Humble Teachers Teach Better Students for Semi-Supervised Object Detection

Yihe Tang [†,*] Weifeng Chen [‡] Yijun Luo [‡] Yuting Zhang [‡]
[†] Carnegie Mellon University, [‡] Amazon Web Services
tangacademic@gmail.com {weifec,yijunl,yutingzh}@amazon.com

Abstract

We propose a semi-supervised approach for contemporary object detectors following the teacher-student dual model framework. Our method [1] is featured with 1) the exponential moving averaging strategy to update the teacher from the student online, 2) using plenty of region proposals and soft pseudo-labels as the student's training targets, and 3) a light-weighted detection-specific data ensemble for the teacher to generate more reliable pseudo-labels. Compared to the recent state-of-the-art – STAC, which uses hard labels on sparsely selected hard pseudo samples, the teacher in our model exposes richer information to the student with soft-labels on many proposals. Our model achieves COCO-style AP of 53.04% *on* VOC07 val *set,* 8.4% *better than STAC, when using* VOC12 *as unlabeled data. On MS-COCO, it outperforms prior work when only a small percentage of data is taken as labeled. It also reaches* 53.8% *AP on* MS-COCO test-dev *with* 3.1% *gain over the fully supervised ResNet-152 Cascaded R-CNN, by tapping into unlabeled data of a similar size to the labeled data.*

1. Introduction

We address the problem of semi-supervised object detection in this paper. Large curated datasets have driven the recent progress in vision tasks like image classification, but data remain scarce for object detection [14, 31, 26, 5, 22, 30]. MS-COCO [25], for example, offers 118,287 annotated images, a relatively small fraction compared to over 14 million labeled images in ILSVRC [35]. Annotation acquisition for detection is also much more costly.

Much effort has been made to solve the semi-supervised learning problem for image classification, where an object always exists and dominates the image. Not all progress for image classification can benefit the detection task significantly as the existence and locations of objects are unknown without bounding box annotations. For example, a direct application of classification-based pretraining [15, 7] is shown to be not so effective in our experiments (Sec. 4.4.2).

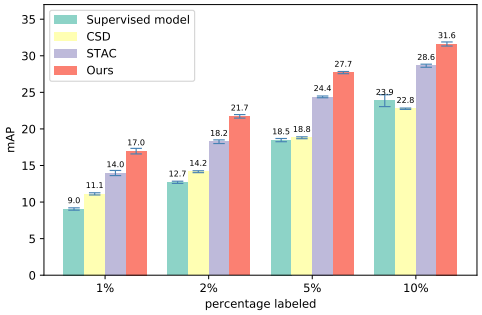


Figure 1: Comparing CSD [19], STAC [40], and our approach trained on full *MS-COCO train* 2017 with 1%, 2%, 5%, and 10% labeled over five runs using the splits in Sec. 4.1. Our approach consistently outperforms others.

In this work, we propose a teacher-student approach called *Humble Teacher*, which fits modern object detection frameworks better. The line of work on teacher-student models has many variants, including self-training [37, 48, 32, 39, 47], the exponential moving average (EMA) based mean teacher [44], and various ways to obtain pseudo-labels and different views of data for consistency regularization [49, 21, 36, 39] between the teacher and student. Recently, Sohn et al. [40] proposed a Self-Training method based on an Augmentation driven Consistency regularization (STAC) via hard pseudo-labels. It adopted FixMatch [39], one of the most successful recent methods for semi-supervised image classification, directly to the classification head of the Faster R-CNN [31] detector, yielding improved semi-supervised detection results.

Our method further advances the semi-supervised object detection for Faster-R-CNN-like models in a few aspects. Unlike self-training with a fixed teacher model, our method updates the teacher model dynamically using EMA updates for object detectors. The teacher and student model use asymmetric data augmentation – stronger augmentations for the student [46, 3, 39, 40] – to process different views of the same image [38]. In this framework, the key to our model's strong performance is to use soft pseudo-

*Work conducted during internship at Amazon Web Services.
[1] Project page: http://yihet.com/humble-teacher

labels on a reasonable number of region proposals, striking a good balance between covering the entire image and focusing more on learning useful foreground instances. It allows the student to distill much richer information from the teacher, compared to sparsely hard-selected high-confident pseudo ground truths in the existing work [40]. The use of soft-labels also keeps the model from over-fitting to the teacher model's potential missing and wrong predictions, which can occur often when using a hard decision threshold. In addition, we ensemble the teacher model under a light-weighted detection-specific data augmentation to obtain more reliable pseudo-labels. Through our study, we find the wisdom from FixMatch and STAC – hard pseudo-labels with sample selection – is not as effective. As our method avoids hard training signals, looks at abundant box instances, seeks for multi-teacher consensus, and uses running average weights as in the mean teacher, we name our method a *Humble Teacher*.

The humble teacher significantly closes the gap between semi-supervised learning and their fully supervised counterpart on VOC. It significantly outperforms the state-of-the-art STAC [40] on MS-COCO (Fig. 1) and VOC by large margins. It also improves the ResNet-152 Cascade R-CNN [5] supervised on *MS-COCO train* significantly with the additional similar-size unlabeled data.

In summary, we propose the humble teacher for semi-supervised object detection. It outperforms the previous state-of-the-art in both low-data and high-data regimes. Its use of soft-labels are pivotal to enable learning with abundant proposals and also make the EMA and teacher ensemble more effective for detection.

2. Related Work

2.1. Semi-supervised Learning in Classification

Significant progress has been made in semi-supervised image classification [44, 4, 3, 51, 39, 21, 46, 17]. One dominant idea in this field is pseudo-labeling [39, 1, 23, 39, 4, 3, 49, 45] — pseudo-labels for unlabeled data are repeatedly generated by a pre-trained model, and the model is then updated by training on a mix of pseudo-labels and human annotated data. The state-of-the-art FixMatch [39] retains only the highly confident hard pseudo-labels for training, and adopts different data augmentation strategies for label creation and training. Our method draws inspiration from it to use separately augmented inputs for pseudo-labeling and training. Our method is different in that we adopt two separate models — a student network that learns from pseudo-labels, and a teacher model that annotates pseudo-labels with the aid of a task-specific ensemble. Moreover, we use soft pseudo-labels while [39] uses hard labels. We additionally update our student and teacher models using two different strategies.

Another popular approach is the consistency regularization [21, 44]. It penalizes the inconsistency between two softmax predictions from different perturbations, such as differently augmented inputs [21], prediction and temporal ensemble prediction [21]. Our adoption of using soft label is partially inspired by consistency regularization, and extends the soft label idea beyond class probability to also bounding box regression offsets, where we keep the predicted offsets of all classes as the soft labels.

A consistency regularization approach, the Mean Teacher [44], is worth mentioning. The Mean Teacher adopts a teacher-student approach and the teacher is updated from the student by the exponential moving averaging. It applies consistency constraints [44] between softmax predictions of the teacher and the student. Besides being designed for a different detection task, our method looks similar to the Mean Teacher, but there is a critical difference that significantly improves our performance. Instead of feeding two strongly augmented copies to the teacher and the student for consistency regularization, our teacher sees the original image to make as-accurate-as-possible predictions as pseudo-labels, and our student sees the strongly augmented image to learn more generalizable features. FixMatch [39] already demonstrates the big gain of pseudo-labeling compared with the consistency regularization.

2.2. Semi-supervised Learning in Object Detection

The pioneering work [34] explores a self-learning approach in object detection based on Mahalanobis metric. Several works [12, 16, 43] have made progress in utilizing image-level labels to aid semi-supervised object detection. Adopting ideas similar to those in semi-supervised image classification also leads to progress [52]. Recently, Sohn et al. [40] established a new state-of-the-art by combining self-learning and consistency regularization. Our work is inspired by it but differs in many ways and attains much better performance. First, their approach only has a single network, while we adopt a framework with separate teacher and student networks as in the Mean Teacher [44]. Second, we generate pseudo-labels from the teacher and train the student simultaneously, while they generate all the pseudo-labels only once and then train on the fixed pseudo-labels. Third, we use soft labels as the pseudo-labels, while they use hard labels.

Jeong et al. recently proposed CSD [19] which horizontally flips an image and enforces its output to be consistent with that from the original image. CSD inspires our task-specific data ensemble of flipping images for teacher network. Our idea differs from CSD in the way the flipped images are used: we average the outputs from the original and flipped images to create better pseudo-labels, while CSD uses flipped images to enforce a consistency loss. Additionally, CSD [19] and its follow-up work ISD [20] focus

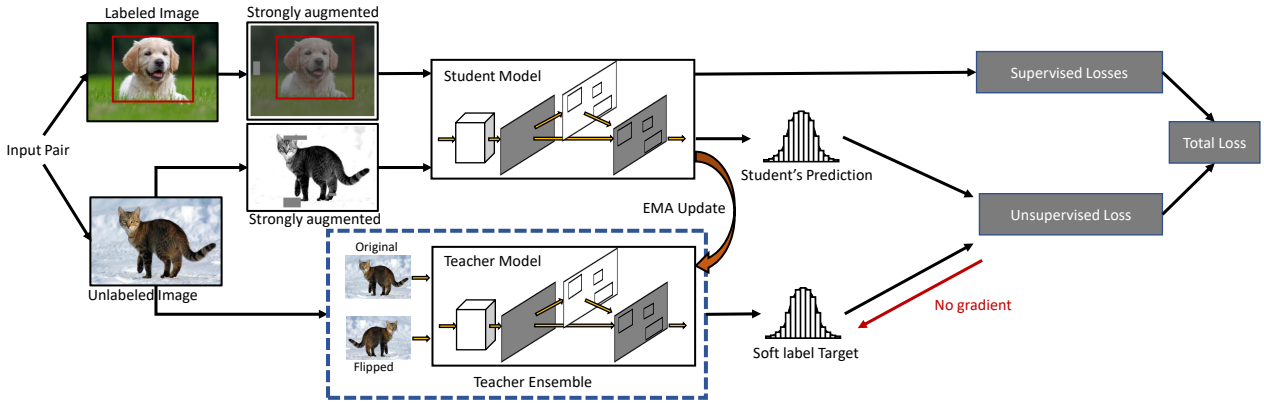


Figure 2: An overview of our Humble Teacher approach. The teacher model produces soft pseudo-labels for the student to learn from, and is updated via exponential moving average (EMA).

on the grid-sampled boxes in single-stage object detectors, while our approach applies to the bounding box proposals in two-stage object detectors such as Faster R-CNN [31].

3. Approach

3.1. Overview

Our approach learns a two-stage object detector from both labeled and unlabeled images. During training, the framework takes a mixed batch of equal numbers of labeled and unlabeled images as input and feeds them into the supervised branch and the unsupervised branch respectively. The final loss L is the sum of the supervised loss L_S and the unsupervised loss L_U,

$$L = L_S + \frac{n_U}{n_S}\beta L_U, \tag{1}$$

where n_U, n_S are the numbers of unlabeled and labeled images, and β is set to 0.5 by default.

The supervised branch It is a standard supervised two-stage detector like Faster R-CNN [31]. The regular detection losses are applied — the RPN's classification loss $L_{\text{cls}}^{\text{rpn}}$ and localization loss $L_{\text{loc}}^{\text{rpn}}$, as well as the ROI head's classification loss $L_{\text{cls}}^{\text{roi}}$ and localization loss $L_{\text{loc}}^{\text{roi}}$. The total supervised loss is

$$L_S = L_{\text{cls}}^{\text{rpn}} + L_{\text{loc}}^{\text{rpn}} + L_{\text{cls}}^{\text{roi}} + L_{\text{loc}}^{\text{roi}}. \tag{2}$$

The unsupervised branch It adopts a teacher-student framework as shown in Fig 2. The teacher, student, and the supervised network share the same architecture (we use Faster R-CNN [31] in our experiments) and are initialized with the same weights. The student shares the same weights with the supervised network but not with the teacher. An unlabeled image is processed independently by both the student and the teacher networks. The teacher network utilizes a task-specific ensemble to predict a pseudo-label from a weakly augmented version of the image (random flipping). It only predicts the pseudo-label and does not back-propagate gradients. The student takes a strongly augmented version of the same image as input to make predictions. An unsupervised loss L_U is then calculated between the student predictions and the pseudo-labels in RPN and ROI heads.

Augmentation Augmentation plays an important role in our model. For training, the image first goes through random flipping and resizing as the weak augmentation. The teacher network takes the weakly augmented image as its input (Sec. 3.4). Upon the same weakly augmented image, we further randomly change the color, sharpness, contrast, add Gaussian noise and apply cutouts [10]. We refer to the final image as strongly augmented from the original image. Our strong augmentation strategy follows [40] largely, but we did not use random rotation because bounding boxes will no longer be tightly wrapping around the rotated objects, making the setting undesirably complicated. Using strongly augmented images increases the difficulty of the student's task and can encourage it to learn better representations [38]. In contrast, using weak augmentations for the teacher can increase the chance for the teacher to generate correct pseudo-labels. Our detailed augmentation method is described in the supplementary material.

Inference Stage We use the teacher model for inference and produce final object detection results. No data augmentation is applied to the input image at the inference stage.

3.2. Soft Labels and Unsupervised Loss

The unsupervised branch uses soft labels predicted by the teacher model as training targets in the classification and regression tasks. For the classification task, the soft label target is the predicted distribution of the class probabilities. For the bounding box regression task, the soft label target is the offsets of all possible classes when the head is perform-

ing class-dependent bounding box regression [13]. We apply unsupervised loss in both the RPN (first stage) and ROI heads (second stage) of our object detector. The choice of using soft labels deviates from common practices of using hard labels [40, 39], where the object categories and offsets are selected when the pseudo-labels are generated.

In the first stage, the unsupervised loss is applied to both the classification objectness and the bounding box regression of the RPN for all anchors S_A. Let $\mathbf{s}_{\text{cls}}^{rpn,i}$ and $\mathbf{s}_{\text{reg}}^{rpn,i}$ denote the classification probability and bounding box regression output by the student RPN for the i-th proposal, and let $\mathbf{t}_{\text{cls}}^{rpn,i}$ and $\mathbf{t}_{\text{reg}}^{rpn,i}$ be those of the teacher RPN. Note that the weak augmentations for teacher and student are shared and in sync. The remaining strong augmentation steps do not impact the image geometry. Consequently, the anchor set is the same for the teacher and the student. The unsupervised loss for the RPN is defined as

$$L_U^{\text{rpn}} = \sum_{i \in S_A} D_{KL}(\mathbf{t}_{\text{cls}}^{rpn,i} \| \mathbf{s}_{\text{cls}}^{rpn,i}) + \| \mathbf{t}_{\text{reg}}^{rpn,i} - \mathbf{s}_{\text{reg}}^{rpn,i} \|_2, \quad (3)$$

where D_{KL} is the KL divergence.

In the second stage, the teacher model's RPN generates a set of region proposals, where the standard RPN NMS is applied [31]. The teacher model keeps the top-N proposals ranked by the predicted objectness score for the pseudo-label generation. It is different from the supervised branch, which follows the standard RPN training mode of Faster R-CNN to randomly sample a fix ratio of positive and negative region proposals. We set $N = 640$ by default and use S_P to denote the set of top-N proposals from the teacher. S_P are fed to the ROI heads of both teacher and student. The student's RPN proposals are not used in its ROI head training as the teacher's proposals are often of higher quality than those from the student. This design also eliminates the need to match proposals between the teacher and student, which could lead to complicated details.

For each region proposal, the student learns the raw probability and class-dependent regression outputs from the teacher. Let $\mathbf{s}_{\text{cls}}^{roi,i}$, $\mathbf{s}_{\text{reg}}^{roi,i}$, $\mathbf{t}_{\text{cls}}^{roi,i}$, $\mathbf{t}_{\text{reg}}^{roi,i}$ denote the classification probabilities and all-class bounding box regression outputs by the student and teacher ROI head for the i-th proposal respectively, our final ROI consistency loss is

$$L_U^{\text{roi}} = \Sigma_{i \in S_P} D_{KL}(\mathbf{t}_{\text{cls}}^{roi,i} \| \mathbf{s}_{\text{cls}}^{roi,i}) + \| \mathbf{t}_{\text{reg}}^{roi,i} - \mathbf{s}_{\text{reg}}^{roi,i} \|_2. \quad (4)$$

The final unsupervised loss L_U is the sum of L_U^{roi} and L_U^{rpn}.

The use of all top-N regions proposals results in abundant box instances for pseudo-labels. They are likely to cover the actual objects, boxes moderately overlapped with objects, and background regions, leading to a more comprehensive representation of the detection score distribution over the entire image. These benefits are unattainable when using hard labels. Many regions are neither strictly foreground nor background, and the hard labels cannot represent such intermediate states. The hard label setting, such as in [40], naturally needs a sample selection process like NMS and score-based thresholding to get definite pseudo ground truths.

3.3. Exponential Moving Average for the Teacher Model Update

The teacher model weights W_{teacher} are updated from the student model weights W_{student} by exponential moving average (EMA) [44]. At each iteration, we have

$$W_{\text{teacher}} = \alpha W_{\text{teacher}} + (1 - \alpha) W_{\text{student}}, \quad (5)$$

where we set $\alpha = 0.999$. Therefore, the teacher only slightly updates itself from the student each time. The gradually updated teacher is more resilient to the sudden weight turbulence of the student due to a wrong label prediction of the teacher model — even if the student is fed with a wrong label, its influence on the teacher model is mitigated by the exponential moving average. Besides resiliency to occasional wrong pseudo-labels, EMA is also known to lead to better generalization [18].

It is worth noting that we follow Faster R-CNN [31] to fix the running mean and variance of the BatchNorm layers in the training.

3.4. Teacher Ensemble with Horizontal Flipping

We ensemble the teacher model by taking as input both the image and its horizontally flipped version. The underlying intuition is that object classes should remain the same when the image is flipped, and the average prediction from both the original and the flipped copy can be more accurate than the prediction from a single image. Our design is inspired by prior research on ensemble methods [33, 29, 53], and by human pose estimation literature in which combining predictions from the original and the flipped image has lead to better pose estimation [28, 41, 8]. Experiments in Sec. 5.4 show that our teacher ensemble leads to superior semi-supervised object detection performance.

More specifically, let f_B be the backbone feature of the original image, $\hat{f_B}$ be the backbone feature of the flipped image, and P be the set of proposals detected by RPN on the original image. We do not use RPN to propose regions for the flipped image but instead flip the proposal coordinates in P horizontally to obtain $\hat{P}$ as the proposals for the flipped image. Then, for the ROI head, its softmax class probability output P_{cls} and regression offset output σ_{reg} from the ensemble are:

$$f = \text{ROIAlign}(f_B, P), \quad (6)$$

$$\hat{f} = \text{ROIAlign}(\hat{f_B}, \hat{P}), \quad (7)$$

$$P_{\text{cls}} = 0.5(C(f) + C(\hat{f})), \quad (8)$$

$$\sigma_{\text{reg}} = 0.5(R(f) + T(R(\hat{f}))). \quad (9)$$

Model	Labeled Dataset	Unlabeled Dataset	AP50	AP
Supervised model	VOC07	N/A	76.3	42.60
Supervised model	VOC07 + VOC12	N/A	82.17	54.29
CSD‡	VOC07	VOC12	76.76	42.71
STAC [40]	VOC07	VOC12	77.45	44.64
Humble teacher (ours)	VOC07	VOC12	**80.94**	**53.04**
CSD‡	VOC07	VOC12 + MS-COCO20 (2017)	77.10	43.62
STAC [40]	VOC07	VOC12 + MS-COCO20 (2017)	79.08	46.01
Humble teacher (ours)	VOC07	VOC12 + MS-COCO20 (2017)	**81.29**	**54.41**

Table 1: Results on Pascal VOC, evaluated on the *VOC07 test* set. Our model consistently outperforms others in all experiment setups. CSD‡ is our ResNet-50-based re-implementation, which achieves better performance than the original CSD [19].

Note that C is the classification head including softmax at the end, and R is the regression head. T is the transformation that flips the x axis of all bounding boxes. We apply this ensemble mechanism only to create pseudo-labels in the ROI heads but not RPN heads, because the corresponding anchors in a flipped pair of images may not be symmetric in the RPN head.

4. Experiments

4.1. Dataset and Evaluation

We evaluate our approach on two detection datasets: Pascal VOC [11] and MS-COCO [25]. For Pascal VOC, we evaluate the performance on the *VOC07 test*. During training, we first use *VOC07 trainval* as the labeled dataset and *VOC12 trainval* as the unlabeled dataset. *VOC07 trainval* and *VOC12 trainval* have 5,011 and 11,540 images respectively, resulting in a roughly 1:2 labeled to unlabeled ratio. Following the practice in [19, 40], besides *VOC12 trainval*, we also bring *MS-COCO20* [19, 40] in as additional unlabeled data. *MS-COCO20* filters out the MS-COCO images that contain objects whose classes are not included in the 20 Pascal VOC classes. We conduct additional experiments using both the *VOC12 trainval* and *MS-COCO20 train* as unlabeled data, totaling 129,827 unlabeled images, leading to a 1:26 labeled to unlabeled ratio.

For MS-COCO, we use version 2017 in all experiments. We report the results on the *MS-COCO val* dataset. For training, we follow [40] to split *MS-COCO train* into the labeled and the unlabeled datasets. We set up four labeling percentages: 1%, 2%, 5%, and 10% as in [40], and the remaining images are used as unlabeled data. For each percentage, we randomly sample five different splits using the provided code from [40]. The same splits are used throughout our experiments and ablation studies. In addition, we also set up an experiment using the entire *MS-COCO train* as labeled dataset, and *MS-COCO unlabeled* as unlabeled dataset. *MS-COCO train* has a total of 118,287 images and *MS-COCO unlabeled* has 123,403 in total, leading to a roughly 1:1 labeled to unlabeled ratio. We run this experiment to demonstrate that our approach is able to further improve upon a model trained on a large labeled dataset like MS-COCO.

4.2. Model Configurations

We use Faster R-CNN with ResNet-50 backbone and FPN as our default base model. We re-implement CSD with ResNet-50 backbone for fair comparison, and it achieves better performance than the original model in [19]. We also evaluate our method on a larger base model Cascade R-CNN with ResNet-151 backbone and FPN [5, 24]. When training on Cascade R-CNN, we apply our unsupervised loss on the ROI head at each stage .

Before training on unlabeled data, the model first goes through a *burn-in* stage, i.e. pre-training the detection network on the labeled data following standard training protocols [31]. This model is the base supervised model, and its weights are copied into the student and the teacher networks to initiate the semi-supervised training.

4.3. Results on Pascal VOC

We benchmark our method on PASCAL VOC under two experiment setups — **(a)** *VOC07* as labeled set and *VOC12* as unlabeled set, and **(b)** the same as (a) but with *MS-COCO20* as additional unlabeled data. We also report the performance of the same model trained fully supervised on *VOC07* and *VOC07+VOC12*. Tab. 1 compares our results with the best existing methods under AP50 and MS-COCO style AP metrics.

Our approach consistently outperforms the best existing results by a large margin in all setups. It outperforms the state-of-the-art STAC [40] by 8.4% and 8.4% in AP respectively in setup (a) and (b). Notably, our method trained on the labeled *VOC07* and the unlabeled *VOC12*

Percentage labeled	1%	2%	5%	10%
Supervised model	9.05±0.16	12.70±0.15	18.47±0.22	23.86±0.81
CSD‡	11.12±0.15 (+2.07)	14.15±0.13 (+1.45)	18.79±0.13 (+0.32)	22.76±0.09 (−1.10)
STAC [40]	13.97±0.35 (+4.92)	18.25±0.25 (+5.55)	24.38±0.12 (+5.91)	28.64±0.21 (+4.78)
Humble teacher (ours)	**16.96±0.38 (+7.91)**	**21.72±0.24 (+9.02)**	**27.70±0.15 (+9.23)**	**31.61±0.28 (+7.74)**

Table 2: The mAP (50:95) results on *MS-COCO val 2017* by models trained on different percentage of labeled *MS-COCO train 2017*. All models are with the ResNet-50 backbone. CSD‡ is our re-implementation with better performance. Our method consistently outperforms others.

significantly outperforms the based model fully supervised on *VOC07* alone, and with the additional unlabeled *MS-COCO20* it further improves performance. Our best performing model is narrowing the gap from 9.65% to 1.25% in COCO style mAP between the model fully supervised on *VOC07+VOC12* and the model trained on labeled *VOC07* and unlabeled *VOC12*. These results suggest that our method is particularly effective in improving model performance with cheap unlabeled data.

Moreover, our model outperforms CSD and STAC more on the 0.5:0.95 AP than on AP50 regarding both absolute gain and relative error reduction. It indicates that the humble teacher could localize objects more accurately. This may be attributed to the use of soft labels over the full set of region proposals, which leads to more guidance for the student model to learn on image regions without definite labels even given the ground truth annotations. Such guidance has been shown to be helpful for localization [50].

4.4. Results on MS-COCO

4.4.1 MS-COCO of Different Labeled Percentages

We first investigate if the proposed humble teacher improves performance under a low data regime. We follow the setup of STAC [40] and report the performance when four percentages of labeled *MS-COCO train* is provided: 1%, 2%, 5% and 10%, while the remaining images are used as unlabeled data. Comparison with the best existing approaches on *MS-COCO val* in terms of mAP (50:95) is shown in Tab. 2. Our method consistently outperforms the best existing approach over all four labeled percentages. Notably, unlike CSD, the amount of improvement does not diminish, and the improvement is consistent though the percentage of labeled data increases.

4.4.2 MS-COCO Train + MS-COCO Unlabeled

Next, we investigate if the proposed semi-supervised learning strategy improves upon an object detector fully supervised on the entire *MS-COCO train*. We use the *MS-COCO unlabeled* [25], a set of 123,403 unlabeled images differed from those in *MS-COCO train*. We experiment with two setups, one is with Faster R-CNN [31] and another with

Model (Faster R-CNN with Resnet-50)	AP
Base supervised model	37.63
MOCOv2 + MS-COCO Unlabeled [7]	35.29
MOCOv2 + ImageNet-1M [7]	40.80
MOCOv2 + Instagram-1B [7]	41.10
Proposal learning [42]	38.4
CSD‡	38.52(+0.89)
STAC [40]	39.21(+1.58)
Humble teacher (ours)	**42.37(+4.74)**
Model (Cascade R-CNN with ResNet-152)	AP
Base supervised model	50.23
Humble teacher (ours)	**53.38 (+3.15)**

Table 3: The mAP (50:95) results on ***MS-COCO val 2017*** by models trained on *MS-COCO train 2017* + *MS-COCO unlabeled*. CSD‡ is with a ResNet-50 backbone.

Model (Cascade R-CNN with ResNet-152)	AP
Base supervised model	50.7
Humble teacher (ours)	**53.8 (+3.1)**

Table 4: The mAP (50:95) results on ***MS-COCO test-dev 2017*** by models trained on *MS-COCO train 2017* + *MS-COCO unlabeled*.

Casacade R-CNN [5]. The results are evaluated on *MS-COCO val*. In the Faster R-CNN case, the baseline model supervised on the full *MS-COCO train* achieves 37.63% AP. Our method achieves a 4.74% improvement in AP over the baseline (Tab. 3), and significantly outperforms other self-supervised methods such as Proposal Learning [42], CSD [19] and STAC [40]. In the Cascade R-CNN case, our method achieves a 3.15% improvement in AP over the high-performing fully supervised baseline (Tab. 3). Further evaluation on the *MS-COCO test-dev* shows a 3.1% AP improvement over the supervised Cascade R-CNN (Tab. 4). These results suggest that our method has the potential to directly apply to any object detectors and improve their performance by combining both labeled and unlabeled data.

We also compare against supervised finetuning with pretrained MOCOv2 [7], a state-of-the-art contrastive learn-

ing method for image classification pretraining. The goal is to show that a simple application of contrastive learning [7, 15, 6] does not work as well as our method in improving object detection from unlabeled data. More specifically, we follow the MOCOv2 setup to pre-train the ResNet-50 backbone in Faster R-CNN on each of the three unlabeled dataests: (1) *MS-COCO unlabeled*, (2) ImageNet-1M [9] and (3) Instagram-1B [27]. The pre-trained backbones are then copied to Faster R-CNN, which is further trained on *MS-COCO train* to perform object detection. Results in Tab. 3 suggest that object detection performance improves as the size of the unlabeled data increases. However, even the best-performing one (MOCOv2 pre-trained on Instagram-1B) still underperforms our method, although it uses 7,600 times more unlabeled data than our method.

5. Ablation Study

5.1. Number of Proposals for Unsupervised Loss

We first study how the number of region proposals fed into ROI head in unsupervised learning affects the performance. As shown in Fig. 3a, we experiment with different numbers of proposals up to 6000 given the GPU memory limit. We found that using too few region proposals hurts performance, possibly because of a poor coverage of objects and useful context. Having too many region proposals may include too many background samples, distracting the unsupervised learning from the important foreground regions [19]. Given the large performance drop when the proposals are too few or too many, we believe that using a balanced number of proposals with soft labels is the key to the superior performance of our method.

5.2. Update Rules

This section studies the benefits of our EMA update at every iteration. The teacher model is updated from the student model. We study three rules with different update frequencies: (1) EMA update at every iteration, (2) copy weights from student to teacher every 10K iterations, and (3) no update at all, i.e. keeping the teacher model fixed throughout the training. We still use Faster R-CNN with ResNet-50 for all the rules and trained on 10% labeled *MS-COCO train 2017*. Tab. 5 reports the mean and standard deviations over five runs using the same five splits described in Sec. 4.1.

Updating every 10K iterations outperforms no update at all. It suggests that keeping the teacher model up to date than using a fixed teacher is beneficial to model performance. EMA update at every iteration leads to even bigger performance gain. The results suggest that EMA updates are crucial for our student-teacher model to work well. One possible explanation is that the negative effect of incorrect pseudo-labels is mitigated by EMA update at every iteration, since the weight updates from one example batch are being averaged over time and sample batches.

The success of EMA is based on the assumption that EMA-updated teacher produces more accurate predictions than the student. To validate this assumption, we compare the object detection results on the 10% labeled *MS-COCO train 2017* setup using the student and the EMA-updated teacher model. Fig. 3b shows that the EMA-updated teacher is better than the student and therefore explains the success of our student-teacher paradigm.

5.3. Soft Labels versus Hard Labels

Next, we turn to the comparison between soft labels and hard labels in our semi-supervised framework. We use the same Faster R-CNN with ResNet-50 setup as before, and train on the 10% labeled *MS-COCO train 2017*, using the same EMA update and teacher-student framework. We then compare a version that trains on soft labels and another that trains on hard labels. Note that the hard labels are generated by thresholding on the prediction confidence. We experiment with a range of thresholds and select 0.7 which leads to the best performance. Moreover, given it is unclear how to combine the hard label from an original image and its flipped version, we exclude the task specific data ensemble from both experiments for fairness of comparison. Tab. 6 reports the results. Contrary to the findings in semi-supervised image classification [39], using soft labels help us achieve much better performance than hard labels in semi-supervised object detection, clearly demonstrating the critical role of soft labels plays in our method.

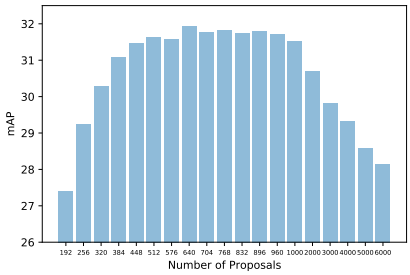
One possible explanation to the better performance due to the soft label is its strength to handle the highly im-

Model	AP
No update	27.26±0.21
Copy weights from student to teacher every 10K iters	28.61±0.18
EMA update at every iter	**31.61±0.28**

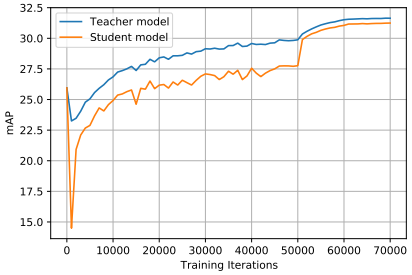
Table 5: Comparison between different update rules on *MS-COCO train 2017* with 10% data labeled. The mean and standard deviation over five data splits are reported (the same five splits of *MS-COCO train 2017* as in Sec. 4.1).

Model	AP
With hard label	27.97±0.13
With soft label	**30.97±0.16**

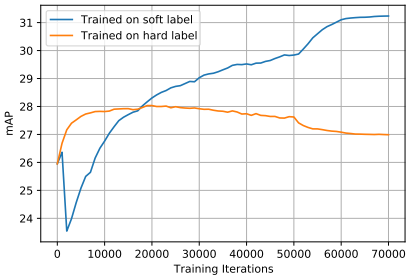
Table 6: Comparison between training on soft label and hard label when 10% labeled *MS-COCO train 2017* is provided. The mean and standard deviation over five data splits are reported (the same five splits of *MS-COCO train 2017* described in Sec. 4.1).



(a) Comparison between models with different number of region proposals used in unsupervised loss. The student-teacher framework is jointly trained on the 10% labeled and 90% unlabeled *MS-COCO train 2017* split.



(b) Comparison between teacher and student performance on the 10% labeled *MS-COCO train 2017* setup. The student-teacher framework is jointly trained on the 10% labeled and 90% unlabeled *MS-COCO train 2017* split.



(c) Teacher models' performance on unlabeled data. Both models are trained on 10% labeled *MS-COCO train 2017* with the remaining 90% as unlabeled.

Figure 3: Ablation study on hyperparameters and hard/soft labels.

balanced class distribution in object detection. This imbalanced issue is reflected in two aspects. First, background class dominates foreground classes during region proposal [31]. Second, foreground classes are not evenly distributed during ROI classification, as evident in the case of MS-COCO [25]. Using hard labels in such an imbalanced setup has the risk of pushing the probability of being dominant classes to 1 and the probability of being minority classes to 0, resulting in significant confirmation bias [2]. In contrast, soft labels carry richer information and retain the probability of being any possible classes, and suffer from less confirmation bias.

To validate this hypothesis, we experiment with the 10% labeled *MS-COCO train 2017* setup, and run object detection every 1,000 iterations using the teacher model on the remaining 90% unlabeled images and evaluate the detection mAP. Fig. 3c reports the mAP as training proceeds. We see that training on soft labels yields much higher mAP, and the mAP keeps increasing as the training goes on, while training on hard labels yields diminishing mAP. These results indicate that soft-label-trained teachers produce pseudo-labels that suffer from less confirmation bias.

5.4. Teacher Ensemble

We study the effectiveness of Teacher Ensemble. FixMatch [39] and ReMixMatch [3] claim a data ensemble of random augmentations may hurt the teacher model performance, and is worse than weak augmentations (resizing and randomly flipping) applied to the inputs of the teacher. We find this partially true, and show that our teacher ensemble improves performance in semi-supervised object detection.

Our experiment is based on the same Faster R-CNN with ResNet-50 trained on 10% labeled *MS-COCO train 2017*. We compare three setups: (1) without ensemble, (2) with a random augmented ensemble on teacher model, and (3) with task-specific data ensemble on teacher model. Tab. 7 reports the results.

Model	AP
No ensemble	30.97±0.16
Random augmented ensemble	30.79±0.31
Task-specific ensemble	**31.61±0.28**

Table 7: Effects of using different ensemble strategies on the teacher model on *MS-COCO train 2017* with 10% data labeled. The mean and standard deviation over five data splits are reported (the same five splits of *MS-COCO train 2017* described in Sec. 4.1).

Consistent with the findings in FixMatch [39], the random augmentation ensemble indeed hurts performance. Nonetheless, with our task-specific data ensemble (ensembling a pair of flipped and original images), the performance improves by 0.64% AP, suggesting that a carefully constructed ensemble is advantageous to the overall performance of our semi-supervised object detection method.

6. Conclusions

We developed a semi-supervised object detection algorithm, "Humble Teacher" that obtained state-of-the-art performance on multiple benchmarks. We demonstrated the effectiveness of our teacher-student model design and showed the importance of iteration-wise EMA teacher update. We found that soft label coupled with a balanced number of teacher's region proposals is the key toward superior performance. We also found that a carefully constructed data ensemble for the teacher improves the overall performance.

Acknowledgements We want to thank Luis Goncalves, Zhaowei Cai, Qi Dong, Aruni RoyChowdhury, R. Manmatha, Zhuowen Tu, and Vijay Mahadevan for insightful discussions.

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 *2020 International Joint Conference on Neural Networks*, pages 1–8. IEEE, 2020. 2

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

[3] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019. 1, 2, 8

[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*, pages 5049–5059, 2019. 2

[5] 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*, pages 6154–6162, 2018. 1, 2, 5, 6

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 7

[7] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 1, 6, 7

[8] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7103–7112, 2018. 4

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. Ieee, 2009. 7

[10] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 3

[11] 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*, 88(2):303–338, 2010. 5

[12] 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*, pages 9508–9517, 2019. 2

[13] Ross Girshick. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015. 4

[14] 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*, pages 580–587, 2014. 1

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020. 1, 7

[16] Judy Hoffman, Sergio Guadarrama, Eric S Tzeng, Ronghang Hu, Jeff Donahue, Ross Girshick, Trevor Darrell, and Kate Saenko. LSDA: Large scale detection through adaptation. In *Advances in Neural Information Processing Systems*, pages 3536–3544, 2014. 2

[17] 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*, pages 5070–5079, 2019. 2

[18] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018. 4

[19] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Advances in Neural Information Processing Systems*, pages 10759–10768, 2019. 1, 2, 5, 6, 7

[20] 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

[21] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016. 1, 2

[22] Hei Law and Jia Deng. CornerNet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision*, pages 734–750, 2018. 1

[23] 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. 2

[24] 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*, pages 2117–2125, 2017. 5

[25] 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 *Proceedings of the European Conference on Computer Vision*, pages 740–755. Springer, 2014. 1, 5, 6, 8

[26] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *Proceedings of the European Conference on Computer Vision*, pages 21–37. Springer, 2016. 1

[27] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly

supervised pretraining. In *Proceedings of the European Conference on Computer Vision*, pages 181–196, 2018. 7

[28] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Proceedings of the European Conference on Computer Vision*, pages 483–499. Springer, 2016. 4

[29] 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*, pages 4119–4128, 2018. 4

[30] 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*, pages 779–788, 2016. 1

[31] 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*, pages 91–99, 2015. 1, 3, 4, 5, 6, 8

[32] Ellen Riloff. Automatically generating extraction patterns from untagged text. In *Proceedings of the National Conference on Artificial Intelligence*, pages 1044–1049, 1996. 1

[33] Lior Rokach. Ensemble-based classifiers. *Artificial Intelligence Review*, 33(1-2):1–39, 2010. 4

[34] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. In *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05) - Volume 1*, volume 1, pages 29–36, 2005. 2

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 1

[36] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in Neural Information Processing Systems*, 29:1163–1171, 2016. 1

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

[38] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019. 1, 3

[39] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 596–608. Curran Associates, Inc., 2020. 1, 2, 4, 7, 8

[40] 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

[41] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European Conference on Computer Vision*, pages 529–545, 2018. 4

[42] Peng Tang, Chetan Ramaiah, Yan Wang, Ran Xu, and Caiming Xiong. Proposal learning for semi-supervised object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2291–2301, 2021. 6

[43] 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*, pages 2119–2128, 2016. 2

[44] 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*, pages 1195–1204, 2017. 1, 2, 4

[45] Keze Wang, Xiaopeng Yan, Dongyu Zhang, Lei Zhang, and Liang Lin. Towards human-machine cooperation: Self-supervised sample mining for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1605–1613, 2018. 2

[46] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*, 2019. 1, 2

[47] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves ImageNet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020. 1

[48] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, 1995. 1

[49] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4L: Self-supervised semi-supervised learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1476–1485, 2019. 1, 2

[50] Yuting Zhang, Kihyuk Sohn, Ruben Villegas, Gang Pan, and Honglak Lee. Improving object detection with deep convolutional networks via Bayesian optimization and structured prediction. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 249–258, June 2015. 6

[51] Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. *CMU CALD Tech Report CMU-CALD-02-107*, 2002. 2

[52] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. Rethinking pre-training and self-training. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3833–3845. Curran Associates, Inc., 2020. 2

[53] Xu Zou, Sheng Zhong, Luxin Yan, Xiangyun Zhao, Jiahuan Zhou, and Ying Wu. Learning robust facial landmark detection via hierarchical structured ensemble. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 141–150, 2019. 4