CROSSMATCH: IMPROVING SEMI-SUPERVISED OBJECT DETECTION VIA MULTI-SCALE CONSISTENCY

Anonymous authors
Paper under double-blind review

ABSTRACT

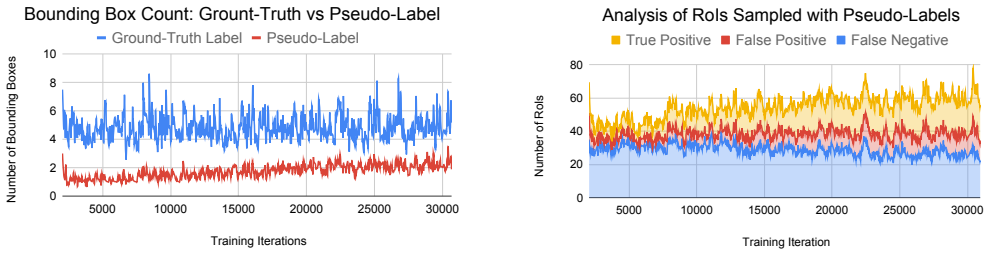
We present a novel method, CrossMatch, for semi-supervised object detection. Inspired by the fact that teacher/student pseudo-labeling approaches result in a weak and sparse gradient signal due to the difficulty of confidence-thresholding, CrossMatch leverages *multi-scale feature extraction* in object detection. Specifically, we enforce consistency between different scales across the student and teacher networks. To the best of our knowledge, this is the first work to use multi-scale consistency in semi-supervised object detection. Furthermore, unlike prior work that mostly uses hard pseudo-labeling methods, CrossMatch further densifies the gradient signal by enforcing multi-scale consistency through both hard and soft labels. This combination effectively strengthens the weak supervision signal from potentially noisy pseudo-labels. We evaluate our method on MS COCO and Pascal VOC under different experiment protocols, and our method significantly improves on previous state of the arts. Specifically, CrossMatch achieves 17.33 and 21.53 mAP with only 0.5% and 1% labeled data respectively on MS COCO, outperforming other state-of-the-art methods by $\sim$3 mAP.

1 INTRODUCTION

Recently, Semi-Supervised Learning (SSL) has led to significant breakthroughs for image classification by effectively making use of large-scale unlabeled data (which is easier to obtain) alongside a smaller annotated dataset. Although great success has been achieved for SSL in the image classification context using these methods, translating these techniques to SSL for object detection has only recently begun to be explored (Liu et al., 2021; Sohn et al., 2020b; Zhou et al., 2021; Tang et al., 2021; Yang et al., 2021). In translating these techniques to object detection, prior work mainly focuses on pseudo-labelling (Lee et al., 2013; Arazo et al., 2020; Rizve et al., 2021; Iscen et al., 2019; Xie et al., 2020b; Rosenberg et al., 2005), which involves extracting a dense set of bounding box predictions, followed by confidence thresholding and non-maximum suppression (NMS).

However, a direct translation of pseudo-labeling does not fully consider the characteristics of the object detector training. For example, since a high confidence threshold is used to ensure the quality of pseudo-labels, only a small number of bounding boxes eventually become pseudo-labels (Figure 1a). Missing and incorrect pesudo-labels result in false negative and false positive region of interests (RoIs) being mistakenly sampled (Figure 1b). As a result, the overall gradient signal produced by such pseudo-labeling is *sparse* and *weak*, which is a potential source of error accumulation.

In this work, we present our method, CrossMatch, to address these problems by seeking to strengthen the gradient signal in two ways. First, we identify a key source of rich information already inherent in many object detection approaches: *multi-scale* feature extraction. For example, the Feature Pyramid Network (FPN) (Lin et al., 2017a) generates a set of features at multiple scales for each image. Even though the region proposal network (Ren et al., 2015) uses the entire feature pyramid to generate RoIs at multiple scales, the network only assigns one chosen level of the feature pyramid to RoIs for class-level prediction. We argue that feature representations at different scales extracted by FPN is a non-negligible resource for diverse representation to train the network under the semi-supervised settings. Thus, we propose a novel method that enforces consistency between multiple scales. In contrast to the default level-assignment approach, CrossMatch makes predictions with multiple levels



(a) (b)

Figure 1: An illustration of sparse and weak gradient signals from hard pseudo label. (a): The number of pseudo-labels generated per image is much smaller than the number of ground-truth labels, which indicates the supervision signal from pseudo-labels is *sparse*. (b): In the training process, only a small number of true positive RoIs are sampled with pseudo-labels. A significant amount of foreground RoIs are not sampled (false negative) and a number of background RoIs are mis-sampled as foregrounds (false positive). This indicates the supervision signal from pseudo-labels is also *weak*.

of features from the feature pyramid and computes losses with those predictions accordingly. For inference, we still use the default heuristic-based method for feature selection so that no additional computation overhead is generated. To the best of our knowledge, our proposal is the first to use such multi-scale losses in the context of semi-supervised learning.

Second, unlike prior work (Tang et al., 2021), which attempts to completely replace hard pseudo-labels with soft probability targets, CrossMatch enforces multi-scale consistency through both soft and hard labels. We argue that combining soft targets with hard labels can be an effective approach to alleviate the weak and sparse supervision signals from hard pseudo-labels. Our approach adopts the teacher-student (Tarvainen & Valpola, 2017; Liu et al., 2021) framework and train the student network with both hard and soft artificial labels. Since soft labels can be easily obtained for all RoIs without matching them to ground-truth bounding boxes, the student network can effectively distill meaningful knowledge through dense RoIs from the teacher model. To prevent the gradients from being dominated by easy background RoIs, we use a refined design of OHEM (Shrivastava et al., 2016) to perform RoI sampling for training with soft labels.

We evaluate our method on MS-COCO (Lin et al., 2014) and Pascal VOC (Everingham et al., 2010) following the experimental protocol used in state-of-the-art literature (Sohn et al., 2020b). It is worth mentioning that our method achieves 17.33 and 21.53 mAP on Faster-RCNN ResNet-50-FPN architectiure with only 0.5% and 1% labeled images respectively on MS-COCO, which significantly outperforms other state-of-the-art methods by around 3 mAP.

In summary, the contributions of this paper are as follows:

- We propose a novel method to train the student network with multi-scale consistency. This practice provides diverse representations of features at multiple scales to better leverage unlabeled images. To the best of our knowledge, this is the first work that leverages multi-scale consistency in semi-supervised object detection.
- We propose a simple yet effective approach to enforce multi-scale consistency through a combination of soft and hard labels to further densify the gradient.
- Our method achieves state-of-the-art performance especially in the most challenging scenarios. It significantly outperforms other competing methods by 2-3 mAP when the percentage of labeled images is below 5% for MS-COCO.

2 Related Work

Semi-Supervised Image Classification: The frontier of semi-supervised image classification has been significantly pushed in recent years. One dominant approach is pseudo-labelling (Sohn et al., 2020a; Lee et al., 2013; Arazo et al., 2020; Bachman et al., 2014; Rizve et al., 2021; Iscen et al., 2019). The quality of pseudo-labels is usually enforced by applying confidence-thresholding to filter out potentially noisy samples. Consistency regularization (Tarvainen & Valpola, 2017; Laine & Aila, 2016; Berthelot et al., 2019; 2020; Sajjadi et al., 2016; Xie et al., 2020a) regularizes the network to be less sensitive to input or model perturbations by enforcing consistent predictions (e.g. via MSE or

KL-Divergence loss) across such perturbations. For both methods, the challenge is how to produce high-quality artificial targets. One typical approach is the Teacher-Student framework (Tarvainen & Valpola, 2017), where a Teacher evolves during the training by exponentially averaging the network weights of the Student. Recent work (Sohn et al., 2020a; Xie et al., 2020a) proposes another idea of using strong-weak augmentation pairs. Artificial labels are generated with weakly-augmented images and the network is trained with strongly-augmented images along with artificial labels.

Semi-Supervised Object Detection: A few existing work (Zhou et al., 2021; Sohn et al., 2020b; Tang et al., 2021; Liu et al., 2021; Tang et al., 2016; Misra et al., 2015; Gao et al., 2019; Li et al., 2020; Jeong et al., 2019; Yang et al., 2021) have studied object detection under the semi-supervised setting. NOTE-RCNN (Gao et al., 2019) proposes an iterative method for bounding box mining and detector re-training; Watch & Learn (Misra et al., 2015) studies object detection for videos under sparse labels (Oh et al., 2011) using visual and semantic knowledge transfer; CSD (Jeong et al., 2019) uses horizontal flip to enforce consistency between original and flipped images; S^4OD proposes a heuristic-based method for selecting unlabeled web images (Li et al., 2020). STAC (Sohn et al., 2020b) recently established a new benchmark for semi-supervised object detection. It trains an offline Teacher network using labeled images and generates pseudo-labels for the Student networks. Unbiased Teacher (Liu et al., 2021), Instant-Teaching (Zhou et al., 2021) and ISMT (Yang et al., 2021) propose to train detectors with hard pseudo-labels. Unbiased Teacher uses exponential moving average to update the Teacher network whereas Instant-Teaching incorporates a co-rectify regime to refine pseudo-labels. ISMT uses a memory bank to update pesudo-labels in addition to the Teacher-Student framework. In contrast to these methods, Humble Teacher (Tang et al., 2021) proposes to train the Student network with soft targets instead of hard pseudo-labels. Different from these approaches, we propose a novel multi-scale training regime to better leverage dense gradients from unlabeled images under the SSL setting. Further inspired by these methods, we also exploit the effective usage of both hard and soft labels, together with a refined OHEM scheme, to combine the benefits of both.

Multi-Scale Object Detection: The Feature Pyramid Network (Lin et al., 2017a) (and its variations (Ghiasi et al., 2019; Wang et al., 2020; Pang et al., 2019; Liu et al., 2018)) is an essential building block to perform multi-scale object detection. Such architectures leverage the feature hierachy in ConvNets to produce the feature pyramid to detect objects at multiple scales. Raw features are extracted from backbone and can be further fused by a top-down path (Lin et al., 2017a), a top-down and bottom-up path (Liu et al., 2018), multiple merging cells (Ghiasi et al., 2019), or a 3D convolution across scales (Wang et al., 2020). AugFPN (Guo et al., 2020) proposes Consistent Supervision which attaches auxiliary heads to features **before** fusion and pass supervision signals directly to different feature maps. The key difference between our multi-scale consistency and Consistent Supervision from AugFPN is that we do not use any auxiliary detection heads and we train the model with multi-scale feature maps **after** the fusion. We include a more detailed comparison in Appendix A.3.

3 METHODOLOGY

Problem Definition. The problem of semi-supervised object detection is defined as follows: we are given a training set of labeled images $\mathcal{D}_s = \{x_i^s, y_i^s\}_{i=1}^{N_s}$ and a set of unlabeled images $\mathcal{D}_u = \{x_i^u\}_{i=1}^{N_u}$, where N_s and N_u are the size of labeled and unlabeled set respectively. Typically, we expect $N_s << N_u$. The label y^s contains information about both object categories and coordinates of bounding boxes. To train an object detector under this setting, the general form of loss is decoupled into a supervised loss (L_s) and an unsupervised loss (L_u):

$$\mathcal{L} = \mathcal{L}_s + \beta \mathcal{L}_u \tag{1}$$

where β is a tunable weight scaling for unsupervised loss.

General Framework. Our method adopts the Teacher-Student framework (Tarvainen & Valpola, 2017; Liu et al., 2021) which has been proven to be an effective approach in semi-supervised learning. Specifically, we maintain two copies of the network weights standing for **Teacher** and **Student**. The Student network is updated based on both the supervised loss and unsupervised loss as shown in Equation 1, whereas the Teacher model is only updated by computing the exponential moving average (EMA) of the Student model weights:

$$\theta_{Teacher} \leftarrow \alpha \theta_{Teacher} + (1 - \alpha)\theta_{Student}. \tag{2}$$

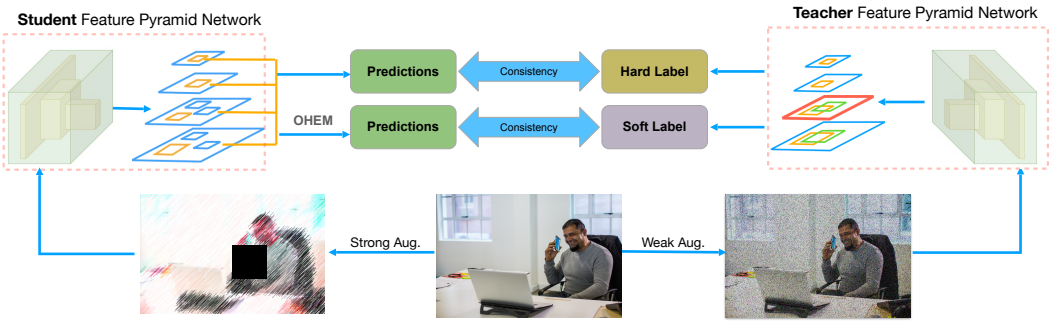


Figure 2: Overview of ***CrossMatch***. CrossMatch enforces consistency between different scales of feature maps. The teacher model takes weakly augmented images and produces artificial (in both soft and hard form, with a refined OHEM employed for the regions used for soft) with a **heuristically selected level of features** (denoted by red). The student model takes strongly augmented images and constantly makes predictions with **every level of the feature pyramid**. The consistency constraint is enforced through both soft and hard pseudo-labels.

Another important aspect of recent success in semi-supervised learning for both image classification (Sohn et al., 2020a; Xie et al., 2020a) and object detection (Liu et al., 2021; Zhou et al., 2021; Tang et al., 2021; Li et al., 2020) is the Weak-Strong Augmentation Scheme. When incorporating this scheme with our Teacher-Student framework, weakly augmented images are passed to the Teacher model for more reliable artificial labels and strongly augmented images are used as inputs to the Student model for higher diversity of input images.

Since the model is usually underfit in the initial training phase, we allow the model to warm-up for a few iterations, which is known as the Burn-In stage in Unbiased Teacher (Liu et al., 2021). During the Burn-In stage, the Student network is trained with only supervised loss and no EMA is applied on the Teacher model. At the end of Burn-In stage, the Teacher model is initialized by the Student model weights and updated by EMA only in the rest of training. Following Faster R-CNN (Ren et al., 2015), the supervised loss has four major components: the RPN classification loss $\mathcal{L}_{cls}^{rpn}$, the RPN regression loss $\mathcal{L}_{reg}^{rpn}$, the RoI classification loss $\mathcal{L}_{cls}^{roi}$, and the RoI regression loss $\mathcal{L}_{reg}^{roi}$:

$$\mathcal{L}_s = \mathcal{L}_{cls}^{rpn}(x, y) + \mathcal{L}_{reg}^{rpn}(x, y) + \mathcal{L}_{cls}^{roi}(x, y) + \mathcal{L}_{reg}^{roi}(x, y). \tag{3}$$

where x and y are input images and the ground-truth annotations (class labels and bounding boxes) respectively.

3.1 CrossMatch

To address the issue of sparse and weak gradient signal in pseudo-labeling methods (Figure 1), we introduce CrossMatch. The intuition behind our method is that we can propagate gradient signals through multiple paths from multi-scale methods. Specifically, our proposed method for semi-supervised object detection enforces consistency *between different levels of the feature pyramid generated by the feature pyramid network (FPN)* (Lin et al., 2017a). See Figure 2 for illustration.

The FPN generates a set of multi-scale feature maps $\{P_2, P_3, P_4, P_5, P_6\}$, which are used in both stages of two-stage detectors[1]. The region proposal network (RPN) uses these feature maps to produce multi-scale region of interests (RoIs). Once the RoIs are collected, two-stage detectors like Faster-RCNN (Ren et al., 2015) assigns only one particular level of features to each RoI determined by a heuristic function (Equation 4) based on spatial dimensions of RoIs (and ignores other levels):

$$k = \lfloor k_0 + log_2(\sqrt{wh}/224) \rfloor \tag{4}$$

We argue that different levels of features on the feature pyramid can be viewed as feature representations at different scales and provide valuable information under semi-supervised settings. Therefore propose a novel method to enforce consistency between different scales. The fact that this includes smaller scales, but potentially in a soft manner (see Section 3.3), would result in a denser (but still

[1] P_6 is often excluded in the second stage

weak) signal that can be used for semi-supervised training. Specifically, the Student network makes predictions using each level of $\{P_2, P_3, P_4, P_5\}$ and the Teacher network generates artificial labels using the features determined by the heuristic function. Although we can also use all scales for the Teacher, the heuristic function ensures the appropriate scale is used to generate reliable artificial labels. We empirically found that using all scales for the Teacher is computationally intensive and did not improve performance. The unsupervised loss function is computed by averaging the loss between each of Student predictions and artificial labels, which is described in the next sections.

Unlike prior work that applies the consistency constraint with either soft labels (Laine & Aila, 2016; Tarvainen & Valpola, 2017; Berthelot et al., 2019; 2020) or hard labels (Sohn et al., 2020a), and in order to further densify the gradient signal, CrossMatch enforces consistency between different scales by using both soft and hard labels. We show in experiments that this is a crucial design for semi-supervised object detection especially with limited labeled data.

3.2 Consistency with Hard Labels

In this section, we describe the training process of enforcing cross-scale consistency with hard artificial labels.

Each unlabeled image is first weakly augmented and forwarded to the Teacher model for hard artificial label generation. The Teacher model assigns features to RoIs based on equation 4 as discussed earlier. Notice that RoIs with different spatial dimension may still be assigned to different levels but ultimately only one particular level is matched with one RoI. We apply the confidence thresholding to filter out low-confident RoIs and use non-maximum suppresion (NMS) to exclude highly overlapped RoIs. This process helps reduce confirmation bias and error accumulation from noisy hard artificial labels.

After label generation, the same image, which is transformed by strong augmentation operations, is forwarded to the Student model. The training process is similar to train the model with labeled data except that the Student model makes predictions using each level of the feature from the feature pyramid. The loss is computed in the exact same manner as the supervised loss where the supervision signal passes through both RPN and RoI heads of the Student network. Notice that we do not include the bounding box regression loss with hard labels because we found it did not help the training process as it is more difficult to obtain and filter accurate regression thresholds. This finding is consistent to prior work (Liu et al., 2021) and we therefore exclude the regression loss in the training procedure.

$$\mathcal{L}_p = \mathcal{L}_{cls}^{rpn}(x, y) + \mathcal{L}_{cls}^{roi}(x, y)) \tag{5}$$

As pointed out by prior work (Liu et al., 2021), the foreground-background imbalance issue in supervised object detection (Shrivastava et al., 2016; Lin et al., 2017b) still exists in semi-supervised object detection. Following existing work (Liu et al., 2021), we use multi-class focal loss (Lin et al., 2017b) in replacement of vanilla cross-entropy loss for RoI-Level predictions.

While this process is similar to pseudo-label methods in prior work (Zhou et al., 2021; Tang et al., 2021; Liu et al., 2021; Li et al., 2020), the key difference is that our method generates hard labels with one certain level of features and the loss is computed with Student output predicted with multiple levels of features. This approach introduces a consistency constraint between different scales and we demonstrate the its efficacy in the experiment section.

3.3 Consistency with Soft Labels

Limitation with Hard Labels Even though hard labelling has achieved recent success in semi-supervised object detection, there are still possible limitations to this approach. The standard object detection training process involves matching RoIs to labels. RoIs that are matched with labels (when the intersection-over-union score is above the threshold) are sampled as positive examples and the rest is treated as backgrounds (or ignored). However, due to confidence thresholding, it is possible that no (or inaccurate) psuedo-labels are generated in some object regions. Unlike semi-supervised image classification where the low-confident images are simply excluded from training, RoIs that are highly overlapped with those false-negative regions can be mistakenly treated as backgrounds which can be a potential source of error accumulation.

To tackle this problem, we propose to combine soft labels with hard labels. For each unlabeled image, we first use the Student's region proposal network (RPN) to produce a list of RoIs $\{R_i\}_{i=1}^{N}$ where N is the total number of proposals. This is a crucial difference from the hard label branch as the supervision signal from soft labels only passes through RoI heads of the Student model. We found the Teacher's RPN is more accurate in terms of generating proposals and thus yields better performance.

Next, for each RoI R_i, Student and Teacher make their own probability predictions on strongly and weakly augmented input images respectively. Again, the Student network makes predictions using all levels of features from the feature pyramid whereas the Teacher model only uses heuristically selected levels. The consistency loss is then computed as the KL Divergence between each of the Student's class probability distribution and the Teacher's probability distribution (p_s^k and p_t respectively):

$$\mathcal{L}_c = \sum_{k=0}^{M} D_{KL}(p_s^k | p_t) \tag{6}$$

where p_s^k stands for the Student prediction made with feature level k and M is the total number of levels of the feature pyramid.

Since the region proposal network typically generates a large number of RoIs and most of them are easy background examples, the imbalance issue also exists in soft labels. To alleviate the problem, we adopt Online Hard Example Mining (OHEM) (Shrivastava et al., 2016) as the sampling strategy in our soft label branch. The general idea is to sample RoIs based on the loss values: we compute the consistency loss for all RoIs but only select top-K of them with largest loss values for back-propagation where K is a tunable hyper-parameter. We make two refinements on the original form of OHEM in our scenario. First, we do not apply NMS scored by loss values because we find doing so significantly hurts the model performance. Moreover, selecting a small portion of RoIs with large loss values affects the stability of training in the initial phase. Therefore, we apply a ramp-down process on the value of K from the total number of RoIs to the desired value of K in a few iterations.

Note that the OHEM sampling is applied between Student predictions from each level of features and the Teacher prediction; it is possible that different RoIs are sampled for different levels of Student features. Therefore, the total number of RoIs involved in the soft label branch may be greater than K.

4 Experiments

Datasets. We evaluate and compare our methods with other state-of-the-art methods using two commonly used benchmark datasets: MS-COCO (Lin et al., 2014) and Pascal VOC (Everingham et al., 2010). Specifically, we use three different experiment protocols: (1)*COCO-standard*: we randomly sample 0.5, 1, 2, 5, and 10% of labeled training data as a labeled set and use the rest of the data as the unlabeled set. (2) *COCO-additional*: we use the entire labeled training set of MS COCO as our labeled set and use additional COCO2017-*unlabel* as our unlabeled set, and *Pascal VOC*: we use the VOC07-*trainval* as our labeled set and use VOC12-*trainval* as the unlabeled set.

Implementation Details. Our implementation of CrossMatch is based on Detectron2 (Wu et al., 2019). For fair comparison, we follow previous work(Liu et al., 2021; Zhou et al., 2021; Tang et al., 2021; Sohn et al., 2020b) to use Faster-RCNN (Ren et al., 2015) with ResNet-50-FPN backbone. We only perform hyper-parameter tuning on our newly-introduced hyper-parameters and keep other hyper-parameters the same as convention (as in Detectron2 (Wu et al., 2019). We provide a comprehensive list of newly-introduced hyper-parameters in Appendix A.1.

Batch size is a critical factor for model performance. To enforce fair comparison, all our experiments are conducted with batch size of 16 (the ratio between labeled and unlabeled data is 1:1), which is consistent with most prior work (Zhou et al., 2021; Tang et al., 2021; Sohn et al., 2020b). We re-run Unbiased Teacher (Liu et al., 2021) using the publicly available codebase with the batch size of 16.

We follow Unbiased Teacher (Liu et al., 2021) for data augmentation policies since exploring better data augmentation operations is not the focus of our paper. Our augmentation policy does not include advanced augmentation operations such as geometric transformation as in STAC or MixUp (Zhang et al., 2017) and Mosaic (Bochkovskiy et al., 2020) as in Instant-Teaching. The full list of augmentation can be found in Appendix A.2. That is to say, the improvement reported in this

Table 1: Experimental results on *COCO-standard* comparing with other state-of-the-art methods. Specifically, we compare our method with CSD (Jeong et al., 2019), STAC (Sohn et al., 2020b), Humble Teacher (Tang et al., 2021), Instant Teaching (Zhou et al., 2021), ISMT (Yang et al., 2021), and Unbiased Teacher (Liu et al., 2021). The results of CSD are re-implemented by Liu et al. (2021).

	COCO-standard					COCO-additional
	0.5%	1%	2%	5%	10%	
Supervised	6.83 ± 0.15	9.05 ± 0.16	12.70 ± 0.15	18.47 ± 0.22	23.86 ± 0.81	37.63
CSD	7.41 ± 0.21 (+0.58)	10.51 ± 0.06 (+1.46)	13.93 ± 0.12 (+1.23)	18.63 ± 0.07 (+0.16)	22.46 ± 0.08 (-1.40)	38.82 (+1.19)
STAC	9.78 ± 0.53 (+2.95)	13.97 ± 0.35 (+4.92)	18.25 ± 0.25 (+5.55)	24.38 ± 0.12 (+5.86)	28.64 ± 0.21 (+4.78)	39.21(+1.58)
Humble Teacher	-	16.96 ± 0.38 (+7.91)	21.72 ± 0.24 (+9.02)	27.70 ± 0.15 (+9.23)	31.61 ± 0.28 (+7.74)	42.37(+4.74)
Instant Teaching	-	18.05 ± 0.15 (+9.00)	22.45 ± 0.15 (+9.75)	26.75 ± 0.05 (+8.28)	30.40 ± 0.05 (+6.54)	40.20 (+2.57)
ISMT	-	18.88 ± 0.74 (+9.83)	22.43 ± 0.56 (+9.73)	26.37 ± 0.24 (+7.9)	30.53 ± 0.52 (+6.67)	39.64(+3.01
Unbiased Teacher	14.36 ± 0.09 (+7.53)	18.33 ± 0.19 (+9.28)	22.23 ± 0.21 (+9.53)	26.65 ± 0.31 (+8.18)	29.56 ± 0.24 (+5.7)	41.30(+3.67)
CrossMatch (Ours)	**17.33 ± 0.18** (+10.50)	**21.53 ± 0.39** (+12.48)	**24.74 ± 0.21** (+12.04)	**28.77 ± 0.24** (+10.3)	**31.78 ± 0.18** (+7.92)	**42.62**(+4.99)

paper does not come from more aggressive data augmentation or larger batch sizes. More details of analysis of our method can be found in Section 5.

4.1 Results on COCO-standard and COCO-additional

We demonstrate the efficacy of our method on MS COCO dataset. For *COCO-standard*, all of our experiments are conducted with 5 runs with different sampling seed and we report the average and standard deviation of these runs. We report our results on MS COCO 2017 *val* set.

As shown in Table 1, CrossMatch consistently outperforms all other competing methods under all experimental protocols under the settings of both *COCO-standard* and *COCO-additional*. Specifically, CrossMatch achieves 17.33 mAP and 21.53 mAP with only 0.5% and 1% labeled data respectively, which advances previous state-of-the-art results by around 3 mAP. Our method trained with 0.5% labeled data even outperforms CSD (Jeong et al., 2019), STAC (Sohn et al., 2020b) and Humble Teacher (Tang et al., 2021) with 1% labeled data. Humble teacher uses additional augmentation operations in training, which puts our method at a disadvantage in comparison. Even so, CrossMatch still outperforms it under all evaluation settings.

Since semi-supervised learning becomes more challenging when the labeled data is very limited and research in this domain typically has a focus on low-labeled scenarios, the experimental results provide a solid validation on the efficacy of our method. Moreover, notice that we do not use any common tricks in object detection such as large batch sizes or more aggressive augmentations, the improvement indeed comes from our proposed components. See Section 5 for more details.

4.2 Results on Pascal VOC

We further compare CrossMatch with other methods on Pascal VOC. Following prior work (Jeong et al., 2019; Sohn et al., 2020b), we use VOC07 as the labeled set and VOC12 (plus COCO-20[2]) as unlabeled set. Since AP_{50} is indicated as a saturated metric by existing work (Cai & Vasconcelos, 2018), we further include $AP_{50:95}$ as an additional metric for comparison. Our results are evaluated on VOC07 *test* set. Results shown in Table 2 is computed with **COCO-Style AP calculation**. Some prior work (Tang et al., 2021) reported Pascal-Style AP calculation results, which is substantially different from the COOC-Style AP. We include our results with Pascal-Style AP and compare with these methods in Appendix A.3.

As shown in Table 2, CrossMatch outperforms several state-of-the-art methods by a large margin in terms of both AP_{50} and $AP_{50:95}$, which indicates its generalizability across multiple datasets. CrossMatch achieves 8.24 and 9.87 absolute mAP improvement over the supervised baseline when using VOC12 and VOC12 plus COCO20 as unlabeled set respectively.

[2]COCO-20 denotes the data sampled from MS-COCO whose corresponding categories are in Pascal VOC

Table 2: Results on *VOC* with COCO-Style AP Calculation.

	Backbone	Labeled	Unlabeled	AP_{50}	$AP_{50:95}$
Supervised (ours)	ResNet50-FPN	VOC07	None	72.63	42.13
CSD (Jeong et al., 2019)	ResNet101-R-FCN	VOC07	VOC12	74.70 (+2.07)	-
STAC (Sohn et al., 2020b)	ResNet50-FPN			77.45 (+4.82)	44.64 (+2.51)
Unbiased Teacher (Liu et al., 2021)	ResNet50-FPN			75.95 (+2.32)	48.86 (+6.73)
ISMT (Yang et al., 2021)	ResNet50-FPN			77.23 (+4.6)	46.23 (+4.1)
CrossMatch (Ours)	ResNet50-FPN			78.25 (+5.62)	**50.37** (+8.24)
CSD (Jeong et al., 2019)	ResNet101-R-FCN	VOC07	VOC12 + *COCO20cls*	75.10 (+2.47)	-
STAC (Sohn et al., 2020b)	ResNet50-FPN			79.08 (+2.78)	46.01 (+3.41)
Unbiased Teacher (Liu et al., 2021)	ResNet50-FPN			77.52 (+4.89)	49.71 (+7.58)
ISMT (Yang et al., 2021)	ResNet50-FPN			77.75 (+5.12)	49.59 (+7.46)
CrossMatch (Ours)	ResNet50-FPN			79.72 (+7.09)	**52.0** (+9.87)

Table 3: Ablation Study on 1% MS-COCO labeled data.

Soft Labels	**Hard Labels**	**Multi-Scale Features**	$AP_{50:95}$
✓			17.07
✓		✓	19.41
	✓		18.33
	✓	✓	19.32
✓	✓		20.53
✓	✓	✓	21.17

5 Ablation Study

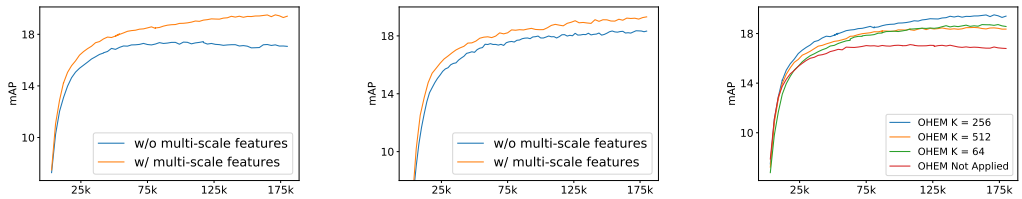
In this section, We conduct extensive analysis to understand the efficacy of our method where we mainly focued on our proposed modules. All experiments are conducted with 1% labeled data protocol from MS-COCO.

Importance of Each Component. To analyze the importance of each component, we evaluate the model trained with different combinations of soft labels, hard labels, and multi-scale consistency. The results can be found in Table 3. It can be inferred from Table 3 that multi-scale features is critical for CrossMatch no matter what form of labels is used. Specifically, it achieves 2.34 and 0.99 absolute mAP improvement with soft and hard label respectively. Even with soft-hard label combination, the model still benefits from multi-scale features by 0.64 mAP. We further demonstrate its efficacy on each branch in Figure 3a and 3b. Both soft label branch and hard label branch generalizes better with multi-scale features. It is worth noting that the soft label branch shows a trend of overfitting without multi-scale features but achieves even better results than hard label branch with multi-scale features.

Combining soft and hard labels is another effective practice in CrossMatch. Unlike previous work, which only uses one of them, CrossMatch proposes to combine them during training. Table 3 suggests that simply combining soft and hard labels provides promising results. The model trained with soft and hard labels without multi-scale features could achieve 20.53 mAP, which already outperforms all other methods with 1% labeled data on MS-COCO as shown in Table 1.

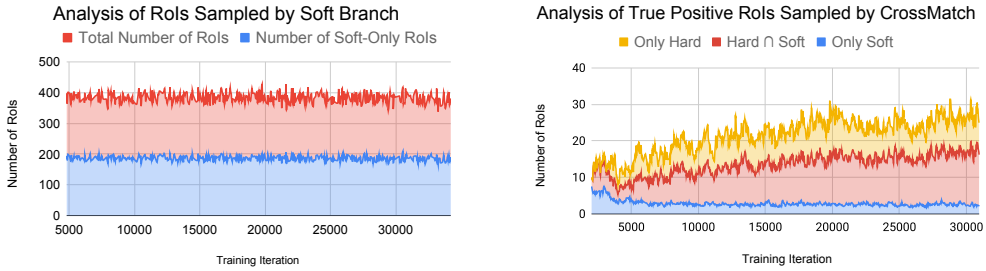
Analysis of Soft-Hard Label Combination. We now analyze the soft-hard label combination in terms of RoI sampling in detail. As shown in Figure 4a, approximately 50% of the RoIs sampled by the soft branch are not sampled by the hard branch (including both positive and negative RoIs). This shows that the soft branch brings in more diverse RoIs into the training process. Even though most unique RoIs from soft branch are backgrounds (by comparing Figure 4a and 4b), the soft branch tends to sample more challenging background RoIs instead of easy ones because of OHEM, which further enhances the gradient signal.

In terms of the true positive RoIs (Figure 4b), there is an overlap between RoIs sampled by soft and hard branches. Training with these RoIs works in a similar fashion as knowledge distillation (Hinton



(a) Soft Label Branch (b) Hard Label Branch (c) Effects of OHEM

Figure 3: (a) and (b): comparison between results with and without multi-scale features for soft label branch and hard label branch respectively. (c): comparison between different values of K for OHEM



(a) Analysis of RoIs Sampled by Soft Branch (b) Analysis of True Positive RoIs

Figure 4: (a): A quantitative analysis of RoIs sampled by soft branch. It can be inferred that around 50% of the RoIs sampled by soft branch is not sampled by hard branch. (b) A quantitative analysis of true positive RoIs. There is an overlap between RoIs sampled by soft branch and hard branch although each branch also selects its own unique samples.

et al., 2015) where the student model not only learns from accurate hard labels but also distills the rich and implicit information from the teacher model. Notice that the soft branch also brings in some unique true positive RoIs(that are overlooked by the hard branch, which further includes additional valuable supervision signal in the training process.

Importance of OHEM in the Soft Label Branch. We analyze the performance of OHEM in soft label branch. OHEM selects top-K RoIs for back-propagation based on the actual loss values. It first sorts all RoIs based on the loss values, then only picks top-K RoIs with highest values of loss.

To eliminate interference factors, we conduct experiments only using the soft label branch with multi-scale features in this section. As shown in Figure 3c, OHEM is a critical design choice for the soft label branch. Without OHEM, the soft label branch suffers from overfitting even with multi-scal features. On the other hand, the choice of K it also important. Too large of a K (*e.g.,* K=512) makes the gradient dominated by easy background examples in the training process whereas too small of a K (*e.g.,* K=64) forces the model to focus on only hard and potentially noisy examples. Both of the cases leads to sub-optimal performance.

6 Conclusion

In this paper, we identify the issue of *weak and noisy* pseudo-labels resulting from applying pseudo-labeling based techniques within current object detection models. In order to strengthen the gradient signal, we propose a novel, yet simple, training strategy to train an object detector with multi-scale features from the feature pyramid network. Based on our analysis, we further propose to densify the gradient signal by combining both hard and soft consistency across artificial labels, where soft labels come from *dense* ROIs that we show through analysis leverage additional ROIs compared to only hard. We further apply a refined OHEM method for the source of soft artificial labels. We show through analysis and experimentation that our method works particularly well when the number of labeled images is limited and significantly outperform other methods under this scenario.

7 REPRODUCIBILITY

To ensure minimal effort to reproduce our work, we list development details of CrossMatch. This work is developped with the publicly available codebase of Unbiased Teacher (Liu et al., 2021) and all core designs are described with details in Section 3. Starting from the codebase of Unbiased Teacher and adding proposed components of our work should be straightforward. We further include a comprehensive list of our newly-introduced hyper-parameters in Appendix A.1 and exact data augmentation operations in Appendix A.2, which further improves the transparency of our work.

We will make a confidential post (only visible to reviewers and ACs) containing an anonymous link to our code on OpenReview after the deadline and make our code publicly available if the paper is accepted.

REFERENCES

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 (IJCNN)*, pp. 1–8. IEEE, 2020.

Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. *arXiv preprint arXiv:1412.4864*, 2014.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*, 2019.

David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. *International Conference on Learning Representations*, 2020.

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

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*, pp. 6154–6162, 2018.

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.

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/CVF International Conference on Computer Vision*, pp. 9508–9517, 2019.

Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7036–7045, 2019.

Chaoxu Guo, Bin Fan, Qian Zhang, Shiming Xiang, and Chunhong Pan. Augfpn: Improving multi-scale feature learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12595–12604, 2020.

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

Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5070–5079, 2019.

Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. 2019.

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

Dong-Hyun Lee et al. 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.

Yandong Li, Di Huang, Danfeng Qin, Liqiang Wang, and Boqing Gong. Improving object detection with selective self-supervised self-training. In *European Conference on Computer Vision*, pp. 589–607. Springer, 2020.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

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*, pp. 2117–2125, 2017a.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017b.

Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8759–8768, 2018.

Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. *arXiv preprint arXiv:2102.09480*, 2021.

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

Sangmin Oh, Anthony Hoogs, Amitha Perera, Naresh Cuntoor, Chia-Chih Chen, Jong Taek Lee, Saurajit Mukherjee, JK Aggarwal, Hyungtae Lee, Larry Davis, et al. A large-scale benchmark dataset for event recognition in surveillance video. In *CVPR 2011*, pp. 3153–3160. IEEE, 2011.

Jiangmiao Pang, Kai Chen, Jianping Shi, Huajun Feng, Wanli Ouyang, and Dahua Lin. Libra r-cnn: Towards balanced learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 821–830, 2019.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *arXiv preprint arXiv:2101.06329*, 2021.

Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. 2005.

Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *arXiv preprint arXiv:1606.04586*, 2016.

Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 761–769, 2016.

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. *Advances in Neural Information Processing Systems*, 33, 2020a.

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*, 2020b.

Yihe Tang, Weifeng Chen, Yijun Luo, and Yuting Zhang. Humble teachers teach better students for semi-supervised object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3132–3141, 2021.

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*, pp. 2119–2128, 2016.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *arXiv preprint arXiv:1703.01780*, 2017.

Xinjiang Wang, Shilong Zhang, Zhuoran Yu, Litong Feng, and Wayne Zhang. Scale-equalizing pyramid convolution for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13359–13368, 2020.

Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33, 2020a.

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*, pp. 10687–10698, 2020b.

Qize Yang, Xihan Wei, Biao Wang, Xian-Sheng Hua, and Lei Zhang. Interactive self-training with mean teachers for semi-supervised object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5941–5950, 2021.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Qiang Zhou, Chaohui Yu, Zhibin Wang, Qi Qian, and Hao Li. Instant-teaching: An end-to-end semi-supervised object detection framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4081–4090, 2021.

A Appendix

A.1 Hyper-parameter Details

We include details of our hyper-parameters in this section. It can be viewed in Table 4.

Table 4: Meanings and values of the hyper-parameters used in experiments.

Hyper-parameter	Description	*MS COCO*	*Pascal VOC*
δ	Confidence threshold for hard artificial labels	0.7	0.7
λ_h	Weight for hard label loss	1.0	1.0
λ_s	Weight for soft label loss	1.0	1.0
K	Number of RoIs sampled by OHEM per level for soft label	256	512
α	EMA rate	0.9996	0.9996
b_l	Batch size for labeled data	8	8
b_u	Batch size for unlabeled data	8	8
γ	Learning rate	0.01	0.01

A.2 Data Augmentation Details

Table 5 shows details of our data augmentation. Following Unbiased Teacher, we use random horizontal flip as the weak augmentation and include color jittering, gray scale, Gaussian blur, and cutout with different probabilities as the strong augmentation. Notice that we do not use more complex augmentation operations such as geometric transformations as in STAC or Mix-up and Mosaic as in Instant-Teaching. We believe that with additional strong data augmentation, experimental results of CrossMatch can be further improved.

A.3 Results on Pascal VOC with Pascal-style AP

The Pascal-Style AP calculation is substantially different from the COCO-Style AP calculation(which is more commonly used). In this section, we compare with prior work that evaluates their model with the Pascal-Style AP. As shown in Table 6, our method significantly outperforms Humble Teacher (Tang et al., 2021) by 2.32 mAP and 2.11 mAP when using VOC-12 and VOC12 plus COCO-20 as unlabeled data. The results demonstrate that the strong performance of our method on Pascal VOC is independent to evaluation metrics.

Table 5: Detail of data augmentations. Probability in the table indicates the probability of applying the corresponding image process.

Weak Augmentation			
Process	Probability	Parameters	Descriptions
Horizontal Flip	0.5	-	None
Strong Augmentation			
Process	Probability	Parameters	Descriptions
Color Jittering	0.8	(brightness, contrast, saturation, hue) = (0.4, 0.4, 0.4, 0.1)	Brightness factor is chosen uniformly from [0.6, 1.4], contrast factor is chosen uniformly from [0.6, 1.4], saturation factor is chosen uniformly from [0.6, 1.4], and hue value is chosen uniformly from [-0.1, 0.1].
Grayscale	0.2	None	None
GaussianBlur	0.5	(sigma_x, sigma_y) = (0.1, 2.0)	Gaussian filter with $\sigma_x = 0.1$ and $\sigma_y = 2.0$ is applied.
CutoutPattern1	0.7	scale=(0.05, 0.2), ratio=(0.3, 3.3)	Randomly selects a rectangle region in an image and erases its pixels. We refer the detail in **?**.
CutoutPattern2	0.5	scale=(0.02, 0.2), ratio=(0.1, 6)	Randomly selects a rectangle region in an image and erases its pixels. We refer the detail in **?**.
CutoutPattern3	0.3	scale=(0.02, 0.2), ratio=(0.05, 8)	Randomly selects a rectangle region in an image and erases its pixels. We refer the detail in **?**.

Table 6: Results on *VOC* using Pascal-style AP Calculation.

	Backbone	Labeled	Unlabeled	AP_{50}	$AP_{50:95}$
Supervised	ResNet50-FPN	VOC07	None	72.63	42.13
Humble Teacher (Tang et al., 2021)	ResNet50-FPN	VOC07	VOC12	80.94 (+8.31)	53.04 (+10.91)
CrossMatch (Ours)	ResNet50-FPN			81.48 (+8.85)	**55.36** (+13.23)
Humble Teacher (Tang et al., 2021)	ResNet50-FPN	VOC07	VOC12	81.29 (+8.66)	54.41 (+12.28)
CrossMatch (Ours)	ResNet50-FPN		*COCO20cls*	82.62 (+9.99)	**56.52** (+14.39)

A.4 Comparison between Our Method and AugFPN

AugFPN proposes Consistent Supervision to propagate supervision signals through multiple levels of the feature pyramid. Though the focus of AugFPN is on regular object detection, here we clarify the difference between our multi-scale consistency method and Consistent Supervision in AugFPN. Feature Pyramid Network (FPN) first extracts feature maps $\{C_2, C_3, C_4, C5\}$ from the backbone and passes them through lateral connections for channel alignment to generate $\{M_2, M_3, M_4, M_5\}$. The final feature pyramid $\{P_2, P_3, P_4, P_5\}$ is created through a top-down path from $\{M_2, M_3, M_4, M_5\}$. AugFPN attaches auxiliary classification and regression head to $\{M_2, M_3, M_4, M_5\}$ and computes an auxiliary loss based on network predictions (with ground-truth labels). In contrast, Our multi-scale consistency is different from AugFPN in the following ways. First, we do not use an auxiliary prediction heads as in AugFPN. Next, instead of propagating gradient signals through $\{M_2, M_3, M_4, M_5\}$, our method uses $\{P_2, P_3, P_4, P_5\}$ as the multi-scale features. Finally, our losses are computed with soft and hard artificial labels rather than ground-truth labels as in AugFPN.