Sparse Semi-DETR: Sparse Learnable Queries for Semi-Supervised Object Detection (Supplementary Material)

Tahira Shehzadi[1,2,3], Khurram Azeem Hashmi[1,2,3], Didier Stricker[1,2,3], Muhammad Zeshan Afzal[1,2,3]
[1]DFKI, [2]RPTU Kaiserslautern-Landau, [3]MindGarage-RPTU

The supplementary document offers an extensive overview of our approach, detailed insights into our implementation details, and a comprehensive analysis of results.

A1.1 Additional Details of Sparse Semi-DETR

The Sparse Semi-DETR framework is an extension of Semi-DETR (the first semi-supervised DETR-based framework). Labeled data is used for student network training, employing a supervised loss. The Sparse Semi-DETR framework processes unlabeled data through two distinct pathways: the teacher network, which receives weakly augmented data, and the student network, which is fed with strongly augmented data. The teacher network utilizes the unlabeled data to produce pseudo-labels. Meanwhile, the student model undergoes parameter refinement via back-propagation. In contrast, the teacher model's parameters are updated, following the exponential moving average (EMA) of the student model.

Additional Details of Semi-DETR. Semi-DETR is a DETR-based semi-supervised framework that introduces cross-view query consistency and stage-wise hybrid matching strategies. (1) In CNN-based semi-supervised object detection (SSOD) frameworks [1, 3–9, 11], consistency regularization is easily implemented by minimizing differences between teacher and student model outputs, given the same input but with different augmentations. However, this approach is not directly applicable in DETR-based SSOD frameworks due to the lack of clear correspondence between input object queries and output predictions. To address this, a novel cross-view query consistency module is proposed. It processes RoI features through MLPs, and generates cross-view query embeddings. These embeddings are combined with original object queries and fed into a decoder. (2) Semi-DETR initially uses a one-to-many assignment in early training, allowing multiple predictions per pseudo-label. It speeds up convergence and improves label quality but can cause redundant predictions. It then switches to one-to-one assignment, reducing redundancy and aiming for an NMS-free final model. However, its effectiveness on small objects is limited. Our Sparse semi-DETR refines object queries, enhancing small object detection and accuracy.

A1.2 Additional Details of Implementation.

The implementation of the Sparse Semi-DETR approach is based on MMdetection framework [2]. We integrate data pre-processing methodologies from Soft-Teacher [8]. We train the network on 8 GPUs (RTXA6000), which takes roughly two training days to complete 120k training iterations. Elaborating on training hyperparameters for different benchmarks: (1) COCO-Partial Setup: We train the network using 8 GPUs for 120k iterations, with each GPU handling five images. It employs one-to-many assignment strategy for first 60k iterations and then one-to-one assignment strategy for 60k-120k iterations. (2) COCO-Full Setup: For this benchmark, we train for 240k iterations, employing one-to-many assignment strategy for first 180k iterations and then one-to-one assignment strategy for 180k-240k iterations. We use 8 GPUs with eight images per GPU. (3) Pascal VOC Setup: Here, first 40k iterations adopt a one-to-many assignment strategy and then one-to-one assignment strategy for 40k-60k iterations. Across all our experimental setups, we've kept the confidence threshold constant at 0.4. We use the Adam optimizer and set the learning rate to 0.001. We avoid using learning rate decay for a fair comparison with Semi-DETR [10]. Complete implementation details are provided in Table 1.

Data Augmentation. We adopt the same data augmentation scheme as in Semi-DETR, detailed in Table 2. We employ weak augmentation on unlabeled data for generating pseudo labels, while strong augmentation is utilized for both labeled and unlabeled data during the model's training.

A1.3 Additional Details of Results.

Additional Details of Query Refinement Module. We perform additional experiments to assess the efficacy of our query refinement approach as follows:

1. Is the attention module crucial in query refinement? Could we apply attention to just low or high-resolution

training setting	COCO-Partial	COCO-Full	VOC	Ablation
batch size	5*8	8*8	5*8	5*8
labeled to unlabeled data ratio	1:4	1:1	1:4	1:4
learning rate	0.001	0.001	0.001	0.001
first stage iterations	0-60K	0-180K	0-40K	0-60K
second stage iterations	60k-120K	180k-240K	40k-60K	60k-120K
iterations	120K	240K	60K	120K
unsupervised loss weight α	4.0	2.0	4.0	4.0
EMA rate	0.996	0.999	0.999	0.999
confidence threshold	0.4	0.4	0.4	0.4

Table 1. Training settings for different datasets. Here, 'Ablation' means the training setting of the ablation studies in the paper.

Augmentation	Labeled image training	Unlabeled image training	Pseudo-label generation
Scale Jitter	shortest edge $\in [480, 800]$	shortest edge $\in [480, 800]$	shortest edge $\in [480, 800]$
Solarize Jitter	$p = 0.25$, ratio$\in (0, 1)$	$p = 0.25$, ratio$\in (0, 1)$	-
Brightness	$p = 0.25$, ratio$\in (0, 1)$	$p = 0.25$, ratio$\in (0, 1)$	-
Contrast Jitter	$p = 0.25$, ratio$\in (0, 1)$	$p = 0.25$, ratio$\in (0, 1)$	-
Sharpness Jitter	$p = 0.25$, ratio$\in (0, 1)$	$p = 0.25$, ratio$\in (0, 1)$	-
Translation	-	$p = 0.3$, translation ratio$\in (0, 1)$	-
Rotate	-	$p = 0.3$, angle$\in (0, 30°)$	-
Shift	-	$p = 0.3$, angle$\in (0, 30°)$	-
Cutout	num$\in (1, 5)$, ratio$\in (0.05, 0.2)$	num$\in (1, 5)$, ratio$\in (0.05, 0.2)$	-

Table 2. Data augmentations used in our approach. p indicate the probability of choosing a certain type of augmentation.

features exclusively, or should it be applied to both high and low-resolution features for optimal results?

2. Is the integration of a similarity module crucial in query refinement? How would training be impacted if we disregarded similarity features and considered all features comprehensively?

Impact of Attention module: In our Query Refinement, we refine the queries by applying the attention module on F_{t2} features and combining them with F_{t1} features. In this experiment, we study the impact of the attention module in Query Refinement, as highlighted in Table 3. Figure 1 (a) illustrates the concatenating high-resolution features after extracting similar features from the low-resolution without applying an attention network to either set of features. Secondly, as indicated in Figure 1 (b), we apply the attention network on both sets of features. In Figure 1 (c), we extract similar features from the F_{t2} and apply the attention network on just F_{t1} features. In Figure 1 (d), we apply the attention network on F_{t2} features and find similarity with F_{t1} features that gives the best results. The model can focus on capturing essential information by applying the attention module to F_{t2} features. When these enhanced F_{t2} features are compared for similarity with F_{t1} features, the model can get refined detail. It enables the model to make more accurate predictions, leading to better overall performance.

Impact of Similarity module: We reduce the number of queries in query refinement by filtering similar query features in low-resolution features. As indicated in Table 4, removing the similarity module results in a performance decline of 0.3 mAP, increasing the number of queries in a one-to-many training strategy. It confirms the importance of the similarity module in our query refinement strategy. The effectiveness of refined queries using the similarity module is because when enhanced low-resolution features are compared for similarity with high-resolution features, the model can effectively correlate the relevant information from both levels of detail, improving performance.

ID	Attention		mAP	AP_{50}	AP_{75}
	F_{t1}	F_{t2}			
(a)	✗	✗	43.4	58.8	46.1
(b)	✓	✓	40.1	57.8	43.8
(c)	✓	✗	42.8	59.7	45.8
(d)	✗	✓	44.3	61.7	47.6

Table 3. Impact of Attention module. Here, F_{t1} and F_{t2} are the high resolution and low resolution features, respectively.

Qualitative comparison with the baseline. We employ Semi-DETR as the baseline and analyze the impact of Query Refinement on the one-to-many assignment strategy, as in-

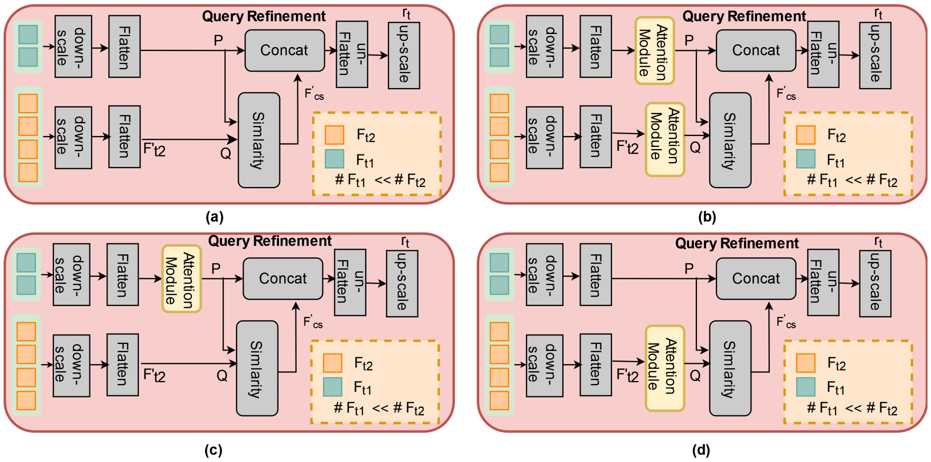

Figure 1. Overview of the Impact of Attention Module in Query Refinement. (a) Query refinement without an attention module, (b) Attention module applied to both low and high-resolution features, (c) Attention module applied to high-resolution features, and (d) Attention module applied to low-resolution features. The best results are achieved for refining queries by applying the attention module to low-resolution features and then combining these with high-resolution features.

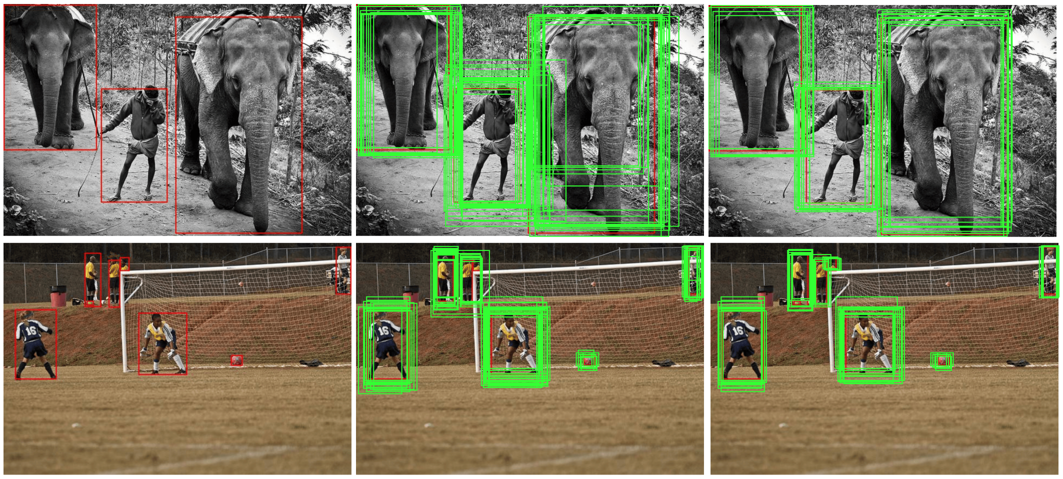
(a) Ground truth (b) Semi DETR (c) Sparse Semi DETR

Figure 2. Qualitative Comparison of positive proposals in One-to-Many assignment strategy: (a) Ground Truth (b) Semi-DETR (c) Sparse Semi-DETR. Our approach, compared to Semi-DETR, generates more refined positive proposals for each ground truth. Here, ground truths are outlined in red, while the positive proposals are highlighted in green. Sparse Semi-DETR performs better in identifying small or hidden objects, as indicated by positive proposals around such items. It employs an attention mechanism, focusing on finer image details, which enhances the detection of hidden objects. Additionally, its similarity module further refines the proposal quality, leading to a notably improved identification accuracy.

Similarity		# Queries (r_t)	mAP	AP_{50}	AP_{75}
F_{t1}	F_{t2}	$\#F_{t1} << \#F_{t2}$			
✗	✗	# $(F_{t1} + F_{t2})$	44.0	61.1	47.5
✗	✓	# $(2 * F_{t1})$	44.3	61.7	47.6

Table 4. Impact of Similarity module. Here, F_{t1} and F_{t2} are the high resolution and low resolution features, respectively.

dicated in Figure 2. Sparse Semi-DETR generates more accurate and refined positive proposals for detecting small or hidden objects. Furthermore, our method significantly reduces the input queries to the decoder compared to Semi-DETR in the one-to-many assignment strategy. As evidenced

Approach	Training time (min)
Semi-DETR	38.56
Sparse Semi-DETR	**34.38** +4.18

Table 5. This is the training time for 1k iterations in one-to-many assignment strategy.

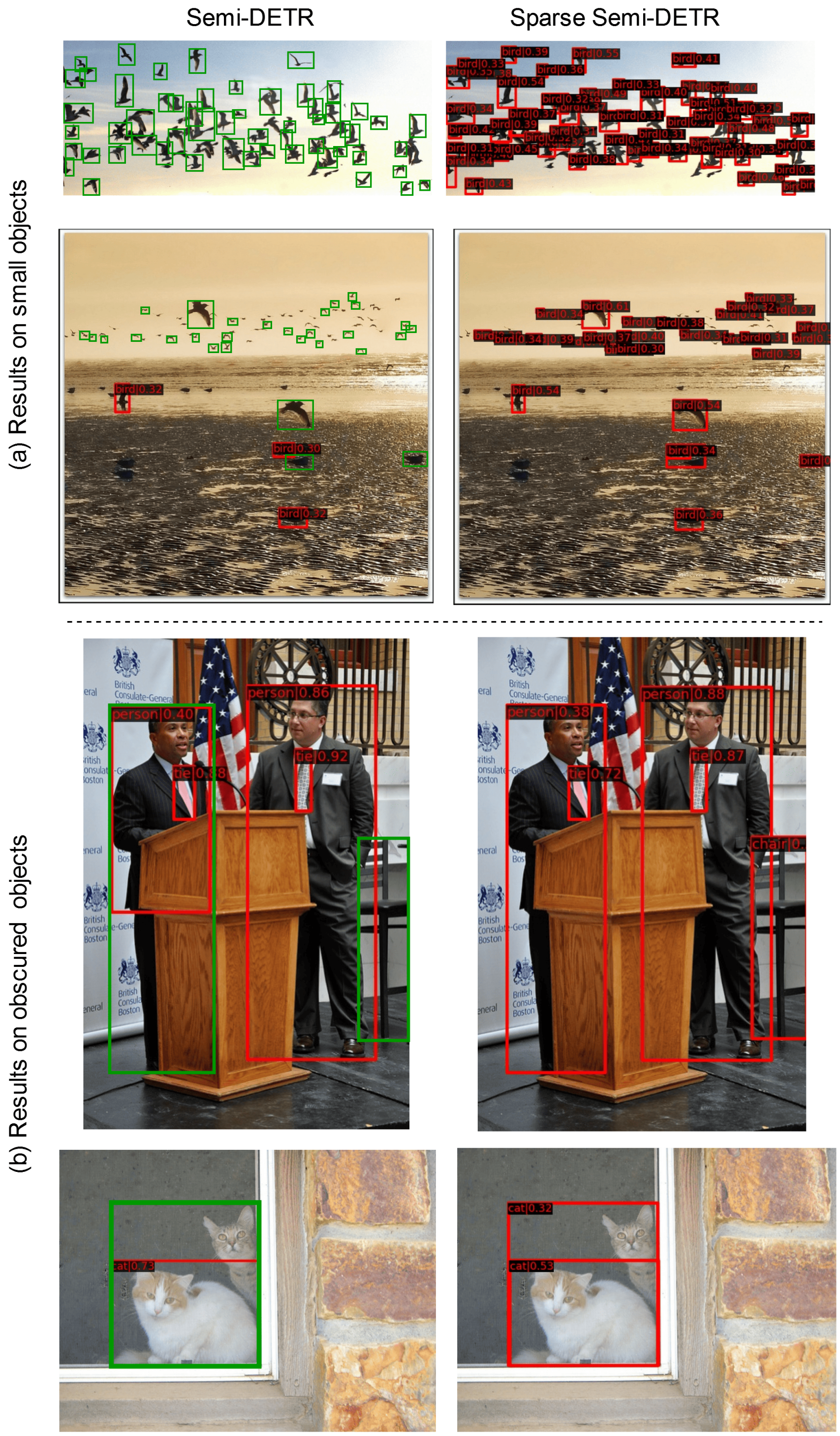

Figure 3. Qualitative comparison on the COCO test set. The prediction results are in red, and the green boxes refer to the prediction difference in Semi-DETR and Sparse Semi-DETR. **(a) Small Objects:** Semi-DETR, on the left, has missed detections of bird objects, indicated with green bounding boxes as false negatives. On the right, red bounding boxes signify correctly identified birds, showcasing Sparse Semi-DETR's more precise and reliable detection capabilities for smaller objects. **(b) Obscured Objects:** The green boxes indicate the regions where the Semi-DETR has either failed to detect an object (false negatives) as the chair or incorrectly estimated the region of the objects, like the person. Sparse Semi-DETR detects obscured objects more precisely, improving performance in complex visual environments.

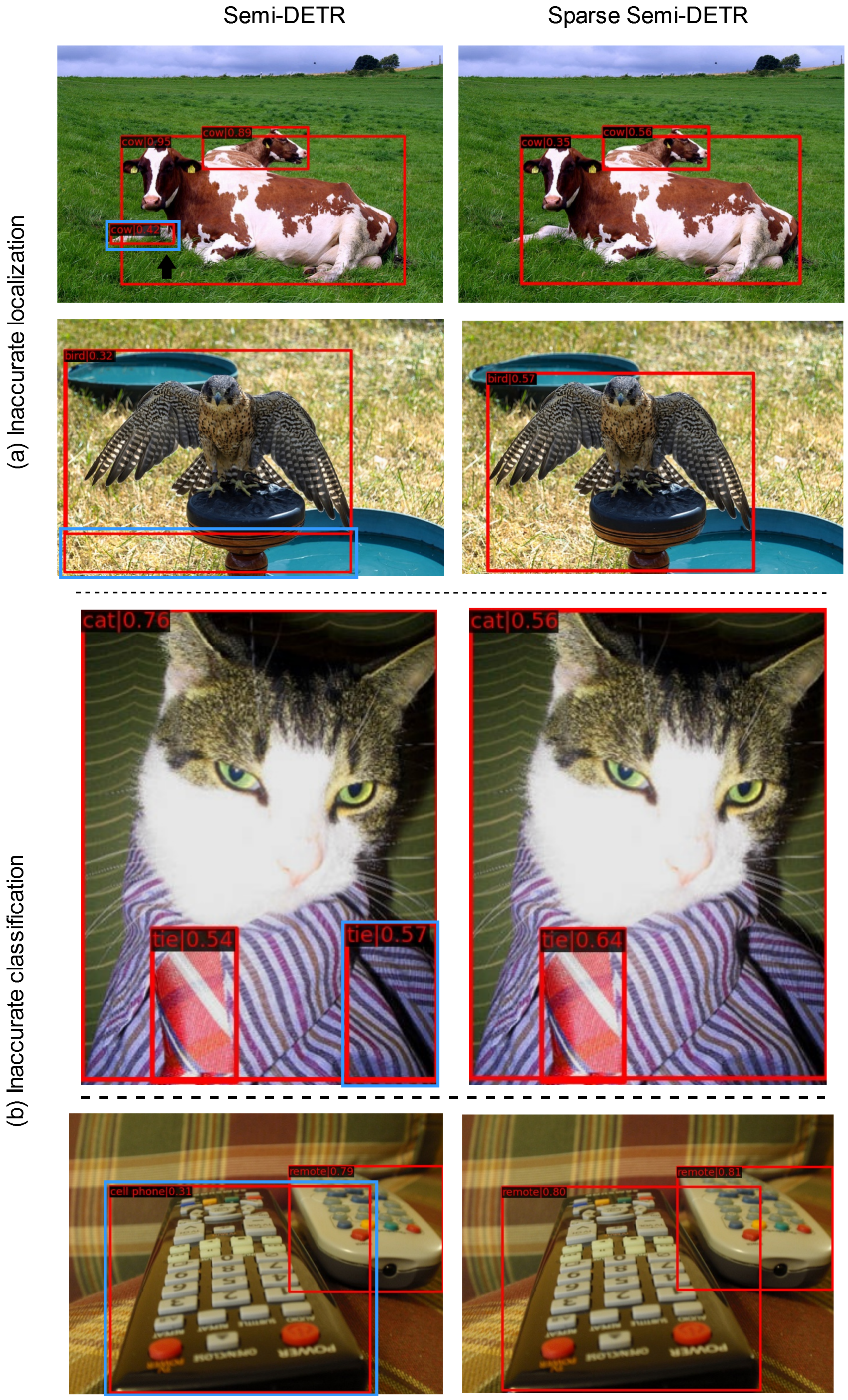

Figure 4. Qualitative comparison on the COCO test data. The prediction results are in red, and the blue boxes highlight the prediction difference in Semi-DETR and Sparse Semi-DETR. **(a) Inaccurate localization:** Semi-DETR incorrectly places multiple bounding boxes around the individual cow and bird objects, indicating it misidentified them as several entities instead of one. Sparse Semi-DETR, however, shows reduced duplications in bounding boxes. **(b) Inaccurate classification:** Semi-DETR partially misidentified a cat wearing a tie as the tie itself and confused a 'remote' object with a 'cell phone' as represented with a blue bounding box. Sparse Semi-DETR reduces the misidentification issues present in Semi-DETR, such as not confusing a cat wearing a tie as the tie itself and correctly identifying a 'remote' without mistaking it as a 'cell phone.'

in Table 5, this refinement of queries has resulted in a training time reduction of 4.18 minutes on 1k iterations, amounting to a relative decrease of 10.84%. To further compare our Sparse Semi-DETR with the baseline Semi-DETR, we visualize the predicted bounding boxes on test2017, trained on the COCO 10% label data. In Figure 3 and Figure 4, we plot the predicted bounding boxes in red, while green and blue boxes highlight the differences in the prediction of Semi-DETR and Sparse Semi-DETR. There are four general properties that we could observe in our demonstration.

1. Firstly, Sparse Semi-DETR significantly improves the detection of small objects compared to Semi-DETR, primarily due to its advanced query refinement mechanism. As shown in Figure 3 (a), Sparse Semi-DETR is particularly beneficial for identifying small subjects such as birds, where Semi-DETR often struggles because of its inadequate query feature representation. By capturing refined details, Sparse Semi-DETR ensures more precise and reliable detection of these smaller objects, enhancing overall performance in object detection tasks.
2. Secondly, for obscured objects, Sparse Semi-DETR provides a distinct advantage over Semi-DETR through its refined query mechanism as indicated in Figure 3 (b). It allows Sparse Semi-DETR to understand better details of partially hidden objects, which is often challenging for Semi-DETR due to its less robust query features. As a result, Sparse Semi-DETR achieves more precise detection of obscured objects, leading to improved performance in complex visual environments.
3. Thirdly, Sparse Semi-DETR exhibits a significant advantage in removing duplicate predictions after the second stage. It is because of a reliable pseudo-label filtering module that filters out some duplications and selects more accurate pseudo-labels. A notable example is the detection of cow objects, as shown in Figure 4 (a). While Semi-DETR tends to provide two predictions for the same object, Sparse Semi-DETR demonstrates remarkable proficiency in duplicate removal.
4. Fourthly, in the semi-supervised setting, Semi-DETR often faces challenges in accurately categorizing objects, even when the location is correctly identified. For example, Semi-DETR labels a 'remote' object as a 'cell phone' despite accurately providing its location as indicated in Figure 4 (b). This misclassification often arises from a disparity between the features used for object detection (regression) and those used for classification. In contrast, Sparse Semi-DETR stands out by adeptly distinguishing between closely related categories. It leverages its innovative attention and similarity module, which dynamically selects the most relevant features for each task, ensuring a more unified and accurate performance in both classification and localization.

References

[1] Binghui Chen, Pengyu Li, Xiang Chen, Biao Wang, Lei Zhang, and Xian-Sheng Hua. Dense learning based semi-supervised object detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4805–4814, 2022. 1

[2] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 1

[3] Qiushan Guo, Yao Mu, Jianyu Chen, Tianqi Wang, Yizhou Yu, and Ping Luo. Scale-equivalent distillation for semi-supervised object detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14502–14511, 2022. 1

[4] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Neural Information Processing Systems*, 2019.

[5] Gang Li, Xiang Li, Yujie Wang, Yichao Wu, Ding Liang, and Shanshan Zhang. Pseco: Pseudo labeling and consistency training for semi-supervised object detection. In *Computer Vision – ECCV 2022*, pages 457–472, Cham, 2022. Springer Nature Switzerland.

[6] Yen-Cheng Liu, Chih-Yao Ma, and Zsolt Kira. Unbiased teacher v2: Semi-supervised object detection for anchor-free and anchor-based detectors, 2022.

[7] X. Wang, X. Yang, S. Zhang, Y. Li, L. Feng, S. Fang, C. Lyu, K. Chen, and W. Zhang. Consistent-teacher: Towards reducing inconsistent pseudo-targets in semi-supervised object detection. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3240–3249, Los Alamitos, CA, USA, 2023. IEEE Computer Society.

[8] 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. *CoRR*, abs/2106.09018, 2021. 1

[9] Qize Yang, Xihan Wei, Biao Wang, Xian-Sheng Hua, and Lei Zhang. Interactive self-training with mean teachers for semi-supervised object detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5937–5946, 2021. 1

[10] Jiacheng Zhang, Xiangru Lin, Wei Zhang, Kuo Wang, Xiao Tan, Junyu Han, Errui Ding, Jingdong Wang, and Guanbin Li. Semi-detr: Semi-supervised object detection with detection transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23809–23818, 2023. 1

[11] Qiang Zhou, Chaohui Yu, Zhibin Wang, Qi Qian, and Hao Li. Instant-teaching: An end-to-end semi-supervised object detection framework. *CoRR*, abs/2103.11402, 2021. 1