

CA-SSL: Class-Agnostic Semi-Supervised Learning for Detection & Segmentation

Lu Qi^{1,3}, Jason Kuen², Zhe Lin², Jiuxiang Gu², Fengyun Rao¹, Dian Li¹,
Weidong Guo^{1*}, Zhen Wen¹, Ming-Hsuan Yang³, Jiaya Jia⁴

¹QQ Browser Lab, Tencent

²Adobe Research

³The University of California, Merced

⁴The Chinese University of Hong Kong

Abstract. To improve instance-level detection/segmentation performance, existing self-supervised and semi-supervised methods extract either task-unrelated or task-specific training signals from unlabeled data. We show that these two approaches, at the two extreme ends of the task-specificity spectrum, are suboptimal for the task performance. Utilizing too little task-specific training signals causes underfitting to the ground-truth labels of downstream tasks, while the opposite causes overfitting to the ground-truth labels. To this end, we propose a novel **Class-Agnostic Semi-Supervised Learning** (CA-SSL) framework to achieve a more favorable task-specificity balance in extracting training signals from unlabeled data. CA-SSL has three training stages that act on either ground-truth labels (labeled data) or pseudo labels (unlabeled data). This decoupling strategy avoids the complicated scheme in traditional SSL methods that balances the contributions from both data types. Especially, we introduce a warmup training stage to achieve a more optimal balance in task specificity by ignoring class information in the pseudo labels, while preserving localization training signals. As a result, our warmup model can better avoid underfitting/overfitting when fine-tuned on the ground-truth labels in detection and segmentation tasks. Using 3.6M unlabeled data, we achieve a significant performance gain of 4.7% over ImageNet-pretrained baseline on FCOS object detection. In addition, our warmup model demonstrates excellent transferability to other detection and segmentation frameworks.

Keywords: Semi-supervised, Class-agnostic, Instance-level Detection

1 Introduction

Deep learning [20, 22, 23, 28, 48, 51–53] has enabled instance-level detection (object detection [16, 17, 45], instance segmentation [19, 35], etc.) methods to achieve previously unattainable performance. Such success cannot be achieved without large datasets with instance annotations such as COCO [34], Cityscapes [12] and Open Images [27]. However, annotating instances is laboriously expensive due to the great intricateness needed for annotating instance-level bounding boxes, masks, and/or semantic classes. Due to such a limitation, the datasets

* Weidong Guo is the corresponding author.

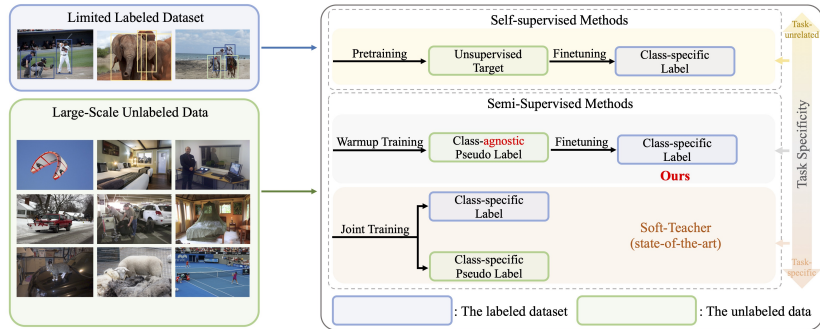


Fig. 1. Schematic comparison between existing semi-supervised methods and CA-SSL. Instead of joint training with labeled and unlabeled data, we train on unlabeled and labeled dataset in different stages: **pseudo labeler training**, **warmup training**, and **finetuning**. Furthermore, we use class-agnostic pseudo labels during warmup training.

in instance-level detection domain are relatively small in scale, compared to other domains. As such, instance-level detection models generally have degraded generalization performance in real-world applications [11, 37, 39, 47].

To alleviate the heavy cost of data annotation, numerous methods have been developed to leverage unlabeled images for instance-level detection [12, 34]. Unlike labeled data that can provide explicit and indisputable supervision signals, the extraction of training signals from unlabeled data remains as an open issue. Two widely-adopted approaches for learning from unlabeled data are self-supervised and semi-supervised learning. Self-supervised methods [8, 14, 18, 21, 44, 59, 62] usually rely on training signals like the relative distances between the augmented samples of a positive/negative pair, while semi-supervised strategies [24, 30, 43, 49, 54, 57, 64, 68] directly leverage the pseudo labels generated by a detector pretrained on ground-truth labels. In terms of task specificity, these two approaches are at the two extreme ends of the spectrum – self-supervised learning utilizes hardly any task-specific training signals from unlabeled data, whilst semi-supervised learning utilizes too much of them. Consequently, the model tends to underfit/overfit (depending on the amount of task specificity in the training signals) the ground-truth labels of downstream task during the finetuning or final-training stage. This motivates us – *is there a good middle ground between the two extremes that has a more optimal amount of task specificity?*

In instance-level detection/segmentation tasks, the datasets usually have two types of annotations: localization-based annotations (*e.g.*, boxes, masks) and class labels for those annotations. While semi-supervised methods utilize the information from both kinds of annotations to do pseudo labeling on unlabeled data, we show that such a practice is not optimal for training and would hurt the final performance. Given the pseudo labels that closely mimic ground-truth labels, the model is likely to take an easier optimization path and potentially arrive at a less-favorable local optimum at the period of training. To mitigate the issue, we can disregard either one of the annotation types for the purpose of pseudo labeling. A related example is the conventional practice of pretraining the model on ImageNet [15] with just image-level class labels. Conversely, the

approach of using localization-based annotations (while ignoring class labels) has not been explored previously, which we believe is a promising direction given the nature of detection/segmentation tasks.

In this work, we propose a novel Class-Agnostic Semi-Supervised Learning (CA-SSL) framework for instance-level detection and segmentation. CA-SSL is a framework consisting of three cascaded training stages (pseudo labeler training, warmup training, and finetuning) with two detector types (pseudo labeler and target detector). Each stage employs a specific type of training data and a type of detector. Concretely, we first train a pseudo labeler on the labeled dataset with class-agnostic annotations and then use it to generate pseudo labels on unlabeled images. After that, we perform warmup training for the target detector on the unlabeled data with class-agnostic pseudo labels. Finally, we finetune the warmed-up target detector on class-specific ground-truth annotations for a particular instance-level task. Unlike existing state-of-the-art SSL method Soft-Teacher [64] that jointly uses labeled and unlabeled data within its single training stage, these two data types are assigned separately to the different stages of CA-SSL, as shown in Fig. 1. This decoupling strategy provides the warmup model with a good initial solution (learned from unlabeled data) that guides it to maintain a good generalization performance during finetuning.

In our experiments, we carry out evaluations by considering upper-bound model performance on multiple large-scale unlabeled data splits that have different dataset scales, consisting of images from COCO unlabeled [34], Open Images [29] and Places365 [66]. Through extensive experiments, we demonstrate that our method can obtain consistent performance gains when using different unlabeled splits ranging from 0.12M to 3.6M images. Owing to the superior effectiveness of our method at consuming large-scale unlabeled data, we are the first successful attempt to improve task performance using an unlabeled dataset with an enormous amount of 3.6M images, in the history of semi-supervised object detection. Moreover, our class-agnostic warmup model trained with warmup training demonstrates excellent transferability to other instance-level detection and segmentation frameworks. The contributions for this paper are threefold:

- We propose a novel class-agnostic semi-supervised learning framework for instance-level detection/segmentation tasks. By leveraging cascaded training stages and class-agnostic pseudo labels, it achieves a more optimal amount of task specificity in the training signals extracted from unlabeled data.
- We conduct extensive ablative and comparative experiments on object detection, demonstrating the effectiveness of our method. To the best of our knowledge, we are the first to use unlabeled data at an unprecedented scale of 3.6M for semi-supervised object detection.
- We demonstrate that our class-agnostic warmup model trained with warmup training can significantly improve the performance on other instance-level detection/segmentation tasks (instance segmentation, keypoint detection, entity segmentation, panoptic segmentation) and frameworks.

2 Related work

Instance-level detection/segmentation. Instance-level detection tasks, including object detection [13, 16, 17, 32, 33, 40, 45, 56], instance segmentation [7, 19, 35, 42, 63, 65], and key point detection [19, 50, 61, 67], require detecting objects with different instance-level representations such as bounding box, pixel-wise mask, and keypoints. Recently, numerous class-specific panoptic segmentation [3, 10, 31] and class-agnostic entity segmentation [41] methods have been developed to perform dense image segmentation by treating all segmentation masks as instances. Most of instance-level detection research works generally focus on designing more advanced architectures or detection methods that work well on existing labeled datasets. Instead, we aim to design a training framework that better utilizes unlabeled images, without modifying the underlying architecture or method. This facilitates better understanding of how far current methods can scale with the help of large-scale unlabeled data.

Semi-supervised detection. Semi-supervised learning approaches mainly focus on two directions for instance-level detection. One is concerned with the consistency-based methods [24, 54], which are closely related to self-supervised approaches. They usually construct a regularization loss by designing some contrastive pretext task [8, 18, 21, 44, 59, 62]. Another direction is on pseudo labeling [30, 43, 49, 57, 64, 68]. As the name implies, they leverage a pretrained detector to generate pseudo labels on unlabeled images. The pseudo labels are usually almost identical to the ground-truth labels. Thus, both two types of labels can be used for joint training with similar losses. Our framework is also based on pseudo labeling, but we decouple the semi-supervised pipeline into three cascaded training stages, where each stage employs a specific type of training data (labeled dataset or unlabeled data). Such a design avoids the complicated and careful weighting strategy required to balance unlabeled and labeled data in the joint training scheme of semi-supervised methods [64], where the common issue is to effectively balance the contributions of noisy pseudo labels and ground-truth annotations.

Class-Agnostic detection and segmentation. Class-Agnostic localization [25, 41, 45, 46, 58] has been widely used in detection and segmentation. One of the most prominent examples is the two-stage detector [45]. It mainly has a class-agnostic region proposal network (RPN) and a detection head. RPN predicts numerous high-quality class-agnostic proposals for further classification and localization refinement. Inspired by this design, our method use class-agnostic pseudo labels on unlabeled data for warming up class-specific target detector. As demonstrated by recent works on open-world detection/segmentation [25, 41, 46], it can improve the model’s generalization on unseen objects. Aside from the benefits shown by existing works, in this paper, we present the first evidence that class-agnostic training can significantly bridge the quality gap between pseudo labels and human (ground-truth) labels. This is important because the quality of pseudo labels directly impacts the effectiveness of learning from unlabeled data.

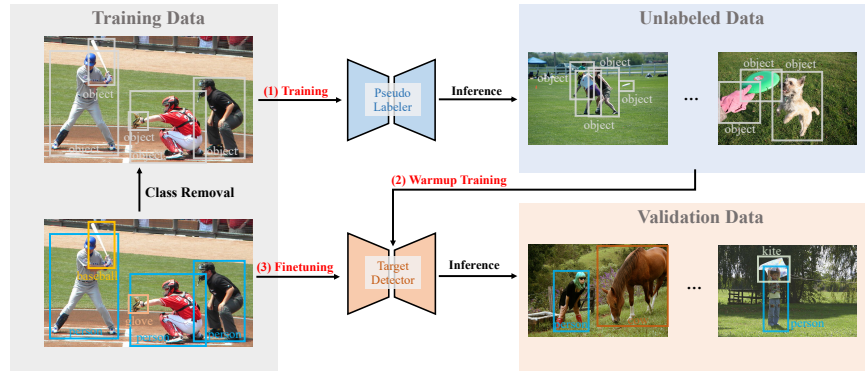


Fig. 2. Illustration of CA-SSL framework. The numbered texts indicate the three cascade stages of our framework. It mainly involves two detectors, namely *pseudo labeler* and *target detector*. The bridge between these two detectors is the unlabeled images. We use the pseudo labeler to predict class-agnostic pseudo labels on the unlabeled images, which are then used as training data for warmup training of target detector.

3 Methodology

Fig. 2 provides an overview of the proposed **Class-agnostic Semi-Supervised Learning (CA-SSL)** framework. The framework consists of three stages, including pseudo labeler training, warmup training, and finetuning. In the first stage, we use the labeled data but with only class-agnostic annotations to train a pseudo labeler. This labeling detector then predicts class-agnostic pseudo labels on unlabeled images. In warmup training, these numerous unlabeled images with their pseudo labels are used to train a target detector. This process is akin to pretraining in self-supervised learning with unlabeled data. We refer to it as warmup training because only localization-based pseudo labels are used as training data, while class labels are ignored. Finally, we fine-tune the warmed-up target detector on the labeled dataset with class-specific annotations.

In the following sections, we first introduce the entire process of our training framework. After that, we explain in detail our proposal to adopt mask-based annotations at Entity [41] level in pseudo labeler and warmup training, as an alternative to the common practice of using box-based object annotations. This annotation adoption is to enable our framework work more effectively beyond instance-level segmentation tasks such as panoptic segmentation. Finally, we describe the base detection framework used by CA-SSL’s training stages.

3.1 Training Stages

We design our training stages based on the finding that there is only a small quality gap between class-agnostic ground-truth and pseudo labels. We study the feasibility of class-agnostic detection for pseudo labeling, by contrasting it to the conventional class-specific detection task, in terms of class-agnostic [41] and class-specific [34] AP metrics respectively. Table 1 shows that the validation performance gap between training on either ground-truth data (COCO

Train data	Annotation	Num	AP ^{det}	AP ^{det-a}
COCO train2017	Ground-truth	118k	41.0	41.9
COCO unlabeled	Pseudo labels	123k	35.9	40.0

Table 1. COCO val2017 validation results from training on ground-truth and pseudo labels, across class-specific (AP^{det}) and -agnostic (AP^{det-a}) tasks. ‘Num’ is the number of training images. AP^{det} and AP^{det-a} indicate the class-specific and -agnostic object detection mAPs respectively.

train2017) or pseudo labels (COCO unlabeled), for both class-specific and -agnostic settings¹ on COCO validation set [34]. The groundtruth-unlabeled AP gaps of class-specific and class-agnostic models are 5.1% (41.0-35.9) and 1.9% (41.9-40.0) respectively. The much smaller gap indicates that the class-agnostic pseudo labels have much a better quality than the class-specific ones, thus enabling the model to closely approach the AP of training on ground-truth labels. In many cases, class-agnostic pseudo labels can eliminate the ambiguities caused by confusing predefined classes such as *cyclist* and *person*, while generalizing well to other kinds of objects unseen, which have been well studied in \mathcal{D}^t [41]. The class-agnostic model with stronger generalization ability provides a greater variety of proposals which help localize objects better during finetuning. We show that such properties of class-agnostic pseudo labels are more useful for warmup training of the class-specific model, compared to prematurely learning classification-aware features through the class-specific ground-truth annotations of downstream task.

Similar to other semi-supervised learning frameworks [24, 30, 43, 49, 54, 57, 64, 68], our CA-SSL framework is largely based on the conventional object detection training process which we first briefly introduce here. Conventionally, given the input images I and their ground-truth annotations Y from the human-labeled training dataset \mathcal{D}^t , the detection model denoted as h_* is trained with the composite detection loss: $\mathcal{L}_{\text{det}} = \mathcal{L}_{\text{cls}}$ (classification loss) + \mathcal{L}_{loc} (localization loss). The detection model h_* is learned through a function $\mathcal{H}(\cdot)$ that determines the neural network hypothesis spaces, depending on the task at hand. Next, we provide the details of the three stages of CA-SSL framework.

Pseudo labeler training. The goal of this stage is to train a pseudo labeler on \mathcal{D}^t to generate high-quality class-agnostic pseudo labels \mathcal{D}^p from the unlabeled dataset split \mathcal{D}^u . To keep the training and inference consistent, we train the pseudo labeler on the class-agnostic annotations obtained from \mathcal{D}^t . We directly remove class information from the annotations in \mathcal{D}^t using the class-agnostic conversion function $\alpha(\cdot)$ and regard each label as a class-free “object”. To train the pseudo labeler, we use a recent class-specific detection framework (see Sec. 3.3) and replace its multi-class classifier with a binary classifier. Given the labeled

¹ We use FCOS [56] with ResNet50 backbone, a widely-used one-stage detector, to explore the performance gap between using class-specific and -agnostic labels. We follow its 36 epoch training setting widely adopted in detectron2 [60] or mmdetection [5]

dataset \mathcal{D}^t , the pseudo labeler h_L is trained as follows:

$$h_L = \operatorname{argmin}_{h \in \mathcal{H}(L)} \sum_{\{I_i, Y_i\} \in \mathcal{D}^t} \mathcal{L}_{\text{det}}^a(h(I_i), \alpha(Y_i)), \quad (1)$$

where $\mathcal{L}_{\text{det}}^a$ representation the class-agnostic version of \mathcal{L}_{det} and $\mathcal{H}(L)$ indicates the neural network hypotheses conditioned on the labeling detection task.

Once the training is done, we apply the pseudo labeler to the unlabeled images and then filter the prediction results using merely a single (constant) score threshold δ . Without semantic class labels in the prediction results, class-agnostic pseudo labels avoid the long-tail problem suffered by in class-specific predictions. Some related ablations are in the supplementary file. As a result, there is no need for a complicated strategy that applies class-dynamic score thresholds as in existing works [30, 43, 49, 57, 64, 68]. Given the pseudo labeler h_L and score threshold δ , our class-agnostic pseudo labeling process to obtain the pseudo labels Y^p and pseudo-label dataset \mathcal{D}^p is represented by the following:

$$Y_i^p = \{y_j \in h_L(I_i^u) | \text{score}(y_j) > \delta\} \quad \forall I_i^u \in \mathcal{D}^u, \quad (2)$$

$$\mathcal{D}^p = \{(I_i^u \in \mathcal{D}^u, Y_i^p \in Y^p) | Y_i^p \neq \emptyset\}, \quad (3)$$

where ‘ p ’ indicates the association with pseudo labels and $\text{score}(\cdot)$ returns the objectness score of any prediction.

Warmup training. We perform warmup training of the target detector only on the pseudo-label dataset \mathcal{D}^p . We do not make use of any ground-truth dataset in this stage, which is different from the state-of-the-art semi-supervised approach Soft-Teacher [64] that carries out joint training on ground-truth and unlabeled dataset splits. Given that, we do not require a divide-and-conquer strategy to handle different dataset splits, such as applying different loss weights to noisy pseudo labels and clean ground-truth labels [1, 2]. To obtain the warmed-up target ‘T’ detection model h_T , we perform warmup training as follows:

$$h_T = \operatorname{argmin}_{h \in \mathcal{H}(T)} \sum_{\{I_i^p, Y_i^p\} \in \mathcal{D}^p} \mathcal{L}_{\text{det}}^a(h(I_i^p), \lceil Y_i^p \rceil), \quad (4)$$

where $\lceil \cdot \rceil$ transforms Y^p to binary training targets. Note that warmup training is related to the pretraining step of self-supervised learning. The weights of the model are well-initialized for better adaptation to downstream tasks. Compared to self-supervised methods [6, 21, 44, 59, 62], the unlabeled data with class-agnostic pseudo labels provides relatively more informative and task-specific supervision signals (class-agnostic localization) that significantly facilitate instance-level detection and segmentation tasks.

Finetuning. After obtaining the warmup model h_T from warmup training, we finetune it for the downstream task using the class-specific ground-truth annotations of \mathcal{D}^t . Instance-level detection/segmentation tasks are typically class-specific tasks. Thus, the output channel of target detector’s semantic classifier should be adapted to the number of pre-defined classes for the downstream

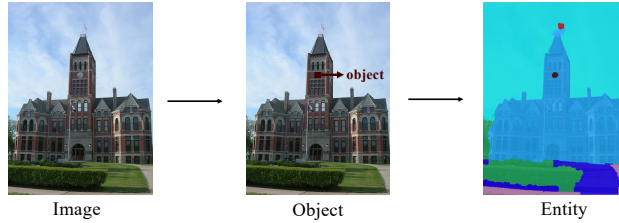


Fig. 3. An example showing that switching from *box-based object* annotations to *mask-based Entity* [41] annotations results in a more substantial set of labels for pseudo labeler and warmup training.

task at hand. There are two ways to initialize multi-class classification layer: (1) random initialization; (2) initialize each output channel with the one from the classification layer of h_T . We empirically find these two strategies produce similar results. The finetuning process to obtain the final ‘F’ downstream-task model h_F is represented by:

$$h_F = \underset{h \in \mathcal{H}(F; h_T)}{\operatorname{argmin}} \sum_{\{I_i, Y_i\} \in \mathcal{D}^t} \mathcal{L}_{\text{det}}(h(I_i), Y_i), \quad (5)$$

where $\mathcal{H}(F; h_T)$ indicates that the neural network hypotheses are conditioned on both the task ‘F’ and pretrained model h_T . Our approach of warming up the target detector with only unlabeled data guarantees it less prone to overfitting to the downstream task’s images and ground-truth labels, since they have not been exposed to the model in the warmup training stage.

3.2 Switching From Objects to Entities

Warmup training with large-scale unlabeled data is a costly process. Thus, it is desirable to design our CA-SSL framework such that the trained class-agnostic model can serve a good range of tasks that expand to detection and also include segmentation. Instance segmentation and panoptic segmentation are two widely-used instance-level segmentation tasks that require more fine-grained visual information for predicting pixelwise masks for each instance.

We draw inspiration from Entity Segmentation [41] on how to boost the applicability and usefulness of our CA-SSL framework. Instead of just focusing on objects, we propose to perform warmup training for the target detector based on the semantically-coherent and class-agnostic mask regions known as Entities. Entities include not just object regions but also stuff regions such as *sky* and *road* which come with the panoptic segmentation task in mainstream datasets (*e.g.*, COCO [34]). With this, even the unlabeled images with little-to-no object regions can still provide substantial pseudo-label training signals through the stuff regions. In Fig. 3, there is only a single object (*clock*), while multiple Entities like *building* and *sky* are present. Furthermore, stuff regions have close relationships with objects, and thus we hypothesize that the training on stuff pseudo labels strongly benefits the downstream task even if it is an object-based task [38].

3.3 Base Detector

We adopt the CondInst [55], a widely-used instance segmentation framework, as our base detector for the pseudo labeler and target detector. Different from Mask R-CNN [19], the CondInst is in fully convolutional manner with two parts: a dense one-stage detector FCOS for detection and a segmentation head for mask prediction. The FCOS has a backbone, FPN neck, and a detection head. The detection head has three output branches: the classification, regression, and kernel branch. The first two branches perform instance-level classification and regression to achieve object detection. Whereas, the kernel branch generates dynamic convolution weights which are used to convolve with high-res feature maps to generate binary instance masks within the segmentation head. Such a network architecture keeps the detection and segmentation parts fairly independent, making it easy to transfer its arbitrary parts to other networks and tasks for finetuning, as proved by our ablation study in Table 9 and Table 10. *E.g.*, we can easily initialize the backbone and FPN neck of Mask R-CNN with our CondInst target detector without any modifications. Overall, the CondInst base detector is trained with the following:

$$\mathcal{L}^e = \mathcal{L}_{\text{det}}^{\{a,c\}} + \mathcal{L}_{\text{seg}}, \quad (6)$$

where \mathcal{L}_{seg} is usually the dice loss between predicted segmentation mask and ground truth. We choose $\mathcal{L}_{\text{det}}^{\{a,c\}}$ as class-agnostic $\mathcal{L}_{\text{det}}^a$ or class-specific $\mathcal{L}_{\text{det}}^c$ depending on the detector and training stage described in subsection 3.1.

Also, CondInst learns to group the pixels that belong to the same semantic/instance region, and otherwise push them far apart. This can be seen as a form of implicit contrastive learning which focuses on spatial-level representation learning, rather than global vector-based representation. This leads to segregated features which are useful for detection and segmentation tasks.

4 Experiments

Datasets. MS-COCO [34] is used as the main evaluation dataset. In addition to MS-COCO `train2017` and `val2017` splits, we curate 3.6 million unlabeled images from COCO `unlabeled` [34], Places365 [66], and Open Images [29]. To better demonstrate the data scalability of CA-SSL, we construct four unlabeled data subsets (`tiny`, `small`, `base`, `large`) with 120K, 660K, 1.74M, and 3.66M images respectively. Unless specified, we report the experimental results from using `tiny` as the unlabeled subset, which is from COCO `unlabeled` and has nearly the same scale as `train2017` as shown in Table 1.

Training Setup. For fair comparisons with other methods (some may require less training time), we train all models to reach their respectively upper-bound performances, by increasing the number of training epochs accordingly. Unless specified, we train with 60 and 36 epochs in warmup training and finetuning respectively for CA-SSL to achieve the upper-bound performance. Upper-bound performance evaluation is the preferred way to gauge the true performance of

warmup training data	warmed-up model				finetuned model		
	AP ^{det-a}	AP ^{seg-a}	AP ^{det}	AP ^e	AP ^{det}	AP ^{seg}	PQ
ImageNet	-	-	-	-	46.8	43.0	41.4
Objects(G,A)	41.9	35.3	-	-	46.9	-	-
Objects(P,S)	-	-	35.9	-	47.1	-	-
Objects(P,A)	40.0	33.4	-	-	48.1	45.0	42.5
Entities(P,A)	-	-	-	40.2	48.2	45.2	43.7

Table 2. Effect of warmup training data choice. AP^{det} and AP^{seg} indicate the APs for class-specific object detection and instance segmentation tasks. Their class-agnostic counterparts have names appended with ‘-a’. AP^e and PQ are the evaluation metrics of class-agnostic Entity detection [41] and panoptic segmentation. ‘G’ and ‘P’ indicate whether ground-truth (COCO `train2017` with 118K images) or pseudo labels (COCO unlabeled data with 123K images) are used. They are different datasets but with comparable numbers of images. ‘A’ and ‘S’ indicate whether class-agnostic or class-specific labels are used. Note that all the results here are obtained via COCO `val2017`.

epochs	AP ^e	AP ^e ₅₀	AP ^e ₇₅	AP ^e _s	AP ^e _m	AP ^e _l
12 (1×)	38.4	59.6	40.5	14.5	35.0	52.6
24 (2×)	39.6	60.5	41.9	15.0	36.3	54.1
36 (3×)	39.9	60.8	42.4	15.1	36.9	54.6
48 (4×)	40.2	60.8	42.7	15.4	37.1	54.8
60 (5×)	40.2	60.9	42.8	15.4	37.0	54.9

Table 3. Performances of pretrained model with different number of warmup training epochs. The AP^e₅₀ and AP^e₇₅ are APs of the Entity detection task based on 0.5 and 0.75 IoU thresholds. The AP^e_s, AP^e_m and AP^e_l indicate the mAP performance on small, middle, and large entities.

different methods that may require different training costs. We apply either *weak* data augmentation (conventional multi-scale training) or *strong* data augmentation (jittering of scale, brightness, contrast, etc. [9, 64]) to the unlabeled images before generating pseudo labels.

Implementation. We adopt the Condinst [55] framework with Swin Transformer Tiny (T) backbone [36] as our base model. Please refer to our supplementary file for the hyper-parameter settings in the warmup training and finetuning.

4.1 Experimental Results

Warmup Training Data. Table 2 shows the impact of the choice of warmup training data on the downstream task performances after the finetuning stage. From the first two rows, we observe that ImageNet supervised-pretrained model and the warmup model trained on class-agnostic ground-truth labels suffer from the worst downstream performances. Whereas, in the last three rows, the models that leverage pseudo labels ‘P’ for training consistently achieve stronger downstream performances. It can also be clearly seen that using class-agnostic labels ‘A’ during warmup training is better than using class-specific labels ‘S’, resulting in a downstream AP^{det} gap of 1.0% (48.1-47.1). Class-Agnostic warmup training helps to prevent the model from overfitting to the ground-truth labels during finetuning. In particular, Entity-based warmup training data provides the best

F \ W	1×	2×	3×	4×	5×
1×	45.3	46.3	46.4	46.8	47.1
2×	47.0	47.4	47.7	47.3	47.7
3×	47.2	47.7	47.6	47.9	48.2
4×	47.3	47.6	47.7	47.9	48.0
5×	47.5	47.7	47.9	47.9	48.0

Table 4. Exploration to the best training epoch combination in the warmup training and finetuning stages on downstream object detection performance AP^{det} . $E \times$ indicates the $E \times 12$ training epochs. *E.g.*, $1 \times$ and $5 \times$ represent 12 and 60 training epochs, respectively. ‘F’ and ‘W’ represent the **F**inetuning and **W**armup training stages.

pseudo labeler (AP^e)	38.0	39.1	40.6	41.4	42.1
warmed-up target detector (AP^e)	36.1	37.3	38.8	39.6	40.2

Table 5. Relationship between the performance of pseudo labeler and the performance of the warmed-up target detector right after warmup training. The different columns correspond to the various backbones³ used by the pseudo labeler. The warmup target detector is always based on **Tiny** backbone.

overall downstream task performance, while strongly raising the downstream PQ performance by 1.2% (43.7-42.5) due to the inclusion of both object and stuff elements. Note that the last two rows of Table 2 use identical COCO **unlabeled** images but with different annotation types: box-based objects and mask-based entities [41] (objects & stuffs), as illustrated in Sec. 3.2.

Upper-Bound Performance. We investigate the number of training epochs required to achieve the upper-bound performance in both the warmup training and finetuning stages. Table 3 shows the class-agnostic performance under different numbers of warmup training epochs. When increasing the number of epochs from 12 to 48, the performance of trained model improves from 38.4 to 40.2 in terms of class-agnostic AP^e . The improvement saturates after 40 epochs. Table 4 shows the the downstream class-specific object detection performance under different combinations of warmup training and finetuning epochs. With 60 epochs ($5 \times$) in the warmup training and 36 epochs ($3 \times$) in finetuning stages, we obtain the best performance of 48.2 AP^{det} . Therefore, we adopt this particular setting in the rest of the experiments.

Pseudo labeler. We investigate the impact of pseudo labeler in the first training stage. Table 5 shows how the performance of pseudo labeler affects the performance of the target detector (after warmup training is performed) on the class-agnostic entity detection task. The results here conclude that a stronger pseudo labeler consistently leads to a stronger target detector in warmup training.

Data Augmentation. Table 6 shows the impact of data augmentation scheme adopted in the warmup training and finetuning stages, on the intermediate and final detection performances. Using strong data augmentation independently for any of the two stages brings some performance improvement, while combining the two provides the largest AP^{det} improvement of 0.9% (49.1-48.2).

¹ ResNet-101, Swin-Tiny, Swin-Small, Swin-Base, and Swin-Large.

F \ W	weak	strong
weak	(40.2, 48.2)	(40.6 , 48.6)
strong	(40.2, 48.6)	(40.6 , 49.1)

Table 6. Ablation study on data augmentation. ‘W’ and ‘F’ represent the warmup training and finetuning stages, while ‘weak’ and ‘strong’ refer to weak and strong data augmentation. (A, B) indicate the class-agnostic AP^e and class-specific AP^{det} in warmup training and finetuning.

Setting	Num	warmup model (AP ^e)	fine-tuned model (AP ^{det})
Tiny	123K	40.2	48.2
Small	660k	40.7	48.8
Base	1.74M	41.4	49.7
Large	3.66M	42.0	50.6

Table 7. Ablation study on unlabeled splits with different dataset scales.

Type	Method	Unlabeled Images	Model	AP ^{det}
Self-supervised	MoCo-v2 [6]	1.28M	R-50-FPN	39.7 $\xrightarrow{+0.1}$ 39.8
	DenseCL [59]	1.28M	R-50-FPN	39.7 $\xrightarrow{+0.6}$ 40.3
	DetCo [62]	1.28M	R-50-MaskRCNN	38.9 $\xrightarrow{+1.2}$ 40.1
	DetCon [21]	1.28M	R-50-FPN	41.6 $\xrightarrow{+1.8}$ 42.7
	PreDet [44]	50.00M	R-50-MaskRCNN	44.9 $\xrightarrow{+2.2}$ 47.1
Semi-supervised	Proposal learning [54]	0.12M	R-50-FPN	37.4 $\xrightarrow{+1.0}$ 38.4
	STAC [49]	0.12M	R-50-FPN	39.5 $\xrightarrow{-0.3}$ 39.2
	Self-training [68]	2.90M	R-50-FPN (SimCLR)	41.1 $\xrightarrow{+0.8}$ 41.9
	Soft Teacher [64]	1.74M [†]	Swin-T-FCOS	46.8 $\xrightarrow{+2.5}$ 49.3
		3.66M [‡]	Swin-L-HTC++	58.2 $\xrightarrow{+1.7}$ 59.9
	CA-SSL (ours)	1.74M [†]	Swin-T-FCOS	46.8 $\xrightarrow{+4.0}$ 50.8
		3.66M [‡]		46.8 $\xrightarrow{+4.7}$ 51.5
			Swin-L-HTC++	58.2 $\xrightarrow{+2.7}$ 60.9

Table 8. Comparison with the state-of-the-arts under the setting of **train2017** set with 118k images. The ‘unlabeled images’ means the number of unlabeled images we use. In the column of ‘AP^{det}’, the left part is the baseline performance of using only **train2017**, and the right part is the performance of using both **train2017** and unlabeled dataset. \rightarrow indicates the performance gain. The symbols [§](ImageNet), ^{*}(COCO), [†] (our curated sets described in Sec. 4) refer to the same respective sets of unlabeled images.

Scale of Unlabeled Dataset. Table 7 shows how the performance varies by training the model on different unlabeled dataset splits. With the increase of unlabeled dataset scale, the performances of the models from the warmup training and finetuning stages improve consistently. We also notice that the class-agnostic performance of the warmup model correlates well with the performance of the finetuning model. This is expected as the substantial task similarity between those two training stages.

State-of-the-art Comparison. In Table 8, we compare the performance of CA-SSL on the downstream object detection task with those of state-of-the-art methods in strong data augmentation. Our method achieves the most significant performance gains even though our model is based on an already strong baseline

backbone	neck	head	classifier	AP ^{det}	AP ₅₀ ^{det}	AP ₇₅ ^{det}
○	○	○	○	46.8	66.2	50.8
✓	○	○	○	50.6	69.2	55.2
✓	✓	○	○	50.8	69.4	55.5
✓	✓	✓	○	51.5	69.5	55.3
✓	✓	✓	✓	51.5	69.6	55.4

Table 9. Ablation study on the initialization strategy. The pretrained model is divided into four parts here. ○ indicates the particular part’s pretrained weights are not being used, while ✓ indicates otherwise.

with the powerful Swin-Tiny backbone. With 1.74M unlabeled images, we obtain 50.8 AP^{det} with 4.0% improvement over the ImageNet-pretrained baseline. Furthermore, by increasing the number of unlabeled images to 3.66M, we observe an even bigger performance gain of 4.7% that leads to 51.5 AP^{det}. Since there is no obvious sign of performance saturation, we believe that using a super-scale unlabeled dataset (larger than our 3.66M one) can potentially improve model performance significantly.

Moreover, using a stronger detector Swin-Large-HTC++ [4] with CA-SSL still provides a meaningful 2.7% improvement (58.2 → 60.9 AP^{det}), suggesting that CA-SSL is compatible with advanced detectors. With the same detector and similar unlabeled images, Soft Teacher [64] merely achieves a 2.5 % (Swin-Tiny-FCOS) and 1.7 % (Swin-Large-HTC++) gain, despite its relatively significant 3.6% gain with R-50-FPN backbone and 120K unlabeled images. The reasons are twofold. Unlike CA-SSL that decouples the usage of different source data into three stages, Soft Teacher performs joint training and that requires it to be particular about the scale of unlabeled data, in order to achieve a good balance between the contributions from labeled and unlabeled data. Using an unlabeled dataset larger than expected can spoil such a balance. On the other hand, Self-training [68] performs much worse than ours, even with a large amount of 2.9M unlabeled images. This is due to the premature use of class/downstream-specific labels in the pseudo-labeling stage, potentially causing the model overfit easily to ground-truth labels during finetuning. Whereas, CA-SSL mitigates such a problem by not allowing the model to train on labeled images and class-specific labels during warmup training.

Note that the performance improvements of our proposed semi-supervised learning do not come from Transformer architecture. With R-50-FPN detector and similar 0.12M unlabeled images, CA-SSL improves the performance at a much larger margin than other semi-supervised methods.

Initialization Strategy. We study the effects of including/excluding the multiple parts (backbone, neck, head, and classifier) of the pretrained model from the warmup training stage for initializing the finetuning model. Table 9 shows that the main source of improvement (+3.8%) comes from backbone initialization, while the other parts make smaller improvements. This ablation study motivates us to transfer our pretrained model to other instance-level detection/segmentation tasks that may not use the downstream frameworks as ours.

Task	Type	Framework	Pretrained w/	Task Perf.
DET	E	FCOS [56]	ImageNet	46.8
			Ours	51.5 (+4.7)
	P	RetinaNet [33]	ImageNet	42.8
			Ours	47.9 (+5.1)
INS	E	CondInst [55]	ImageNet	44.0
			Ours	46.5 (+2.5)
	P	Mask R-CNN [19]	ImageNet	41.9
			Ours	44.9 (+3.0)
Entity	E	CondInst [55]	ImageNet	35.1
			Ours	38.2 (+3.1)
Point	P	Mask R-CNN [19]	ImageNet	66.8
			Ours	67.8 (+1.0)
Panop	P	PanopticFPN [26]	ImageNet	39.5
			Ours	41.5 (+2.0)

Table 10. Evaluation on representative instance-level detection/segmentation tasks with Swin-Tiny backbone. “E” and “P” indicate initializing the framework with *entire* or *part(s)* of our pretrained model. “ImageNet” and “Ours” refer to ImageNet and our weights trained in the warmup stage. “Task Perf” refers to the task-specific evaluation metrics, including AP for object detection, instance segmentation and key-point detection, AP^e for entity segmentation, and PQ for panoptic segmentation.

Generalization to Other Frameworks/Tasks. Table 10 shows the strong performance improvements resulted from initializing the downstream models with a single model h_T pretrained with CA-SSL, on various instance-level detection and segmentation tasks. Some frameworks like Mask R-CNN have heads that are incompatible with those of our Condinst base detector. Using just our FPN backbone pretrained weights to initialize Mask R-CNN, we achieve 2.7 AP^{seg} and 1.0 AP^{point} improvements on instance segmentation and key-point detection. This demonstrates the strong generalization ability of our semi-supervised learning method.

5 Conclusion

In this work, we propose a class-agnostic semi-supervised learning framework to improve instance-level detection/segmentation performance with unlabeled data. To extract the training signals with a more optimal amount of task specificity, the framework adopts class-agnostic pseudo labels and includes three cascaded training stages, where each stage uses a specific type of data. By performing warmup training on a large amount of class-agnostic pseudo labels on unlabeled data, the class-agnostic model has strong generalization ability and is equipped with the right amount task-specific knowledge. When finetuned on different downstream tasks, the model can better avoid overfitting to the ground-truth labels and thus can achieve better downstream performance. Our extensive experiments show the effectiveness of our framework on object detection with different unlabeled splits. Moreover, the pretrained class-agnostic model demonstrates excellent transferability to other instance-level detection frameworks and tasks.

References

1. et al, B.: Mixmatch: A holistic approach to semi-supervised learning. In: NeurIPS (2019) [7](#)
2. et al, R.: Not all unlabeled data are equal: Learning to weight data in semi-supervised learning. In: NeurIPS (2020) [7](#)
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020) [4](#)
4. Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., et al.: Hybrid task cascade for instance segmentation. In: CVPR (2019) [13](#)
5. Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., et al.: Mmdetection: Open mmlab detection toolbox and benchmark (2019) [6](#)
6. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. In: arXiv (2020) [7](#), [12](#)
7. Chen, X., Girshick, R., He, K., Dollár, P.: Tensormask: A foundation for dense object segmentation. In: ICCV (2019) [4](#)
8. Chen, X., He, K.: Exploring simple siamese representation learning. In: CVPR (2021) [2](#), [4](#)
9. Chen, Y., Li, Y., Kong, T., Qi, L., Chu, R., Li, L., Jia, J.: Scale-aware automatic augmentation for object detection. In: CVPR (2021) [10](#)
10. Cheng, B., Schwing, A.G., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation. In: NeurIPS (2021) [4](#)
11. Chu, R., Sun, Y., Li, Y., Liu, Z., Zhang, C., Wei, Y.: Vehicle re-identification with viewpoint-aware metric learning. In: ICCV (2019) [2](#)
12. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: CVPR (2016) [1](#), [2](#)
13. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In: NeurIPS (2016) [4](#)
14. Dai, Z., Cai, B., Lin, Y., Chen, J.: Up-detr: Unsupervised pre-training for object detection with transformers. In: CVPR (2021) [2](#)
15. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009) [2](#)
16. Dollár, P., Zitnick, C.L.: Fast edge detection using structured forests. In: PAMI (2015) [1](#), [4](#)
17. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014) [1](#), [4](#)
18. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR (2020) [2](#), [4](#)
19. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. In: ICCV (2017) [1](#), [4](#), [9](#), [14](#)
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) [1](#)
21. Hénaff, O.J., Koppula, S., Alayrac, J.B., Oord, A.v.d., Vinyals, O., Carreira, J.: Efficient visual pretraining with contrastive detection. In: ICCV (2021) [2](#), [4](#), [7](#), [12](#)
22. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: CVPR (2018) [1](#)
23. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: CVPR (2017) [1](#)

24. Jeong, J., Lee, S., Kim, J., Kwak, N.: Consistency-based semi-supervised learning for object detection (2019) [2](#), [4](#), [6](#)
25. Kim, D., Lin, T.Y., Angelova, A., Kweon, I.S., Kuo, W.: Learning open-world object proposals without learning to classify. In: arXiv (2021) [4](#)
26. Kirillov, A., Girshick, R., He, K., Dollár, P.: Panoptic feature pyramid networks. In: CVPR (2019) [14](#)
27. Krasin, I., Duerig, T., Alldrin, N., Veit, A., Abu-El-Haija, S., Belongie, S., Cai, D., Feng, Z., Ferrari, V., Gomes, V., et al.: Openimages: A public dataset for large-scale multi-label and multi-class image classification. In: Dataset available from <https://github.com/openimages> (2016) [1](#)
28. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NeurIPS (2012) [1](#)
29. Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Mallocci, M., Duerig, T., et al.: The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. IJCV (2020) [3](#), [9](#)
30. Li, Y., Huang, D., Qin, D., Wang, L., Gong, B.: Improving object detection with selective self-supervised self-training. In: ECCV (2020) [2](#), [4](#), [6](#), [7](#)
31. Li, Y., Zhao, H., Qi, X., Chen, Y., Qi, L., Wang, L., Li, Z., Sun, J., Jia, J.: Fully convolutional networks for panoptic segmentation with point-based supervision. In: arXiv (2021) [4](#)
32. Lin, T., Dollár, P., Girshick, R.B., He, K., Hariharan, B., Belongie, S.J.: Feature pyramid networks for object detection. In: CVPR (2017) [4](#), [14](#)
33. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV (2017) [4](#), [14](#)
34. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014) [1](#), [2](#), [3](#), [5](#), [6](#), [8](#), [9](#)
35. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: CVPR (2018) [1](#), [4](#)
36. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV (2021) [10](#)
37. Morrison, D., Tow, A.W., McTaggart, M., Smith, R., Kelly-Boxall, N., Wade-McCue, S., Erskine, J., Grinover, R., Gurman, A., Hunn, T., et al.: Cartman: The low-cost cartesian manipulator that won the amazon robotics challenge. In: ICRA (2018) [2](#)
38. Mottaghi, R., Chen, X., Liu, X., Cho, N.G., Lee, S.W., Fidler, S., Urtasun, R., Yuille, A.: The role of context for object detection and semantic segmentation in the wild. In: CVPR (2014) [8](#)
39. Qi, L., Jiang, L., Liu, S., Shen, X., Jia, J.: Amodal instance segmentation with kins dataset. In: CVPR (2019) [2](#)
40. Qi, L., Kuen, J., Gu, J., Lin, Z., Wang, Y., Chen, Y., Li, Y., Jia, J.: Multi-scale aligned distillation for low-resolution detection. In: CVPR (2021) [4](#)
41. Qi, L., Kuen, J., Wang, Y., Gu, J., Zhao, H., Lin, Z., Torr, P., Jia, J.: Open-world entity segmentation. In: arXiv (2021) [4](#), [5](#), [6](#), [8](#), [10](#), [11](#)
42. Qi, L., Zhang, X., Chen, Y., Chen, Y., Sun, J., Jia, J.: Pointins: Point-based instance segmentation. In: arXiv (2020) [4](#)
43. Radosavovic, I., Dollár, P., Girshick, R., Gkioxari, G., He, K.: Data distillation: Towards omni-supervised learning. In: CVPR (2018) [2](#), [4](#), [6](#), [7](#)

44. Ramanathan, V., Wang, R., Mahajan, D.: Predet: Large-scale weakly supervised pre-training for detection. In: ICCV (2021) 2, 4, 7, 12
45. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: NeurIPS (2015) 1, 4
46. Sharma, A., Khan, N., Mubashar, M., Sundaramoorthi, G., Torr, P.: Class-agnostic segmentation loss and its application to salient object detection and segmentation. In: ICCV (2021) 4
47. Shu, G.: Human detection, tracking and segmentation in surveillance video (2014) 2
48. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015) 1
49. Sohn, K., Zhang, Z., Li, C.L., Zhang, H., Lee, C.Y., Pfister, T.: A simple semi-supervised learning framework for object detection. In: arXiv (2020) 2, 4, 6, 7, 12
50. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: CVPR (2019) 4
51. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: arXiv (2016) 1
52. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR (2015) 1
53. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: CVPR (2016) 1
54. Tang, P., Ramaiah, C., Wang, Y., Xu, R., Xiong, C.: Proposal learning for semi-supervised object detection. In: WACV (2021) 2, 4, 6, 12
55. Tian, Z., Shen, C., Chen, H.: Conditional convolutions for instance segmentation. In: ECCV (2020) 9, 10, 14
56. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: ICCV (2019) 4, 6, 14
57. Wang, K., Yan, X., Zhang, D., Zhang, L., Lin, L.: Towards human-machine cooperation: Self-supervised sample mining for object detection. In: CVPR (2018) 2, 4, 6, 7
58. Wang, W., Feiszli, M., Wang, H., Tran, D.: Unidentified video objects: A benchmark for dense, open-world segmentation. In: arXiv (2021) 4
59. Wang, X., Zhang, R., Shen, C., Kong, T., Li, L.: Dense contrastive learning for self-supervised visual pre-training. In: CVPR (2021) 2, 4, 7, 12
60. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. <https://github.com/facebookresearch/detectron2> (2019) 6
61. Xiao, B., Wu, H., Wei, Y.: Simple baselines for human pose estimation and tracking. In: ECCV (2018) 4
62. Xie, E., Ding, J., Wang, W., Zhan, X., Xu, H., Sun, P., Li, Z., Luo, P.: Detco: Unsupervised contrastive learning for object detection. In: ICCV (2021) 2, 4, 7, 12
63. Xie, E., Sun, P., Song, X., Wang, W., Liu, X., Liang, D., Shen, C., Luo, P.: PolarMask: Single shot instance segmentation with polar representation. In: CVPR (2020) 4
64. Xu, M., Zhang, Z., Hu, H., Wang, J., Wang, L., Wei, F., Bai, X., Liu, Z.: End-to-end semi-supervised object detection with soft teacher. In: ICCV (2021) 2, 3, 4, 6, 7, 10, 12, 13
65. Zhang, R., Tian, Z., Shen, C., You, M., Yan, Y.: Mask encoding for single shot instance segmentation. In: CVPR (2020) 4

- 66. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. In: TPAMI (2017) [3](#), [9](#)
- 67. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points (2019) [4](#)
- 68. Zoph, B., Ghiasi, G., Lin, T.Y., Cui, Y., Liu, H., Cubuk, E.D., Le, Q.V.: Rethinking pre-training and self-training. In: NeurIPS (2020) [2](#), [4](#), [6](#), [7](#), [12](#), [13](#)