Positive-Unlabeled Data Purification in the Wild for Object Detection

Jianyuan Guo[1,2], Kai Han[1], Han Wu[2], Chao Zhang[3*], Xinghao Chen[1],
Chunjing Xu[1], Chang Xu[2], Yunhe Wang[1*]

[1] Noah's Ark Lab, Huawei Technologies. [2] School of Computer Science, Faculty of Engineering, University of Sydney.
[3] Key Lab of Machine Perception (MOE), Dept. of Machine Intelligence, Peking University.

{jianyuan.guo, kai.han, yunhe.wang}@huawei.com; chzhang@cis.pku.edu.cn c.xu@sydney.edu.au

Abstract

Deep learning based object detection approaches have achieved great progress with the benefit from large amount of labeled images. However, image annotation remains a laborious, time-consuming and error-prone process. To further improve the performance of detectors, we seek to exploit all available labeled data and excavate useful samples from massive unlabeled images in the wild, which is rarely discussed before. In this paper, we present a positive-unlabeled learning based scheme to expand training data by purifying valuable images from massive unlabeled ones, where the original training data are viewed as positive *data and the unlabeled images in the wild are* unlabeled *data. To effectively utilized these purified data, we propose a self-distillation algorithm based on hint learning and ground truth bounded knowledge distillation. Experimental results verify that the proposed positive-unlabeled data purification can strengthen the original detector by mining the massive unlabeled data. In particular, our method boosts the mAP of FPN by +2.0% on COCO benchmark.*

1. Introduction

As a fundamental but challenging task in computer vision, object detection has attracted increasing attention from both academia and industry due to its significant role in numerous fields, including autonomous driving and surveillance video analysis. Recently, many object detection methods [4, 42, 28, 31, 41, 15, 9, 25, 14] have been proposed and achieved great progress, pushing the related applications forward to the real world. However, most existing methods rely on a great volume of fine annotated images as training data to obtain a high-performance detector as illustrated in Figure 1(c) and keep benefiting from more available labeled data [30, 10, 24]. When the labeled data are scarce, detectors are easy to severely overfit and fail to generalize. However, it is a time-consuming and expensive process to build a qualified object detection dataset. According to [45, 50], it takes annotators about 42 seconds to perform one annotation task after they receive a thorough training on that project. A label composed of bounding box coordinates and corresponding category needs to be annotated for multiple objects in one image, each with a different label.

Many works have observed heavy dependence of object detection task on huge amount of training data. Some of them try to tackle it from few-shot learning perspective [59, 22, 47, 11, 12, 53], which eliminates this phenomenon by transferring knowledge from data-abundant base classes to the data-scarce novel classes as illustrated in Figure 1(a). However, the inherent data imbalance problem proposes an even greater challenge for these novel methods, *e.g.* the "train2017" set of COCO benchmark [30] contains 26k annotation boxes for category "person" while only 198 for "hair drier". Such cases of data scarce problem can hardly be handled by few-shot learning.

Another trend of eliminating label dependence in object detection belongs to semi-supervised learning [44, 40, 46, 48, 34, 49, 52]. This line of work either generates labeled/unlabeled data by splitting a fully annotated dataset, or directly using unlabeled data that have the same distribution with labeled ones as shown in Figure 1(b). The former usually needs a predefined data scale and inevitably limit the performance of detectors, while the latter often results in the largely neglected problem of dataset bias, which is a well known limitation of semi-supervised learning [35].

In contrast to the existing methods which only utilize limited data, this paper aims at both improving the performance of object detectors and eliminating label dependence by exploiting massive unlabeled data from the wild. As shown in Figure 1(d), the detector not only exploits as much well annotated data (*e.g.* COCO [30] and VOC [10]) as possible, but also is provided with unlimited unlabeled data from the wild (*e.g.* Flickr [19], ImageNet[1] [5] and

*Corresponding author.

[1] Although ImageNet has image-level labels, there is no extra box-level annotation, we choose to ignore all labels and treat the images as unlabeled.

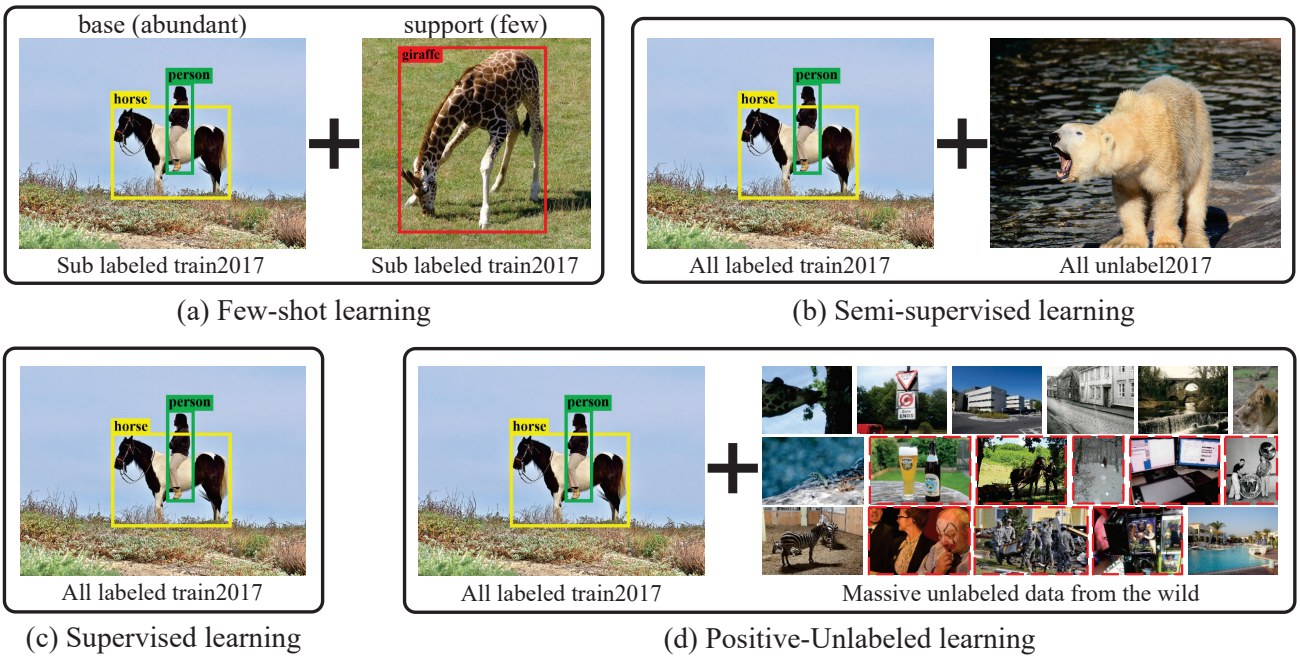

Figure 1. Different types of object detection setting and corresponding training images. The "train2017" and "unlabel2017" indicate the 118k images and 123k images provided by COCO benchmark [30], respectively. Images marked with red-dotted boxes denote the positive images from the unlabeled set.

Open Images [24]). In such setting, data from the wild may contain potentially *positive* images, in which objects are the same with that appeared in original detection benchmark, and these data can further help enlarge the training set. Nevertheless, the trained detector could also be contaminated by other *negative* images that contain unrelated objects from the unlabeled data. A popular solution to utilize the unlabeled data is the self-training based semi-supervised methods [40, 44]. However, directly training the detector under the self-training framework [26] which uses a teacher model to generate pseudo labels on unlabeled images may inject extra noises to the augmented dataset. To tackle this challenge, we propose a two-stage framework to adaptively utilize the positive data and meanwhile, avoid noise from negative data for improving the detector's performance. Firstly, a PU (Positive-unlabeled) classifier for data purification is trained with the given labeled data and massive unlabeled data from the wild. Then the expanded training set is constructed by combining the original training data and the *positive* data from massive unlabeled data purified by the trained PU classifier. In addition, we develop a self-distillation based training scheme to utilize the expanded dataset where the teacher's architecture is the same with the student's. We demonstrate the efficiency of our positive-unlabeled data purification for object detection (PUDet) on COCO benchmark [30] in general two-stage detection framework FPN [28]. In particular, the proposed method can improve the mAP of baseline FPN from 37.4% to 39.4% without annotating any data manually.

2. Related Work

Fully-supervised object detection. Object detection is considered as one of the most challenging problems in computer vision which aims at determining *what* and *where* the object is when given an image. Following the success of CNNs, promising improvements in accuracy have been made in object detection with sufficient amount of annotated data [41, 31, 29, 36, 25, 9, 42, 60, 15, 4, 28, 38]. These methods take advantage of the well-annotated object detection datasets and concentrate on consistently revising detector architecture or loss function to achieve a satisfying performance. On top of that, increasing attention has been paid to employ unlabeled training data in a semi-supervised way to improve object detectors. In this work, we explore a realistic situation by using unlabeled data to improve the detection performance.

Semi-supervised object detection. Semi-supervised learning [44, 55, 54] is of great significance by using readily available unlabeled data to improve supervised learning tasks. There are several works exploring semi-supervised object detection [39, 20, 51, 49]. Generally, these methods can be divided into self-training based and consistency regularization based approaches. Jeong *et al*. [20] uses consistency constraints to let the predictions of unlabeled images be consistent with their flipped counterparts. Tang *et al*. [51] combines self-supervised learning and consistency losses to learn proposal features from both labeled and unlabeled data. Radosavovic *et al*. [39] ensembles predictions

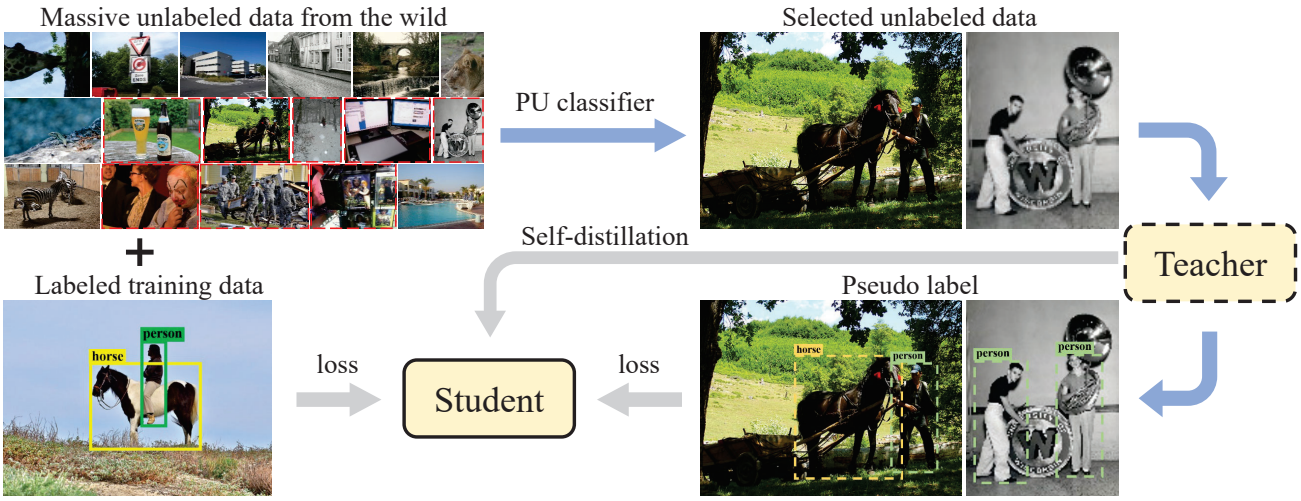

Figure 2. The proposed two stage framework for object detection. We first train a PU classifier to select useful images from the massive unlabeled data from the wild. Then we utilize the self-training based method and propose self-distillation to promote the detector's performance.

from multiple transformations of unlabeled data, using a single model to automatically generate new training annotations. Sohn *et al*. [49] uses a pretrained detector to generate pseudo labels on unlabeled images. In training phase, they leverage abundant augmentation strategies, *e.g*. color transformation, global geometric transformation, box-level geometric transformation and cutout, to train a better detector. More specifically, CSD [20] takes "VOC07" as labeled data and "VOC12" as unlabeled data, both provided by VOC benchmark [10]. [39, 51, 49] take "train2017" as labeled data and "unlabel2017" as unlabeled data, both provided by COCO benchmark [30].

Webly-supervised learning. There are several works that have explored to use web-crawled images or videos to learn CNN features for different tasks. Chen *et al*. [3] collects both easy and hard web images to train CNNs for classification and detection tasks. Wei *et al*. [57] and Jin *et al*. [21] exploit web images to progressively learn features for semantic segmentation in a simple-to-complex manner. However, all these methods require extra human efforts to divide web images into different groups. Luo *et al*. [32] leverages several models to distill the student together with color information to select webly data for salient object detection.

We are particularly interested in a natural question: Are previous methods applicable in "real-world" settings? We argue that conventional semi-supervised learning does not address this question in a satisfying way. In above setting, the unlabeled data have similar distribution with the labeled data, in other words, the unlabeled images only contain objects of the same categories that have appeared in the labeled images [35]. They only need to assign coordinate and category generated by teacher detector to accomplish the data purification. In this work, our intuition is to use the massive web images as unlabeled set, which creates a closer scenario to realistic semi-supervised object detection. Those unlabeled images can also contain other puzzling objects (see more cases in Figure 3), so that directly leveraging the self-training based method in our setting can lead to suboptimal results. To this end, we propose a positive-unlabeled learning based scheme to consummate the data purification framework and a self-distillation scheme to further boost the performance of object detector for targeting at this challenging task.

3. PUDet

3.1. Positive-unlabeled Classifier for Training Data Purification

Preliminary. There are massive web images in the wild for different tasks (*e.g*. ImageNet, Flickr and Open Image) which have not been exploited for object detection task. To probe into the utilization of these data, we first propose to achieve data purification by positive-unlabeled (PU) learning [23, 27, 8, 58], we treat typical classification benchmark ImageNet as the massive unlabeled data, and "train2017" from COCO benchmark as the labeled data.

Problem setting. The goal of data purification is to screen out the images containing the positive objects from massive unlabeled ones. For this purpose, a classifier $\mathcal{N}_{pu}$ is trained using given labeled "train2017" and unlabeled ImageNet. Given $x \in \mathcal{X} \subseteq \mathbb{R}^d$ as input image, $y \in \mathcal{Y} = \{+1, -1\}$ as output label, n_l and n_u are the numbers of labeled set L and unlabeled set U, respectively. In this section, the positive-unlabeled setting refers to the "labeled" and "unlabeled" data.

PU classification. We first review the formulation of positive-negative (PN) classification. Consider a binary classification problem from x to y, and given three sets of samples: labeled set L, negative set N, and unlabeled set U, then we have:

$$\mathcal{X}_{\mathrm{p}} = \{x_i^p\}_{i=1}^{n_l} \overset{\text{i.i.d.}}{\sim} p_{\mathrm{p}}(x) = p(x \mid y = +1), \tag{1}$$

$$\mathcal{X}_{\mathrm{n}} = \{x_i^n\}_{i=1}^{n_n} \overset{\text{i.i.d.}}{\sim} p_{\mathrm{n}}(x) = p(x \mid y = -1). \tag{2}$$

$$\mathcal{X}_{\mathrm{u}} = \{x_i^u\}_{i=1}^{n_u} \overset{\text{i.i.d.}}{\sim} p(x) = \pi_{\mathrm{p}} p_{\mathrm{p}}(x) + \pi_{\mathrm{n}} p_{\mathrm{n}}(x) \tag{3}$$

where $\pi_{\mathrm{p}} = p(y = +1)$ indicates the probability of class prior and $\pi_{\mathrm{n}} = p(y = -1) = 1 - \pi_p$. We assume π_{p} as known in all our experiments as it can be estimated from labeled and unlabeled data [33, 18]. We denote $g : \mathbb{R}^d \to \mathbb{R}$ as an arbitrary decision function, and $\ell : \{+1, -1\} \to \mathbb{R}$ as an arbitrary loss function such that $\ell\{g(x), y\}$ indicates the loss computed by the input x and the ground truth y. In normal PN learning, $\mathcal{X}_{\mathrm{n}}$ rather than $\mathcal{X}_{\mathrm{u}}$ would be available and the decision function g can be optimized from $\mathcal{X}_{\mathrm{p}}$ and $\mathcal{X}_{\mathrm{n}}$:

$$\widetilde{R}_{\mathrm{pn}}(g) = \pi_{\mathrm{p}} \widetilde{R}_{\mathrm{p}}^{+}(g) + \pi_{\mathrm{n}} \widetilde{R}_{\mathrm{n}}^{-}(g) \tag{4}$$

where $\widetilde{R}_{\mathrm{p}}^{+}(g) = \frac{1}{n_l} \sum_{i=1}^{n_l} \ell(g(x_i^p), +1)$ and $\widetilde{R}_{\mathrm{n}}^{-}(g) = \frac{1}{n_n} \sum_{i=1}^{n_n} \ell(g(x_i^n), -1)$. However, it is initially unknown which images are negative, in the given set of "train2017" and ImageNet. In this positive-unlabeled (PU) setting, $\mathcal{X}_{\mathrm{n}}$ is unavailable, the key idea is to utilize unlabeled ImageNet data to evaluate the risk for negative samples in the PU risk. As shown in [8, 7], $\pi_{\mathrm{n}} p_{\mathrm{n}}(x) = p(x) - \pi_{\mathrm{p}} p_{\mathrm{p}}(x)$, and $R_{\mathrm{n}}^{-}(g)$ can be approximated indirectly by $\pi_{\mathrm{n}} R_{\mathrm{n}}^{-}(g) = R_{\mathrm{u}}^{-}(g) - \pi_{\mathrm{p}} R_{\mathrm{p}}^{-}(g)$. Thus the decision function g can be optimized as following:

$$\widetilde{R}_{\mathrm{pu}}(g) = \pi_{\mathrm{p}} \widetilde{R}_{\mathrm{p}}^{+}(g) - \pi_{\mathrm{p}} \widetilde{R}_{\mathrm{p}}^{-}(g) + \widetilde{R}_{\mathrm{u}}^{-}(g) \tag{5}$$

where $\widetilde{R}_{\mathrm{p}}^{-}(g) = \frac{1}{n_l} \sum_{i=1}^{n_l} \ell(g(x_i^p), -1)$ and $\widetilde{R}_{\mathrm{u}}^{-}(g) = \frac{1}{n_u} \sum_{i=1}^{n_u} \ell(g(x_i^u), -1)$. Nevertheless, the PU classifier in Eq. 5 relies on unbiased risk estimators. According to [23], we adjust the estimator to alleviate the overfitting problem as following:

$$\widetilde{R}_{\mathrm{pu}}(g) = \pi_{\mathrm{p}} \widetilde{R}_{\mathrm{p}}^{+}(g) + \max\{0, \widetilde{R}_{\mathrm{u}}^{-}(g) - \pi_{\mathrm{p}} \widetilde{R}_{\mathrm{p}}^{-}(g)\}. \tag{6}$$

The training set T of the PU classifier can be formulated as:

$$T = L \cup U = \{x_l, +1\}_{p=1}^{n_l} \cup \{x_u, y_u\}_{u=1}^{n_u}. \tag{7}$$

Data purification. Given the labeled set L (*e.g.* "train2017" from COCO) and unlabeled set U (*e.g.* ILSVRC 2012), we firstly train a PU classifier $\mathcal{N}_{pu}$ based on Eq. 6. Then we can utilize $\mathcal{N}_{pu}$ to select images containing the objects that have appeared in L from the massive unlabeled dataset U, and construct the selected unlabeled dataset U^P for following object detector training. After selecting the unlabeled data from the massive data, we develop a self-distillation scheme for training object detector, via the self-training way (pseudo label). First, we train a teacher detector $\mathcal{D}_t$ using all available labeled data (*e.g.* "train2017" from COCO). Then pseudo labels of selected unlabeled images are generated by the teacher detector. Last, we train the student detector using both labeled and unlabeled data. We also use a threshold $\tau = 0.5$ to control the quality of pseudo labels in object detection inspired by [48, 49]. High threshold can filter lower-quality pseudo labels out, while lower threshold can bring more instances during training phase. Considering that the data purification has excluded the noisy images, a lower threshold τ is helpful to generate more instances which are comprised of box coordinates and corresponding categories. A more specific procedure is depicted in Algorithm 1.

Algorithm 1 Data purification and corresponding pseudo label generation.

Input: Network $\mathcal{N}_{pu}$, labeled set L, unlabeled set U
Step 1: Train the PU Classifier $\mathcal{N}_{pu}$
while not converge **do**
    Collect mini batch $\{x_l, x_u\}^N$ from $L \cup U$
    Optimize $\mathcal{N}_{pu}$ following Eq. 6;
end while
Step 2: Construct selected unlabeled dataset
Utilizing $\mathcal{N}_{pu}$ to select images from U, construct set U^p
Step 3: Generate pseudo label
Train a teacher detector $\mathcal{D}_t$ on labeled set L
Generate pseudo labels on U^p using $\mathcal{D}_t$
Obtain extended dataset $L \cup U^p$;
Output: Teacher detector $\mathcal{D}_t$, dataset $L \cup U^p$

3.2. Self-distillation on Expanded Data

In this work, we adopt the FPN [28] as object detection framework, which is composed of four modules: (i) a backbone for extracting semantic features, *e.g.* ResNet-50 [16]; (ii) a feature pyramid network for fusing multi-level features; (iii) a region proposal network (RPN) for generating region proposals; and (iv) a head that contains two branches for object classification and bounding box regression. As shown in Figure 2, we use FPN for both teacher and student detector to guarantee the exactly same architecture, with the only difference that the teacher detector has been pretrained. In Sec. 3.1, we first train the teacher detector on the original set L to generate the pseudo labels for the selected unlabeled data U^P. The student detector is then trained on the expanded dataset $L \cup U^P$. In the training phase of the student detector, we propose the self-distillation scheme to

further exploit the pretrained teacher detector.

We propose to distill features from *feature pyramid network* and *head* to fully exploit the pretrained teacher detector. Both the RPN and the head take the output of feature pyramid network as input features, so the quality of these features is critical for object detectors to achieve higher performances. We adopt the hint based learning [43, 13] to transfer the knowledge inside the feature representation from teacher detector to student detector. And we apply knowledge distillation to the classification branch in the head, by using teacher detector's classification logits to guide the student. Then we use the ℓ_1 loss to distill the bound box regression branch in the head. p and t are used to indicate the predictive probability and box coordinate, respectively. The overall learning objective can be written as following:

$$\mathcal{L} = \mathcal{L}_{det} + \gamma \mathcal{L}_{hint} + \lambda \mathcal{L}_{kd}, \tag{8}$$

$$\begin{aligned} \mathcal{L}_{det}(p_i, t_i, p_i^*, t_i^*) = & \frac{1}{N_{cls}} \sum_i \mathcal{L}_{cls}(p_i, p_i^*) \\ & + \frac{1}{N_{reg}} \sum_i p_i^* \mathcal{L}_{reg}(t_i, t_i^*), \end{aligned} \tag{9}$$

$$\begin{aligned} \mathcal{L}_{kd}(p_i', t_i', p_i^t, t_i^t, p_i^*, t_i^*) = & \mathcal{L}_{kdc}(p_i', p_i^t, p_i^*) \\ & + \mathcal{L}_{kdr}(t_i', t_i^t, t_i^*), \end{aligned} \tag{10}$$

where the hyper-parameters γ and λ are used to control the different loss items. $\mathcal{L}_{det}$ is the regular detection training loss. And the knowledge distillation losses $\mathcal{L}_{hint}$, $\mathcal{L}_{kdc}$ and $\mathcal{L}_{kdr}$ will be described in the following.

Multi-layer hint for feature adaption. Simplest distillation transfers knowledge by using a soft target distribution on the final output. However, Adriana *et al.* [43] demonstrates that using the intermediate feature representation from teacher network as *hint* can improve the final performance of student network. And considering that both the RPN and the head in object detectors need to use the intermediate features from feature pyramid network for later classification and regression, we propose to utilize hint learning to distill the multi-layer features generated by feature pyramid network. Define $\mathcal{S} = [\mathcal{S}^{(1)}, \mathcal{S}^{(2)}, \mathcal{S}^{(3)}, \mathcal{S}^{(4)}, \mathcal{S}^{(5)}]$ as student detector's output and $\mathcal{T} = [\mathcal{T}^{(1)}, \mathcal{T}^{(2)}, \mathcal{T}^{(3)}, \mathcal{T}^{(4)}, \mathcal{T}^{(5)}]$ as teacher detector's output. We select the imitation region on each layer following [56] and perform hint learning accordingly, with the binary mask denoted as $I_{h,w}$. Although the architectures of student and teacher are the same, we still add five adaptation layers to each feature for better generation according to [1]. The hint learning objective is to minimize:

$$\mathcal{L}_{hint} = \frac{1}{2N_I} \sum_{i=1}^{5} \sum_{h=1}^{H} \sum_{w=1}^{W} \sum_{c=1}^{C} I_{h,w} (\phi_i(\mathcal{S}_{h,w,c}^{(i)}) - \mathcal{T}_{h,w,c}^{(i)})^2 \tag{11}$$

where $N_I = C \sum_{i=1}^{5} \sum_{h=1}^{H} \sum_{w=1}^{W} I_{h,w}$, which is the normalization factor. $\phi_i(\cdot)$ is the i-th adaptation layer, which is a 3×3 convolution layer in our implementation.

Ground truth bounded distillation for classification. Here we propose the ground truth bounded distillation for guiding the classification branch of the student detector. The RPN will generate region proposals with different prior size and pool corresponding features accordingly, so that the classification branch in head will then classify the proposals based on these features. To make sure teacher and student will classify the same region, we feed the region proposals generated by teacher detector into the student's head and let the student generate extra p' based on teacher's proposals. Given the logits z of each proposal, we distil the knowledge by a temperature T for teacher and student as:

$$p'(c \mid x, \theta) = \frac{exp(z_c'/T)}{\sum_{j=1}^{C} exp(z_j'/T)}, c \in \mathcal{Y} \tag{12}$$

$$p^t(c \mid x^t, \theta^t) = \frac{exp(z_c^t/T)}{\sum_{j=1}^{C} exp(z_j^t/T)}, c \in \mathcal{Y} \tag{13}$$

where θ and θ^t denote the parameters of the student and teacher detectors, respectively. z' and z^t indicate the logits (based on the same region proposals) of student and teacher detectors, respectively. $\mathcal{Y} = \{1, 2, ..., C\}$ is the classes of detection benchmark. T is a temperature that controls the distribution of output labels. To avoid introducing interference caused by teacher's wrong judgements to the student, we use the ground truth of categories to bound the process of distillation. To quantify the knowledge between student and teacher detectors in their classification outputs, we use the Kullback Leibler (KL) divergence written as:

$$\mathcal{L}_{kdc} = \sum_i q_i T^2 \sum_{c=1}^{C} p^t(c \mid x, \theta^t) \log \frac{p^t(c \mid x, \theta^t)}{p(c \mid x, \theta)} \tag{14}$$

Where $q_i = 1$ if the teacher correctly classifies the proposal. We multiply the knowledge distillation loss by T^2 to ensure the scale of the gradient magnitudes. Considering that object detection is a challenging task where the prediction error is already high, a larger T may introduce more noise to the student detector [17, 61, 6]. Empirically, we set $T = 2$ in all our experiments.

Ground truth bounded distillation for bounding box regression. Classification and bounding box regression are two core tasks in object detection; thus training a better regression branch is critical for a higher performance. In addition to the above distillation for classification branch, we also propose to distill the bounding box regression branch of the student detector bounded by ground truth. We use t' to indicate the extra parameterized coordinates of the predicted bounding box based on the teacher's region proposals. Given the ground truth t^* and teacher detector's output

t^t, the objective can be written as:

$$\mathcal{L}_{kdr} = \frac{1}{\sum_i q_i} \sum_i q_i \min\{\|t'_i - t^t_i\|_1, \|t'_i - t^*_i\|_1\} \quad (15)$$

where q_i is the binary label of i-th region proposal with respect to the foreground boxes. Instead of only distilling student by the soften output from teacher detector, we penalize the student by teacher's output when the ℓ_1 loss between them is smaller than that between student and ground truth, and vice versa. This knowledge distillation loss can encourage the student to be better than the teacher by preventing the case that teacher's regression outputs provide sub-optimal guidance.

4. Experiments

We investigate the efficiency of the proposed data purification method and self-distillation algorithm by conducting elaborate experiments on COCO [30] benchmark, which contains 80 object classes and 118k labeled images for training (train2017). We evaluate the detection performance on COCO minival (val2017) by the Average Precision (AP) with different IoU thresholds from 0.5 to 0.95 with an interval of 0.05, *i.e.*, mAP. We also report the AP with thresholds of 0.5 and 0.75, *i.e.*, AP_{50} and AP_{75}, as well as AP on objects with small, medium and large sizes, *i.e.*, AP_{S}, AP_{M} and AP_{L}.

4.1. Implementation Details

Our implementation is based on mmdetection [2] with Pytorch framework [37]. For data purification, we use ImageNet dataset [5] as the massive unlabeled data pool, which contains over 1.2M images. We refer to it as the unlabeled set U with n_u=1.2M in our experiment, and refer to the "train2017" in COCO as the labeled set L with n_l=118k, according to Eq. 7. We use the proposed PU classifier to select 118k images from ImageNet (U^P) together with "train2017" to evaluate the efficiency of our data purification. We also randomly select 236k images from ImageNet (U^R) to serve as a comparison. The π_{p} in Eq. 3 can be calculated by $\pi_{\mathrm{p}} = p(y = +1) = 118k/1280k \approx 0.1$ according to [23]. To perform the data purification, we first train a ResNet-50 based FPN on "train2017", the backbone of which is used as the PU classifier $\mathcal{N}_{pu}$. Then we train the $\mathcal{N}_{pu}$ on gathered dataset $T = L \cup U$. The model is trained for 100 epochs and optimized by SGD. We set the learning rate as 0.01 and divide it by 10 after 60 epochs. The momentum and weight decay is set to be 0.001 and 0.9, respectively. The λ and γ in Eq. 8 are set to be 1 empirically.

We choose the ResNet-50 based FPN [28] as our baseline detection model. Labeled and selected unlabeled data are gathered together to train the detector. All input images are resized for a fixed number of 800 pixels for the shorter side. We use SGD optimizer with a batch size of 32 images to train the detector for 12 epochs, known as $1\times$ schedule. The initial learning rate is 0.04 and is divided by 10 at the 8^{th} and 11^{th} epoch. The momentum is set to 0.9 and the weight decay is 0.0001. We use horizontal flipping as data augmentation for both labeled and unlabeled images, and color transformation is applied for unlabeled images additionally. No data augmentation is adopted during testing phase. The model is trained on 8 NVIDIA Tesla V100 GPUs.

4.2. Main Results

We report the comparisons between our method and fully supervised baselines, as well as two recently proposed methods for deep semi-supervised learning [49, 51] in Table 1. The first row indicates the results of FPN [28] under fully supervised setting, and the second row is the result after adding our self-distillation scheme. The last three rows show the results of training the detector by self-distillation on $L \cup U^P$. As can be observed, our proposed data purification can obviously bring better performances compared to the baseline method. These results show that training detectors on both labeled and unlabeled data can easily outperform training detectors on labeled data only, confirming our intuition to boost detector without extra annotations. In addition, our data purification surpasses the randomly selected one by 1.2% even though the randomly selected data have more images and boxes during training phase, which demonstrates that the PU classifier can filter out dirty images and select high quality positive images from massive unlabeled dataset.

PLLD [51] proposes to learn proposal features from RPN by adding random noise to original feature maps, and use the unlabeled data provided by COCO dataset to further boost the two-stage detector. Different with PLLD, we concentrate on exploring semi-supervised learning under more complicated situation that the unlabeled data can have large amount of dirty images. And we think the features from neck and head are more decisive for object detection. Results demonstrate that our approach outperforms PLLD by 1% in mAP with less unlabeled data. STAC [49] leverages abundant data augmentation strategies to improve the performance upon semi-supervised learning. In particular, we achieve better results under a more complicated scenario, and only use horizontal flipping and color transformation to the unlabeled data.

To evaluate the effects of our proposed PU classifier for data purification, we use a simple threshold on the confidence of classes predicted by the teacher detector on the unlabeled set as a baseline. It can be seen from Table 1 that the expanded training data selected by PU classifier outperforms its randomly selected counterpart by 1.2% mAP on COCO minival with only half of the extra images. This

Table 1. Comparisons between training with proposed data purification and random selection manner on COCO minival. 'thr' means threshold for pseudo label generating. The predicted box is considered as pseudo label if its confidence score is higher than 'thr'. '# boxes' is the number of instances in unlabeled dataset accordingly.

Training data	n_u	thr	# boxes	mAP	AP_{75}	AP_{50}	AP_S	AP_M	AP_L
Positive (baseline)	-	-	-	37.4	58.3	40.5	21.8	41.0	47.8
+ Self-Distillation	-	-	-	38.4	59.1	41.9	22.2	42.0	50.1
PLLD [51]	123k	-	-	38.4	59.7	41.7	22.6	41.8	50.6
STAC [49]	123k	0.5	-	39.2	-	-	-	-	-
+ Randomly selected	236k	0.5	455k	37.4	57.8	40.8	21.9	41.0	48.3
	236k	0.7	301k	38.0	58.4	41.3	21.7	41.6	48.9
	236k	0.9	182k	38.2	58.6	41.7	22.2	42.0	49.4
+ PU selected	118k	0.5	444k	**39.4**	**60.1**	**42.9**	**23.4**	**43.0**	**50.9**
	118k	0.7	296k	39.3	59.9	42.8	23.2	42.8	**50.9**
	118k	0.9	175k	39.0	59.7	42.7	22.8	42.7	50.6

Table 2. Evaluation results of different n_u on COCO minival.

Unlabeled data	n_u	thr	# boxes	mAP
Randomly Selected	118k	0.5	304K	37.7
Randomly Selected	118k	0.9	103K	37.9
Randomly Selected	1.3M	0.5	3287K	36.6
Randomly Selected	1.3M	0.9	1117K	37.0
PU Selected	236k	0.5	802K	39.5
PU Selected	236k	0.9	299K	39.3

result demonstrates the necessity and effectiveness of the proposed PU classifier.

In addition, we find that randomly selecting unlabeled data and using PU classifier to select unlabeled data achieve their best results at different thresholds. The detector trained by labeled and randomly selected data achieves 38.2% mAP when the threshold is set as 0.9, and the mAP decreases to 37.4% when the threshold is set as 0.5. Even though lower threshold can bring more instances during training phase, the randomly selected data may contain other dirty images that could withhold the pretrained FPN of generating high quality pseudo labels. Detector trained by labeled and PU selected data achieves 39.4% mAP when the threshold is set as 0.5, which indicates that our data purification is effective enough to filter out dirty images. Lower threshold can bring more instances thus induces a better performance. Another interesting fact is that the 118k images selected by PU classifier have similar number of instances (# boxes in Table 1) compared to 236k randomly selected images. This phenomenon also demonstrates the efficiency of our proposed data purification method.

We also report the result of training with different numbers of unlabeled images as shown in Table 2. We can find that using 118K randomly selected data achieves 37.9% mAP while using all unlabeled data only ahieves 37.0% mAP. These comparisons indicate that more unlabeled data with different distributions inject more severe bias, leading to a worse performance. And this result is different with the recently proposed method [62], one main reason is the different strategies on constructing mini-batch. [62] uses a batch size of 512 with 256 from COCO (118K images) and 256 from the pseudo dataset (i.e. 1.3M images in ImageNet), and the "epoch" COCO is trained for is different from that of ImageNet. In our setting, we first construct mini-batch using only ImageNet images then construct mini-batch using only COCO images.

4.3. Ablation Study

In this section, we conduct the ablation study on the proposed self-distillation loss and present the visualization of selected unlabeled images by two different methods. The study analyzes the impact of hint learning on FPN features, ground truth bounded distillation on the classification branch and the regression branch.

Self-distillation loss. We investigate the proposed self-distillation loss, which is depicted in Eq. 8, in two settings: (i) student detector has smaller backbone compared to teacher detector, *i.e.* student is ResNet-18 based FPN and teacher is ResNet-50 based FPN and (ii) student detector has the same overall architecture as teacher detector. The results are shown in Table 3. In the first setting, hint learning improves result of the baseline model from 33.4% to 34.6%. Distilling the classification branch further boosts the mAP by 0.7%, and distilling the regression branch finally helps the detector achieve an mAP of 35.8% on COCO minival set. In the second setting, the proposed self-distillation loss enhances the ResNet-50 based FPN by 1.0% mAP, which validates that our proposed self-distillation loss is of significant benefit for student detector to fully exploit the potential of pretrained teacher detector.

Quality of data purification. We visualize the selected images from unlabeled set and the corresponding pseudo labels with scores predicted by the teacher detector in Figure 3. It can be seen that if we directly use self-training based algorithm that simply use a threshold to select images on the massive unlabeled data, the pretrained teacher

Table 3. Ablation study of the proposed self-distillation on COCO "val2017" (minival). All models are trained on "train2017" only.

Model	Teacher	hint	kdc	kdr	mAP	AP_{50}	AP_{75}
Res18 FPN (baseline)	-	-	-	-	33.4	53.6	35.9
Res18 FPN	Res50 FPN	✓			34.6	54.6	37.2
		✓	✓		35.3	55.4	38.2
		✓	✓	✓	**35.8**	**56.0**	**38.8**
Res50 FPN (baseline)	-	-	-	-	37.4	58.3	40.5
Res50 FPN	Res50 FPN	✓			37.8	**59.1**	41.0
		✓	✓		38.2	58.7	41.8
		✓	✓	✓	**38.4**	**59.1**	**41.9**

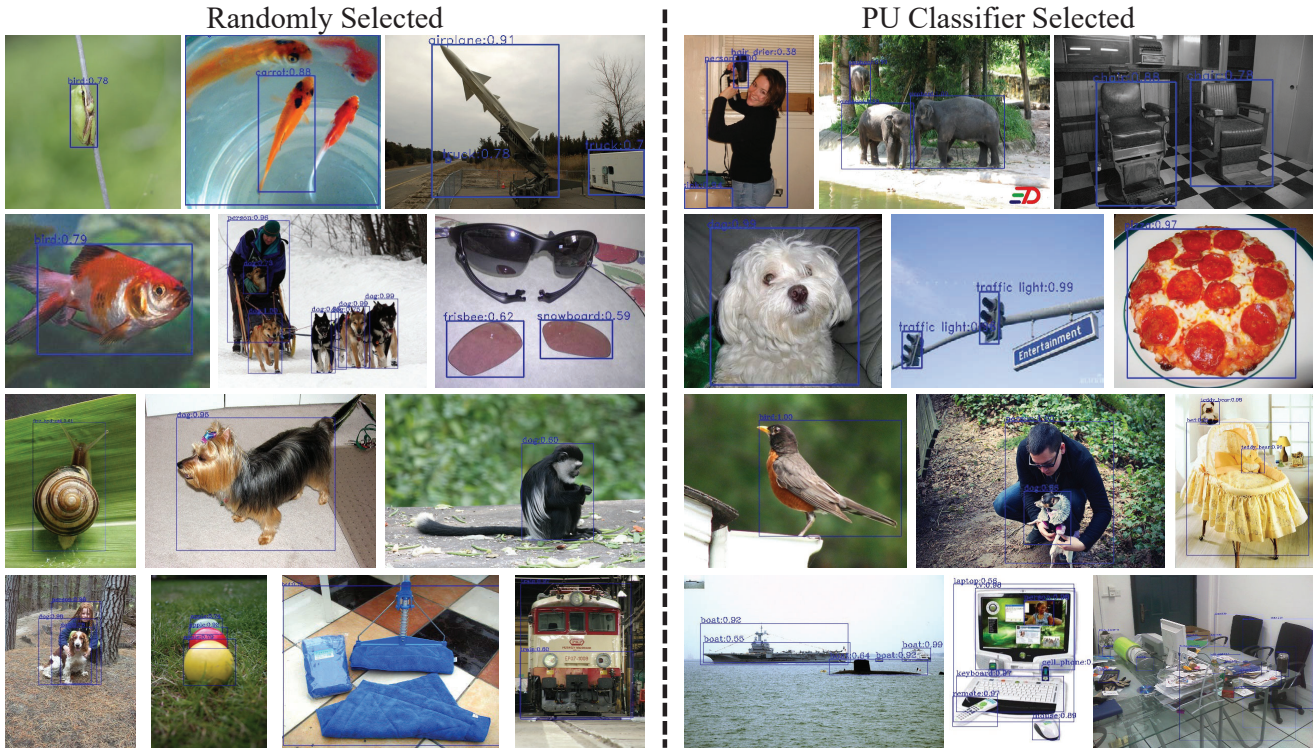

Figure 3. Visualization of the generated pseudo labels on unlabeled images. Images on the left are selected randomly while images on the right are selected by our proposed PU classifier (Best viewed in color).

detector will generate wrong annotations on the negative images. For example, the COCO dataset doesn't have the category "goldfish". However, the pretrained FPN classifies the "goldfish" into "carrot" with a confidence of 0.88. Although one can use a higher threshold to alleviate this situation, the number of training boxes will inevitably decrease accordingly. The qualities of PU classifier selected images are much better than the randomly selected ones.

5. Conclusion

In this work, we explore a more complicated and realistic situation on semi-supervised object detection, where we intend to exploit all available labeled data and excavate useful samples from unlabeled web images. While semi-supervised learning has achieved great success in classification task, not much effort has been put on the above situation. To accomplish the data purification, we propose a positive-unlabeled learning based scheme that can filter out the dirty images from the unlabeled data. Further more, we present a self-distillation algorithm which can enhance the object detector by exhausting a single model. Experiments on COCO benchmark demonstrate the efficiency of our proposed two stage framework.

Acknowledgement. This work is supported in part by the National Nature Science Foundation of China under Grant 62071013 and 61671027, in part by National Key R&D Program of China under Grant 2018AAA0100300, and in part by the Australian Research Council under Projects DE180101438 and DP210101859.

References

[1] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *NIPS*, 2017. 5

[2] 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:1906.07155*, 2019. 6

[3] Xinlei Chen and Abhinav Gupta. Webly supervised learning of convolutional networks. In *ICCV*, 2015. 3

[4] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *NIPS*, 2016. 1, 2

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 6

[6] Shangchen Du, Shan You, Xiaojie Li, Jianlong Wu, Fei Wang, Chen Qian, and Changshui Zhang. Agree to disagree: Adaptive ensemble knowledge distillation in gradient space. *Advances in Neural Information Processing Systems*, 2020. 5

[7] Marthinus Du Plessis, Gang Niu, and Masashi Sugiyama. Convex formulation for learning from positive and unlabeled data. In *ICML*, 2015. 4

[8] Marthinus C Du Plessis, Gang Niu, and Masashi Sugiyama. Analysis of learning from positive and unlabeled data. In *Advances in neural information processing systems*, 2014. 3, 4

[9] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Object detection with keypoint triplets. *arXiv preprint: 1904.08189*, 2019. 1, 2

[10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 1, 3

[11] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017. 1

[12] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018. 1

[13] Jianyuan Guo, Kai Han, yunhe Wang, Han Wu, Xinghao Chen, Chunjing Xu, and Chang Xu. Distilling object detectors via decoupled features. *arXiv preprint: 2103.14475*, 2021. 5

[14] Jianyuan Guo, Kai Han, Yunhe Wang, Chao Zhang, Zhaohui Yang, Han Wu, Xinghao Chen, and Chang Xu. Hit-detector: Hierarchical trinity architecture search for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 1

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1, 2

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[17] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint:1503.02531*, 2015. 5

[18] Shantanu Jain, Martha White, and Predrag Radivojac. Estimating the class prior and posterior from noisy positives and unlabeled data. In *NIPS*, 2016. 4

[19] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, 2008. 1

[20] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *NIPS*, 2019. 2, 3

[21] Bin Jin, Maria V Ortiz Segovia, and Sabine Susstrunk. Webly supervised semantic segmentation. In *CVPR*, 2017. 3

[22] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *ICCV*, 2019. 1

[23] Ryuichi Kiryo, Gang Niu, Marthinus C du Plessis, and Masashi Sugiyama. Positive-unlabeled learning with non-negative risk estimator. In *NIPS*, 2017. 3, 4, 6

[24] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint:1811.00982*, 2018. 1, 2

[25] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *ECCV*, 2018. 1, 2

[26] 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*, 2013. 2

[27] Xiao-Li Li, Philip S Yu, Bing Liu, and See-Kiong Ng. Positive unlabeled learning for data stream classification. In *Proceedings of the SIAM International Conference on Data Mining*, 2009. 3

[28] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1, 2, 4, 6

[29] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 2

[30] 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 *ECCV*, 2014. 1, 2, 3, 6

[31] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016. 1, 2

[32] Ao Luo, Xin Li, Fan Yang, Zhicheng Jiao, and Hong Cheng. Webly-supervised learning for salient object detection. *Pattern Recognition*, 2020. 3

[33] Aditya Menon, Brendan Van Rooyen, Cheng Soon Ong, and Bob Williamson. Learning from corrupted binary labels via class-probability estimation. In *ICML*, 2015. 4

[34] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *TPAMI*, 2018. 1

[35] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *NIPS*, 2018. 1, 3

[36] Jiangmiao Pang, Kai Chen, Jianping Shi, Huajun Feng, Wanli Ouyang, and Dahua Lin. Libra r-cnn: Towards balanced learning for object detection. In *CVPR*, 2019. 2

[37] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 6

[38] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini-batch object detector. In *CVPR*, 2018. 2

[39] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari, and Kaiming He. Data distillation: Towards omni-supervised learning. In *CVPR*, 2018. 2, 3

[40] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. Semi-supervised learning with ladder networks. In *NIPS*, 2015. 1, 2

[41] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 1, 2

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 1, 2

[43] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint:1412.6550*, 2014. 5

[44] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. *WACV/MOTION*, 2005. 1, 2

[45] Olga Russakovsky, Li-Jia Li, and Li Fei-Fei. Best of both worlds: human-machine collaboration for object annotation. In *CVPR*, 2015. 1

[46] Tomoya Sakai, Marthinus Christoffel du Plessis, Gang Niu, and Masashi Sugiyama. Semi-supervised classification based on classification from positive and unlabeled data. In *ICML*, 2017. 1

[47] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017. 1

[48] 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. *arXiv preprint:2001.07685*, 2020. 1, 4

[49] 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:2005.04757*, 2020. 1, 2, 3, 4, 6, 7

[50] Hao Su, Jia Deng, and Li Fei-Fei. Crowdsourcing annotations for visual object detection. In *AAAI Workshops*, 2012. 1

[51] Peng Tang, Chetan Ramaiah, Ran Xu, and Caiming Xiong. Proposal learning for semi-supervised object detection. *arXiv preprint:2001.05086*, 2020. 2, 3, 6, 7

[52] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NIPS*, 2017. 1

[53] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NIPS*, 2016. 1

[54] Keze Wang, Xiaopeng Yan, Dongyu Zhang, Lei Zhang, and Liang Lin. Towards human-machine cooperation: Self-supervised sample mining for object detection. In *CVPR*, 2018. 2

[55] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *CVPR*, 2017. 2

[56] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng. Distilling object detectors with fine-grained feature imitation. In *CVPR*, 2019. 5

[57] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Xiaohui Shen, Ming-Ming Cheng, Jiashi Feng, Yao Zhao, and Shuicheng Yan. Stc: A simple to complex framework for weakly-supervised semantic segmentation. *TPAMI*, 2016. 3

[58] Yixing Xu, Yunhe Wang, Hanting Chen, Kai Han, Chunjing Xu, Dacheng Tao, and Chang Xu. Positive-unlabeled compression on the cloud. *arXiv preprint:1909.09757*, 2019. 3

[59] Xiaopeng Yan, Ziliang Chen, Anni Xu, Xiaoxi Wang, Xiaodan Liang, and Liang Lin. Meta r-cnn: Towards general solver for instance-level low-shot learning. In *ICCV*, 2019. 1

[60] Shan You, Tao Huang, Mingmin Yang, Fei Wang, Chen Qian, and Changshui Zhang. Greedynas: Towards fast one-shot nas with greedy supernet. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2020. 2

[61] Shan You, Chang Xu, Chao Xu, and Dacheng Tao. Learning from multiple teacher networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017. 5

[62] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D Cubuk, and Quoc V Le. Rethinking pre-training and self-training. *arXiv preprint:2006.06882*, 2020. 7