SparseDet: Improving Sparsely Annotated Object Detection with Pseudo-positive Mining

Saksham Suri*[1] sakshams@cs.umd.edu
Saketh Rambhatla*[1] rssaketh@umd.edu
Rama Chellappa[2] rchella4@jhu.edu
Abhinav Shrivastava[1] abhinav@cs.umd.edu
University of Maryland, College Park[1] Johns Hopkins University[2]

Abstract

Training with sparse annotations is known to reduce the performance of object detectors. Previous methods have focused on proxies for missing ground truth annotations in the form of pseudo-labels for unlabeled boxes. We observe that existing methods suffer at higher levels of sparsity in the data due to noisy pseudo-labels. To prevent this, we propose an end-to-end system that learns to separate the proposals into labeled and unlabeled regions using Pseudo-positive mining. While the labeled regions are processed as usual, self-supervised learning is used to process the unlabeled regions thereby preventing the negative effects of noisy pseudo-labels. This novel approach has multiple advantages such as improved robustness to higher sparsity when compared to existing methods. We conduct exhaustive experiments on five splits on the PASCAL-VOC and COCO datasets achieving state-of-the-art performance. We also unify various splits used across literature for this task and present a standardized benchmark. On average, we improve by 2.6*,* 3.9 *and* 9.6 *mAP over previous state-of-the-art methods on three splits of increasing sparsity on COCO. Our project is publicly available at* cs.umd.edu/~sakshams/SparseDet.

1. Introduction

The performance of object detectors is sensitive to the quality of labeled data [1–3]. Existing object detection methods assume that the training data is pristine and a drop in performance is observed if this assumption fails. Noise in the data used for training object detectors can arise due to incorrect class labels or incorrect/missing bounding boxes. In this work, we deal with the problem of training object detectors with sparse annotations, i.e., missing region or bounding boxes. This problem is of utmost importance, as obtaining crowd-sourced datasets [4, 5] can be expensive and laborious. The alternative is to use computer-assisted protocols to collect annotations which have been shown to be noisy and incomplete [6]. This problem of training object detectors with incomplete bounding box annotations is called Sparsely Annotated Object Detection (SAOD).

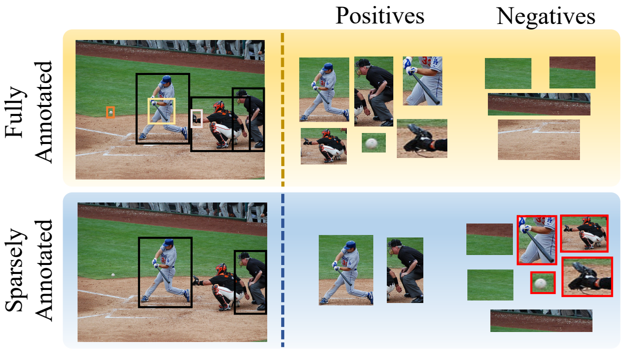

Figure 1: (Top) Most Object Detection datasets have exhaustive annotations for foreground/positives. During training, the unlabeled regions can be safely considered as background/negatives. Sparsely Annotated Object Detection datasets (bottom) have missing annotations. This results in foreground regions (shown in red) being considered as negatives during training, adversely affecting the performance of the classifier.

To understand why training with sparse annotations is detrimental to the performance, consider the example shown in Figure 1 (top). If the annotation were exhaustive, then the negative samples to the classifier contain *true* background regions. But with sparse annotations, as shown in Figure 1 (bottom), a few positive regions will inevitably be considered as negatives (shown in red), thereby wrongfully penalizing the classifier leading to lower performance. Existing methods[2, 7–10] prevent this by predicting pseudo-labels and removing the foreground regions from the negatives to the classifier. However, at higher levels of sparsity, the quality of pseudo-labels is greatly affected, resulting in the same problem noted above.

*First two authors contributed equally

Crowd sourced object detection datasets [5, 11, 12] are ensured to be almost exhaustively labeled. Hence, for SAOD, researchers artificially create sparsely annotated datasets from the original ones. There is no general consensus on the correct way to create the sparsely annotated datasets, *a.k.a.* splits, and hence each method reports results on one or two different splits. Split can be created by considering the dataset as a whole or per image (*i.e.* annotations can be removed by considering all the images or only a single image at once). They can also be created by removing annotations in a class-agnostic or class-aware fashion (*i.e.* remove $p\%$, of annotations per category or across all the categories). These variations result in splits with different data distributions making some harder than the others (refer to Table 1). A proper benchmark that analyzes the performance of SAOD methods across these different types of splits is missing. This makes it difficult to compare methods and assess their effectiveness for a specific use case.

To tackle the issues discussed above, as our first major contribution, we present SparseDet, a novel SAOD framework that achieves state-of-the-art performance across multiple SAOD benchmarks in practice. SparseDet operates on an image and its augmented counterpart. The combination of features extracted from the two views is used to generate region proposals. Standard detection training methods, consider a region proposal as positive if its intersection over union (IoU) with any ground truth is greater than 0.5 and the rest are treated as negatives. This strategy works when the annotations are exhaustive, which implies that the remaining regions are from background. But due to missing annotations, some of these regions could belong to foreground instances. To prevent considering all region proposals without annotations as negatives, SparseDet partitions all the region proposals into labeled, *unlabeled* and background. The labeled and background regions are processed as usual. Features extracted from unlabeled regions are then trained with a self-supervised loss. Previous approaches like Co-Mining [10], consider two partitions, labeled and background, and generate pseudo-labels. This is a disadvantage at high sparsity as the generated pseudo-labels can be very noisy. The self-supervised loss in our approach enforces consistency between the features of the two views for the unlabeled regions and prevents penalizing the classifier due to false negatives.

Our second major contribution is unifying evaluation. The standard practice is to simulate sparsely annotated training data on COCO [5] and PASCAL-VOC [11] train sets and evaluate them on their corresponding standard validation set. As discussed above, a survey of recent SAOD approaches [2, 7–10, 13–16] reveals that there are at least ***five*** different ways to create splits, each differing in the strategy (refer to Section 4.2) used to achieve the desired level of sparsity. However, these splits have not been made public, making it difficult to replicate results for comparison. Additionally, each of these strategies has a different property for simulating sparse data, *e.g*., different distribution of annotations per class, resulting in a different training set. As a result, methods trained on different splits cannot be compared with one another. To mitigate these issues we standardize the generation of these splits that enables the evaluation of any SAOD methods on all of them for fair comparison. Additionally, we propose a new benchmark that assesses the semi-supervised learning capabilities of SAOD methods, *i.e*., leveraging unlabeled data to improve performance. We present our approach as a baseline. We will make the data for the new benchmark along with all the SAOD splits public to facilitate future research. We briefly summarize our contributions below.

- We propose a novel formulation, SparseDet, for SAOD which is an end-to-end approach that identifies labeled, unlabeled and background regions and deals with them in appropriate manner.
- We show state-of-the-art performance on sparsely annotated object detection across various splits. On average, we improve by 2.6, 3.9 and 9.6 mAP over previous state-of-the-art methods on three splits of increasing sparsity on COCO.
- We standardize the experimental setup for SAOD by evaluating methods on all the splits to facilitate future research. Additionally, we propose a new benchmark that evaluates the semi-supervised learning capabilities of SAOD methods.

In Section 2, we discuss the related works on SAOD and related fields. We describe our approach in detail in Section 3. We describe our experimental setup and present results in Section 4, and conclude in Section 5.

2. Related Work

Semi-supervised object detection: Semi-supervised object detection (SSOD) is an active field that also deals with training object detectors with missing annotations. Existing works on semi-supervised object detection, have focused mainly on consistency regularization [17, 18] or pseudo-labeling-based approaches [19–22]. The main idea behind these approaches is to perturb the images, or features, and apply a consistency regularization loss to enforce consistency between the predictions using a student teacher framework. However, typical SSOD methods assume a small exhaustively labeled set and a large unlabeled set for training. This is different from Sparsely annotated object detection (SAOD), which assumes a large training set which is sparsely labeled.

Sparsely annotated object detection: One of the initial works addressing this problem, by Niitani *et al.* [7], proposes utilizing logical relations between the co-occurrences of objects and pseudo-labeling. Yoon *et al.* [23] use object tracking to densely label objects across sparsely annotated frames along with single stage detectors to mitigate the negative effects of missing annotations. Wu *et al.* [2] propose a re-weighting approach where the gradients corresponding to region of interest are weighed as a function of overlap with ground truth instances. Improving upon the previous work, Zhang *et al.* [8] propose an automatic re-calibration strategy for single stage detectors where the negative branch is changed to take into account low confidence background predictions which might correspond to missing annotations.

Finally, one of the most recent works, Co-mining [10] uses a co-training strategy by using two views of an image and predictions from one view along with the ground truth as supervision for the other view and vice versa. Our method doesn't solely rely on pseudo-labels and leverages a self-supervised loss to prevent propagating negative gradients to the model due to false negatives.

Fully supervised object detection: After the success of AlexNet [24] on the image classification challenge (ILSVRC 2012) [25], research on designing deep neural networks for object detection gathered more interest. First among the successful methods are the two-stage **Region-based convolutional networks** (R-CNN) family of detectors. Two-stage object detectors consists of 1) a region proposal stage which produces a set of candidate object bounding boxes followed by 2) a classification stage which classifies each candidate region as either belonging to a foreground object or "background". R-CNN processes a large amount of region proposals by cropping the input image and using a CNN backbone to extract features making it extremely slow. **Fast R-CNN** [26] was proposed to overcome this limitation. Fast R-CNN computes one convolution feature map for the whole image. RoI Pooling was introduced to pool the feature for each region of interest into a fixed spatial dimension. RoI pooling shares the computation among all the region proposals speeding up training and inference. Fully connected layers are applied on the fixed RoI pooled feature maps which are then passed to two sister heads, for classification and bounding box regression. The whole network is trained end-to-end with a multi-task loss avoiding the multi-stage training in R-CNN [27]. While Fast R-CNN improved the efficiency of its predecessor, test time computation bottleneck is still an issue because of the region proposal method employed. **Faster R-CNN** [28] proposed a region proposal network (RPN), an elegant solution that trains deep networks to predict region proposals for practically no additional computational overhead. RPN shares convolutional layers with Fast R-CNN [26] and at test time the cost of generating proposals is minimal. Faster R-CNN paved the way for more sophisticated and efficient two stage detection architectures [29–33]. Faster R-CNN has also been extended to achieve state-of-the-art instance segmentation [34], panoptic segmentation [35, 36], and 3D mesh generation [37] *etc*. A few limitations of two stage object detectors and anchor boxes has also inspired the family of single-stage [38–40] and anchor-free object detection systems [41, 42].

3. Approach

We present SparseDet for Sparsely Annotated Object Detection. Given N images, we denote the set of labeled regions in the dataset as $\mathcal{B}_\text{l} = \{b_i, c_i\}_{i=1}^{\text{N}_\text{l}}$ where (b_*, c_*) are the bounding box and class labels respectively. The unlabeled regions are denoted as $\mathcal{B}_\text{ul} = \{b_k\}_{k=1}^{\text{N}_\text{ul}}$ and the background regions as $\mathcal{B}_\text{bg} = \{b_q\}_{q=1}^{\text{N}_\text{bg}}$. Note that unlike existing SAOD approaches, we do not assume images to contain at least one labeled region in this work. The unlabeled $\mathcal{B}_\text{ul}$ and background $\mathcal{B}_\text{bg}$ sets are not known a-priori.

3.1. Overview

The proposed approach is shown in Figure 2 and consists of a backbone network that extracts features from the original and augmented views of an image. The common RPN (C-RPN), concatenates the features, to generate a set of region proposals. A region proposal b can belong to one of three groups, namely 1) labeled regions $b \in \mathcal{B}_\text{l}$, 2) unlabeled foreground regions $b \in \mathcal{B}_\text{ul}$, or 3) background regions $b \in \mathcal{B}_\text{bg}$. For a given set of ground-truth annotations, the first group, i.e. labeled regions can be automatically identified. The problem then reduces to identifying and separating the second group i.e. the unlabeled regions, from the background regions. Given all the region proposals, a pseudo-positive mining (PPM) step identifies the unlabeled regions and segregates them from the background regions. The labeled and unlabeled regions are trained using supervised and self-supervised losses respectively. We describe each stage in detail below.

3.2. Feature Extraction

Given an image I, an augmented version of I denoted as $\mathcal{A}(I)$ is computed. In this work, we use random contrast, brightness, saturation, lighting and bounding box erase in a cascaded fashion to generate $\mathcal{A}(I)$. A backbone network is employed to extract two features, f_o and f_a, from I and $\mathcal{A}(I)$ respectively.

3.3. Common RPN (C-RPN)

Two stage object detectors [28, 30, 43] use a region proposal network (RPN) [28] to generate regions of interest (RoIs). We propose C-RPN which concatenates f_o and f_a to obtain the RoIs. This is different from existing approaches that use f_o and f_a to generate two separate sets of RoIs. Operating on two sets increases the difficulty of identifying the labeled, unlabeled and background regions which

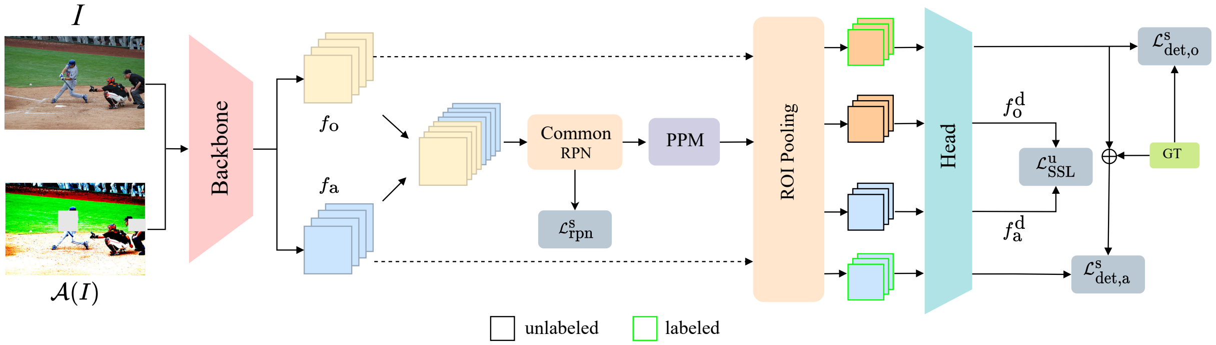
I
$\mathcal{A}(I)$
Backbone
f_{o}
f_{a}
Common RPN
$\mathcal{L}^{\mathrm{s}}_{\mathrm{rpn}}$
PPM
ROI Pooling
Head
$f^{\mathrm{d}}_{\mathrm{o}}$
$f^{\mathrm{d}}_{\mathrm{a}}$
$\mathcal{L}^{\mathrm{u}}_{\mathrm{SSL}}$
$\mathcal{L}^{\mathrm{s}}_{\mathrm{det,o}}$
GT
$\mathcal{L}^{\mathrm{s}}_{\mathrm{det,a}}$
□ unlabeled
🟩 labeled

Figure 2: Illustration of SparseDet for sparsely annotated object detection. Following feature extraction from the original and augmented image, a common set of proposals is generated by the common RPN (C-RPN). Using an end to end approach, we identify and mine proposals corresponding to missing annotations using pseudo positive mining (PPM). We train the network end-to-end using a combination of supervised and self-supervised losses. The unlabeled instances (black) are supervised with a self-supervised loss and the labeled instances (green) are supervised with ground truth annotations.

are required for subsequent stages. The sparse ground truth is used as supervision to train the C-RPN with a binary cross entropy loss for foreground classification and smooth L1 [44] loss for bounding box regression as shown below:

$$\mathcal{L}^{\mathrm{s}}_{\mathrm{rpn}}\left(\mathbf{c},\mathbf{b},\mathbf{c}^{*},\mathbf{b}^{*}\right)=\mathcal{L}_{\mathrm{bce}}\left(\mathbf{c},\mathbf{c}^{*}\right)+\lambda\mathcal{L}_{\mathrm{reg}}\left(\mathbf{b},\mathbf{b}^{*}\right) \tag{1}$$

where $\mathbf{c}$, $\mathbf{b}$ are the objectness scores and bounding boxes, and $\mathbf{c}^{*}$, $\mathbf{b}^{*}$ are the corresponding ground truth.

3.4. Pseudo Positive Mining (PPM)

Given the RoIs from C-RPN, the next step is to seggregate the unlabeled regions from the labeled and background regions. A standard practice in training detectors is to consider all proposals with an objectness score greater than τ_{obj} and IoU greater τ_{fg} with any ground truth as foreground and proposals with IoU less than τ_{bg} as background. In the presence of missing annotations, detectors are wrongly penalized because the IoU of RoIs corresponding to missing annotations with any ground truth is less than τ_{bg}. To avoid this, we mine "potential" foregrounds and consider them as unlabeled regions ($\mathcal{B}_{\mathrm{u}}$). The proposed PPM module is based on our observation that when trained with sparse annotations, the RPN can reliably distinguish foreground from background regions. We pick all RoIs that have an objectness score greater than τ_{ppm} and IoU less than τ_{bg} with any ground truth as the unlabeled regions. The remaining RoIs are assigned as background.

3.5. Losses

The pseudo positive mining step segregates the RoIs into labeled, unlabeled and background regions. An RoI pooling layer [28] extracts RoI pooled features for the labeled and background regions using the feature f_{o}. The detection head processes the RoI pooled features to predict class-wise probabilities and bounding box adjustments for each region. The ground truth is used to supervise the predictions by applying the cross entropy loss for classification and smooth L1 [44] for bounding box regression as shown below:

$$\mathcal{L}^{\mathrm{s}}_{\mathrm{det,o}}\left(\mathbf{c},\mathbf{b},\mathbf{c}^{*},\mathbf{b}^{*}\right)=\mathcal{L}_{\mathrm{ce}}\left(\mathbf{c},\mathbf{c}^{*}\right)+\lambda\mathcal{L}_{\mathrm{reg}}\left(\mathbf{b},\mathbf{b}^{*}\right) \tag{2}$$

where $\mathbf{c}$, $\mathbf{b}$ are the class and bounding box predictions, and $\mathbf{c}^{*}$, $\mathbf{b}^{*}$ are the corresponding ground truth class labels and bounding boxes.

All detections with a score greater than τ_{m} are combined with the ground truth followed by a class specific NMS to obtain the merged ground truth. It is ensured that no ground truth annotation is discarded during this step. The labeled and background regions along with f_{a} are used to extract RoI pooled features which are then fed to the detector head. The predictions of the detector head are supervised with the merged ground truth with the following losses:

$$\mathcal{L}^{\mathrm{s}}_{\mathrm{det,a}}\left(\mathbf{c},\mathbf{b},\mathbf{c}^{*},\mathbf{b}^{*}\right)=\mathcal{L}_{\mathrm{bce}}\left(\mathbf{c},\mathbf{c}^{*}_{\mathbf{m}}\right)+\lambda\mathcal{L}_{\mathrm{reg}}\left(\mathbf{b},\mathbf{b}^{*}_{\mathbf{m}}\right) \tag{3}$$

where $\mathbf{c}$, $\mathbf{b}$ are the class and bounding box predictions from the detector head, and, $\mathbf{c}^{*}_{\mathbf{m}}$, $\mathbf{b}^{*}_{\mathbf{m}}$ are the corresponding merged ground truths.

Finally, a class agnostic NMS is performed on the unlabeled regions $\mathcal{B}_{\mathrm{u}}$ (obtained after PPM described in Sec. 3.4). The unlabeled regions along with f_{o} and f_{a} are passed through the ROI pooling layer and the detection head to obtain $f^{\mathrm{d}}_{\mathrm{o}}$ and $f^{\mathrm{d}}_{\mathrm{a}}$ respectively. A self-supervised loss is applied that enforces the detection head features of the original and augmented regions to be consistent with each other as shown below:

$$\mathcal{L}^{\mathrm{u}}_{\mathrm{SSL}}\left(f^{\mathrm{d}}_{\mathrm{a}},f^{\mathrm{d}}_{\mathrm{o}}\right)=\|f^{\mathrm{d}}_{\mathrm{a}}-f^{\mathrm{d}}_{\mathrm{o}}\|^{2}_{2} \tag{4}$$

The network is trained end-to-end with both supervised and unsupervised losses as shown below:

$$\mathcal{L}=0.5*\left(\mathcal{L}^{\mathrm{s}}_{\mathrm{det,o}}+\mathcal{L}^{\mathrm{s}}_{\mathrm{det,a}}\right)+\mathcal{L}^{\mathrm{s}}_{\mathrm{rpn}}+\mathcal{L}^{\mathrm{u}}_{\mathrm{SSL}} \tag{5}$$

Table 1: **Sparsely annotated object detection** results on three splits of COCO dataset. "Oracle" corresponds to training models using all annotations. Results are reported on the COCO validation set using AP[0.50:0.95].

Method	**Split-1**			**Split-2**			**Split-3**			100%
	30%	50%	70%	30%	50%	70%	30%	50%	70%	
Oracle										40.91
Pseudo Label [7]	-	27.50	-	-	-	-	-	-	-	-
BRL [8]	-	32.70	-	-	-	-	-	-	-	-
Co-mining [10]	36.35	32.84	24.93	36.72	33.04	24.83	36.76	32.54	24.96	-
Ours	**38.22**	**35.92**	**32.68**	**39.76**	**36.94**	**35.33**	**39.56**	**37.15**	**35.48**	-

Discussion: In the absence of ground truth annotations, i.e. for completely unlabeled images, the supervised losses ($\mathcal{L}^{\mathrm{s}}_{\mathrm{det,o}}$, $\mathcal{L}^{\mathrm{s}}_{\mathrm{det,a}}$, $\mathcal{L}^{\mathrm{s}}_{\mathrm{rpn}}$) cannot be computed. Our proposed approach leverages self-supervised consistency loss that does not need ground truth. This helps our approach leverage these unlabeled regions unlike contemporary SAOD methods. We claim that this is the first method to use self-supervised losses for SAOD. Even though Co-Mining [10] claims to use self-supervised learning it is technically co-learning. Ours is the first approach to use pseudo-labeling and a self-supervised loss to handle the sparse annotations.

4. Experimental Evaluation

In this section, we describe the experiments to evaluate our proposed approach. In Sections 4.1 and 4.2, we describe data, splits, and metrics. In Section 4.3, we mention the implementation details followed by the baselines in Section 4.4. We compare with contemporary methods in Section 4.5, followed by an ablation study in Section 4.6.

4.1. Data and Metrics

We conduct all our experiments on the COCO [5] and PASCAL-VOC [11, 12] (2007+2012) datasets. The COCO [5] dataset consists of 118000 and 5000 images for training and validation respectively. Experiments on the PASCAL-VOC07 [11] are conducted on 5011 trainval images and performance is computed on 4952 images of the test set. The PASCAL-VOC12 version consists of 11530 (trainval) images for training and evaluation is done on the PASCAL-VOC07 test set. Following past literature [2, 7–10], we create five different splits (Section 4.2) and report results on them. For splits on the COCO [5] dataset, we use the standard COCO style Average Precision (AP). For splits on the PASCAL-VOC [11] dataset, we use the standard PASCAL-VOC style AP$_{50}$, which is Average Precision computed at an IoU threshold of 0.5.

4.2. SAOD Splits

An extensive review of sparsely annotated object detection methods reveals that there are atleast five popular types of splits in use for creating training data. Most methods report results on a subset of these making it harder to compare across methods.

We standardize the evaluation of SAOD methods by evaluating exhaustively on all the splits facilitating future research. We briefly describe the splits below.

Split-1 [9, 10]**:** In this split, for each object category, $p\%$ of annotations are randomly removed from the COCO [5] train set and results are reported on the validation set. This split simulates sparsity at dataset level in a class aware fashion. Note, this split can contains images with no annotations. We experiment with $p = \{30, 50, 70\}$.

Split-2: For each image in the COCO [5] train set, all annotations of $p\%$ of all categories in that image removed and results are reported on the COCO validation set. This split can be considered as image level and class aware. We experiment with $p = \{30, 50, 70\}$. Note, this split might contain images with no annotations.

Split-3: This split, which uses the COCO 2017 [5] train set for training and the validation set for evaluation, deletes $p\%$ annotations in a class agnostic fashion for each image ensuring at least one annotation. This split is image level but class agnostic. For the experiments, we use $p = \{30, 50, 70\}$.

Split-4 [8, 10]**:** This split requires evaluating models on three different settings namely *easy*, *hard* and *extreme* ensuring at least one annotation per image constructed using PASCAL-VOC 2007+12 [11, 12] trainval set. Results are reported on the PASCAL-VOC 2007 [11] test set. For each image in the training set, the easy setting randomly removes one annotation while the *hard* setting randomly removes half of the annotations. The *extreme* setting retains only one annotation per image. All the sets ensure each image consists of atleast one annotation. This split is a small scale version for an image level class agnostic split.

Split-5 [2]**:** This split uses the PASCAL-VOC 2007 [11] train set for training and the PASCAL-VOC 2007 test set for evaluation and drops $p\%$ annotations per class. For each image in this construction, instances of randomly selected categories are exhaustively annotated while the remaining categories do not have any annotations. This is a small scale version of Split-1 as annotations are dropped in a class aware manner across the full dataset maintaining atleast 1 annotation per image. In this case we use $p = \{30, 40, 50\}$.

Table 2: **Sparsely annotated object detection** results on two splits of PASCAL-VOC dataset. "Oracle" corresponds to training models using all annotations. Results are reported on VOC 07 test set usiing AP_{50}.

Method	Split-4			Split-5		
	Easy	Hard	Extreme	30%	40%	50%
Oracle	83.09			77.47		
BRL [8]	73.50	71.70	66.20	-	-	-
Co-mining [10]	79.59	78.38	69.60	74.42	73.30	69.89
Ours	**82.15**	**81.50**	**75.59**	**76.84**	**75.88**	**74.35**

Table 3: Results on the proposed **SSL+SAOD** setup. VOC12 [12] is used as the unlabeled data and p is the removal percentage

Method	AP		
	30%	40%	50%
Co-mining [10]	46.53	45.98	43.21
Ours	**48.34**	**47.82**	**46.76**

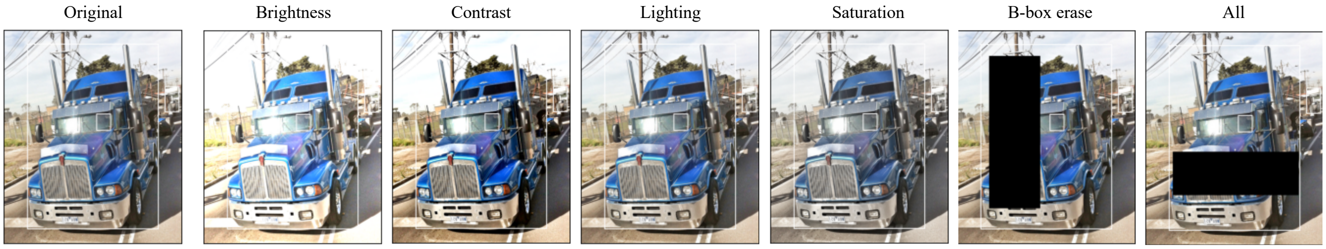

Figure 3: Illustration of the effects of various augmentations used in this work.

4.3. Implementation details

For all our experiments, we use a Faster RCNN [28] architecture with a ResNet-101 [45] FPN backbone [30] implemented using Detectron2 [46] framework. For the image augmentations, we apply a For augmentation, a cascade of random contrast, brightness, saturation, lighting and bounding box erase augmentations. The effect of this augmentation is shown in Figure 3. We train all our models with a batch size of 8 for 270000 and 18000 iterations on the COCO and PASCAL-VOC splits respectively with a learning rate of 0.01. The learning rate is decreased ten fold twice at $\{210000, 250000\}$ and $\{12000, 15000\}$ for COCO and PASCAL-VOC respectively. We adopt a warm-up strategy for 1000 and 100 iterations for the COCO and PASCAL-VOC respectively. Following existing detector implementations we set τ_{bg}, τ_{fg}, τ_{obj} and τ_{ppm} to $0.2, 0.4$, 0.5 and 0.8 respectively. We set τ_m to 0.9. During inference, we compute the backbone features (f_{o} and f_{a}) for both the original and augmented versions of the input to obtain the RoIs and only f_{o} is used for the final detections.

4.4. Baselines

We compare our method against Co-Mining [10], BRL [8], and Pseudo Label [7].

We choose these methods as they outperform previous approaches for the SAOD task. We use the public implementation of these methods and usa a ResNet 101 FPN [30] backbone for all the experiments in order to perform a fair comparison.

4.5. Comparison with state-of-the-art

In this section, we compare our method with contemporary methods. We evaluate all the models in two different setups. We name the first setup *Sparsely annotated* setup which evaluates models on SAOD. The second setup, contains labeled and unlabeled images and regions.

***Sparsely annotated* setup:** We show results of this setup in Table 1 and Table 2. Splits-2 and 3 ensure at least one annotation per image is retained. On the other hand, Split-1 doesn't ensure this and has significantly more unlabeled data than the other splits. Splits-4 and 5 have no unlabeled images at all. In both the tables, the rows named "oracle" refers to the models trained using all annotations.

From Table 1, we observe that our approach outperforms the other baselines and is closer to the oracle performance on the 30% splits. Co-mining [10], performs competitively on all the splits at 30%.

On Split-1, our method obtains a performance improvement of 1.87, 3.08 and 7.75 on 30%, 40% and 50% sparsity respectively over Co-mining [10].

On Split-2, our method obtains an improvement of 3.04, 3.9 and 10.5 at 30%, 40% and 50% sparsity respectively over the previous state of the art Co-mining. On Split-3, an again we see consistent improvement of 2.8, 4.61 and 10.52 at 30%, 40% and 50% sparsity respectively over Co-mining.

In particular, on the hardest setting (70%) in Split 1-3, we demonstrate that current SAOD methods struggle at higher sparsity in the labeled data. This is because the performance of current SAOD approaches rely on the quality

Table 4: Ablation of various components of proposed approach.

C-RPN	**PPM**	$\mathcal{L}^{\mathbf{u}}_{\mathbf{SSL}}$	$\mathcal{L}^{\mathbf{s}}_{\mathbf{det,a}}$	**AP**	Δ
✗	✗	✗	✗	41.77	–
✗	✗	✗	✓	42.67	0.90
✓	✗	✗	✓	45.30	3.53
✓	✓	✗	✓	45.44	3.67
✓	✓	✓	✗	45.51	3.74
✗	✓	✓	✓	44.86	3.09
✓	✓	✓	✓	**46.00**	4.23

of pseudo labels which degrades at higher sparsity. Our proposed PPM prevents penalizing the classifier irrespective of the quality of pseudo labels and utilizes self-supervised loss to benefit from mined regions. On average, all the methods achieve lower performance on Split-1 compared to the other splits due to the nature of its creation, *i.e.*, class aware dataset level fashion of sparsity.

From Table 2, we see a similar trend as above. All the methods perform competitively on the easier settings of both splits. However, the performance gap between the our approach and Co-mining [10] increases at higher sparsity.

On Split-4, we get an improvement of 2.56, 3.12 and 5.99 (AP_{50}) percentage points on the *easy*, *hard* and *extreme* settings respectively over Co-mining. On Split-5, an improvement of 2.42, 2.58 and 4.46 points was observed. For details on more experiments refer to supplementary material.

***Single Instance Object Detection (SIOD)* setup**: [47] proposed SIOD where only one instance per category is annotated in every image. While this setup has its benefits [47], note that it is a special case of SAOD. For this setup, we obtain an AP of 32.89 (compared to 31.9 of [47]) which is an improvement of ~1 mAP.

***SSL+SAOD* setup:** We propose a semi-supervised learning benchmark for SAOD. This benchmark entails training models on a sparsely annotated labeled and an unlabeled set. As shown in Figure 4, this setup introduces two kinds of sparsity in the data, namely, sparsity in labeled regions (left) and images (right). We believe this is a realistic setup and SAOD methods must be capable of tackling both these kinds of sparsity in the data. For this setup, we use Split-5 (Section 4.2) with increasing sparsity as the labeled set and VOC12 trainval as the unlabeled set. We use the COCO-style AP metric to report results on this setup.

In Table 3, we compare against Co-mining [10] and observe an improvement of 1.81, 1.84 and 3.55 mAP on the 30%, 40% and 50% sparsity levels respectively. We observe that the gap in performance increases with sparsity, consistent with our observation for Tables 1 and 2. This can be attributed to the inability of methods like Co-mining to handle unlabeled images and high sparsity in the labeled data. We will make this benchmark public and propose this method as a baseline. We encourage researchers to report results on this benchmark in the future.

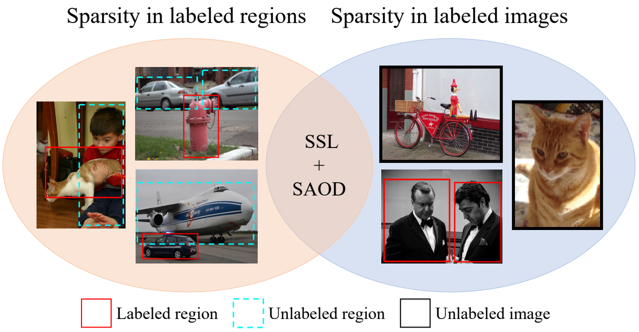

Figure 4: Types of sparsity in labeled data; sparsity in labeled regions (left) and images (right). Our proposed SSL+SAOD is a realistic setup that presents challenges from both kinds of sparsity.

4.6. Ablation Experiments

In this section we conduct ablation experiments to understand the various components. For the ablation experiments, we use a ResNet-101 as the backbone network and train on Split-5 with $p = 50\%$. We evaluate on the VOC07 test set and report the COCO style AP metric.

From Table 4, a baseline model trained on the ablation set attains 41.77 (row 1). Pseudo-labeling $\left(\mathcal{L}^{s}_{\text{det},a}\right)$ improves the performance by 0.9 (row 2). Co-mining [10] relies extensively on pseudo-labels. This results in a drop in performance at higher sparsities due to noisy pseudo-labels. Addition of the C-RPN improves our performance by 3.53 points (row 3). C-RPN reduces the overhead of computing two sets of proposals and learns a better notion of objects due to the combined processing of features from the two views.

The combination of C-RPN and PPM improves the performance by 3.67 (row 4). We do not observe major improvements with the introduction of PPM because its task is to identify and segregate the unlabeled regions from the backgrounds. After the segregation, PPM does no further processing to improve performance. The power of PPM can be observed when trained in conjunction with consistency regularization loss $\left(\mathcal{L}^{\text{u}}_{\text{SSL}}\right)$ which achieves the best performance of 46 (row 7); an improvement of 4.23 points over the baseline. We distinguish ourselves from pseudo-labeling approaches like Co-mining in one important as-

Table 5: Analysis of the threshold used for PPM.

Threshold	0.6	0.7	0.8	0.9	0.95
AP	45.96	45.96	**46.00**	45.77	45.30

Table 6: Removing test time augmentations.

Method	**Split-1**			**Split-2**			**Split-3**		
	30%	50%	70%	30%	50%	70%	30%	50%	70%
TTA	38.22	35.92	32.68	39.76	36.94	35.33	39.56	37.15	35.48
w/o TTA	38.32	35.91	32.67	39.77	36.95	35.27	39.55	37.12	35.48

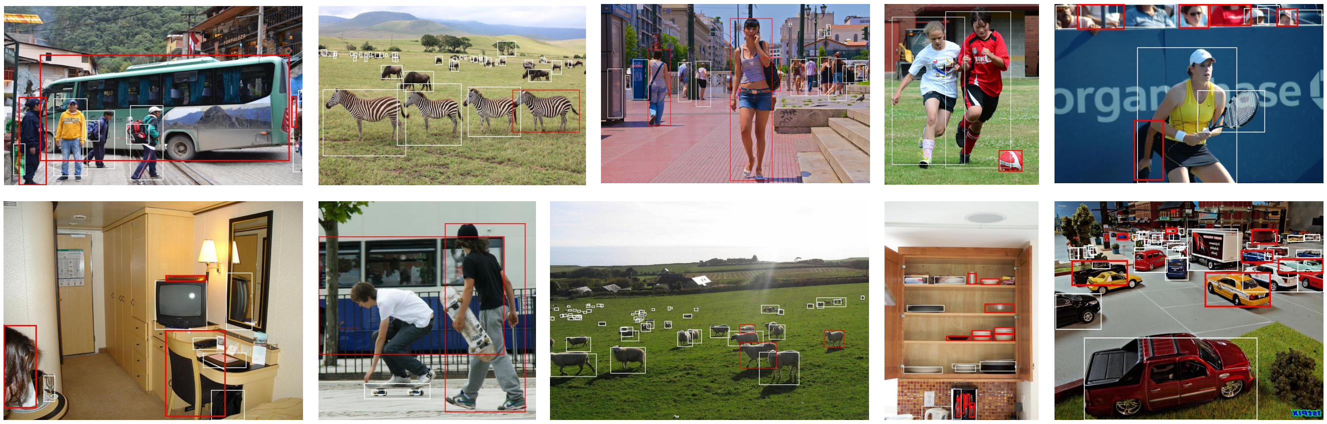

Figure 5: Qualitative results showing the unlabeled regions identified by the PPM. The red boxes correspond to the available ground truth. A class agnostic NMS was performed on the regions and the result is shown in white.

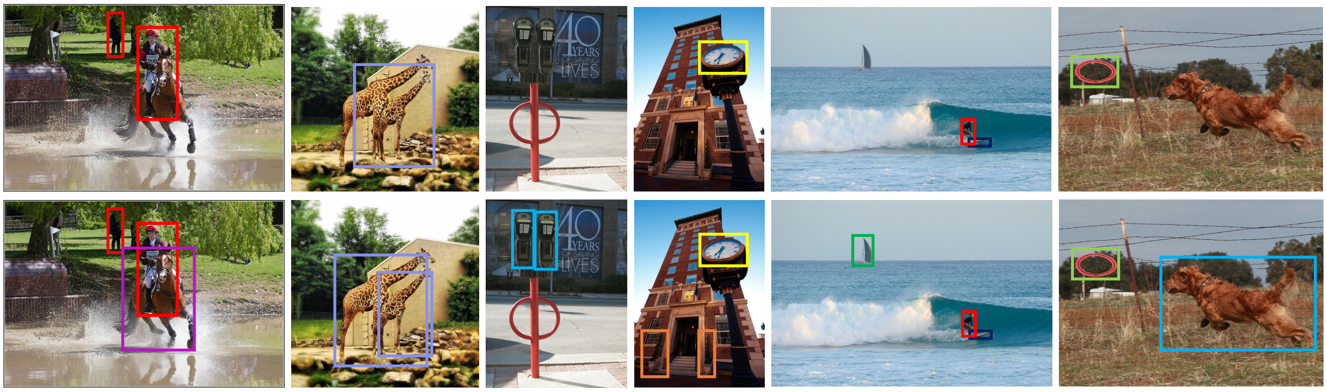

Figure 6: Qualitative results comparing the output of a model trained using available ground truths (top) to a model trained using our approach (bottom). Predictions with a class confidence score greater than 0.9 are shown. Red: Person, Cyan: Dog, Purple: Horse, Yellow: Clock, Green: Stop sign, Blue: Parking meter, Violet: Giraffe, Orange: Potted plant, Black: Surfboard, Dark green: Boat

pect. While, Co-mining [10] relies solely on pseudo-labels, we leverage additional components from self-supervised learning along with pseudo-labeling. In row 5, due to C-RPN, PPM and $\mathcal{L}^{u}_{SSL}$, we show an improvement of 3.74 over the baseline. Our proposed components are orthogonal to pseudo-labeling as using them together results in an additional improvement of $\sim$0.5 on mAP. At higher sparsity in the labeled dataset, the proposed C-RPN, PPM and $\mathcal{L}^{u}_{SSL}$ are less affected than pseudo-labels resulting in the large improvements on these splits. Finally, we show the effect of C-RPN by generating a single set of proposals from the original image and using it for both the branches. We observe a drop in performance (row 6) highlighting the effectiveness of C-RPN.

PPM mines potential positives which can be mistaken for negatives due to missing annotations. We rely on the objectness score of the RPN to identify these regions. In Table 5, we vary the threshold of PPM. For a low threshold, a few hard negatives might also be identified as pseudo-positive leading to a drop in performance. With a high threshold, a few potential positives might not be mined. We observe that a threshold of 0.8 provides a good trade-off and is there-

Table 7: Comparison with a different backbone. Results are reported on the COCO validation set on Splits 1-3 using AP and on VOC 2007 test set on Splits-4,5 using AP_{50}.

Method	**Split-1**			**Split-2**			**Split-3**			**Split-4**			**Split-5**		
	30%	50%	70%	30%	50%	70%	30%	50%	70%	Easy	Hard	Extreme	30%	40%	50%
Ours (C4)	37.67	35.95	33.16	39.22	36.81	34.98	38.84	36.76	35.26	80.56	80.43	74.45	77.38	76.13	75.32
Ours (FPN)	38.22	35.92	32.68	39.76	36.94	35.33	39.56	37.15	35.48	82.15	81.50	75.59	76.84	75.88	74.35

fore used for all experiments unless stated otherwise. In all our experiments, PPM is performed after an initial warmup of 9000 and 30000 iterations on the PASCAL-VOC and COCO datasets respectively.

4.6.1 Inference without augmentations

Our method requires passing an input with augmentations during inference as well. It should be noted that this is not a test-time augmentation (TTA), a technique that typically involves passing images at a higher resolution. We perform inference by removing the augmentation and extracting the region proposals using C-RPN. We show the results in Table 6 on the three splits of COCO. We do not observe a significant improvement in performance due to the augmentation. For the analysis on the effect of augmentations refer to supplementary material.

4.6.2 Effect of backbone

We analyze the effect of backbone on our approach and show results for the normal convolution based (C4) and FPN based (FPN) backbones in Table 7. The first row corresponds to C4 while the second row corresponds to FPN. While the gap in performance is lower in most cases, we observe a significant improvement using the FPN on the low sparsity settings of all the split.

4.7. RPN Recall experiments for object discovery

PPM identifies foreground regions mistakenly assigned as background during training to avoid penalizing the network. To study the effect of PPM on novel [48] classes, we train a network using our approach on randomly chosen 6000 images of the COCO dataset, containing annotations for only 20 classes of PASCAL-VOC. We evaluate the recall@0.5 of the RPN over the remaining 60 classes. A model trained using the standard technique on this dataset achieves a recall@0.5 of 77.46% and 29.06% on the known classes (20 categories) and unknown classes (60 categories) respectively. Our proposed approach, with PPM, achieves a recall@0.5 of 78.47% and 35.20% respectively. This ability to localize objects not seen during training can be beneficial for object discovery methods like [49–52] which use RPN proposals to learn/discover new categories.

4.8. Qualitative Results

In Figure 5, we show the pseudo positives mined by PPM. In each figure, the red boxes correspond to the ground truth annotations and the white boxes correspond to the post NMS pseudo positive boxes mined by PPM. We observe that the PPM correctly mines proposals which correspond to missing object annotations. Without PPM, these regions will be used as negatives to the classifier resulting in a reduced class confidence score leading to a drop in mAP. We show the detection results of a model trained on the 50% Split-1 of our approach in Figure 6. The images on the top corresponds to the model trained using sparse annotations and the bottom image shows the output of our approach.

5. Conclusion

We present SparseDet, a novel end-to-end system for Sparsely Annotated Object Detection (SAOD). We propose a simple yet effective technique for identifying the unlabeled regions using pseudo-positive mining and apply self-supervised loss on them. Through qualitative results we highlight the ability of PPM to mine pseudo-positives. We standardize the evaluation setup and show the effectiveness of our approach with an exhaustive set of experiments on multiple splits of SAOD. While we outperform existing state-of-the-art on all metrics and splits, we observe the gap in performance increases with sparsity demonstrating the short coming of methods that rely solely on pseudo-labeling. We propose a new benchmark, that evaluates the semi-supervised learning capabilities of SAOD approaches. We will release the data for the new benchmark along with all the SAOD splits and encourage researchers to evaluate future SAOD methods on these.

Acknowledgements. This project was partially funded by DARPA SAIL-ON (W911NF2020009), DARPA SemaFor (HR001119S0085), and IARPA SMART (2021-21040700003) programs. Rama Chellappa was supported by an ONR MURI Grant N00014-20-1-2787. We would like to thank our colleagues from the PI lab for their feedback on this work. The views and conclusions contained herein are those of the authors and do not represent views, policies, and/or endorsements of funding agencies.

References

[1] Simon Chadwick and Paul Newman. Training object detectors with noisy data. *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1319–1325, 2019. 1

[2] Zhe Wu, Navaneeth Bodla, Bharat Singh, Mahyar Najibi, Rama Chellappa, and Larry S. Davis. Soft sampling for robust object detection. In *BMVC*, 2019. 1, 2, 3, 5

[3] Junnan Li, Caiming Xiong, Richard Socher, and Steven C. H. Hoi. Towards noise-resistant object detection with noisy annotations. *ArXiv*, abs/2003.01285, 2020. 1

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

[5] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 1, 2, 5

[6] Alina Kuznetsova, Hassan Rom, Neil Gordon Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4. *International Journal of Computer Vision*, 128:1956–1981, 2020. 1

[7] Yusuke Niitani, Takuya Akiba, Tommi Kerola, Toru Ogawa, Shotaro Sano, and Shuji Suzuki. Sampling techniques for large-scale object detection from sparsely annotated objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6510–6518, 2019. 1, 2, 5, 6

[8] Han Zhang, Fangyi Chen, Zhiqiang Shen, Qiqi Hao, Chenchen Zhu, and Marios Savvides. Solving missing-annotation object detection with background recalibration loss. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1888–1892. IEEE, 2020. 3, 5, 6

[9] Yuewei Yang, Kevin J Liang, and Lawrence Carin. Object detection as a positive-unlabeled problem. In *BMVC*, 2020. 5

[10] Tiancai Wang, Tong Yang, Jiale Cao, and X. Zhang. Co-mining: Self-supervised learning for sparsely annotated object detection. In *AAAI*, 2021. 1, 2, 3, 5, 6, 7, 8

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html, . 2, 5

[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, . 2, 5, 6

[13] Ke Yan, Jinzheng Cai, Adam P. Harrison, Dakai Jin, Jing Xiao, and Le Lu. Universal lesion detection by learning from multiple heterogeneously labeled datasets. *ArXiv*, abs/2005.13753, 2020. 2

[14] Hansheng Li, Xin Han, Yuxin Kang, Xiaoshuang Shi, Mengdi Yan, Zixu Tong, Qirong Bu, Lei Cui, Jun Feng, and Lin Yang. A novel loss calibration strategy for object detection networks training on sparsely annotated pathological datasets. In *MICCAI*, 2020.

[15] Yongqiang Zhang, Mingli Ding, Yancheng Bai, Mengmeng Xu, and Bernard Ghanem. Beyond weakly supervised: Pseudo ground truths mining for missing bounding-boxes object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 30:983–997, 2020.

[16] Ke Yan, Jinzheng Cai, Youjing Zheng, Adam P. Harrison, Dakai Jin, You-Bao Tang, Yuxing Tang, Lingyun Huang, Jing Xiao, and Le Lu. Learning from multiple datasets with heterogeneous and partial labels for universal lesion detection in ct. *IEEE Transactions on Medical Imaging*, 40:2759–2770, 2021. 2

[17] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. *Advances in neural information processing systems*, 32:10759–10768, 2019. 2

[18] 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. 2

[19] 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. 2

[20] 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.

[21] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. *arXiv preprint arXiv:2106.09018*, 2021.

[22] 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*, pages 5941–5950, 2021. 2

[23] Jihun Yoon, Seungbum Hong, and Min-Kook Choi. Semi-supervised object detection with sparsely annotated dataset. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 719–723. IEEE, 2021. 3

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. 3

[25] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115 (3):211–252, 2015. doi: 10.1007/s11263-015-0816-y. 3

[26] Ross Girshick. Fast r-cnn. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pages 1440–1448, Washington, DC, USA, 2015. IEEE Computer Society. ISBN 978-1-4673-8391-2. doi: 10.1109/ICCV.2015.169. URL http://dx.doi.org/10.1109/ICCV.2015.169. 3

[27] 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. 3

[28] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. 3, 4, 6

[29] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *NIPS*, pages 379–387, 2016. URL http://dblp.uni-trier.de/db/conf/nips/nips2016.html#DaiLHS16. 3

[30] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017. 3, 6

[31] Bharat Singh and Larry S Davis. An analysis of scale invariance in object detection-snip. *CVPR*, 2018.

[32] Bharat Singh, Mahyar Najibi, and Larry S Davis. SNIPER: Efficient multi-scale training. *NeurIPS*, 2018.

[33] Mahyar Najibi, Bharat Singh, and Larry S Davis. AutoFocus: Efficient multi-scale inference. *ICCV*, 2019. 3

[34] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. doi: 10.1109/ICCV.2017.322. 3

[35] Alexander Kirillov, Ross B. Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, pages 6399–6408. Computer Vision Foundation / IEEE, 2019. 3

[36] Alexander Kirillov, Kaiming He, Ross B. Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, pages 9404–9413. Computer Vision Foundation / IEEE, 2019. 3

[37] Justin Johnson Georgia Gkioxari, Jitendra Malik. Mesh r-cnn. In *ICCV*, 2019. 3

[38] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016. 3

[39] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, 2017.

[40] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection, 2015. URL http://arxiv.org/abs/1506.02640. cite arxiv:1506.02640. 3

[41] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019. 3

[42] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 3

[43] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:386–397, 2020. 3

[44] Ross B. Girshick. Fast R-CNN. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015. 4

[45] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90. 6

[46] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 6

[47] Hanjun Li, Xingjia Pan, Ke Yan, Fan Tang, and Wei-Shi Zheng. Siod: Single instance annotated per category per image for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14197–14206, 2022. 7

[48] Akshay Raj Dhamija, Manuel Günther, Jonathan Ventura, and Terrance E. Boult. The overlooked elephant of object detection: Open set. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1010–1019, 2020. doi: 10.1109/WACV45572.2020.9093355. 9

[49] Huy V. Vo, Patrick Pérez, and Jean Ponce. Toward unsupervised, multi-object discovery in large-scale image collections. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 9

[50] Huy V. Vo, Francis Bach, Minsu Cho, Kai Han, Yann LeCun, Patrick Pérez, and Jean Ponce. Unsupervised image matching and object discovery as optimization. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 8287–8296, 2019.

[51] Sai Saketh Rambhatla, Rama Chellappa, and Abhinav Shrivastava. The pursuit of knowledge: Discovering and localizing novel categories using dual memory. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[52] Sai Saketh Rambhatla, Ishan Misra, Rama Chellappa, and Abhinav Shrivastava. The pursuit of knowledge: Discovering and localizing novel categories using dual memory. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023. 9