Open-Set Semi-Supervised Object Detection Supplementary Materials

1 COCO-additional.

To examine OOD filtering in a large-scale scenario, we consider COCO-additional which aims to improve the fully-supervised object detector with the additional large-scale dataset (*e.g.,* `COCO2017-unlabeled`). As presented in Table 3, UT (with data augmentation from SoftTeacher [20]) can achieve 44.06 mAP, and using the proposed OOD filtering can further improve UT and achieve 45.14 mAP and shows state-of-the-art result against the existing SS-OD works [18,22,9,17,20] on COCO-additional. This demonstrates that removing OOD objects from the large-scale unlabeled data can still improve the existing SSOD framework. However, it is worth noting that our proposed OOD filtering mechanism is not restricted to UT, and we believe that is also complementary to other SSOD methods [15,18,22,9,17,20].

2 Qualitative Results with the OOD filtering

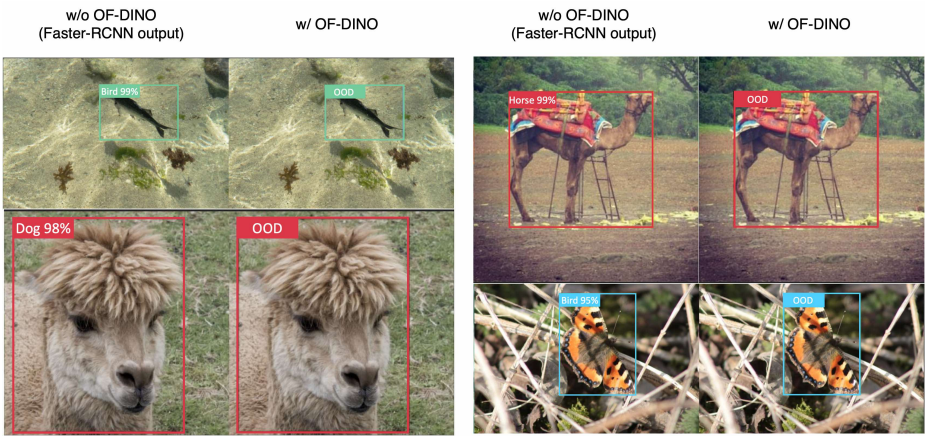

Fig. 1. Comparison between the pseudo-labels with and without DINO-based OOD filtering (OF-DINO).

To show the effectiveness of the OOD filtering, we show the pseudo-labels generated with and without OOD filtering in Figure 1. Without the OOD filtering, some OOD objects (*e.g.,* fish) are predicted as inlier (*e.g.,* bird) with

high confidence. Such an issue is alleviated by the OOD filtering, which effectively suppresses the OOD objects in pseudo-labels and thus improves the accuracy of the object detector.

3 Experiment results when using ViT-B as OOD detector

To understand how the ViT-B pretrained on ImageNet21k performs on OSSOD tasks, we examine the ViT-B model on the experiment setups presented in the main paper. Specifically, we consider different degrees of supervision in Table 1, different numbers of ID objects in Table 2, large-scale SSOD setting (*e.g.,* COCO-additional) in Table 3, large-scale OSSOD setting (*e.g.,* COCO-OpenImage) in Table 4. We observe using the ViT-B model can consistently lead to better results in all experiment scenarios, while, as we mentioned in the main paper, ViT-B requires large-scale supervised pre-training dataset (ImageNet21k), which is not suitable for our label-efficient settings. However, it does demonstrate that the method can scale with larger-scale pre-trained models, including when using in-the-wild unlabeled data (e.g. OpenImages).

Table 1. Mean average precision of COCO-open under **different degrees of supervision**. We first pre-select 20 COCO classes as ID classes and 60 COCO classes as OOD classes, and $1/2/4k$ images are *randomly* selected from the purely-ID set as the labeled set. The remaining images from pure-ID, pure-OOD, and mixed sets are used as the unlabeled set. We run each method 3 times and report the standard deviation.

Num. of Labeled Images	1,000	2,000	4,000
Label-only	10.20± 0.34	11.84± 0.33	16.35± 0.28
UT	11.77±0.38 (+1.57)	13.87±0.68 (+2.03)	18.23±0.47 (+1.88)
UT + OF-DINO	16.80±0.53 (+6.60)	18.10±0.71 (+6.26)	22.56±0.51 (+6.21)
UT + OF-ViT	**17.10±0.46 (+6.90)**	**19.32±0.53 (+7.48)**	**23.01±0.67 (+6.66)**

Table 2. Mean average precision of COCO-Open when **varying the number of ID objects**. We first randomly sample 20/40/60 COCO classes as ID classes and remaining COCO classes as OOD classes, and $4k$ images are randomly selected from the purely-ID set as the labeled set. The remaining images from pure-ID, pure-OOD, and mixed sets are used as the unlabeled set. We run each method 3 times and report the standard deviation.

Num. of ID/OOD objects	20/60	40/40	60/20
Label-only	16.89±2.6	15.98±0.49	16.64±0.59
UT	18.37±1.67 (+1.48)	20.28±0.85 (+4.29)	23.09±0.25 (+6.45)
UT + OF-DINO	23.43±2.19 (+6.54)	22.91±0.28 (+6.93)	24.89±0.34 (+8.25)
UT + OF-ViT	**25.20±2.00 (+8.31)**	**25.10±1.01 (+9.12)**	**26.11±0.40 (+9.47)**

Table 3. Comparison to other SSOD methods in COCO-additional.

	mAP
Supervised	40.90
Proposal Learning [17]	38.40
CSD [9]	38.82
STAC [15]	39.21
Instant-Teaching [22]	40.20
MOCOv2 + Instagram-1B [18]	41.10
Humble Teacher [18]	42.37
SoftTeacher [20]	44.05
Unbiased Teacher* [13]	44.06
Unbiased Teacher* + OF-DINO	45.14
Unbiased Teacher* + OF-ViT	**45.16**

Table 4. Experimental results of COCO-OpenImage.

	OpenImage GT labels	mAP
COCO		40.90
COCO + OpenImage	✓	42.91
Unbiased Teacher [13]		41.81
Unbiased Teacher + OF-DINO		43.14
Unbiased Teacher + OF-ViT		**43.48**

4 Experiments on STAC

Table 5. Generalization of our findings to other methods, namely STAC [15] performance comparison between closed-set SSOD and open-set SSOD. For closed-set SSOD, we randomly select 1%/2% from the training set (*i.e.,* 1172/2234 labeled images). For the open-set SSOD, we randomly samples 20 classes as ID classes and the remaining classes as OOD classes; hence the differences in performance of "Labeled only". We then sample the same amount of labeled images in both cases for a fair comparison.

	closed-set SSOD		open-set SSOD	
Percentage of labeled images	1%	2%	1%	2%
Num. of labeled images	1,172	2,344	1,172	2,344
Labeled only	9.05	12.70	11.20	12.18
STAC [15]	13.97	18.25	13.22	15.34
Δ	+4.92	+5.55	+2.02	+3.16

In addition to Unbiased Teacher [13] experimented with in the main paper, we also consider another SSOD method, STAC [15], to show that our findings are general. As shown in Table 5, we observe STAC also suffers from open-set issues when experimenting on OSSOD tasks. Compared with the traditional closed-set SSOD task, the performance gain of STAC is smaller in open-set conditions.

To address the open-set issues, we also apply our proposed OOD detection method on STAC. As presented in Table 6, when the OOD detector (DINO) is applied, we can improve STAC from 18.60 mAP to 19.80 mAP. This shows that our proposed OOD filtering method is not restricted to any particular SSOD method and can potentially improve other SSOD methods.

Furthermore, similar to Unbiased Teacher [13], other existing SSOD methods [18,22,21,20] also applied confidence thresholding to select pseudo-labels, so they are also prone to suffer from the semantic expansion issue as we described in the main paper.

Table 6. OOD filtering improves STAC on COCO-Open. We randomly sample 40 COCO classes as ID classes and remaining COCO classes as OOD classes, and $4k$ images are randomly selected from the purely-ID set as the labeled set. The remaining images from pure-ID, pure-OOD, and mixed sets are used as the unlabeled set.

	Label-only	STAC	STAC + OF-DINO
mAP	16.54	18.60 (+2.06)	19.80 (+3.26)

5 Complete Comparison of OOD Detection

Table 7. Evaluation of OOD detection for object detection tasks. We sample 20 classes from COCO as in-distribution (ID) objects, and 4000 pure-ID images are selected as labeled images. All methods are evaluated on `COCO2017-val`. Value before the slashes indicates ignoring background patches when computing AUROC and FPR, and value after the slashes regradining background patches as OOD objects when computing AUROC and FPR.

Model	Methods	OoD Scores γ_{ood}	AUROC ↑	FPR50↓	FPR75↓	FPR95↓
Online OOD Detector (Faster-RCNN branch)	Vanilla	MSP [5]	67.0 / 71.0	22.5 / 15.5	58.4 / 52.4	92.3 / 91.1
		Energy [12]	75.5 / 68.2	13.2 / 22.8	36.8 / 49.0	83.6 / 87.8
		Entropy	75.9 / 68.4	12.2 / 22.3	38.5 / 51.1	83.1 / 87.7
		Mahalanobis [10]	50.2 / 61.6	51.5 / 32.8	83.0 / 65.7	98.1 / 93.7
		Euclidean	56.3 / 61.5	40.2 / 31.8	74.3 / 66.9	96.1 / 94.1
	OE [6]	MSP	67.0 / 73.3	25.3 / 15.2	55.0 / 45.9	89.1 / 85.6
	One-vs-all [14]	MSP	73.0 / 76.0	13.4 / 10.7	45.7 / 40.2	90.0 / 84.8
	GODIN [7]	Cosine $h(x)$	77.8 / 73.5	12.5 / 14.6	33.8 / 45.0	77.4 / 84.5
	GSD [19]	Feat. angle	78.7 / 71.3	11.8 / 19.3	32.1 / 48.8	73.9 / 83.4
Offline OOD Detector (DINO)	Ours	Inv. abstaining conf.	83.6 / 86.0	8.5 / **5.9**	22.4 / 18.7	61.7 / 57.9
		Energy	89.6 / 85.9	4.0 / 7.0	12.2 / 18.8	47.5 / 56.8
		Entropy	88.9 / 84.7	**3.5** / 7.3	12.6 / 20.3	51.1 / 59.9
		Mahalanobis [10]	81.8 / 75.7	11.7 / 17.6	25.6 / 35.9	57.6 / 68.9
		Euclidean	**90.8** / **86.1**	3.6 / 7.3	**10.7** / **18.5**	**38.6** / **51.6**
Offline OOD Detector (ViT)	Ours	Inv. abstaining conf.	87.5 / 88.2	4.5 / 4.0	15.3 / 14.7	54.7 / 51.5
		Energy	93.3 / 88.5	2.1 / 5.9	6.0 / 14.5	32.4 / 45.3
		Entropy	93.2 / 88.1	1.5 / 5.8	5.9 / 15.1	33.9 / 46.3
		Feat. angle	93.2 / 87.6	2.1 / 7.2	6.1 / 15.5	33.4 / 46.0.

To compare OOD detection methods, we list a more complete comparison as presented in Table 7. We list other observations as follows:

Mahalanobis Distance under limited amount of data setting With limited amount of training data (*i.e.,* 4k ID images), Mahalanobis distance becomes unreliable compared with other OOD metrics, and this is contrary to the prior works on OOD detection [4,10], where a large amount of data is used for training

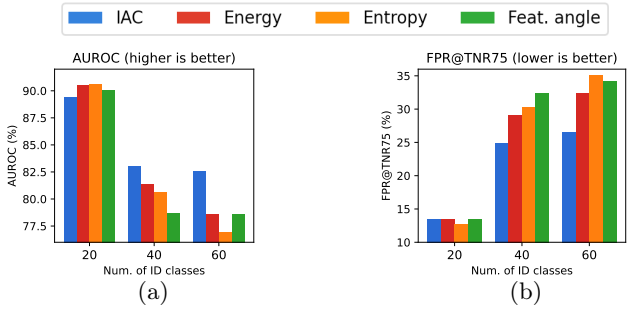

Fig. 2. Comparison of OOD metrics of OF-DINO under different number of ID classes, and we present (a) AUROC and (b) FPR@TNR75 to evaluate the performance of the OOD detection. Among different OOD metrics, inverse abstaining confidence (IAC) suffers less when the number of ID classes increases. Note that 20/40/60 classes from the `COCO2017-train` are selected as ID classes (and the remaining classes as OOD classes), and $4k$ images of pure-ID set are selected as the labeled set.

the OOD detectors. As computing Mahalanobis distance requires the covariance matrix and the class-wise mean vectors, and estimating the covariance matrix for high-dimension features is difficult and inaccurate, this is even more difficult when the data is scarce (especially when the number of instances is closer to or even less than the number of feature dimensions). This is also why we observe that Euclidean distance, which is equivalent to the Mahalanbois distance with the identity convariance matrix, leads to even better results than the Mahalanobis distance.

Another weakness of Mahalanobis distance is that it requires one to obtain class-wise mean and co-variance features by deriving the features for all ID images in the labeled set, so it is computationally slow (also pointed out in MOS [8]) and thus less preferable for large-scale open-set semi-supervised learning, which requires detecting OOD samples in each training iteration.

Inverse abstaining confidence under different number of ID/OOD classes. Compared with other offline OOD detection metrics, using inverse abstaining confidence is more robust when varying the number of ID classes. To be more specific, as shown in Table 7, the Energy score and Shannon entropy perform on par or even better than the inverse abstaining confidence in the case of using 20 ID classes. However, as shown in Figure 2, when we increase the number of ID classes, the inverse abstaining confidence degrades much less than the Energy score and Shannon entropy. Such a property makes the inverse abstaining confidence more suitable for different OSSOD scenarios.

6 Comparison between COCO-OpenImage and COCO-additional

In the main paper, we considered two large-scale unlabeled sets, `OpenImagev5` and `COCO2017-unlabeled`, to improve the object detection trained on the labeled

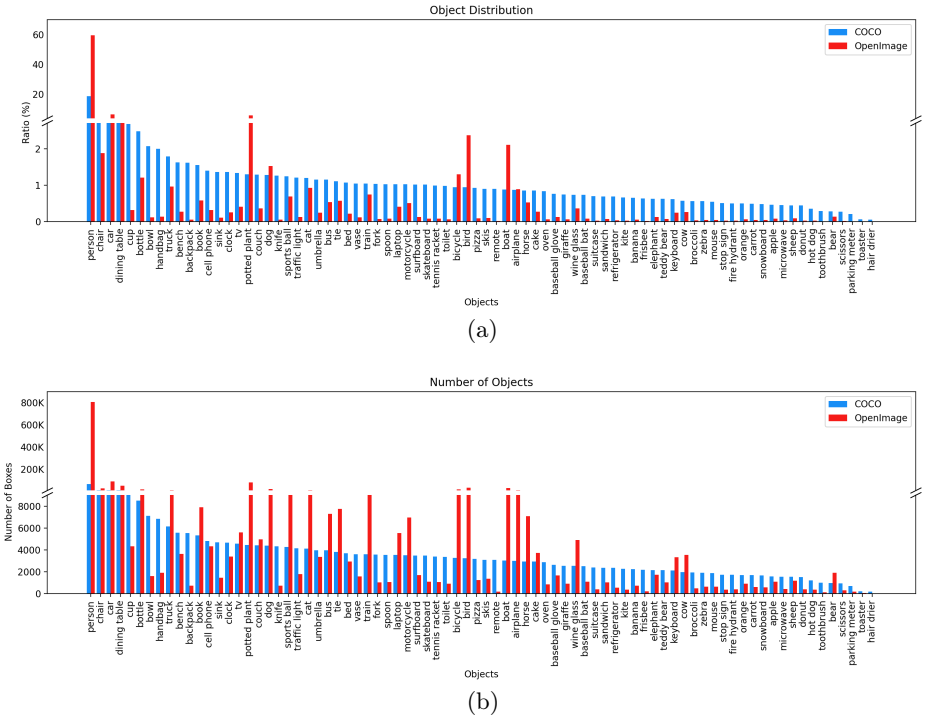

Fig. 3. (a) Distribution of object categories and (b) number of objects in `COCO2017-train` and `OpenImagev5`.

`COCO2017-labeled`. Our OOD filtering framework improves the supervised object detector from 40.90 mAP to 43.14 mAP by using `OpenImagev5` as an unlabeled set and achieves 45.14 mAP when the `COCO2017-unlabeled` is used as an unlabeled set.

As `OpenImagev5` has more unlabeled images than `COCO2017-unlabeled` (1.7M vs. 120k), we are curious *why the model using the* `OpenImagev5` *as an unlabeled set cannot outperform the model using the* `COCO-unlabeled` *as an unlabeled set.*

We attribute this trend to the following factors:

(i) **Mismatch in class distribution.** We first compare the class distribution of both datasets, and we find these two datasets have very different object distributions as shown in Figure 3a. The mismatch in the class distribution in the unlabeled set is prone to affect the frequency or confidence of objects predicted for the evaluation set, and this potentially leads to performance degradation in the evaluation set.

(ii) **Some COCO objects are rare in OpenImage.** When `OpenImagev5` is used as an unlabeled set to train the object detector in a semi-supervised manner, we observe, as shown in Table 8, the performance on *some* objects

Table 8. Performance degradation of minor objects in semi-supervised learning. We apply Unbiased Teacher with the proposed OOD filtering (DINO) and use the `OpenImagev5` as an unlabeled set, and detection performance of some rare objects are degraded due to the scarcity of these objects in `OpenImagev5`. Note that `OpenImagev5` contains 1.7M images, and `COCO2017-train` has 117k images.

Objects	**Supervised-only** Labeled: `COCO2017-train` -	**UT+OF-DINO** Labeled: `COCO2017-train` Unlabeled: `OpenImagev5`	mAP difference	Number of boxes in `COCO2017-train`	Number of boxes in `OpenImagev5`
sheep	51.81	51.49	-0.33	1529	1188
carrot	22.52	22.34	-0.18	1683	594
hair drier	2.59	1.53	-1.06	189	27
zebra	66.63	65.99	-0.63	1916	621
snowboard	35.48	33.90	-1.57	1654	574
knife	19.60	19.45	-0.15	4326	726
banana	23.56	23.08	-0.48	2243	723
orange	32.29	31.38	-0.91	1699	900
hot dog	32.23	31.20	-1.03	1222	362
toaster	40.40	34.28	-6.13	217	60
giraffe	68.41	68.04	-0.37	2546	920
tennis racket	49.31	49.10	-0.22	3394	1047
microwave	54.14	53.75	-0.40	1547	432

are even lower than the supervised model due to the scarcity of these objects. Specifically, even though `OpenImagev5` has more images than `COCO2017-train`, the number of some COCO objects in entire `OpenImage` are even fewer than the objects in `COCO2017-train`, as shown in Figure 3b. This suggests the objects are very rare and infrequently appear, and such a property potentially limits the further improvement by using `OpenImagev5`. Note that `COCO2017-unlabeled` follows the same class distribution as `COCO2017-labeled` (described in COCO official page), and both datasets have similar amount of images (120k vs. 117k).

7 Label Correspondence between COCO and OpenImage

To construct the baseline trained with ground-truth labels from `OpenImage`, we manually label the correspondence between 80 classes in `MS-COCO` and 601 classes in `OpenImage`. We provide the object correspondence in Table 9. Among 601 classes in `OpenImage`, 139 GOI classes have matching COCO classes, and the remaining 462 classes do not correspond to any COCO classes. We thus remove the labels of these classes in the training of the supervised baseline.

8 Implementation Details

Our implementation is based on the Detectron2 framework. As our framework is built on the Unbiased Teacher [13], we follow its implementation details, including training iterations, threshold, unsupervised loss weight, and other hyper-parameters for a fair comparison.

Table 9. Object classes correspondence between `MS-COCO` and `OpenImages`. 139 OpenImages objects have the matching COCO objects, while the remaining 462 OpenImage objects do not correspond to any COCO object.

COCO-objects	OpenImage-objects	COCO-objects	OpenImage-objects
person	Person, Boy, Woman, Man, Girl	wine glass	Wine glass
bicycle	Bicycle	cup	Coffee cup, Measuring cup, Mug
car	Car, Ambulance, Limousine, Taxi	fork	Fork
motorcycle	Motorcycle	knife	Knife, Kitchen knife
airplane	Airplane	spoon	Spoon, Ladle
bus	Bus	bowl	Mixing bowl, Bowl
train	Train	banana	Banana
truck	Truck, Van	apple	Apple
boat	Boat, Barge, Gondola, Canoe	sandwich	Submarine sandwich, Sandwich
traffic light	Traffic light	orange	Orange
fire hydrant	Fire hydrant	broccoli	Broccoli
stop sign	Stop sign	carrot	Carrot
parking meter	Parking meter	hot dog	Hot dog
bench	Bench	pizza	Pizza
cat	Cat	cake	Cake
dog	Dog	chair	Chair
horse	Horse	couch	Studio couch, Couch, Sofa bed, Loveseat
sheep	Sheep	potted plant	Lavender (Plant), Plant, Houseplant, Flowerpot
cow	Cattle, Bull	bed	Bed
elephant	Elephant	dining table	Kitchen & dining room table, Table, Coffee table
bear	Bear, Brown bear, Panda, Polar bear	toilet	Toilet, Bidet
zebra	Zebra	tv	Computer monitor, Television
giraffe	Giraffe	laptop	Laptop
backpack	Backpack	mouse	Computer mouse
umbrella	Umbrella	remote	Remote control
handbag	Handbag	keyboard	Computer keyboard
tie	Tie	cell phone	Mobile phone
suitcase	Suitcase	microwave	Microwave oven
frisbee	Flying disc	oven	Oven, Gas stove
skis	Ski	toaster	Toaster
snowboard	Snowboard	sink	Sink
donut	Doughnut	refrigerator	Refrigerator
kite	Kite	book	Book
baseball bat	Baseball bat	clock	Wall clock, Clock, Alarm clock, Digital clock, Watch
baseball glove	Baseball glove	vase	Vase
skateboard	Skateboard	scissors	Scissors
surfboard	Surfboard	teddy bear	Teddy bear
tennis racket	Tennis racket, Racket	hair drier	Hair dryer
bottle	Beer, Bottle, Wine	toothbrush	Toothbrush
bird	Bird, Magpie, Woodpecker, Blue jay, Raven, Eagle, Falcon, Owl, Duck, Canary, Goose, Swan, Parrot, Sparrow	sports ball	Rugby ball, Football, Ball, Cricket ball, Volleyball (Ball), Golf ball, Tennis ball

Model Architecture. We experiment on the Faster-RCNN with FPN [11], and ResNet-50 pretrained on ImageNet-1K is used as the feature backbone. For the offline OOD detectors, we consider DINO [1] and VITB [3] as base models.

Training. For the object detectors, we use the SGD optimizer with a momentum rate 0.9 and a learning rate 0.01, and we use a constant learning rate scheduler for COCO-Open and learning rate decay for COCO-additional and COCO-OpenImage. Each batch contains 8 labeled images and 8 unlabeled images for COCO-Open, and 32 labeled images and 32 unlabeled images for COCO-OpenImage and COCO-additional. To fine-tune DINO/VITB models, we randomly select 64 patches from each image and train $10k/20k/40k$ iterations for $1k/2k/4k$ labeled images setups of COCO-Open. For COCO-additional and COCO-OpenImage, we also randomly select 64 patches from each image and train for $160k$ iterations. We follow the prior work [16] to use SGD optimizer with a learning rate $1e-3$ and $5e-3$ for DINO and VIT models. We apply the

inverse abstaining confidence as the OOD score due to its robustness to different number of object categories. As for thresholds for confidence thresholding and OOD filtering, we use $\delta = 0.5$ for the confidence thresholding and $\delta_{ood} = 0.5$ for the OOD filtering.

Data augmentation. For COCO-Open, We follow the data augmentation used in Unbiased Teacher [13], which applies a random horizontal flip for weak augmentation and randomly adds color jittering, grayscale, Gaussian blur, and cutout patches [2] for the strong augmentation. For COCO-additional and COCO-OpenImage, we additionally consider scale jitter used in SoftTeacher [20] to further improve the performance. Image-level or box-level geometric augmentations, such as rotation, translation, and Mosaic [22], are not used in our method.

9 Training of our online and offline frameworks

In the main paper, we present our proposed OSSOD framework integrated with online and offline OOD detectors. We thus present the training details of offline OOD detectors in Alg. 1 and online OOD detectors in Alg. 2.

Algorithm 1: Learning of an **offline OOD Detector** and UT [13]

Data: Labeled set: $D_s = \{x_s, y_s\}$; Unlabeled set: $D_u = \{x^u\}$
for *Iters. of supervised training an object detector* **do**
  Compute supervised object detector loss $\mathcal{L}_{sup}$ with D_s
  $\theta^s_{obj} \leftarrow \theta^s_{obj} - \nabla_{\theta^s_{obj}} \mathcal{L}_{sup}$
$\theta^t_{obj} \leftarrow \theta^s_{obj}$
for *Iters. of training an offline OOD detector* **do**
  Get background proposal boxes $\tilde{y}_t$ from Teacher object detector
  Select foreground GT boxes from y^s and crop I_{fg} from x_s
  Select background proposal boxes from $\tilde{y}_t$ and crop I_{bg} from x_s
  Compute multi-class cross-entropy loss $\mathcal{L}_{ood}$ based on $\{I_{bg}, I_{fg}\}$
  $\theta_{ood} \leftarrow \theta_{ood} - \nabla_{\theta_{ood}} \mathcal{L}_{ood}$
for *Iters. of semi-supervised training of an object detector* **do**
  Predict $\tilde{y}_u = f(x_u; \theta_t)$
  Apply confidence thresholding $\hat{y}_u \leftarrow h(\tilde{y}_u; \delta)$
  Apply OOD filtering $\bar{y}_u \leftarrow h(\hat{y}_u; \delta_{ood})$
  Compute unsupervised object detector loss $\mathcal{L}_{unsup}$ with $\{D_u, \bar{y}_u\}$
  Compute supervised object detector loss $\mathcal{L}_{sup}$ with D_s
  $\mathcal{L}_{ssod} = \mathcal{L}_{sup} + \lambda \mathcal{L}_{unsup}$
  $\theta^s_{obj} \leftarrow \theta^s_{obj} - \nabla_{\theta^s_{obj}} \mathcal{L}_{ssod}$
  $\theta^t_{obj} \leftarrow \alpha \theta^t_{obj} + (1 - \alpha) \theta^s_{obj}$
Result: Learned weights $\theta^t_{obj}/\theta^s_{obj}$ of Teacher/Student object detector

Algorithm 2: Learning of an **online OOD Detector** and UT [13]

Data: Labeled set: $D_s = \{x_s, y_s\}$; Unlabeled set: $D_u = \{x^u\}$

Add an OOD detection head on both Teacher and Student object detectors

for *Iters. of semi-supervised training of an object detector* **do**

- Predict $\tilde{y}_u = f(x_u; \theta_t)$
- Apply confidence thresholding $\hat{y}_u \leftarrow h(\tilde{y}_u; \delta)$
- Apply OOD filtering $\bar{y}_u \leftarrow h(\hat{y}_u; \delta_{ood})$
- Compute unsupervised object detector loss $\mathcal{L}_{unsup}$ with $\{D_u, \bar{y}_u\}$
- Compute supervised object detector loss $\mathcal{L}_{sup}$ with D_s
- Compute OOD loss $\mathcal{L}_{ood}$ (Refer to definition in original papers)
- $\mathcal{L} = \mathcal{L}_{sup} + \lambda \mathcal{L}_{unsup} + \lambda_{ood} \mathcal{L}_{ood}$
- $\theta^s_{obj} \leftarrow \theta^s_{obj} - \nabla_{\theta^s_{obj}} \mathcal{L}$
- $\theta^t_{obj} \leftarrow \alpha \theta^t_{obj} + (1 - \alpha) \theta^s_{obj}$

Result: Learned weights $\theta^t_{obj}/\theta^s_{obj}$ of Teacher/Student object detector

Limitations and future works. While addressing OSSOD, we do not address other issues such as covariate shift and mismatch in object category distributions between datasets. The offline OOD detector is an individual module from the object detector, so it requires more computational resources in the training stage. However, this concern does not exist as we remove the offline OOD detector and only use the object detector in the inference stage. Our key message is that combining an offline OOD detection module and an SSOD method is a simple yet effective solution to address OSSOD tasks. Based on this integrated framework, there will be more advanced techniques for both SSOD and OOD detection methods, which can potentially improve the performance on OSSOD tasks.

References

1. Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021. 8
2. Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 9
3. Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 8
4. Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 4
5. Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. 4
6. Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019. 4

7. Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 4
8. Rui Huang and Yixuan Li. Mos: Towards scaling out-of-distribution detection for large semantic space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5
9. Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 1, 3
10. Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 4
11. Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 8
12. Weitang Liu, Xiaoyun Wang, John D Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 4
13. 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. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 3, 7, 9, 10
14. Kuniaki Saito, Donghyun Kim, and Kate Saenko. Openmatch: Open-set consistency regularization for semi-supervised learning with outliers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 4
15. Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020. 1, 3
16. Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021. 8
17. Peng Tang, Chetan Ramaiah, Ran Xu, and Caiming Xiong. Proposal learning for semi-supervised object detection. *arXiv preprint arXiv:2001.05086*, 2020. 1, 3
18. 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*, pages 3132–3141, 2021. 1, 3
19. Junjiao Tian, Dylan Yung, Yen-Chang Hsu, and Zsolt Kira. A geometric perspective towards neural calibration via sensitivity decomposition. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 4
20. 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. 1, 3, 9
21. 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3
22. 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 Conference on Computer Vision and Pattern Recognition (CVPR), 2021. 1, 3, 9