61.5	43.9	33.8	67.3	46.9	74.4	8.6	52.3	31.3	63.5	48.8	47.9	48.3	26.3	40.1	33.5	66.7	67.8	46.3
PFN [29]	76.4	15.6	74.2	54.1	26.3	73.8	31.4	92.1	17.4	73.7	48.1	82.2	81.7	72.0	48.4	23.7	57.7	64.4	88.9	72.3	58.7
R2-IOS [28]	87.0	6.1	90.3	67.9	48.4	86.2	68.3	90.3	24.5	84.2	29.6	91.0	71.2	79.9	60.4	42.4	67.4	61.7	94.3	82.1	66.7
Fathi *et al.* [13]	69.7	1.2	78.2	53.8	42.2	80.1	57.4	88.8	16.0	73.2	57.9	88.4	78.9	80.0	68.0	28.0	61.5	61.3	87.5	70.4	62.1
WISE (Ours)	59.0	5.6	63.6	41.4	21.9	40.6	34.1	73.8	8.5	38.7	29.1	64.6	58.1	60.4	33.3	25.1	43.8	32.7	64.7	60.7	43.0

Table 3. **Comparison to fully supervised methods.** Per-class comparison against the AP_{50} metric on PASCAL VOC 2012.

Model	COCO 2014			KITTI			CityScapes		
	AP_{25}	AP_{50}	AP_{75}	AP_{25}	AP_{50}	AP_{75}	AP_{25}	AP_{50}	AP_{75}
L-Net Best proposal	18.3	13.6	7.3	46.4	38.1	22.2	27.2	15.5	6.7
WISE (Ours)	25.8	17.6	7.8	63.4	49.8	30.9	28.7	18.2	8.8

Table 4. **Baseline comparisons.** Results across different average precision IoU thresholds.

mentation of the objects. This allows to select better proposals by choosing those with the highest IoU. Note that other object proposal selection strategies have been used in other weakly supervised instance segmentation setups [56, 8].

To assess how much improvement we can make over L-Net, we report the results of "GT-points + E-Net" which uses the ground-truth points instead of L-Net's predictions. We see that L-Net's performance is close to its upper-bound. Further, we provide the results of "PRM + E-Net" which is an extension to WISE that can train using image-level annotations only. Similarly, we observe that the results are not widely different. However, image-level labels might not be suitable for datasets when the number of objects in an image is dense and when the same object class exist in almost every image as the *car* category in CityScapes.

4.3.2 Comparison to Similar Annotation Time

We compare the performance between state-of-the-art methods in Table 2 when the annotation time is fixed. Therefore, we limit the annotation budget to around 8.13 hours which is calculated as $20.0 \times 1,464$ seconds. Bearman *et al.* [4] has shown that it takes 20.0, 22.1, and 239.7 seconds per image for collecting image-level, point-level, and per-pixel labels, respectively. As a result, for the same annotation time budget, we acquire $1,464$ images with image-level labels, $1,325$ images with point-level labels, and 122 images with per-pixel labels. We selected these images uniformly without replacement from the training set. We also reported the result of Mask R-CNN [35] trained on the images with the per-pixel labels. The table shows that our method significantly outperforms other approaches, suggesting that using point-level annotations is a cost-effective labeling method for instance segmentation. Further, Figure 6 illustrates that WISE can capture high quality masks for PASCAL VOC objects, although it can fail in merging two masks of the same object such as in the horse image.

4.3.3 Comparison to Weakly and Fully Supervised Methods

Acquiring point-level labels is almost as cheap as image-level labels, yet they vastly improve results, as shown in Table 2. For a fair evaluation, we compare "PRM + E-Net" which uses image-level labels against current state-of-the-art image-level instance segmentation methods. The concurrent work of [8] performs better with respect to AP_{25}, which is expected as their counting results is better than LCFCN which is what L-Net is based on.

Further, we report WISE results against fully supervised methods in Table 3 for each category with respect to AP_{50}. While WISE achieves competitive results, there is room for improvement between weakly- and strong- supervised methods.

Model	AP_{50}	AP_{75}
Base-DA [11]	46.0	28.1
Mask-RCNN [17]	55.2	35.3
WISE (Ours)	17.4	07.7

Table 5. **COCO 2014.** Comparison to fully supervised methods.

4.4. Experiments on COCO 2014

For COCO 2014 [31], we train on the union of the 80k train images and the 35k subset of validation images, and report the results on *minival* consisting of 5k images, following the experimental setup of He *et al.* [17]. It consists of 80 categories belonging to a wide variety of everyday objects. We obtain ground-truth points by taking the pixel with the largest distance transform for each instance segmentation mask. We use the standard COCO metrics including AP

(averaged over IoU thresholds), AP_{50}, and AP_{75}. Table 4 shows that WISE outperforms our baseline "L-Net + Best Proposal", which suggests that E-Net generates better proposal masks. The qualitative results in Figure 6 show that WISE can successfully capture the mask of diverse objects. Table 5 shows that while our results are poor compared to fully supervised methods, they establish a first strong baseline for instance segmentation with point-level supervision.

4.5. Experiments on KITTI

KITTI [15] is a meaningful benchmark for autonomous driving. Using the setup described in [54], we train our models on the $3,712$ training images where the ground-truth points are the provided bounding box centers. We reported results on the 120 validation images using the *MUCov* and *MWCov* metrics, as described in Silberman *et al.* [48]. Table 4 shows that WISE significantly outperforms the baseline "L-Net + Best Proposal", suggesting that relying on the best "objectness" score for picking the proposal is not the optimal approach. Furthermore, Table 6 shows that WISE achieves competitive results compared to methods that use full supervision. Figure 6 shows quality masks being generated for the cars and persons objects on KITTI images by WISE.

Model	MWCov	MUCov
DepthOrder [55]	70.9	52.2
DenseCRF [54]	74.1	55.2
AngleFCN+Depth [52]	79.7	75.8
Recurrent+attention [43]	80.0	66.9
WISE (Ours)	74.2	58.9

Table 6. **KITTI.** Comparison to fully supervised methods.

4.6. Experiments on CityScapes

CityScapes [9] is a popular autonomous driving benchmark for instance segmentation. It contains $2,975$ high-resolution training images, and 500 validation images that represent street scenes acquired from an on-board camera. The pixels are labeled into 19 classes, but only 8 classes belong to countable objects (used for instance segmentation): person, rider, car, truck, bus, train, motorcycle, and bicycle. The ground-truth point for each object is the pixel with the largest distance transform within its corresponding ground-truth segmentation mask.

Table 4 shows that WISE sets a new strong baseline for the weakly supervised setting, while achieving better results than the comparable baseline "L-Net + Best proposal". Further, Figure 6 illustrates that our method can obtain good masks for various objects of interest. However, fully supervised methods shown in Table 7 outperform our weakly supervised method with a large margin, inspiring future research on this problem setup.

In Table 8, we compare "GT-points + E-Net" against the methods proposed by Remez *et al.* [42] which use bounding box ground-truth labels at test time. Using their evaluation setup, we report the results in Table 8 which shows better results across four categories. This is despite E-Net using weaker labels than Cut & Paste. According to Bearman *et al.* [4], it takes an average of 10.2 seconds to acquire a bounding box, but only 2.4 seconds to get an annotation for a single object instance.

Method	AP
InstanceCut [22]	15.8
DWT [3]	19.8
SGN [32]	29.2
Mask-RCNN [17]	31.5
WISE (Ours)	07.8

Table 7. **CityScapes.** Comparison to fully supervised methods.

Method	Car	Person	T. light	T. sign
Box [42]	62.0	49.0	76.0	76.9
Simple Does it [20]	68.0	53.0	60.0	51.0
GrabCut [46]	62.0	50.0	64.0	65.0
Cut & Paste [42]	67.0	54.0	77.0	79.0
Fully Supervised [42]	80.0	61.0	79.0	81.0
GT-points + E-Net (Ours)	77.6	55.4	77.8	80.1

Table 8. **CityScapes.** Methods with bounding boxes at test time.

5. Conclusion

In this paper, we have introduced a weakly supervised instance segmentation network (WISE). It can train by using point-level annotations and by leveraging pseudo-masks from object proposal methods. WISE uses L-Net to first detect the object locations which are then given as input to E-Net in order to obtain the segmentation masks. E-Net is based on an embedding network that groups pixels in the image-based on their similarity which are then used to select the best matching proposal mask. We have validated our method across a wide variety of datasets. The results show that WISE obtains competitive results against fully supervised methods and outperform weakly-supervised methods with a fixed annotation cost. The results also provide a strong first baseline for instance segmentation with point-level supervision. Although a pretrained proposal method was used in this problem setup, it was not finetuned on any of our datasets. However, an interesting future direction is to address this task with a more challenging setup that requires proposal-free methods.

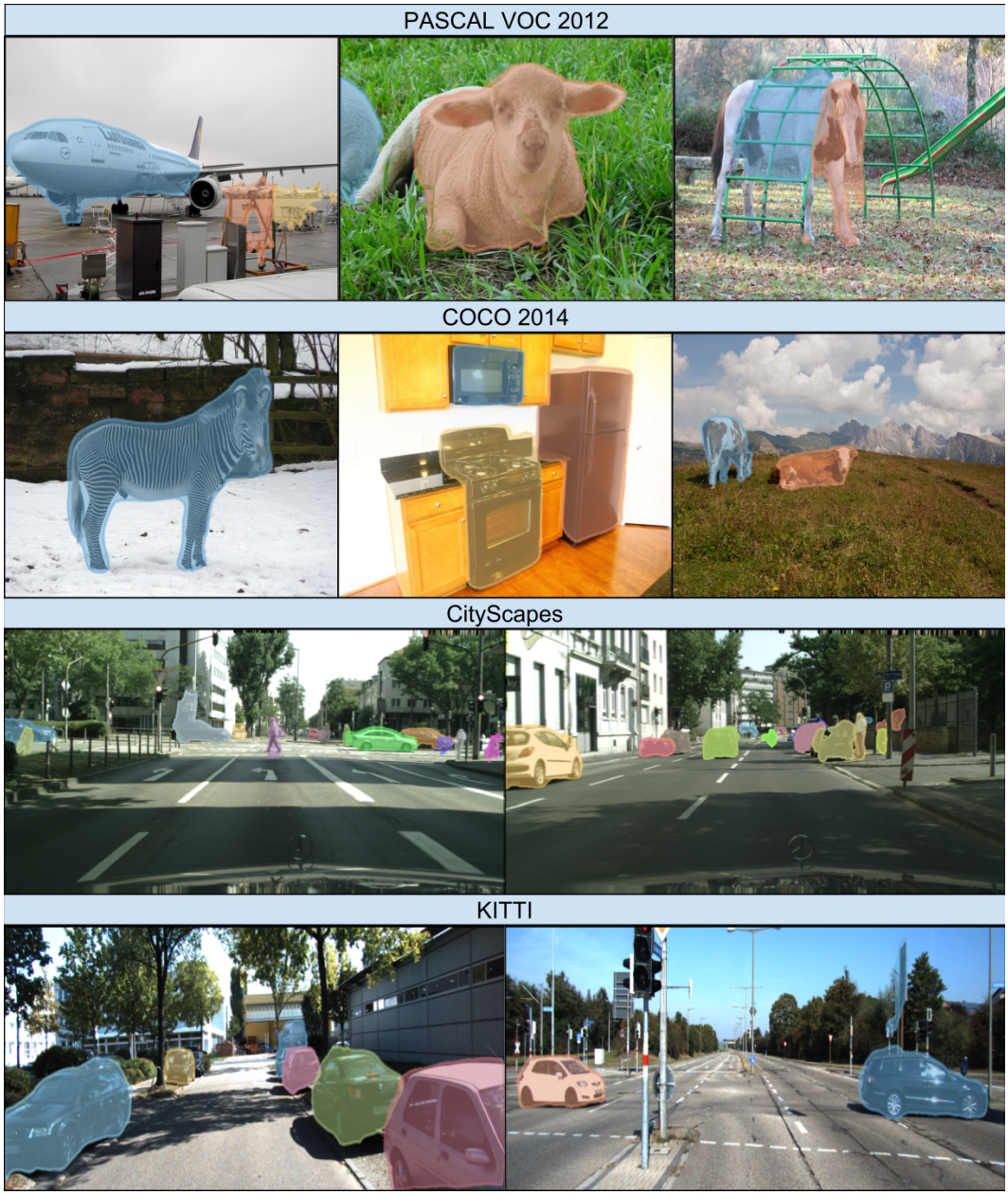

Figure 6. **Qualitative results.** Qualitative results of WISE on the four datasets evaluated.

References

[1] J. Ahn and S. Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. *CVPR*, 2018.

[2] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *CVPR*, 2014.

[3] M. Bai and R. Urtasun. Deep watershed transform for instance segmentation. In *CVPR*, 2017.

[4] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei. Whats the point: Semantic segmentation with point supervision. In *ECCV*, 2016.

[5] H. Bilen and A. Vedaldi. Weakly supervised deep detection networks. In *CVPR*, 2016.

[6] L.-C. Chen, A. Hermans, G. Papandreou, F. Schroff, P. Wang, and H. Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *CVPR*, 2018.

[7] Y.-T. Chen, X. Liu, and M.-H. Yang. Multi-instance object segmentation with occlusion handling. In *CVPR*, 2015.

[8] H. Cholakkal, G. Sun, F. S. Khan, and L. Shao. Object counting and instance segmentation with image-level supervision, 2019.

[9] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[11] N. Dvornik, J. Mairal, and C. Schmid. On the importance of visual context for data augmentation in scene understanding. *arXiv preprint arXiv:1809.02492*, 2018.

[12] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010.

[13] A. Fathi, Z. Wojna, V. Rathod, P. Wang, H. O. Song, S. Guadarrama, and K. P. Murphy. Semantic instance segmentation via deep metric learning. *arXiv preprint arXiv:1703.10277*, 2017.

[14] C.-Y. Fu, M. Shvets, and A. C. Berg. RetinaMask: Learning to predict masks improves state-of-the-art single-shot detection for free. In *arXiv preprint arXiv:1901.03353*, 2019.

[15] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012.

[16] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *ECCV*, 2014.

[17] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *ICCV*, 2017.

[18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[19] J. Hosang, R. Benenson, P. Dollár, and B. Schiele. What makes for effective detection proposals? *T-PAMI*, 2016.

[20] A. Khoreva, R. Benenson, J. H. Hosang, M. Hein, and B. Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *CVPR*, 2017.

[21] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[22] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. Instancecut: from edges to instances with multicut. In *CVPR*, 2017.

[23] A. Kolesnikov and C. H. Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *ECCV*, 2016.

[24] S. Kong and C. Fowlkes. Recurrent pixel embedding for instance grouping. In *CVPR*, 2018.

[25] I. H. Laradji, N. Rostamzadeh, P. O. Pinheiro, D. Vazquez, and M. Schmidt. Where are the blobs: Counting by localization with point supervision. In *ECCV*, 2018.

[26] V. Lempitsky and A. Zisserman. Learning to count objects in images. In *NIPS*, 2010.

[27] Y. Li, X. Zhang, and D. Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *CVPR*, 2018.

[28] X. Liang, Y. Wei, X. Shen, Z. Jie, J. Feng, L. Lin, and S. Yan. Reversible recursive instance-level object segmentation. In *CVPR*, 2016.

[29] X. Liang, Y. Wei, X. Shen, J. Yang, L. Lin, and S. Yan. Proposal-free network for instance-level object segmentation. *arXiv preprint arXiv:1509.02636*, 2015.

[30] D. Lin, J. Dai, J. Jia, K. He, and J. Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, 2016.

[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[32] S. Liu, J. Jia, S. Fidler, and R. Urtasun. Sgn: Sequential grouping networks for instance segmentation. In *ICCV*, 2017.

[33] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[34] K.-K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool. Convolutional oriented boundaries. In *ECCV*, 2016.

[35] F. Massa and R. Girshick. maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch, 2018.

[36] A. Newell, Z. Huang, and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *NIPS*, 2017.

[37] P. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015.

[38] P. O. Pinheiro, R. Collobert, and P. Dollár. Learning to segment object candidates. In *NIPS*, 2015.

[39] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In *ECCV*, 2016.

[40] R. Pohle and K. D. Toennies. Segmentation of medical images using adaptive region growing. In *MIIP*, 2001.

[41] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.

[42] T. Remez, J. Huang, and M. Brown. Learning to segment via cut-and-paste. In *ECCV*, 2018.

[43] M. Ren and R. S. Zemel. End-to-end instance segmentation with recurrent attention. In *CVPR*, 2017.

[44] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[45] B. Romera-Paredes and P. H. S. Torr. Recurrent instance segmentation. In *ECCV*, 2016.

[46] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *SIGGRAPH*, 2004.

[47] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017.

[48] N. Silberman, D. Sontag, and R. Fergus. Instance segmentation of indoor scenes using a coverage loss. In *ECCV*, 2014.

[49] H. O. Song, R. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darrell. On learning to localize objects with minimal supervision. *arXiv preprint arXiv:1403.1024*, 2014.

[50] H. O. Song, Y. J. Lee, S. Jegelka, and T. Darrell. Weakly-supervised discovery of visual pattern configurations. In *NIPS*, 2014.

[51] P. Tang, X. Wang, A. Wang, Y. Yan, W. Liu, J. Huang, and A. Yuille. Weakly supervised region proposal network and object detection. In *ECCV*, 2018.

[52] J. Uhrig, M. Cordts, U. Franke, and T. Brox. Pixel-level encoding and depth layering for instance-level semantic labeling. In *GCPR*, 2016.

[53] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. In *ICCV*, 2013.

[54] Z. Zhang, S. Fidler, and R. Urtasun. Instance-level segmentation for autonomous driving with deep densely connected mrfs. In *CVPR*, 2016.

[55] Z. Zhang, A. G. Schwing, S. Fidler, and R. Urtasun. Monocular object instance segmentation and depth ordering with cnns. In *CVPR*, 2015.

[56] Y. Zhou, Y. Zhu, Q. Ye, Q. Qiu, and J. Jiao. Weakly supervised instance segmentation using class peak response. In *CVPR*, 2018.

[57] Y. Zhu, Y. Zhou, Q. Ye, Q. Qiu, and J. Jiao. Soft proposal networks for weakly supervised object localization. In *ICCV*, 2017.

[58] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.