BoxTeacher: Exploring High-Quality Pseudo Labels for Weakly Supervised Instance Segmentation

Tianheng Cheng\textsuperscript{1,*}, Xinggang Wang\textsuperscript{1}, Shaoyu Chen\textsuperscript{1,*}, Qian Zhang\textsuperscript{2}, Wenyu Liu\textsuperscript{1,†}
\textsuperscript{1} School of EIC, Huazhong University of Science & Technology
\textsuperscript{2} Horizon Robotics

https://github.com/hustvl/BoxTeacher

Abstract

Labeling objects with pixel-wise segmentation requires a huge amount of human labor compared to bounding boxes. Most existing methods for weakly supervised instance segmentation focus on designing heuristic losses with priors from bounding boxes. While, we find that box-supervised methods can produce some fine segmentation masks and we wonder whether the detectors could learn from these fine masks while ignoring low-quality masks. To answer this question, we present BoxTeacher, an efficient and end-to-end training framework for high-performance weakly supervised instance segmentation, which leverages a sophisticated teacher to generate high-quality masks as pseudo labels. Considering the massive noisy masks hurt the training, we present a mask-aware confidence score to estimate the quality of pseudo masks, and propose the noise-aware pixel loss and noise-reduced affinity loss to adaptively optimize the student with pseudo masks. Extensive experiments can demonstrate effectiveness of the proposed BoxTeacher. Without bells and whistles, BoxTeacher remarkably achieves \textsuperscript{35.0} mask AP and \textsuperscript{36.5} mask AP with ResNet-50 and ResNet-101 respectively on the challenging COCO dataset, which outperforms the previous state-of-the-art methods by a significant margin and bridges the gap between box-supervised and mask-supervised methods.

1. Introduction

Instance segmentation, aiming at recognizing and segmenting objects in images, is a fairly challenging task in computer vision. Fortunately, the rapid development of object detection methods\cite{7,40,50} has greatly advanced the emergence of numbers of successful methods\cite{5,6,23,49,54,55} for effective and efficient instance segmenta-

\textsuperscript{*} This work was done when Tianheng Cheng and Shaoyu Chen were interns at Horizon Robotics. \textsuperscript{†} Wenyu Liu is the corresponding author: liuwy@hust.edu.cn

Figure 1. (a) Segmentation Masks from BoxInst. BoxInst (ResNet-50\cite{24}) can produce some fine segmentation masks with weak supervisions from bounding boxes and images. (b) Self-Training with Pseudo Masks on COCO \text_\textsuperscript{val}. We explore the self-training to train a CondInst\cite{49} with the pseudo labels generated by BoxInst. However, the improvements are limited.

- With the fine-grained human annotations, recent instance segmentation methods can achieve impressive results on challenging the COCO dataset\cite{34}. Nevertheless, labeling instance-level segmentation is much complicated and time-consuming, e.g., labeling an object with polygon-based masks requires \textsuperscript{10.3×} more time than that with a 4-point bounding box\cite{11}.

- Recently, a few works\cite{25,31–33,51,53} explore weakly supervised instance segmentation with box annotations or low-level colors. These weakly supervised methods can effectively train instance segmentation methods\cite{23,49,55} without pixel-wise or polygon-based annotations and obtain fine segmentation masks. As shown in Fig. 1(a), BoxInst\cite{51} can output a few high-quality segmentation masks and segment well on the object boundary, e.g., the person, even performs better than the ground-truth mask in details though other objects may be badly segmented. Naturally, we wonder if the generated masks of box-supervised methods, especially the high-quality masks, could be qualified as pseudo segmentation labels to further improve the performance of weakly supervised instance segmentation.

To answer this question, we first employ the naive
self-supervised training to evaluate the performance of using box-supervised pseudo masks. Given the generated instance masks from BoxInst, we propose a simple yet effective box-based pseudo mask assignment to assign pseudo masks to ground-truth boxes. And then we train the CondInst [49] with the pseudo masks, which has the same architecture with BoxInst and consists of a detector [50] and a dynamic mask head. Fig. 1(b) shows that using self-training brings minor improvements and fails to unleash the power of high-quality pseudo masks, which can be attributed to two obstacles, i.e., (1) the naive self-supervision fails to filter low-quality masks, and (2) the noisy pseudo masks hurt the training using fully-supervised pixel-wise loss. Besides, the multi-stage self-supervision is inefficient.

To address these problems, we present BoxTeacher, an end-to-end training framework, which takes advantage of high-quality pseudo masks produced by box supervision. BoxTeacher is composed of a sophisticated Teacher and a perturbed Student, in which the teacher generates high-quality pseudo instance masks along with the mask-aware confidence scores to estimate the quality of masks. Then the proposed box-based pseudo mask assignment will assign the pseudo masks to the ground-truth boxes. The student is normally optimized with the ground-truth boxes and pseudo masks through box-based loss and noise-aware pseudo mask loss, and then progressively updates the teacher via Exponential Moving Average (EMA). In contrast to the naive multi-stage self-supervision, BoxTeacher is more simple and efficient. The proposed mask-aware confidence score effectively reduces the impact of low-quality masks. More importantly, pseudo labeling can mutually improve the student and further enforce the teacher to generate higher-quality masks, hence pushing the limits of the box supervision. BoxTeacher can serve as a general training paradigm and is agnostic to the methods for instance segmentation.

To benchmark the proposed BoxTeacher, we adopt CondInst [49] as the basic segmentation method. On the challenging COCO dataset [34], BoxTeacher surprisingly achieves 35.0 and 36.5 mask AP based on ResNet-50 [24] and ResNet-101 respectively, which remarkably outperforms the counterparts. We provide extensive experiments on PASCAL VOC and Cityscapes to demonstrate its effectiveness and generalization ability. Furthermore, BoxTeacher with Swin Transformer [37] obtains 40.6 mask AP as a weakly approach for instance segmentation. Overall, the contribution can be summarized as follows:

- We improve the weakly supervised instance segmentation by large margins and bridge the gap between box-supervised and mask-supervised methods, e.g., BoxTeacher achieves 36.5 mask AP on COCO compared to 39.1 AP obtained by CondInst.

### 2. Related Work

#### Instance Segmentation

Methods for instance segmentation can be roughly divided into two groups, i.e., single-stage methods and two-stage methods. Single-stage methods [5, 49, 58, 62] tend to adopt single-stage object detectors [35, 50], to localize and recognize objects, and then generate segmentation masks through object embeddings or dynamic convolution [9]. Wang et al. present box-free SOLO [54] and SOLOv2 [55], which are independent of object detectors. SparseInst [13] and YOLACT [5], aiming for real-time inference, achieve great trade-off between speed and accuracy. Two-stage methods [14, 23, 27, 29] adopt bounding boxes from object detectors and RoIAlign [23] to extract the RoI (region-of-interest) features for object segmentation, e.g., Mask R-CNN [23]. Several methods [14, 27, 29] based on Mask R-CNN are proposed to refine the segmentation masks for high-quality instance segmentation. Recently, many approaches [7, 10, 12, 17, 20, 63] based on transformers [18, 52] or the Hungarian algorithm [46] have made great progress in instance segmentation.

#### Weakly Supervised Instance Segmentation

Considering the huge cost of labeling instance segmentation, weakly supervised instance segmentation using image-level labels or bounding boxes gets lots of attention. Several methods [1, 2, 64, 66] exploit image-level labels to generate pseudo masks from activation maps. Khoreva et.al. [28] propose to generate pseudo masks with GrabCut [42] from given bounding boxes. BoxCaseg [53] leverages a saliency model to generate pseudo object masks for training Mask R-CNN along with the multiple instance learning (MIL) loss. Recently, many box-supervised methods [25, 31, 33, 51] combine the MIL loss or pairwise relation loss from low-level features obtain impressing results with box annotations. In comparison with BoxInst [51], BoxTeacher inherits the box supervision [51] but concentrates more on the novel training paradigm and exploiting noisy pseudo masks for high-performance box-supervised instance segmentation with box annotations. Different from DiscoBox [31] based on mean teacher [48], BoxTeacher aims at a simple yet effective training framework with obtaining high-quality pseudo masks and learning from noisy masks.

#### Semi-supervised Learning

Pseudo labeling [3, 21, 41] and consistency regularization [4, 30, 43, 44, 59] have greatly
advanced the semi-supervised learning, which enables the training on large-scale unlabeled datasets. Recently, semi-supervised learning has been widely used in object detection [36, 45, 60] and semantic segmentation [8, 56, 61] and demonstrated its effectiveness. Motivated by high-quality masks from box supervision, we adopt the successful pseudo labeling and consistency regularization to develop a new training framework for weakly supervised instance segmentation. Compared to [22] which has similar motivation but aims for semi-supervised object detection with labeled images and extra point annotations, BoxTeacher addresses box-supervised instance segmentation with box-only annotations. Compared to [26, 47] which adopt multi-stage training and combine weakly supervised and semi-supervised learning, BoxTeacher is a one-stage framework without pre-trained labelers.

3. Naive Self-Training with Pseudo Masks

Revisiting Box-supervised Methods. Note that box-only annotations is sufficient to train an object detector, which can accurately localize and recognize objects. Box-supervised methods [31, 33, 51] based on object detectors mainly exploit two exquisite losses to supervise mask predictions, i.e., the multiple instance learning (MIL) loss and the pairwise relation loss. Concretely, according to the bounding boxes, the MIL loss can determine the positive and negative bags of pixels of the predicted masks. Pairwise relation loss concentrates on the local relations of pixels from low-level colors or features, in which neighboring pixels have the similar color will be regarded as a positive pair and should output similar probabilities. The MIL loss and pairwise relation loss enables the box-supervised methods to produce the complete segmentation masks, and even some high-quality masks with fine details.

Naive Self-Training. Considering that the box-supervised methods can produce some high-quality masks without mask annotations, we adopt self-training to utilize the high-quality masks as pseudo labels to train an instance segmentation method with full supervision. Specifically, we adopt the successful BoxInst [51] to generate pseudo instance masks on the given dataset \( \mathcal{X} = \{X, B^p\} \), which only contains the box annotations. For each input image \( X \), let \( \{B^p, C^p, M^p\} \) denote the predicted bounding boxes, confidence scores, and predicted instance masks, respectively. We propose a simple yet effective **Box-based Pseudo Mask Assignment** algorithm in Alg. 1 to assign the predicted instance masks to the box annotations via the confidence scores and intersection-over-union (IoU) between ground-truth boxes \( B^\theta \) and predicted boxes \( B^p \). The hyper-parameters \( \tau_{io} \) and \( \tau_c \) are set to 0.5 and 0.05, respectively. The assigned instance masks will be rectified by removing the parts beyond the bounding boxes. Then, we adopt the dataset \( \hat{X} = \{X, B^\theta, M^\theta\} \) with pseudo instance masks to train an approach, e.g., CondInst [49].

**Naive Self-Training is Limited.** Fig. 1(b) and Tab. 7 provide the experimental results of using naive self-training pseudo masks. Compared to the pseudo labeler, using self-training brings minor improvements and even fails to surpass the pseudo labeler. We attribute the limited performance to two issues, i.e., the naive self-training fails to exclude low-quality masks and the fully-supervised loss is sensitive to the noisy pseudo masks.

**Algorithm 1: Box-based Pseudo Mask Assignment**

**Input:** predicted boxes \( B^p \in \mathbb{R}^{N \times 4} \), predicted masks \( M^p \in \mathbb{R}^{N \times H \times W} \), confidence score \( C^p \in \mathbb{R}^{N} \), ground-truth boxes \( B^\theta \in \mathbb{R}^{K \times 4} \).

**Parameter:** IoU threshold \( \tau_{io} \), confidence threshold \( \tau_c \).

**Output:** assigned pseudo masks \( M^\theta \in \mathbb{R}^{K \times H \times W} \).

1. Initialize output masks \( M^\theta \) with empty (0), assignment index \( A \in \mathbb{R}^{K} \) with -1;
2. Filter the predictions by the confidence threshold \( \tau_c \);
3. Sort the confidence score \( C^p \) in descending order with output indices \( S \in \mathbb{N}^N \);
4. foreach prediction \( i \in S \) do
5. Initialize \( u \leftarrow -1, v \leftarrow -1 \);
6. for \( j = 1 \) to \( K \) do
7. \( \text{iou}_{ij} = \text{ComputeIoU}(B^p_i, B^\theta_j) \);
8. if \( \text{A}_j > 0 \) then
9. continue;
10. end
11. if \( \text{iou}_{ij} \geq \tau_{io} \) and \( \text{iou}_{ij} \geq u \) then
12. \( u \leftarrow \text{iou}_{ij}, v \leftarrow i; \)
13. end
14. if \( v > 0 \) then
15. Assign mask \( M^p_i \) to mask \( M^\theta_v \);
16. \( A_j \leftarrow i; \)
17. end
18. end
19. end

4. BoxTeacher

In this section, we present BoxTeacher, an end-to-end training framework, which aims to unleash the power of pseudo masks. In contrast to multi-stage self-training, BoxTeacher, consisting of a teacher and a student, simultaneously facilitates the training of the student and pseudo labeling of the teacher. The mutual optimization is beneficial to both the teacher and the student, thus leading to higher performance for box-supervised instance segmentation.
4.1. Architecture

The overall architecture of BoxTeacher is depicted in Fig. 2. BoxTeacher is composed of a teacher and a student, which shares the same model. Given the input image, the teacher \( f_\theta \) straightforwardly generates the predictions, including the bounding boxes, segmentation masks, and mask-aware confidence scores. Similarly, we apply the box-based pseudo mask assignment in Alg. 1 to assign the predicted masks to the ground-truth annotations. Inspired by consistency regularization [36, 44, 45, 59], we adopt strong augmentation for images (e.g., color jittering) fed into the student \( f_\xi \) and the student is optimized under the box supervision and the mask supervision. To acquire high-quality pseudo masks, we adopt the exponential moving average to gradually update the teacher from student [48], i.e.,

\[
 f_\theta \leftarrow \alpha \cdot f_\theta + (1 - \alpha) \cdot f_\xi
\]

(\( \alpha \) is empirically set 0.999).

Mask-aware Confidence Score. Considering that the generated pseudo masks are noisy and unreliable, which may hurt the performance, we define the mask-aware confidence score to estimate the quality of the pseudo masks. Inspired by [54], we denote \( m_i^p \in \mathbb{R}^{H \times W} \) and \( m_i \in \mathbb{R}^{H \times W} \) as the box-based binary masks and sigmoid probabilities of the \( i \)-th pseudo mask with the detection confidence \( c_i \), the mask-aware confidence score \( s_i \) is defined as follows:

\[
 s_i = \sqrt{c_i \cdot \frac{\sum_{x,y} I(m_{i,x,y} > \tau_m) \cdot m_{i,x,y} \cdot m_i^p}{\sum_{x,y} I(m_{i,x,y} > \tau_m) \cdot m_{i,x,y}^p}},
\]

(1)

where \( I(\cdot) \) is the indicator function, \( \tau_m \) is the threshold for binary masks and set to 0.5. The mask-aware score calculates the average probability score of the positive masks inside the ground-truth boxes, and the higher average probability means more confident pixels in the mask. In addition, we explore several kinds of quality scores and compare with the mask-aware score in experiments.

Training Loss. BoxTeacher can be end-to-end optimized with box annotations and the generated pseudo masks. The overall loss is defined as:

\[
 L = L_{\text{det}} + L_{\text{box-sup}} + L_{\text{mask-sup}},
\]

where \( L_{\text{det}} \) is the detection loss, \( L_{\text{box-sup}} \) is the box-supervised loss, and \( L_{\text{mask-sup}} \) is the mask-supervised loss. We inherit the detection loss defined in FCOS [49], and we follow previous works [25, 31, 51] to adopt the max-projection loss and the color-based pairwise relation loss [51] for box-supervised mask loss \( L_{\text{box-sup}} \).

4.2. Noise-aware Pseudo Mask Loss

The goal of BoxTeacher is to take advantage of high-quality pseudo masks in a fully supervised manner while reduce the impact of the noisy or low-quality instance masks. To this end, we present the noise-aware mask loss in Eq. (2). Ideally, BoxTeacher can leverage the pseudo masks to calculate the fully-supervised pixel-wise segmentation loss, e.g., dice loss [38]. Besides, we also propose a novel noise-reduced mask affinity loss \( L_{\text{affinity}} \) to enhance the pixel-wise segmentation with neighboring pixels. Further, we employ the proposed mask-aware confidence scores \( \{s_i\} \) as weights for the pseudo mask loss, which adaptively scales the weights for pseudo masks of different qualities. The total pseudo mask loss is defined as follows:

\[
 L_{\text{mask-sup}} = \frac{1}{N_p} \sum_{i=1}^{N_p} s_i \cdot \lambda_{\text{pixel}} (m_i^p, m_i^q) + \lambda_{\text{affinity}} (m_i^p, m_i^q),
\]

(2)

where \( m_i^p \) and \( m_i^q \) denotes the \( i \)-th predicted masks and pseudo masks, \( N_p \) denotes the number of valid pseudo masks, \( \lambda_{\text{pixel}} \) and \( \lambda_{\text{affinity}} \) are set to 0.5 and 0.1 respectively. To stabilize the training, we adopt a linear warmup strategy for pseudo mask loss at the beginning of the training, i.e., the first 10k iterations.
Noise-reduced Mask Affinity Loss. considering pseudo masks contain much noise while neighboring pixels in local regions (e.g., $3 \times 3$) tend to have similar semantics or labels, we exploit label affinities among pixels within local regions to alleviate label noise. Given the $i$-th pixel sigmoid probability $g_i$ of the pseudo segmentation, we first calculate the refined pixel probability $\tilde{g}_i$ with its neighboring pixels, which is defined as follows:

$$\tilde{g}_i = \frac{1}{2} (g_i + \frac{1}{|\mathcal{P}|} \sum_{j \in \mathcal{P}} g_j),$$

where $\mathcal{P}$ denotes the set of neighboring pixels, e.g., a $3 \times 3$ region. This refinement can reduce the outliers and enhance the pixels with local context. Then, We present a simple noise-reduced mask affinity loss and define the affinity $\mu_{ij}$ between $i$-th and $j$-th pixels as follows:

$$\mu_{ij} = \tilde{g}_i \cdot \tilde{g}_j + (1 - \tilde{g}_i) \cdot (1 - \tilde{g}_j),$$

where $\tilde{g}_i$ and $\tilde{g}_j$ are refined pixels which encode the local context. Then the noise-reduced mask affinity loss for $i$-th pixel is defined as follows:

$$L_{affinity} = -\frac{1}{|\mathcal{P}|} \sum_{j \in \mathcal{P}} \sum_{\mu_{ij} > \tau_u} (\log(p_i \cdot p_j) + \log((1-p_i) \cdot (1-p_j))),$$

where $j \in \mathcal{P}$ are the neighboring pixels of the $i$-th pixel and $\tau_u$ is set to 0.5 as default. $p_i$ denotes the $i$-th pixel of the predicted mask.

5. Experiments

In this section, we mainly evaluate the proposed BoxTeacher on the COCO dataset [34], the Cityscapes dataset [15], and the PASCAL VOC 2012 dataset [19], and provide extensive ablations to analyze the proposed method.

Datasets. The COCO dataset contains 80 categories and 110k images for training, 5k images for validation, and 20k images for testing. The Cityscapes dataset, aiming for perception in driving scenes, consists of 5000 street-view high-resolution images, in which 2975, 500, and 1525 images are used for training, validation, and testing, respectively. Foreground objects in Cityscapes are categorized into 8 classes and fine-annotated with pixel-wise segmentation labels instead of polygons adopted in COCO, thus making the labeling process much costly. The PASCAL VOC 2012 dataset has 20 categories and consists of 10582 images for training and 1449 images for validation. For weakly supervised instance segmentation, we only keep the bounding boxes and ignore the segmentation masks during training.

Implementation Details. The proposed BoxTeacher is implemented based on PyTorch [39] and we mainly adopt CondInst [49] as the meta method for instance segmentation. The backbone networks are initialized with the ImageNet-pretrained weights and the BatchNorm layers are frozen. All BoxTeacher models are trained over 8 GPUs.

Data Augmentation. For images input to the student, we adopt random horizontal flip and the multi-scale augmentation which randomly resizes images from 640 to 800 as the basic augmentation. In addition, we randomly apply color jittering, grayscale, and Gaussian blur for stronger augmentation. While the images fed into the teacher are fixed to $800 \times 1333$ without perturbation.

5.1. Instance Segmentation on COCO

Experimental Setup. Following the training recipes [49–51], BoxTeacher is trained with 16 images per batch. Unless specified, we adopt the standard $1 \times$ schedule (90k iterations) [23, 57] with the SGD and the initial learning rate 0.01. For comparisons with the state-of-art methods, we scale up the learning schedule to $3 \times$ (270k iterations).

Main Results. Tab. 1 shows the main results on COCO test-dev. In comparison with other state-of-the-art methods, we evaluate the proposed BoxTeacher with different backbone networks, i.e., ResNet [24] and Swin Transformer [37], and under different training schedules, i.e., $1 \times$ and $3 \times$. It’s clear that BoxTeacher with ResNet-50 achieves 32.9 mask AP, which outperforms other box-supervised methods [31, 51] even with longer schedules. Compared to recent BoxInst [51], BoxLevelSet [33] and DiscoBox [31], BoxTeacher significantly brings about 3.0 mask AP improvements based on ResNet-50 and ResNet-101 under the same setting. Remarkably, BoxTeacher also bridges the gap between mask-supervised methods and box-supervised methods, e.g., the gap between BoxTeacher based on ResNet-101 and CondInst is reduced to 2.6 AP. With the stronger backbones, e.g., Swin Transformer [37], BoxTeacher can surprisingly obtain 40.6 mask AP on COCO dataset, which is highly competitive as a weakly supervised method for instance segmentation.

5.2. Instance Segmentation on PASCAL VOC

Experimental Setup. BoxTeacher is trained for 15k iterations with 16 images per batch. Following previous works [31–33], we report COCO-style AP and AP under 4 thresholds, i.e., $\{0.25, 0.5, 0.70, 0.75\}$.

Main Results. Tab. 2 shows the comparisons with the state-of-the-art methods on PASCAL VOC 2012. Compared to recent box-supervised methods [31, 33, 51], the proposed BoxTeacher achieves better results under different IoU thresholds, which remarkably outperforms BoxInst and DiscoBox by large margins. Notably, BoxTeacher obtains significant improvements under higher IoU thresholds.
Table 1. Results on COCO Instance Segmentation. Comparisons with state-of-the-art methods on COCO test-dev. With the same backbone or learning schedule, BoxTeacher surprisingly surpasses the counterparts by large margins (more than 2.0 mask AP). †: trained without strong augmentation. ‘R-50’ and ‘R-101’ denote ResNet-50 and ResNet-101, and ‘DCN’ denotes deformable convolution [16,65].

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>Schedule</th>
<th>AP</th>
<th>AP50</th>
<th>AP75</th>
<th>APs</th>
<th>APm</th>
<th>APl</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mask-supervised methods.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mask R-CNN [23]</td>
<td>R-50-FPN</td>
<td>1×</td>
<td>35.5</td>
<td>57.0</td>
<td>37.8</td>
<td>19.5</td>
<td>37.6</td>
<td>46.0</td>
</tr>
<tr>
<td>CondInst [49]</td>
<td>R-50-FPN</td>
<td>1×</td>
<td>35.9</td>
<td>57.0</td>
<td>38.2</td>
<td>19.0</td>
<td>38.6</td>
<td>46.7</td>
</tr>
<tr>
<td>CondInst [49]</td>
<td>R-50-FPN</td>
<td>3×</td>
<td>37.7</td>
<td>58.9</td>
<td>40.3</td>
<td>20.4</td>
<td>40.2</td>
<td>48.9</td>
</tr>
<tr>
<td>CondInst [49]</td>
<td>R-101-FPN</td>
<td>3×</td>
<td>39.1</td>
<td>60.9</td>
<td>42.0</td>
<td>21.5</td>
<td>41.7</td>
<td>50.9</td>
</tr>
<tr>
<td>SOLO [54]</td>
<td>R-101-FPN</td>
<td>6×</td>
<td>37.8</td>
<td>59.5</td>
<td>40.4</td>
<td>16.4</td>
<td>40.6</td>
<td>54.2</td>
</tr>
<tr>
<td>SOLOv2 [54]</td>
<td>R-101-FPN</td>
<td>6×</td>
<td>39.7</td>
<td>60.7</td>
<td>42.9</td>
<td>17.3</td>
<td>42.9</td>
<td>57.4</td>
</tr>
<tr>
<td><strong>Box-supervised methods.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoxInst [51]</td>
<td>R-50-FPN</td>
<td>3×</td>
<td>32.1</td>
<td>55.1</td>
<td>32.4</td>
<td>15.6</td>
<td>34.3</td>
<td>43.5</td>
</tr>
<tr>
<td>DiscoBox [31]</td>
<td>R-50-FPN</td>
<td>3×</td>
<td>32.0</td>
<td>53.6</td>
<td>32.6</td>
<td>11.7</td>
<td>33.7</td>
<td>48.4</td>
</tr>
<tr>
<td>BoxTeacher</td>
<td>R-50-FPN</td>
<td>1×</td>
<td>32.9</td>
<td>54.1</td>
<td>34.2</td>
<td>17.4</td>
<td>36.3</td>
<td>43.7</td>
</tr>
<tr>
<td>BoxTeacher</td>
<td>R-50-FPN</td>
<td>3×</td>
<td>35.0</td>
<td>56.8</td>
<td>36.7</td>
<td>19.0</td>
<td>38.5</td>
<td>45.9</td>
</tr>
</tbody>
</table>

Table 2. Results on PASCAL VOC. Comparisons with the state-of-the-art methods on PASCAL VOC 2012 val. All methods adopt box-only annotations.

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>AP</th>
<th>AP25</th>
<th>AP50</th>
<th>AP70</th>
<th>AP75</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDI [28]</td>
<td>VGG-16</td>
<td>-</td>
<td>-</td>
<td>44.8</td>
<td>-</td>
<td>16.3</td>
</tr>
<tr>
<td>BoxInst [51]</td>
<td>R-50</td>
<td>34.3</td>
<td>59.1</td>
<td>34.2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DiscoBox [31]</td>
<td>R-50</td>
<td>71.4</td>
<td>42.2</td>
<td>35.5</td>
<td>39.3</td>
<td>35.9</td>
</tr>
<tr>
<td>BoxLevelSet [33]</td>
<td>R-50</td>
<td>36.2</td>
<td>76.3</td>
<td>43.9</td>
<td>39.5</td>
<td>35.9</td>
</tr>
<tr>
<td>BoxTeacher</td>
<td>R-50</td>
<td>38.6</td>
<td>77.6</td>
<td>46.1</td>
<td>46.1</td>
<td>38.7</td>
</tr>
<tr>
<td>BBTP [25]</td>
<td>R-101</td>
<td>-</td>
<td>75.0</td>
<td>59.9</td>
<td>30.4</td>
<td>21.6</td>
</tr>
<tr>
<td>Arun et al. [2]</td>
<td>R-101</td>
<td>73.1</td>
<td>57.6</td>
<td>33.5</td>
<td>31.2</td>
<td></td>
</tr>
<tr>
<td>BBAM [32]</td>
<td>R-101</td>
<td>76.8</td>
<td>63.7</td>
<td>39.5</td>
<td>31.8</td>
<td></td>
</tr>
<tr>
<td>BoxInst [51]</td>
<td>R-101</td>
<td>36.4</td>
<td>61.4</td>
<td>-</td>
<td>37.0</td>
<td></td>
</tr>
<tr>
<td>DiscoBox [31]</td>
<td>R-101</td>
<td>72.8</td>
<td>62.2</td>
<td>45.5</td>
<td>37.5</td>
<td></td>
</tr>
<tr>
<td>BoxLevelSet [33]</td>
<td>R-101</td>
<td>38.3</td>
<td>77.9</td>
<td>66.3</td>
<td>46.4</td>
<td>38.7</td>
</tr>
<tr>
<td>BoxTeacher</td>
<td>R-101</td>
<td>40.3</td>
<td>78.4</td>
<td>67.8</td>
<td>48.0</td>
<td>41.3</td>
</tr>
</tbody>
</table>

5.3. Instance Segmentation on Cityscapes

Experimental Setup. Following previous methods [23, 49], we train all models for 24k iterations with 8 images per batch. The initial learning rate is 0.005. Cityscapes contains high-resolution images (2048×1024), and we randomly resize images from 800 to 1024 for the student and keep the original size for the teacher during training. In addition, we also adopt the COCO pre-trained models (1× schedule) to initialize the weights for higher performance.

Main Results. Tab. 3 shows the evaluation results on Cityscapes val. BoxTeacher outperforms previous box-supervised methods significantly, especially with the COCO pre-trained weights. Though performance gap between fully supervised methods and weakly supervised methods become larger than that in COCO, the human labour of labeling pixel-wise segmentation for a high-resolution Cityscapes image is much costly (90 minutes per image). And we hope future research can bridge the gap between box-supervised methods and mask-supervised methods for high-resolution images.

5.4. Ablation Experiments

Effects of Pseudo Mask Loss. In Tab. 4, we explore the different pseudo mask loss for BoxTeacher. Firstly, we apply the box-supervised loss proposed in [51] achieves 30.7 mask AP (gray row). As shown in Tab. 4, directly applying binary cross entropy (bce) loss with pseudo masks leads to severe performance degradation, which can be attributed to the foreground/background imbalance and noise in pseudo masks. Using dice loss to supervise the training with pseudo masks can bring significant improvements in comparison to the baseline. Adding mask affinity loss $L_{affinity}$ provides 0.4
Figure 3. **Visualization Results.** We provide the visualization results of BoxTeacher with ResNet-101 on the COCO test-dev. The proposed BoxTeacher can produce the high-quality segmentation results, even in some complicated scenes.

| Table 3. **Results on Cityscapes Instance Segmentation.** Comparisons with state-of-the-art methods for mask AP on Cityscapes val.†.†: our re-produced results on Cityscapes based on the public code. †: denotes the Cityscapes train with fine annotations while †: denotes using COCO pre-trained weights. For box-supervised methods, we remove the fine-grained mask annotations in Cityscapes.

<table>
<thead>
<tr>
<th>Method</th>
<th>Data</th>
<th>AP</th>
<th>AP(_50)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mask-supervised methods.</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mask R-CNN [23]</td>
<td>fine</td>
<td>31.5</td>
<td>-</td>
</tr>
<tr>
<td>CondInst [49]</td>
<td>fine</td>
<td>33.0</td>
<td>59.3</td>
</tr>
<tr>
<td>CondInst [49]</td>
<td>fine + COCO</td>
<td>37.8</td>
<td>63.4</td>
</tr>
<tr>
<td><strong>Box-supervised methods.</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoxInst† [51]</td>
<td>fine</td>
<td>19.0</td>
<td>41.8</td>
</tr>
<tr>
<td>BoxInst† [51]</td>
<td>fine + COCO</td>
<td>24.0</td>
<td>51.0</td>
</tr>
<tr>
<td>BoxLevelSet† [33]</td>
<td>fine</td>
<td>20.7</td>
<td>43.3</td>
</tr>
<tr>
<td>BoxLevelSet† [33]</td>
<td>fine + COCO</td>
<td>22.7</td>
<td>46.6</td>
</tr>
<tr>
<td>BoxTeacher</td>
<td>fine</td>
<td>21.7</td>
<td>47.5</td>
</tr>
<tr>
<td>BoxTeacher</td>
<td>fine + COCO</td>
<td>26.8</td>
<td>54.2</td>
</tr>
</tbody>
</table>

AP gain based on the dice loss. Moreover, we ablate the loss weights in pseudo mask loss in Tab. 5.

**Effects of Mask-aware Confidence Score.** Tab. 6 explores several different scores to estimate the quality of pseudo masks in an unsupervised manner, i.e., (1) classification scores (cls), (2) matched IoU between predicted boxes and ground-truth boxes (iou), (3) mean entropy of the pixel probabilities of pseudo masks (mean-entropy: \( s = 1 + \frac{1}{HW} \sum_{i,j}^{H,W} (p_{ij} \log p_{ij} + (1-p_{ij}) \log(1-p_{ij})) \)), (4) the proposed mask-aware score (mask-aware). As Tab. 6 shows, using the proposed mask-aware confidence score leads to better performance for BoxTeacher. Notably, measuring the quality of predicted masks is critical but challenging for leverage pseudo masks. Accurate quality estimation can effectively reduce the impact of noisy masks.

**Comparisons with Self-Training Paradigm.** We adopt the box-supervised approach, i.e., BoxInst [51], to generate pseudo masks, which is pre-trained on COCO with box-only annotations. And then we assign the pseudo masks to the ground-truth boxes through the assignment Alg. 1. As shown in Tab. 7, the improvements provided by self-training are much limited and the naive self-training even performs worse than the training with box-only annotations, e.g., CondInst with R-50 and 3× schedule obtains 31.3 AP with pseudo masks, but inferior to the box-supervised version (31.8 AP). Though the self-training scheme enables the supervised training with pseudo masks and achieves comparable performance, we believe the high-quality pseudo masks are not well exploited. Significantly, BoxTeacher achieves higher mask AP compared to both self-training, in an end-to-end manner without complicated steps or procedures for label generation.

**Effects of the Strong Data Augmentation.** In this study, we explore the effect of strong augmentation on the proposed BoxTeacher, and apply augmentation to the input images of the student. Specifically, we defined two levels of data augmentation in A.1 (in Appendix), i.e., strong augmentation and weak augmentation. As Tab. 8 shows, both strong and weak augmentation hurt the performance of CondInst and BoxTeacher under the 1× training schedule. Differently, BoxTeacher is more robust to the augmentations as the mask AP drops 0.4 compared to CondInst. However, BoxTeacher remarkably benefits more from the strong data augmentation when increasing the schedule to 3×. In comparison to CondInst, BoxTeacher with strong augmentation will enforce the consistency between the student and teacher. Interestingly, Tab. 8 indicates that using strong augmentation is merely beneficial to the weakly supervised instance segmentation (+0.6 AP), but has no effect to the fully supervised object detection (+0.1AP), sug-
Acknowledgement. This work was in part supported by NSFC (No. 62276108).
References


