Instant-Teaching: An End-to-End Semi-Supervised Object Detection Framework

Qiang Zhou, Chaohui Yu, Zhibin Wang, Qi Qian, Hao Li
Alibaba Group
{jianchong.zq, huakun.ych, zhibin.waz, qi.qian, lihao.lh}@alibaba-inc.com

Abstract

Supervised learning based object detection frameworks demand plenty of laborious manual annotations, which may not be practical in real applications. Semi-supervised object detection (SSOD) can effectively leverage unlabeled data to improve the model performance, which is of great significance for the application of object detection models. In this paper, we revisit SSOD and propose Instant-Teaching, a completely end-to-end and effective SSOD framework, which uses instant pseudo labeling with extended weak-strong data augmentations for teaching during each training iteration. To alleviate the confirmation bias problem and improve the quality of pseudo annotations, we further propose a co-rectify scheme based on Instant-Teaching, denoted as Instant-Teaching. Extensive experiments on both MS-COCO and PASCAL VOC datasets substantiate the superiority of our framework. Specifically, our method surpasses state-of-the-art methods by 4.2 mAP on MS-COCO when using 2% labeled data. Even with full supervised information of MS-COCO, the proposed method still outperforms state-of-the-art methods by about 1.0 mAP. On PASCAL VOC, we can achieve more than 5 mAP improvement by applying VOC07 as labeled data and VOC12 as unlabeled data.*

1. Introduction

Deep neural networks [24, 43, 19] have significantly improved the performance of diverse computer vision applications, *e.g.*, image classification and object detection. In order to avoid overfitting and achieve better performance, a large amount of accurate human-annotated data is needed to train a deep learning model. However, the assumption of having a sufficient amount of accurate labeled data for training may not hold, especially for object detection tasks, which need annotations with accurate class labels and precise bounding box coordinates. Thus, a natural idea is to leverage abundant unlabeled data to facilitate learning in the original task. To relax the dependency of manually labeled data, a promising approach is called semi-supervised learning (SSL) [8].

SSL has recently received increasing attention from the community, since it provides effective methods of using unlabeled data to facilitate model learning with limited annotated data. Most of the existing SSL methods focus on image classification tasks and there are multiple strategies for semi-supervised learning, *e.g.*, self-training [42, 52] and co-training [5, 35]. Recently, one popular line of research uses consistency losses for semi-supervised learning [27, 37, 25, 48, 34, 41, 49, 51, 4, 3, 44]. They either adopt ensemble learning algorithms to enforce the predictions of the unlabeled data to be consistent across multiple models, or constrain the model predictions to be invariant to noise. Another popular line of SSL research focuses on more effective data augmentations to improve the generalization and robustness of the model, in which some learning-based and more complex data augmentation strategies [3, 51, 10, 4, 44] greatly improve the performance of SSL on image classification tasks.

Although semi-supervised learning has made great progress in the field of image classification, there is a paucity of literature focus on semi-supervised object detection (SSOD). The recently proposed STAC [45] performs best among existing SSOD methods and outperforms the supervised model by a large margin, which is of great significance to the research of SSOD. However, we find that STAC still has some problems. First, its training procedure is complicated and inefficient. Before model training, STAC needs to train a teacher model, and then it uses the teacher model to pre-generate pseudo annotations of unlabeled data. Second, during model training, the pre-generated pseudo annotations will no longer be updated, and the constant label will limit its performance. In this paper, to address the above two problems, we propose a novel end-to-end SSOD framework, Instant-Teaching, which uses instant pseudo labeling with extended weak-strong data augmentations for teaching during each training iteration. Specifically, as shown in Fig. 1, during each training iter-

ation, Instant-Teaching will first generate pseudo annotations of unlabeled data with weak data augmentations in a mini-batch, and then the predicted annotations will instantly be used as the ground-truth of the same image with strong data augmentations for training. The advantage of Instant-Teaching is that as the model converges during training, the quality of pseudo annotations will be improved instantly. The weak-strong data augmentation scheme is inherited from STAC, which has been proven to be effective in combination with pseudo annotations, and we further extend the strong data augmentations to include Mixup and Mosaic. In addition, the confirmation bias [48] is a common problem in SSL. To alleviate this issue, we further propose a co-rectify scheme based on Instant-Teaching, denoted as Instant-Teaching*. Instant-Teaching* simultaneously trains two models that have the same structure but share different weights and these two models help each other to rectify false predictions. During inference, we still only use a single model so that it does not increase inference time.

We test the efficacy of Instant-Teaching* on PASCAL VOC [14] and MS-COCO [31] datasets, and follow the experimental protocols used in the latest state-of-the-art SSOD literature STAC [45] to evaluate the performance. It is worth mentioning that, our Instant-Teaching* framework outperforms state-of-the-art methods at all experimental protocols, and achieves state-of-the-art performance on semi-supervised object detection learning.

The contributions of this paper are as follows:

- We propose a novel SSOD framework, called Instant-Teaching, which uses instant pseudo labeling with extended weak-strong data augmentations for teaching during each training iteration. Instant-Teaching is an end-to-end framework and can effectively leverage the unlabeled data.
- To alleviate the confirmation bias problem and improve the quality of pseudo annotations, we further propose a co-rectify scheme based on Instant-Teaching, denoted as Instant-Teaching*.
- Our extensive experiments on PASCAL VOC and MS-COCO datasets demonstrate the significant efficacy of our Instant-Teaching* framework.

2. Related Work

Object detection is an important computer vision task and has received considerable attention in recent years [17, 16, 39, 18, 29, 7, 20, 32, 38, 30]. One line of research focuses on strong two-stage object detectors [17, 16, 39, 11, 18, 29, 7, 20], which first generate a sparse set of regions of interest (RoIs) with a Region Proposal Network (RPN), and then perform classification and bounding box regression. Another line of research develops fast single-stage object detectors [32, 38, 30, 26, 13, 50]. However, these methods train stronger or faster models on a large amount of accurate human-annotated data, which is expensive and time-consuming to acquire in real applications. In this work, we follow the popular two-stage object detector (Faster-RCNN [39]) to develop our framework. Different from previous methods training models only on labeled data, we train our object detector on both labeled and unlabeled data with our proposed semi-supervised learning strategy.

Semi-supervised learning (SSL) exploits the potential of unlabeled data to facilitate model learning with limited annotated data. Most of the existing SSL methods focus on image classification tasks and most of the works [4, 51, 34, 41, 25, 48, 3, 44] are consistency-based methods, which constrain the model to be invariant to the noise. Pseudo labeling based methods [27, 2, 21, 1, 52] improve the performance of SSL by generating high-quality hard labels (*i.e.*, the $\arg\max$ of the output class probability) of unlabeled data with a predefined threshold and retraining the model. Recently, data augmentations have proven to be a powerful paradigm for boosting SSL on image classification [4, 51, 10, 3, 44]. MixMatch [4] improves SSL by guessing low-entropy labels for data-augmented unlabeled data and mixes labeled and unlabeled data using Mixup. FixMatch [44], UDA [51] and ReMixMatch [3] have shown that RandAugment [10] and CTAugment [3] can significantly facilitate learning of SSL.

Semi-supervised object detection (SSOD) applies semi-supervised learning to object detection. Recently, a few existing works [33, 47, 15, 22, 46, 28, 45, 23] propose to train object detectors on both labeled data and unlabeled data by incorporating SSL into object detection. The methods in [33, 47] depend on additional context (*e.g.*, temporal information from video). The method in [36] proposes data distillation to automatically generate new training annotations by ensembling predictions of multiple transformations of unlabeled data. NOTE-RCNN [15] proposes to iteratively perform bounding box mining and detector retraining. S^{4}OD [28] proposes a selective net as a heuristic for selecting bounding boxes to improve object detection with unlabeled web images. CSD [22] proposes a consistency-based SSL method for object detection, which uses flip augmentation and consistency constraints to enhance detection performance. Based on CSD, ISD [23] proposes to use interpolation regularization to further improve the performance of SSL for object detection. Recently, STAC [45] develops a SSL framework for object detection that combines self-training and consistency regularization based on strong data augmentations, which achieves state-of-the-art results.

Inspired by these methods, this paper exploits the effective usage of pseudo annotations as well as data augmentations and co-rectify scheme to further improve the perfor-

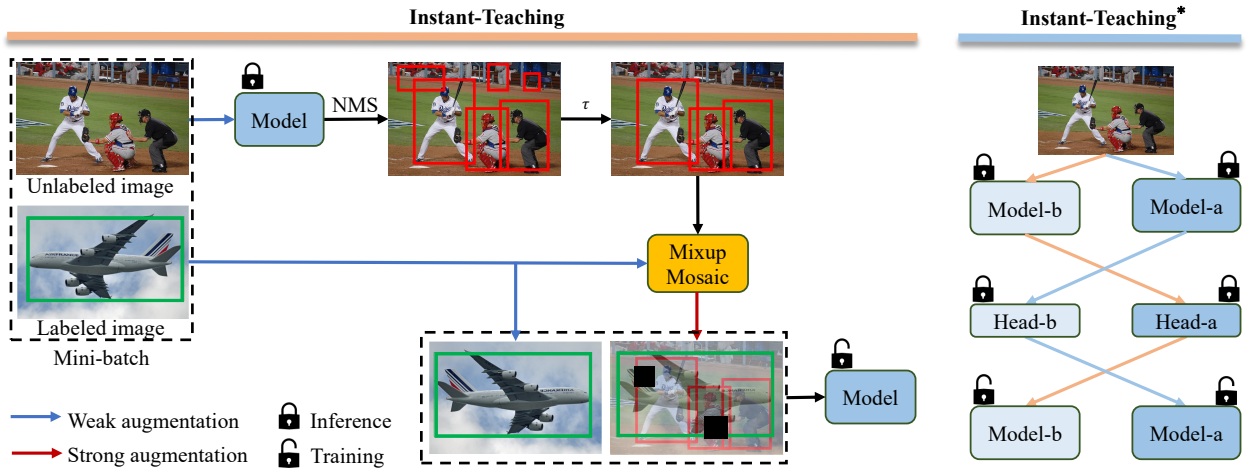

Figure 1. The proposed semi-supervised object detection framework. Instant-Teaching includes instant pseudo labeling with extended weak-strong data augmentations. Instant-Teaching* represents Instant-Teaching combined with our co-rectify scheme.

mance of SSL for object detection in a more efficient and simpler way.

3. Method

In this section, we first give the problem definition of our semi-supervised object detection task (see Section 3.1). Then, we show an overview of our Instant-Teaching* framework (see Section 3.2), which consists of instant pseudo labeling with extended weak-strong data augmentations and co-rectify scheme (see Section 3.3).

3.1. Problem definition

In semi-supervised object detection (SSOD), we are given a set of labeled data $\mathcal{D}_l = \{(\mathbf{x}_i^l, y_i^l)\}_{i=1}^{n_l}$ and a set of unlabeled data $\mathcal{D}_u = \{\mathbf{x}_j^u\}_{j=1}^{n_u}$, where $\mathbf{x}$ and y denote image and ground-truth annotations (class labels and bounding box coordinates) respectively. The goal of SSOD is to train object detectors on both labeled and unlabeled data.

3.2. The overview framework

As shown in Fig. 1, our Instant-Teaching* framework is mainly composed of two modules, namely, instant pseudo labeling with weak-strong data augmentations and co-rectify. It is worth mentioning that, the first module of instant pseudo labeling with weak-strong data augmentations already forms a complete SSOD framework, denoted as Instant-Teaching, which also outperforms state-of-the-art methods.

These two modules have their own focus, among which instant pseudo labeling with weak-strong data augmentations enables our method to be trained end-to-end, and the quality of pseudo annotations is instantly improved as the model converges. Moreover, weak-strong data augmentations enforce the model to maintain consistent predictions between the weakly augmented and the strongly augmented unlabeled data. In this way, the model can learn useful information from the pseudo annotations generated by itself. The co-rectify scheme trains two models with the same structure simultaneously and these two models help each other to rectify false predictions, thus alleviating the common confirmation bias problem and further improving the model performance.

Please note that although our Instant-Teaching* trains two models at the same time, we only use a single model (Model-a) during inference, which does not increase inference time.

3.3. Instant-Teaching*

Instant pseudo labeling. It is beneficial to update the pseudo annotations with a more precise model during the training process, which motivates us to propose instant pseudo labeling. Instant pseudo labeling performs model training and pseudo-label generation at the same time, which is end-to-end and different from the latest STAC [45] framework. STAC needs to train a teacher model in advance to generate pseudo annotations of unlabeled data. Moreover, STAC does not update the generated pseudo annotations during training, which limits its performance.

To be more specific, we decompose each training iteration into two steps. In the first step, we use the current model to generate pseudo annotations of unlabeled data in a mini-batch. Note that in this step, we apply weak augmentations $\alpha(\cdot)$ to unlabeled data (unless otherwise specified, we only use random flip as weak augmentation in all experiments). In the second step, we apply strong augmentations $A(\cdot)$ to the same unlabeled data with pseudo annotations generated in the first step, and update the model parameters with a entire training objective, which consists of a supervised loss and an unsupervised loss. Note that in this

step, to get a fair comparison with STAC, we only apply strong data augmentations to unlabeled data, while still applying weak augmentations to labeled data. In fact, the performance of the model will be relatively poor in the initial training phase. In order to guarantee the quality of the generated pseudo annotations, we always apply non-maximum suppression (NMS) and confidence-based box filtering with a high threshold τ in the first step (unless otherwise specified, we use $\tau = 0.9$ in all experiments).

Overall, the model is trained by jointly minimizing the supervised loss and unsupervised loss as follows:

$$\ell = \ell_s + \lambda_u \ell_u, \tag{1}$$

where we use λ_u to balance the supervised loss ℓ_s and the unsupervised loss ℓ_u.

The supervised loss ℓ_s consists of a classification loss L_{cls} (a standard cross-entropy loss), and a bounding box regression loss L_{reg} (a L_1 loss). ℓ_s can be computed as:

$$\begin{aligned}\ell_s = \sum_l [\frac{1}{N_{cls}} \sum_i L_{cls}(p(c_i \mid \alpha(\mathbf{x}_l)), c_i^*) \\ + \frac{\lambda}{N_{reg}} \sum_i c_i^* L_{reg}(p(\mathbf{t}_i \mid \alpha(\mathbf{x}_l)), \mathbf{t}_i^*)].\end{aligned} \tag{2}$$

In the above equation, l is the index of labeled images in a mini-batch, i is the index of an anchor in a single image, $p(c_i \mid \mathbf{x})$ is the predicted probability of anchor i being an object in image $\mathbf{x}$, $p(\mathbf{t}_i \mid \mathbf{x})$ is the 4-dimensional coordinates of an predicted bounding box, c_i^* and $\mathbf{t}_i^*$ are the human-annotated ground-truth class label and bounding box coordinates respectively.

When computing the unsupervised loss ℓ_u, we first compute the model's predicted class probability distribution and box coordinates based on weakly augmented unlabeled data in a mini-batch: $(c^u, \mathbf{t}^u) = p(c, \mathbf{t} \mid \alpha(\mathbf{x}_u))$. Then we use the hard label $\hat{c}^u = \arg\max(c^u)$ as the final class label of pseudo annotations. In addition, the unsupervised loss is computed on strongly augmented unlabeled data and can be written as:

$$\begin{aligned}\ell_u = \sum_u [\frac{1}{N_{cls}} \sum_i L_{cls}(p(c_i \mid A(\mathbf{x}_u)), \hat{c}_i^u) \\ + \frac{\lambda}{N_{reg}} \sum_i (\max(c_i^u) \geq \tau) L_{reg}(p(\mathbf{t}_i \mid A(\mathbf{x}_u)), \mathbf{t}_i^u)],\end{aligned} \tag{3}$$

where u is the index of unlabeled images in a mini-batch, $\hat{c}_i^u$ and $\mathbf{t}_i^u$ are pseudo annotations of unlabeled data generated by the model itself, and τ is the confidence threshold.

Weak-strong data augmentations. How to encourage the model to learn useful information from the pseudo annotations generated by the model itself is essential to all self-training based SSL methods. Weak-strong data augmentation scheme is a promising practice, which has been proven in semi-supervised image classification [44] and semi-supervised object detection [45]. Weak-strong data augmentations enforce the model to maintain consistent predictions between the weakly augmented and the strongly augmented unlabeled data, and thus encourage the model to learn useful information from the pseudo annotations.

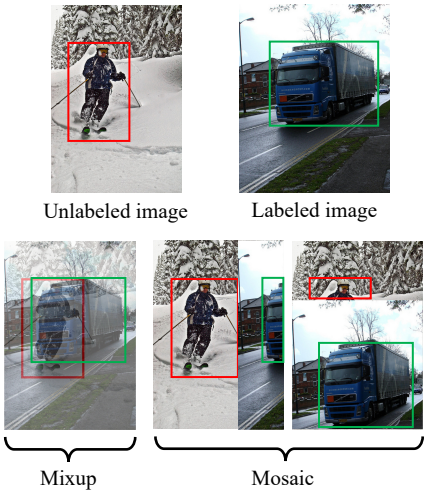

Figure 2. Mixup and Mosaic data augmentations for semi-supervised object detection learning.

Intuitively, the key of weak-strong data augmentation scheme lies in the difference between weak augmentations and strong augmentations. When the weak augmentations remain unchanged, the more complex and appropriate the strong augmentations, the more information the model can learn from the pseudo annotations. Based on this hypothesis, we extend the strong data augmentations of STAC and introduce more complex augmentations for unlabeled data, including Mixup [53] and Mosaic [6]. The experimental results also reveal that our extended weak-strong data augmentations can further improve the performance of semi-supervised object detection.

Specifically, as shown in Fig. 2 (Mixup), given an unlabeled image $\mathbf{x}_u$ and its pseudo annotations (b_u, c_u), where b_u are the 4-dimensional box coordinates and c_u are the one-hot class labels of these pseudo boxes (note that we use hard label when the confidence score is larger than the confidence threshold τ). We first randomly choose one labeled image $\mathbf{x}_l$ with ground-truth annotations (b_l, c_l) from the mini-batch. Next, we mix these two images and their one-hot labels and bounding box coordinates with a mixing coefficient λ_m drawn from the $Beta(\alpha_m, \alpha_m)$ distribution, where $\alpha_m = 1.0$. Finally, we use the mixed image and soft class labels and bounding box coordinates to substitute the image content and pseudo annotations of the unlabeled

image $\mathbf{x}_u$, which can be computed as:

$$\begin{cases} \lambda_m & \sim & Beta(\alpha_m, \alpha_m), \\ \mathbf{x}_u & = & \lambda_m \mathbf{x}_u + (1-\lambda_m)\mathbf{x}_l, \\ c_u & = & \lambda_m c_u \cup (1-\lambda_m) c_l, \\ b_u & = & b_u \cup b_l. \end{cases} \tag{4}$$

As for Mosaic, as shown in Fig. 2 (Mosaic), given an unlabeled image $\mathbf{x}_u$ and a labeled image $\mathbf{x}_l$ in a mini-batch, we randomly perform two kinds of mixing styles (horizontal mixing and vertical mixing) and mix their corresponding annotations accordingly. By applying Mixup and Mosaic data augmentations to unlabeled data, we can improve the model robustness to pseudo annotation noise and alleviate the overfitting problem in model training.

Note that, for a fair comparison, we only perform Mixup and Mosaic data augmentations on unlabeled data and keep labeled data with weak data augmentation unchanged in all our experiments, which is the same as STAC [45].

Co-rectify. Confirmation bias [48] is a common problem in semi-supervised learning. When the model generates incorrect predictions with high confidence, these incorrect predictions will be further strengthened through incorrect pseudo annotations. In other words, the model itself is difficult to rectify these false predictions.

To alleviate this problem, we propose a co-rectify scheme, which trains two models $f_a(\cdot)$ (Model-a) and $f_b(\cdot)$ (Model-b) simultaneously. These two models help each other to rectify the false predictions, as shown in Fig. 1. The key to the success of co-rectify is that the two models will not converge to the same model. We take two measures to ensure that the two models converge independently. First, although the two models have the same structure, they use different initialization parameters. Second, although the two models share the same data in each mini-batch, their data augmentations and pseudo annotations are also different.

We take model $f_a(\cdot)$ as an example and the rectified pseudo annotations of model $f_b(\cdot)$ are constructed in a similar way. When generating pseudo annotations during each training iteration, model $f_a(\cdot)$ first predicts class probabilities c_i and bounding box coordinates $\mathbf{t}_i$ on the weakly augmented unlabeled image $\mathbf{x}_u$. Then, we use the detection head in model $f_b(\cdot)$ to refine the class probabilities c_i^r and bounding box coordinates $\mathbf{t}_i^r$ by taking the predicted boxes $\mathbf{t}_i$ as proposals. Finally, the rectified class probabilities c_i^* are averaged from c_i and c_i^r, and the rectified bounding box coordinates $\mathbf{t}_i^*$ are the weighted average of $\mathbf{t}_i$ and $\mathbf{t}_i^r$. The co-rectify process can be computed as:

$$\begin{cases} (c_i, \mathbf{t}_i) & = & f_a(\mathbf{x}_u), \\ (c_i^r, \mathbf{t}_i^r) & = & f_b(\mathbf{x}_u; \mathbf{t}_i), \\ c_i^* & = & \frac{1}{2}(c_i + c_i^r), \\ \mathbf{t}_i^* & = & \frac{1}{c_i + c_i^r}(\mathbf{t}_i c_i + \mathbf{t}_i^r c_i^r). \end{cases} \tag{5}$$

4. Experiments

We test our proposed semi-supervised object detection framework Instant-Teaching* on the large-scale dataset MS-COCO [31] and report the mAP over 80 object categories. For a fair comparison, we use the same SSL experimental settings as STAC. When performing SSL experiments on the MS-COCO dataset, two experimental settings are used. In the first setting, only a small amount of data in the 118k labeled images is selected as the labeled set. The remainder is used as the unlabeled set. Under this setting, we are able to verify the performance of the SSL algorithm when there is only a small amount of labeled data. In the second setting, the entire 118k images are used as the labeled set and the additional 123k unlabeled images are used as the unlabeled set, which enables us to verify whether SSL algorithm can further improve the performance of the detector when large-scale labeled images already exist. In the first experimental setting, we randomly selected 1%, 2%, 5%, and 10% from the 118k labeled images as the labeled set.

In addition, we also test on PASCAL VOC [14] following [22] and report the mAP over 20 object categories. We use the trainval set of VOC07 as labeled data, which consists of ∼5k images, and the unlabeled data contains the trainval set of VOC12 (∼11k images) and the subset of MS-COCO with the same classes as PASCAL VOC (∼95k images). We evaluate the performance on the test set of VOC07 and report the mAPs at IoU=0.5, IoU=0.75, and IoU=0.5:0.95.

4.1. Implementation details

We implement our Instant-Teaching* framework based on the MMDetection toolbox [9]. To get a fair comparison, we follow STAC to use Faster-RCNN [40] with FPN [29] as our object detector and use ResNet-50 [19] as the feature extractor. The feature weights are initialized by the ImageNet-pretrained model. Instant-Teaching* mainly contains three hyperparameters: λ, λ_u and τ, we set $\lambda = 1.0$, $\lambda_u = 1.0$ and $\tau = 0.9$ unless otherwise specified.

All our experiments maintain the same training parameters as STAC. Specifically, we train the model using an SGD optimizer on 8 GPUs, with an initial learning rate of 0.01, a momentum of 0.9, a weight decay of $1e{-}4$ and a total training step of 180k. The learning rate decays by $10\times$ at 120k and 165k respectively. Moreover, we fix the mini-batch size to 16, in which the ratio between labeled images and unlabeled images is 1:1. Following STAC, for 1%, 2%, 5% and 10% MS-COCO protocols, we use the **quick** learning schedule. For the 100% protocol, we use the **standard** learning schedule. The quick schedule adopts multi-scale training and the standard schedule adopts single-scale training, which is depicted in the Appendix A of STAC [45] and our supplementary materials.

Methods	Backbone	1% COCO	2% COCO	5% COCO	10% COCO	100% COCO
Supervised	R50-FPN	9.05±0.16	12.70±0.15	18.47±0.22	23.86±0.81	37.63
CSD† [22]	R50-FPN	10.20±0.15 (+1.15)	13.60±0.10 (+0.90)	18.90±0.10 (+0.43)	24.50±0.15 (+0.64)	38.87 (+1.24)
STAC[45]	R50-FPN	13.97±0.35 (+4.92)	18.25±0.25 (+5.55)	24.38±0.12 (+5.91)	28.64±0.21 (+4.78)	39.21 (+1.58)
Instant-Teaching (ours)	R50-FPN	**16.00±0.20** (+6.95)	**20.70±0.30** (+8.00)	**25.50±0.05** (+7.03)	**29.45±0.15** (+5.59)	**39.60** (+1.97)
Instant-Teaching* (ours)	R50-FPN	**18.05±0.15** (+9.00)	**22.45±0.15** (+9.75)	**26.75±0.05** (+8.28)	**30.40±0.05** (+6.54)	**40.20** (+2.57)

Table 1. Comparison of mAP for different semi-supervised methods on MS-COCO. CSD† is our implementation of the CSD method based on the Faster-RCNN detector. Instant-Teaching* represents our Instant-Teaching framework with co-rectify scheme. The value in brackets represents the mAP improvement compared to the supervised model.

Methods	Backbone	Unlabeled	$AP^{0.5:0.95}$	$AP^{0.5}$	$AP^{0.75}$
Supervised (Ours)	R50-FPN		43.60	76.70	44.50
CSD [22]	R101-R-FCN	VOC12	-	74.70	-
STAC [45]	R50-FPN		44.64 (+1.04)	77.45	-
Instant-Teaching	R50-FPN		**48.70** (+5.10)	**78.30**	**52.00** (+7.50)
Instant-Teaching*	R50-FPN		**50.00** (+6.40)	**79.20**	**54.00** (+9.50)
CSD [22]	R101-R-FCN	VOC12 & COCO	-	75.10	-
STAC [45]	R50-FPN		46.01 (+2.41)	79.08	-
Instant-Teaching	R50-FPN		**49.70** (+6.10)	79.00	**54.10** (+9.60)
Instant-Teaching*	R50-FPN		**50.80** (+7.20)	**79.90**	**55.70** (+11.20)

Table 2. Comparison of mAP for different semi-supervised methods on VOC07. We report the mAP at IoU=0.50:0.95 ($AP^{0.5:0.95}$), IoU=0.5 ($AP^{0.5}$) and IoU=0.75 ($AP^{0.75}$), which are the standard metrics for object detection [31, 7].

4.2. Results

We will make a detailed comparison with the supervised baseline and state-of-the-art SSOD methods, including CSD [22] and STAC [45]. The detailed results are summarized in Table 1 and Table 2.

As depicted in Table 1, our Instant-Teaching* outperforms state-of-the-art methods by a large margin under all experimental settings of the MS-COCO dataset. Specifically, for the 1% protocol, Instant-Teaching* improves mAP from STAC's 13.97 to 18.05, which achieves 4.08 mAP improvement; for the 2% protocol, Instant-Teaching* improves mAP from STAC's 18.25 to 22.45, which achieves 4.2 mAP improvement. Instant-Teaching* also brings significant improvement in mAP when there are more labeled data: 24.38 to 26.75 on the 5% protocol, 28.64 to 30.40 on the 10% protocol. For the 100% protocol, our Instant-Teaching* still achieves about 1.0 mAP improvement under the high benchmark of 39.21 mAP.

We also observe a similar trend on PASCAL VOC experiments. As depicted in Table 2, when compared with STAC, with VOC07 as labeled data and VOC12 as unlabeled data, our Instant-Teaching* improves mAP from 44.64 to 50.00, which demonstrates 5.36 absolute mAP improvement. When there are more unlabeled data introduced (the subset of MS-COCO), Instant-Teaching* can further improve mAP from STAC's 46.01 to 50.80. We also observe that the improvement of $AP^{0.75}$ of Instant-Teaching* is more prominent compared to that of $AP^{0.5}$. In other words, the improvement of mAP ($AP^{0.5:0.95}$) mainly comes from the improvement of predicted high-quality bounding boxes. We also perform ablation studies on our Instant-Teaching* with different backbones in the Appendix of our arxiv version, demonstrating the scalability of our method.

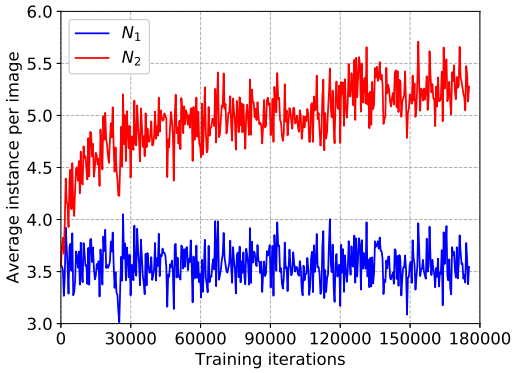

Figure 3. Changes in the number of annotations per image during training. N_1 refers human-annotated instances and N_2 refers total instances including human-annotated and model generated.

5. Ablation Study

5.1. Instant pseudo labeling

As shown in Fig. 3, we report the average number of annotated instances per image during each training iteration, in which N_1 denotes the number of only human-annotated instances and N_2 denotes the number of total instances including human-annotated and model generated (pseudo annotations). It can be observed that the number of high-quality pseudo annotations ($N_2 - N_1$) gradually increases during the training process. Namely, as the model converges, the quantity of high-quality pseudo annotations can be instantly improved.

From Table 3, we can also observe that at the protocol of 5% MS-COCO with 8× unlabeled data, Instant-Teaching improves mAP from STAC's 23.14 to 24.70 using only color jittering and Cutout [12] as the strong data augmentations. Without using more strong data augmentations, our Instant-Teaching already outperforms the state-of-the-art STAC method. These results prove that our instant pseudo labeling can finally achieve higher performance by continuously improving the pseudo annotations.

Methods	Strong data augmentations				mAP
	Color+Cutout	Geometric	Mixup	Mosaic	
STAC[45]	√	√			23.14
Instant-Teaching	√*				21.60 (-1.54)
	√				24.70 (+1.56)
	√		√		25.40 (+2.26)
	√			√	25.00 (+1.86)
	√		√	√	**25.60** (+2.46)

Table 3. Comparison of mAP of Instant-Teaching trained with various data augmentation methods at the protocol of 5% MS-COCO and 8× unlabeled data. √* denotes that we also apply strong augmentations "Color+Cutout" to unlabeled data in the first step during instant pseudo labeling.

Methods	Labeled Size	Unlabeled Size				
		1×	2×	4×	8×	Full
STAC[45]	5% COCO	19.81	20.79	22.09	23.14	24.38±0.12
Instant-Teaching		**23.60**	**24.30**	**25.30**	**25.60**	**25.60±0.14**
STAC[45]	10% COCO	25.38	26.52	27.33	27.95	28.64±0.21
Instant-Teaching		**28.80**	**29.00**	**29.20**	**29.50**	**29.53±0.17**

Table 4. Comparison of mAP of Instant-Teaching trained with various scales of unlabeled data on MS-COCO. $[n]\times$ denotes the scale of unlabeled data is $[n]$ times larger than that of labeled data.

5.2. Strong data augmentation

In weak-strong data augmentation scheme, the choice of strong augmentations directly affects the performance of the final SSOD model. We extend the strong augmentations of STAC from color jittering, geometric transformation and Cutout to include Mixup and Mosaic. Note that, we do not apply geometric transformation, mainly because the online geometric transformation of pseudo annotations is more complicated, and we leave it for future work.

As shown in Table 3, we first also apply strong augmentations (Color+Cutout) to unlabeled data in the first step during the pseudo labeling phase. This method gives us 1.54 mAP drop compared with STAC. The observation verifies our hypothesis, *i.e.*, the key of weak-strong data augmentation scheme lies in the difference between weak augmentations and strong augmentations. Furthermore, we find that using either Mixup or Mosaic can improve the performance of Instant-Teaching. Instant-Teaching can obtain the best performance by using Mixup and Mosaic data augmentations together, increasing mAP from 23.14 of STAC to 25.60. These observations indicate that our extended weak-strong data augmentations can further improve the performance of SSOD.

Note that we only use Mixup and Mosaic data augmentations for unlabeled data for a fair comparison with STAC.

5.3. Size of unlabeled data

In the field of semi-supervised object detection, the importance of the size of unlabeled data should not be ignored. Therefore in this section, we evaluate our method with 5% and 10% labeled data of MS-COCO while varying the size of unlabeled data from 1, 2, 4, and 8 times to that of the labeled data. The results are given in Table 4. We can observe that our method outperforms the state-of-the-art method STAC on all scales of unlabeled data. It is worth mentioning that, for both 5% and 10% labeled data, our Instant-Teaching method trained on 1× unlabeled data achieves 23.60 and 28.80 mAP respectively, which are even higher than STAC trained on 8× unlabeled data (23.14 and 27.95). This demonstrates that Instant-Teaching can efficiently leverage the unlabeled data.

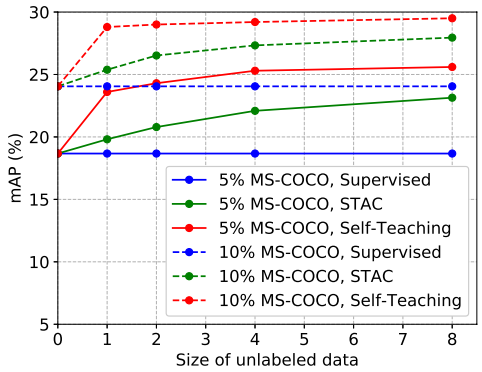

Figure 4. Comparison of mAP w.r.t. the size of unlabeled data.

τ	0.3	0.5	0.7	0.9
mAP (%)	26.30	27.70	28.70	29.80

Table 5. Comparison of mAP with various values of confidence threshold τ.

From Fig. 4 we can observe that our Instant-Teaching (without co-rectify) outperforms the supervised model and the state-of-the-art method STAC by a large margin. We also find that as the size of unlabeled data increases, both STAC and Instant-Teaching suffer a "ceiling effect": as the performance gets closer to the ceiling, the improvement becomes smaller.

5.4. Analysis of τ and λ_u

We analyze the effect of the confidence threshold τ and the unsupervised loss weight λ_u in this section. Our Instant-Teaching method is tested with 10% MS-COCO as labeled data and the remainder as unlabeled data. We first analyze the effect of τ. As shown in Table 5, we test Instant-Teaching with $\lambda_u = 1.0$ and $\tau \in \{0.3, 0.5, 0.7, 0.9\}$. The result shows that the model can achieve better performance by varying the threshold value τ from 0.3 to 0.9, which indicates $\tau = 0.9$ is a better choice to select high-quality pseudo annotations for unlabeled data.

When analyzing the effect of unsupervised loss weight λ_u, we fix $\tau = 0.9$ and vary the value of λ_u from 1/4 to 4. As can be seen in Fig. 5, Instant-Teaching achieves the best performance when $\lambda_u = 1.0$ and the mAP only slightly drop when λ_u becomes larger or smaller, which indicates

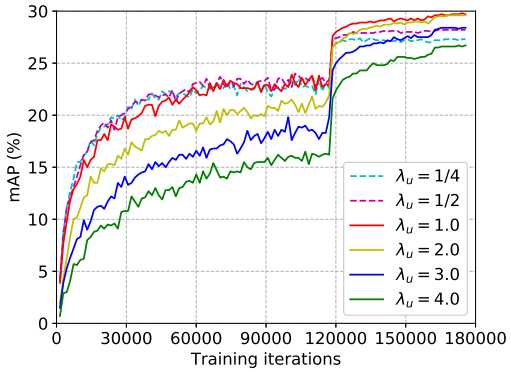

Figure 5. Comparison of mAP with various values of λ_u along training iterations.

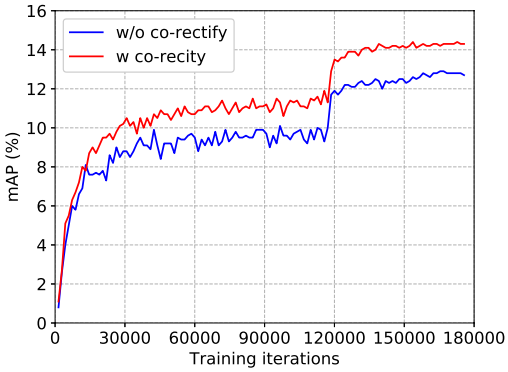

Figure 6. Comparison of mAP of generated pseudo annotations with different training iterations. The model is trained based on Instant-Teaching with and without co-rectify respectively.

that Instant-Teaching is relatively robust to λ_u.

We can also observe that Instant-Teaching achieves a higher mAP with a smaller value of λ_u (*e.g.*, 1/4, 1/2) during the early training iterations. In other words, in the early stages of training, the quality (quantity) of pseudo annotations is low, and the model should pay more attention to the labeled data. In this paper, we use a constant λ_u, and take the dynamic adjustment of λ_u as future work.

5.5. Analysis of co-rectify

We further propose a co-rectify scheme based on Instant-Teaching to alleviate the confirmation bias problem in SSL, which is shown in Fig. 1 (Instant-Teaching*). We analyze the effect of co-rectify using 1% labeled data and the remaining 99% as unlabeled data (1% MS-COCO protocol). The model is trained based on Instant-Teaching with and without our co-rectify scheme respectively. For evaluation, we test on 5k labeled data, which is randomly selected from the 99% unlabeled data of MS-COCO.

Note that, to verify whether the co-rectify scheme is able to generate more high-quality pseudo annotations, we compare the mAP of predicted pseudo annotations with score larger than 0.9 (same as τ during training). As shown in Fig. 6, we can directly observe that the model trained with co-rectify scheme obtains better performance faster, and is able to consistently improve the performance of our Instant-Teaching along the training iterations.

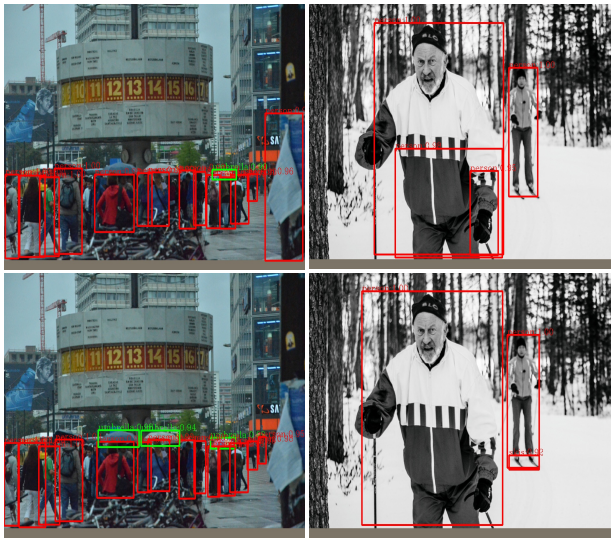

Figure 7. Visualization of predicted pseudo annotations whose confidence scores are larger than 0.9 for unlabeled data. The first row denotes the results of Instant-Teaching (without co-rectify) and the second row denotes the results of Instant-Teaching*.

In addition, we visualize the pseudo annotations for some unlabeled data in Fig. 7. The results are generated at the same training iteration (120k) with and without the co-rectify scheme respectively. We can observe that Instant-Teaching cooperated with co-rectify scheme can filter out some false predictions and generate more high-quality pseudo annotations at the same time. In summary, the co-rectify scheme is able to alleviate the confirmation bias problem and further improve the performance of Instant-Teaching.

6. Conclusion

In this paper, we revisit semi-supervised object detection (SSOD) and propose a simple and effective end-to-end SSOD framework — Instant-Teaching, which uses instant pseudo labeling with extended weak-strong data augmentations for teaching during each training iteration. Based on Instant-Teaching, we further propose a co-rectify scheme to alleviate the confirmation bias problem and further improve the performance. Extensive experiments on MS-COCO and PASCAL VOC demonstrate the significant superiority of our method. Although we evaluate with the two-stage detector Faster-RCNN [40], our proposed Instant-Teaching* is a general SSOD framework and is not restricted to the object detection models. This means Instant-Teaching* can be directly applied to other detectors, *e.g.*, one-stage detectors [32, 50], which we will leave for future work.

References

[1] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020. 2

[2] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Advances in neural information processing systems (NIPS)*, pages 3365–3373, 2014. 2

[3] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *International Conference on Learning Representations (ICLR)*, 2019. 1, 2

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5049–5059, 2019. 1, 2

[5] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on computational learning theory*, pages 92–100, 1998. 1

[6] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 4

[7] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6154–6162, 2018. 2, 6

[8] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. Semi-supervised learning. adaptive computation and machine learning. *MIT Press, Cambridge, MA, USA. Cited in page (s)*, 21(1):2, 2010. 1

[9] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 5

[10] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical data augmentation with no separate search. *arXiv preprint arXiv:1909.13719*, 2(4):7, 2019. 1, 2

[11] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 379–387, 2016. 2

[12] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 6

[13] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *International Conference on Computer Vision (ICCV)*, pages 6569–6578, 2019. 2

[14] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision (IJCV)*, 88(2):303–338, 2010. 2, 5

[15] Jiyang Gao, Jiang Wang, Shengyang Dai, Li-Jia Li, and Ram Nevatia. Note-rcnn: Noise tolerant ensemble rcnn for semi-supervised object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 9508–9517, 2019. 2

[16] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015. 2

[17] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 580–587, 2014. 2

[18] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017. 2

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1, 5

[20] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3588–3597, 2018. 2

[21] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 5070–5079, 2019. 2

[22] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 10759–10768, 2019. 2, 5, 6

[23] Jisoo Jeong, Vikas Verma, Minsung Hyun, Juho Kannala, and Nojun Kwak. Interpolation-based semi-supervised learning for object detection. *arXiv preprint arXiv:2006.02158*, 2020. 2

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012. 1

[25] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *International Conference on Learning Representations (ICLR)*, 2017. 1, 2

[26] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 734–750, 2018. 2

[27] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, 2013. 1, 2

[28] Yandong Li, Di Huang, Danfeng Qin, Liqiang Wang, and Boqing Gong. Improving object detection with selective

self-supervised self-training. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[29] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2117–2125, 2017. 2, 5

[30] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. 2

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision (ECCV)*, pages 740–755. Springer, 2014. 2, 5, 6

[32] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision (ECCV)*, pages 21–37. Springer, 2016. 2, 8

[33] Ishan Misra, Abhinav Shrivastava, and Martial Hebert. Watch and learn: Semi-supervised learning for object detectors from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3593–3602, 2015. 2

[34] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 41(8):1979–1993, 2018. 1, 2

[35] Siyuan Qiao, Wei Shen, Zhishuai Zhang, Bo Wang, and Alan Yuille. Deep co-training for semi-supervised image recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 135–152, 2018. 1

[36] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari, and Kaiming He. Data distillation: Towards omni-supervised learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4119–4128, 2018. 2

[37] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. Semi-supervised learning with ladder networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3546–3554, 2015. 1

[38] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016. 2

[39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 91–99, 2015. 2

[40] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 39(6):1137–1149, 2017. 5, 8

[41] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1163–1171, 2016. 1, 2

[42] H Scudder. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*, 11(3):363–371, 1965. 1

[43] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015. 1

[44] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2, 4

[45] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020. 1, 2, 3, 4, 5, 6, 7

[46] Peng Tang, Chetan Ramaiah, Ran Xu, and Caiming Xiong. Proposal learning for semi-supervised object detection. *arXiv preprint arXiv:2001.05086*, 2020. 2

[47] Yuxing Tang, Josiah Wang, Boyang Gao, Emmanuel Dellandréa, Robert Gaizauskas, and Liming Chen. Large scale semi-supervised object detection using visual and semantic knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2119–2128, 2016. 2

[48] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1195–1204, 2017. 1, 2, 5

[49] Antti Tarvainen and Harri Valpola. Weight-averaged consistency targets improve semi-supervised deep learning results. *International Conference on Neural Information Processing Systems (NIPS)*, 2017. 1

[50] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 9627–9636, 2019. 2, 8

[51] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2

[52] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10687–10698, 2020. 1, 2

[53] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *International Conference on Learning Representations (ICLR)*, 2018. 4