PROB: Probabilistic Objectness for Open World Object Detection Supplementary Material

A. How does PROB not use pseudo-labeling?

In this section, we aim to clarify how and why PROB does not require pseudo-labeling to supervise unknown object detection during training. The need for pseudo-labeling stems from the datasets themselves, which are not densely labeled (i.e., not all the objects in an image are labeled). Consequently, it becomes difficult to differentiate between unknown objects and background as a proposal that does not overlap with any ground-truth object may still contain an unlabeled (or labeled unknown) object. To address this lack of supervision between unknowns and background proposals, existing methods, such as OW-DETR [9], use pseudo-labeling, where a certain number of unmatched object proposals are labeled as 'unknown' using different heuristics (e.g., high backbone activation) during training and are then used to supervise the classification head.

In this work, we propose to tackle this lack of supervision differently. Rather than directly needing to generate supervision between background and unknown objects, we aim to separate this single unsupervised problem into two supervised ones:

1. Objectness detection
2. Object Detection/OOD classification

Specifically, with the embeddings class prediction, we would like to perform object *and then* object class prediction. For class-agnostic object detection $p(o|\boldsymbol{q})$, we introduced our probabilistic objectness head, which aims to classify if a particular query embedding represents an object or background. Meanwhile, for object class predictions, we use the traditional D-DETR classification head (see eq. 1). The classification head, $f^t_{\text{cls}}(\boldsymbol{q})$, therefore, operates under the assumption that *all* the queries are objects, and leaves the task of separating objects and background to the objectness head.

A.1. Training

Training for objectness detection is not trivial as the dataset is not densely labeled. When training a simple classifier on *all* object classes, the classifier will still learn to classify unlabeled objects as background [12, 26]. We, therefore, employ our probabilistic approach which can leverage only positive supervision (i.e., applying the loss only on known objects, or matched query embeddings) without causing it to classify *all* proposals as objects.

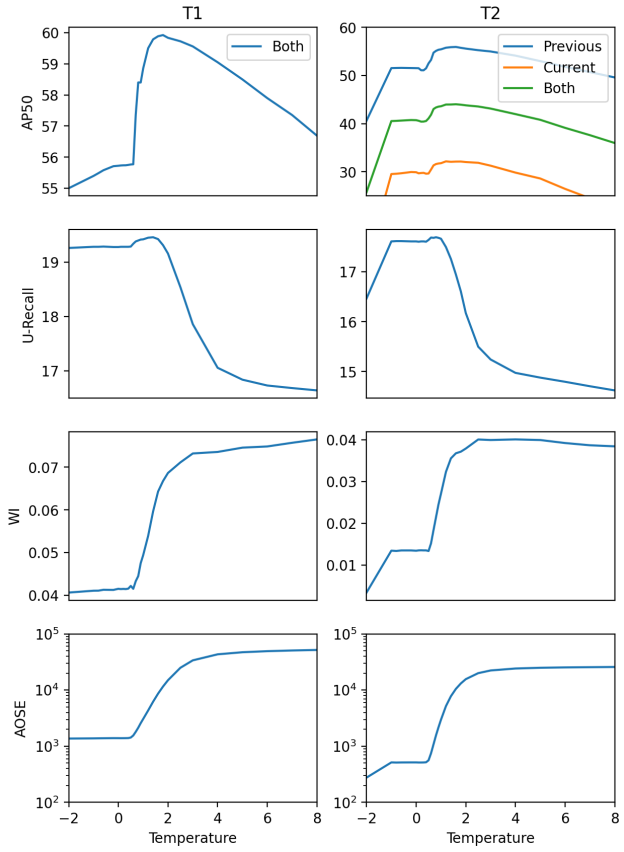

Figure 5. **Objectness Temperature Sweep Experiment**. Effect of objectness temperature on model performance for Task 1 and Task 2. WI and A-OSE monotonically increase with objectness temperature, while U-Recall and AP50 peak at an objectness temperature of 1.0-1.6. See Sec. B.1 for additional details.

Meanwhile, our classification head is trained as a traditional classification head, where the $K+1$ logit represents all proposals not matched to a ground-truth object (both unknown/unlabeled objects and background). This can also be viewed as a type of OOD classification problem, where the classifier needs to classify each query into one of the C know classes or into OOD.

A.2. Inference

During inference, unlike other methods, PROB uses the objectness head to 'filter out' queries that represent background, so the classification head needs only to classify whether an object (not proposal) is a known object class

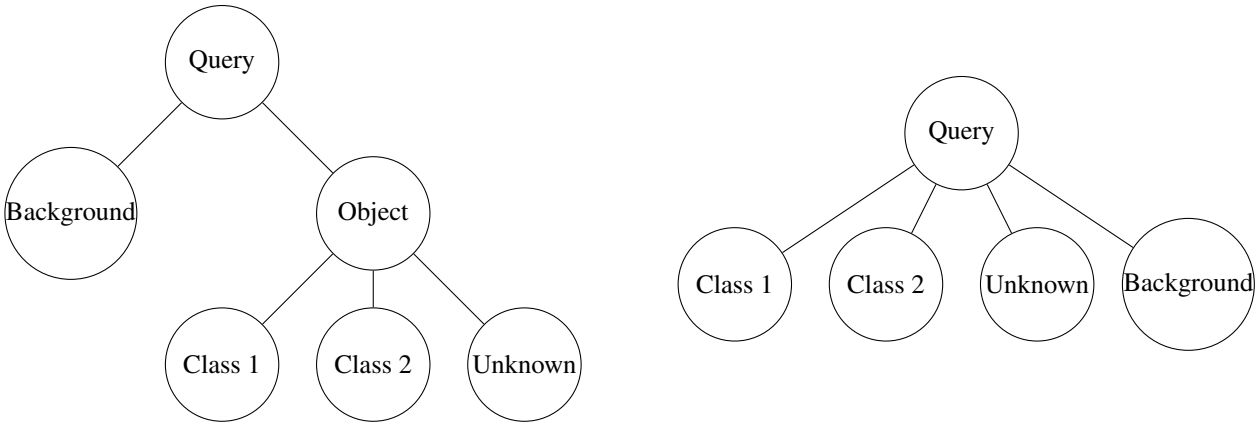

Figure 6. **A Comparison of PROBs and Other Methods Inference Scheme.** (left) PROB performs inference in two stages: the first -top- stage is the objectness detection, and it 'filters out' unknown objects. The second -bottom- stage performs classification into one of the known classes or an unknown object. (right) other methods attempt to directly classify a proposal into either one of the known classes, an unknown object, or a background.

or not (see Fig. 6, left). Given a query that represents *only* background, the objectness head should predict a very low probability of it being an object (i.e., $f^t_{\text{obj}}(\boldsymbol{q}) \approx 0$) and suppresses the prediction of any objects. Conversely, if the query contains an object, then the objectness prediction should be high (i.e., $f^t_{\text{obj}}(\boldsymbol{q}) \approx 1$), and the task of classifying the query into any of the known objects or an unknown object is left to the classification head. This is why we can take *all* unmatched queries as 'unknown objects' during training - as at inference, all background proposals will be suppressed. The revised inference scheme can be conceptualized as a two-step process, where first the objectness head filters out background proposals, and only then the classification head classifies the embeddings into one of the known classes *or* an unknown object (see Fig. 6, left). Meanwhile, at inference, other methods directly classify each query embedding into either one of the known classes, unknown, or background (see Fig. 6, right).

B. Additional Quantitative Results

B.1. Objectness Temperature Variation

In Sec. 4.1, eq. 3, we introduced the proposed objectness prediction. However, there is an additional hyperparameter, specifically 'objectness temperature', which can be varied (see Sec. D). Objectness temperature does not affect training and can be varied at test time if needed. It controls the degree of confidence in objectness prediction, with higher objectness temperature resulting in less confident objectness predictions. In all of the reported experiments, the objectness temperature was set to 1.3. To investigate its effect on model inference, we evaluated the model at different objectness temperatures (see Fig. 5). Fig. 5 shows that U-Recall and known mAP have optimum points; however, the two do not necessarily coincide, as can be seen in Task 2. Meanwhile, there is a definite trade-off between the models' unknown-known object confusion (as quantified by WI and A-OSE) and U-Recall and mAP. A-OSE gets as low as 1300 in Task 1 and 500 in Task 2 (excluding the drop-off at ($\tau = -2$), with the known mAP remaining reasonably high at 55.8 (dropping from 59.5) and 40.7 (dropping from 44.0). The same can be said of wilderness impact (WI). It is worth noting that, for the methods that have lower A-OSE and WI in Tab. 4, there exists an objectness temperature where PROB outperforms them in terms of known mAP (previous, current, and both), U-Recall, WI, and A-OSE.

B.2. Evaluation using WI and A-OSE Metrics

In Sec. 5.1, we referred to additional metrics that quantify unknown-known object confusion. Tab. 4 shows a comparison of the different open-world object detection (OWOD) methods on the M-OWODB dataset [11] in terms of unknown recall (U-Recall), wilderness impact (WI), and absolute open-set error (A-OSE). U-Recall measures the models' ability to detect unknown objects and indicates the degree of unknown objects that are detected by an OWOD method. Meanwhile, WI and A-OSE measure a model's confusion in predicting an unknown instance as a known class. Specifically, WI measures the effect the unknown detections have on the model's precision. WI is problematic as U-Recall grows, the unknown object precision becomes more dominant, causing WI to increase - even if the models have the same unknown object precision. Therefore, the WI will typically increase with U-Recall. Meanwhile, A-OSE measures the total number of unknown instances detected as one of the known classes and is less affected by U-Recall.

When examining Tab. 4, it becomes apparent that PROB outperforms all other OWOD methods in terms of U-Recall

Table 4. **Uknown Object Confusion on M-OWODB.** The comparison is shown in terms of wilderness impact (WI), absolute open set error (A-OSE) and unknown class recall (U-Recall). The unknown recall (U-Recall) metric quantifies a model's ability to retrieve the unknown object instances. PROB achieves improved WI, A-OSE and U-Recall over OW-DETR across tasks, thereby indicating less confusion in detecting unknown instances as known classes with higher unknown instance detection capabilities. See Sec. B.2 for additional details.

Task IDs (→)	**Task 1**			**Task 2**			**Task 3**		
	U-Recall (↑)	WI (↓)	A-OSE (↓)	U-Recall (↑)	WI (↓)	A-OSE (↓)	U-Recall (↑)	WI (↓)	A-OSE (↓)
ORE – EBUI [11]	4.9	0.0621	10459	2.9	0.0282	10445	3.9	0.0211	7990
2B-OCD [32]	12.1	0.0481	-	9.4	0.160	-	11.6	0.0137	-
OW-DETR [9]	7.5	0.0571	10240	6.2	0.0278	8441	5.7	0.0156	6803
OCPL [34]	8.3	**0.0413**	5670	7.6	0.0220	5690	11.9	0.0162	5166
Ours: PROB	**19.4**	0.0569	5195	17.4	0.0344	6452	19.6	0.0151	2641
Ours: PROB($\tau = 0.5$)	19.3	0.0415	**1428**	**17.7**	**0.0133**	**562**	**19.7**	**0.0082**	**387**

Table 5. **Effect of the number of queries on PROB performance.** The comparison is shown in terms of known mean average precision (mAP) and unknown recall (U-Recall) on M-OWODB. **PROB** `-50Q/75Q` varies the number of queries to 50/75, which had little effect on known mAP while having some impact on U-Recall (which is now recall at 50/75/100), which is to be expected.

Task IDs (→)	**Task 1**		**Task 2**				**Task 3**				**Task 4**		
	U-Recall (↑)	mAP (↑) Current known	U-Recall (↑)	mAP (↑) Previously known	Current known	Both	U-Recall (↑)	mAP (↑) Previously known	Current known	Both	mAP (↑) Previously known	Current known	Both
PROB `-50Q`	15.3	58.5	12.3	55.6	31.5	43.5	14.3	42.3	20.0	34.9	35.0	17.3	30.6
PROB `-75Q`	17.8	59.6	15.5	55.8	32.0	43.9	16.2	43.0	21.5	35.9	35.6	18.6	31.5
Final: **PROB**	19.4	59.5	17.4	55.7	32.2	44.0	19.6	43.0	22.2	36.0	35.7	18.9	31.5

while having lower (or similar) A-OSE. While the A-OSE reduction in Tab. 4 is not very large, reducing the objectness temperature (see Sec. B.1) results in a much lower A-OSE. For example, for Task 2, at an objectness temperature of 0.5, PROB's OWOD performance can be seen in Tab. 4. Specifically, A-OSE drops to 562, with only marginal degradation of the known mAP, as can be seen in Fig. 5. WI additionally reduces by almost 50% to 0.0133. This shows that PROB can be easily tuned towards either better unknown or known precision simply by varying the objectness temperature.

B.3. Additional Ablations

In this section, we present an additional ablation study of PROB where we examine the effect of varying numbers of queries on PROB's performance. We report results on M-OWODB split [11] and evaluation metrics, as described in Section 5.2. Specifically, we vary the number of queries to 50, 75, and 100, while keeping all other settings the same as the default configuration.

Tab. 5 summarizes our findings. As expected, decreasing the number of queries results in a slight drop in U-Recall, as the model has access to fewer query embeddings to detect unknown objects. However, we observe that the effect on known mAP is negligible. For instance, in Task 1, using only 50 queries results in a drop of 21% in U-Recall while the known mAP is almost identical to the default setting. Similarly, using 75 queries results in a drop of 8.2% in U-Recall at 75, with no noticeable effect on known mAP. These results demonstrate the robustness of PROB to the number of queries used.

B.4. Incremental Learning

In Sec. 5.3, we reported PROB's incremental learning capabilities. For completeness, we add the per-class AP of the incremental learning experiments reported in Tab. 3 in Tab. 6. As discussed in Sec. 4.2, OWOD methods rely on exemplar replay for mitigating catastrophic forgetting. First introduced by Prabhu *et al*. [23], exemplar replay adds an additional fine-tuning stage on a balanced set of exemplars after every training step and was shown to be extremely effective. Nonetheless, PROB's *active* selection of high/low objectness scoring exemplars further improved both unknown and known object detection performance, as shown in our ablations (see Tab. 2). Interestingly, the known object performance seems to grow over time (with a delta of 0.5, 0.7, and 0.9 in Tasks 2, 3, and 4, respectively), further motivating the utility of PROB's active exemplar selection.

B.5. Open Set

While not the focus of our work, we also include the open-set detection results. The open-set detection task evaluates whether known object mAP degrades in an open dataset. This is done by evaluating the model on a 'closed' (contains only known objects) and 'open' (compose 50/50

Table 6. **State-of-the-art comparison for incremental object detection (iOD) on PASCAL VOC.** We experiment on 3 different settings. The comparison is shown in terms of per-class AP and overall mAP. The 10, 5 and 1 class(es) in gray background are introduced to a detector trained on the remaining 10, 15 and 19 classes, respectively. PROB achieves favorable performance in comparison to existing OWOD approaches in all three settings. See Sec. B.4 for additional details.

10 + 10 setting	aero	cycle	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	bike	person	plant	sheep	sofa	train	tv	mAP
ILOD [27]	69.9	70.4	69.4	54.3	48	68.7	78.9	68.4	45.5	58.1	59.7	72.7	73.5	73.2	66.3	29.5	63.4	61.6	69.3	62.2	63.2
Faster ILOD [22]	72.8	75.7	71.2	60.5	61.7	70.4	83.3	76.6	53.1	72.3	36.7	70.9	66.8	67.6	66.1	24.7	63.1	48.1	57.1	43.6	62.1
ORE – (CC + EBUI) [11]	53.3	69.2	62.4	51.8	52.9	73.6	83.7	71.7	42.8	66.8	46.8	59.9	65.5	66.1	68.6	29.8	55.1	51.6	65.3	51.5	59.4
ORE – EBUI [11]	63.5	70.9	58.9	42.9	34.1	76.2	80.7	76.3	34.1	66.1	56.1	70.4	80.2	72.3	81.8	42.7	71.6	68.1	77	67.7	64.5
OW-DETR [9]	61.8	69.1	67.8	45.8	47.3	78.3	78.4	78.6	36.2	71.5	57.5	75.3	76.2	77.4	79.5	40.1	66.8	66.3	75.6	64.1	65.7
Ours: PROB	70.4	75.4	67.3	48.1	55.9	73.5	78.5	75.4	42.8	72.2	64.2	73.8	76.0	74.8	75.3	40.2	66.2	73.3	64.4	64.0	**66.5**
15 + 5 setting	aero	cycle	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	bike	person	plant	sheep	sofa	train	tv	mAP
ILOD [27]	70.5	79.2	68.8	59.1	53.2	75.4	79.4	78.8	46.6	59.4	59	75.8	71.8	78.6	69.6	33.7	61.5	63.1	71.7	62.2	65.8
Faster ILOD [22]	66.5	78.1	71.8	54.6	61.4	68.4	82.6	82.7	52.1	74.3	63.1	78.6	80.5	78.4	80.4	36.7	61.7	59.3	67.9	59.1	67.9
ORE – (CC + EBUI) [11]	65.1	74.6	57.9	39.5	36.7	75.1	80	73.3	37.1	69.8	48.8	69	77.5	72.8	76.5	34.4	62.6	56.5	80.3	65.7	62.6
ORE – EBUI [11]	75.4	81	67.1	51.9	55.7	77.2	85.6	81.7	46.1	76.2	55.4	76.7	86.2	78.5	82.1	32.8	63.6	54.7	77.7	64.6	68.5
OW-DETR [9]	77.1	76.5	69.2	51.3	61.3	79.8	84.2	81.0	49.7	79.6	58.1	79.0	83.1	67.8	85.4	33.2	65.1	62.0	73.9	65.0	69.4
Ours: PROB	77.9	77.0	77.5	56.7	63.9	75.0	85.5	82.3	50.0	78.5	63.1	75.8	80.0	78.3	77.2	38.4	69.8	57.1	73.7	64.9	**70.1**
19 + 1 setting	aero	cycle	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	bike	person	plant	sheep	sofa	train	tv	mAP
ILOD [27]	69.4	79.3	69.5	57.4	45.4	78.4	79.1	80.5	45.7	76.3	64.8	77.2	80.8	77.5	70.1	42.3	67.5	64.4	76.7	62.7	68.2
Faster ILOD [22]	64.2	74.7	73.2	55.5	53.7	70.8	82.9	82.6	51.6	79.7	58.7	78.8	81.8	75.3	77.4	43.1	73.8	61.7	69.8	61.1	68.5
ORE – (CC + EBUI) [11]	60.7	78.6	61.8	45	43.2	75.1	82.5	75.5	42.4	75.1	56.7	72.9	80.8	75.4	77.7	37.8	72.3	64.5	70.7	49.9	64.9
ORE – EBUI [11]	67.3	76.8	60	48.4	58.8	81.1	86.5	75.8	41.5	79.6	54.6	72.8	85.9	81.7	82.4	44.8	75.8	68.2	75.7	60.1	68.8
OW-DETR [9]	70.5	77.2	73.8	54.0	55.6	79.0	80.8	80.6	43.2	80.4	53.5	77.5	89.5	82.0	74.7	43.3	71.9	66.6	79.4	62.0	70.2
Ours: PROB	80.3	78.9	77.6	59.7	63.7	75.2	86.0	83.9	53.7	82.8	66.5	82.7	80.6	83.8	77.9	48.9	74.5	69.9	77.6	48.5	**72.6**

Table 7. **Performance comparison on open-set object detection task**. PROB had a simular performance to that reported by OW-DETR and ORE.

Evaluated on →	Pascal VOC 2007	Open-Set (WR1)
Faster R-CNN	81.8	77.1
RetinaNet	79.2	73.8
Dropout Sampling [21]	78.1	71.1
ORE [11]	81.3	78.2
OW-DETR [9]	82.1	**78.6**
Ours:PROB	**82.3**	78.0

known and unknown objects). As we focused on discovering unknown objects (not mitigating the misclassification of unknown objects as known objects), no improvement compared to OW-DETR was seen (see Tab. 7).

C. Additional Qualitative Results

In Sec. 5.1, we noted that PROB has favorable qualitative performance on the M-OWODB. Specifically, PROB consistently detects unknown objects across Tasks, and learns them properly when the objects are incrementally learned. For example, in Fig. 7, left, PROB detects both frisbees in Task 1 and 2 as 'unknown objects' and in Task 3, after frisbees are added as a known class, as frisbees. Meanwhile, OW-DETR detected only one frisbee in Task 1, none in Task 2, and both as frisbees in Task 3. In Fig. 7, right, PROB detected both toilets and the sink as 'unknown' while OW-DETR detected only one of the toilets across all tasks. OW-DETR additionally 'forgets' the bottle in Task 2, and remembers it again in Task 3, while PROB consistently detected the bottle. When examining the confidence scores of known and unknown objects, it is clear that PROB makes balanced predictions of known and unknown objects (i.e., predictions with similar confidence scores), while OW-DETR predicts unknown objects with very low confidence. This imbalance may lead to difficulty in detecting unknown objects at test time, as unknown proposals may not meet the detection confidence threshold.

Unknown Object Confidence and Forgetting. PROB detects unknown objects more confidently and does not forget the unknown objects in later tasks. For example, in Fig. 9, most of the unknown object detections of OW-DETR have less than $< 10\%$ confidence. Unlike OW-DETR's unknown detection performance, which seems to degrade/fluctuate over time, PROB's unknown object detection seems to improve. For example, In Fig. 9, left, PROB consistently detected the four unknown objects in the image, with little the bounding box localization and confidence variation across tasks, while OW-DETR's bound box localization, confidence, and final predictions seem erratic, with it not detecting any unknown objects in Task 3. In Fig. 9, right, there is another example of OW-DETR catastrophically forgetting all unknown objects. Here, OW-DETR detected the remote, book, and trash can in Tasks 1 and 2, but forgot all of them in Task 3, unlike PROB.

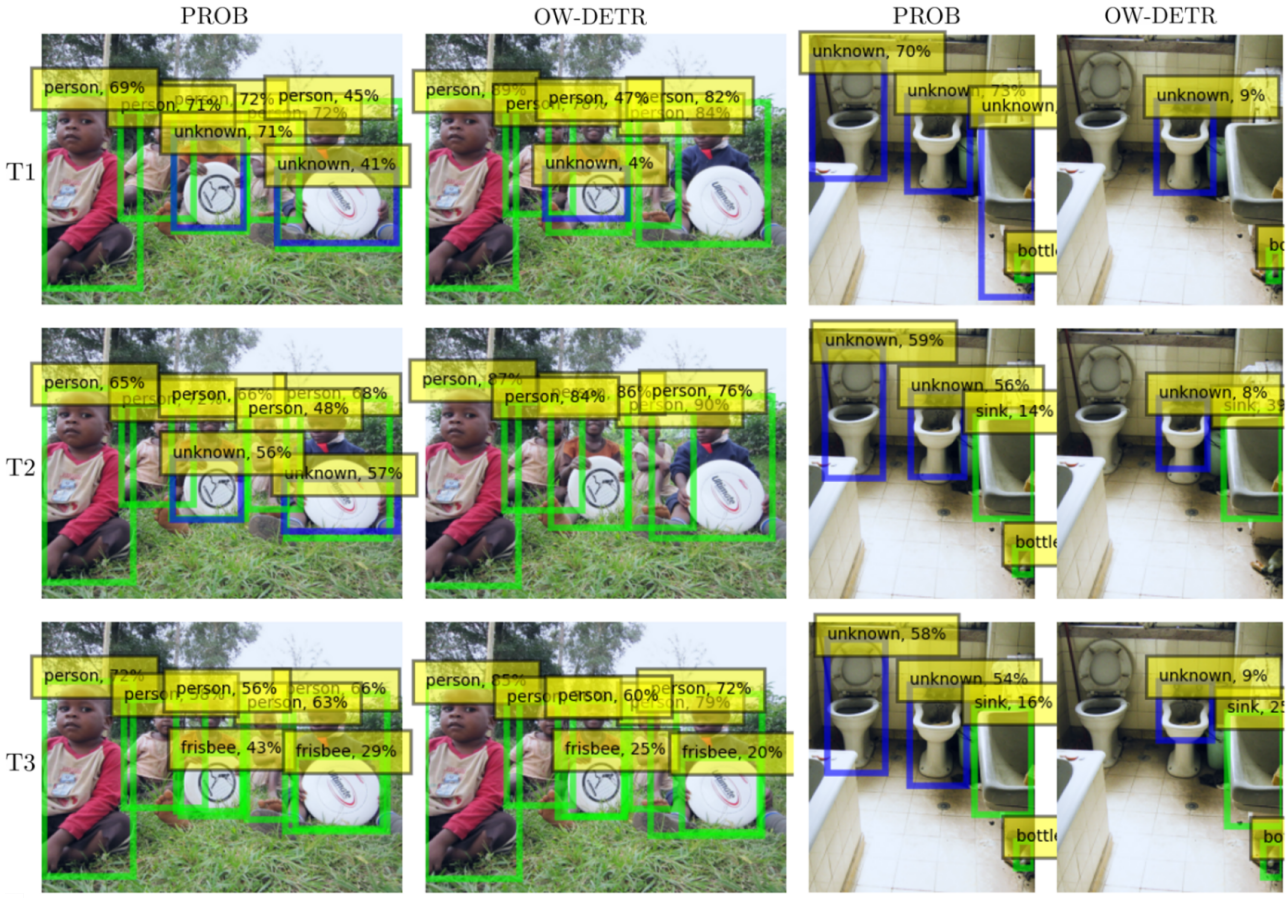

Figure 7. **General OWOD Performance.** PROB appears to have favorable OWOD performance, detecting unknowns and incrementally learning them, while OW-DETR does not. For example, in the right column, PROB detected both toilets across all tasks and also detected the sink in Task 1 before learning it in Task 2, while OW-DETR did not. When OW-DETR did detect an unknown object, it tended to have very low confidence ($\sim 10\%$), while PROB had confidence on par with that of the known objects, showing that PROB had more balanced unknown-known predictions. OW-DETR additionally 'forgot' the bottle in Task 2.

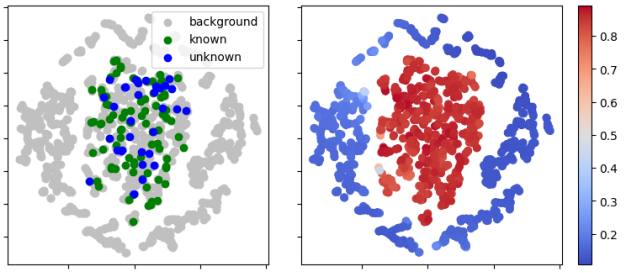

Figure 8. **t-SNE Visualization of Query Embeddings.** (left) – colored by ground truth labeling. As MS-COCO is not densely labeled, the background embeddings may contain unlabeled objects. (right) – colored by predicted objectness.

Unknown Object Detection Consistency. Qualitatively PROB tends to detect unknown objects more consistently, both across tasks (Fig. 10 (a)) and across unknown object instances (Fig. 10 (b)). In Fig. 10 (a), you can see that PROB consistently detects the same objects across tasks (e.g., laptop and keyboard) while OW-DETR detects a different unknown object in every Task. In Fig. 10 (b), you can see that PROB tends to detect all the unknown objects of the same class (e.g., zebras, giraffes, and kites), while OW-DETR seems to miss obvious instances of the same object class. For example, in Fig. 10 (b), top, OW-DETR misses the zebra in the foreground, seemingly only detecting the zebras more in the background. This shows the relatively poor performance of OW-DETR in detecting unknown objects, as it doesn't seem to generalize unknown objects across the same class. This puts into question what object features OW-DETR extracts to make its predictions, as they do not seem robust, unlike PROB which seems to do this quite well.

t-SNE Visualization of Query Embeddings. Fig. 8 shows the t-SNE visualization of the query embeddings, colored by ground-truth labeling and predicted objectness. The visualization demonstrates a clear separation between objects and background. While there are some unlabeled

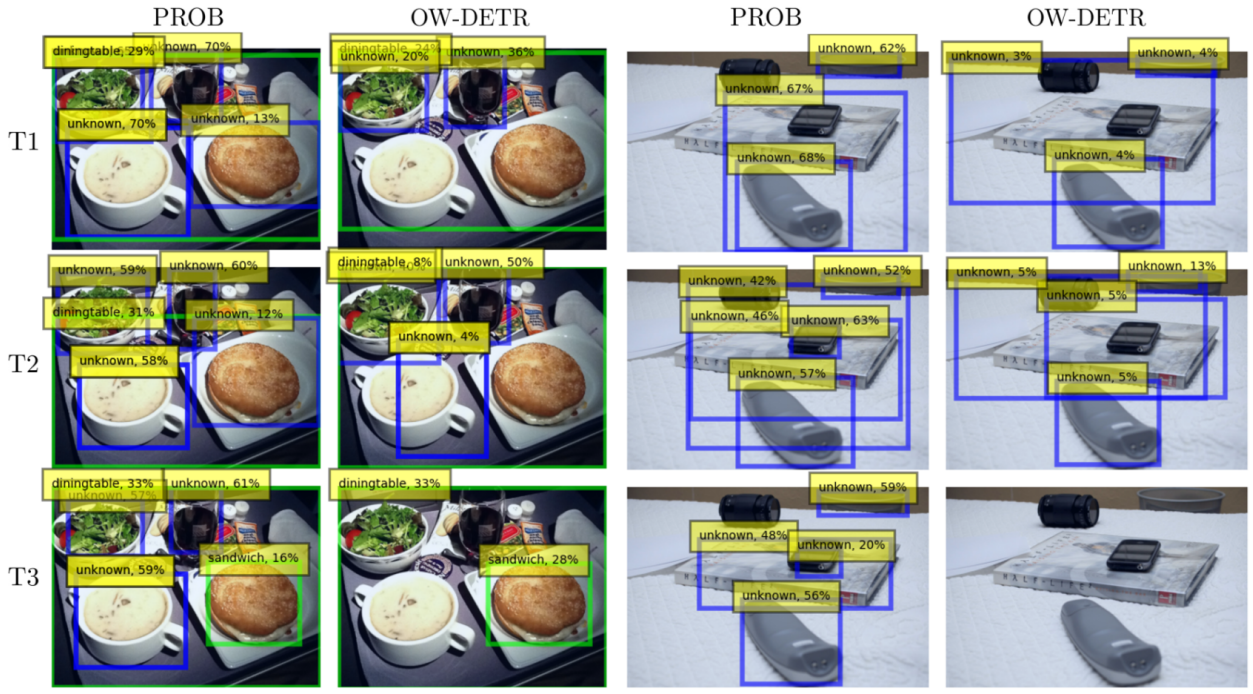

Figure 9. **Unknown object catastrophic forgetting.** PROB does not catastrophically forget unknown objects, while OW-DETR does. For example, OW-DETR catastrophically forgot all previously detected unknown objects when learning for Task 3, while PROB did not. After learning for Task 3, OW-DETR no longer detects the bowls (which it previously detected) while PROB continued to detect them.

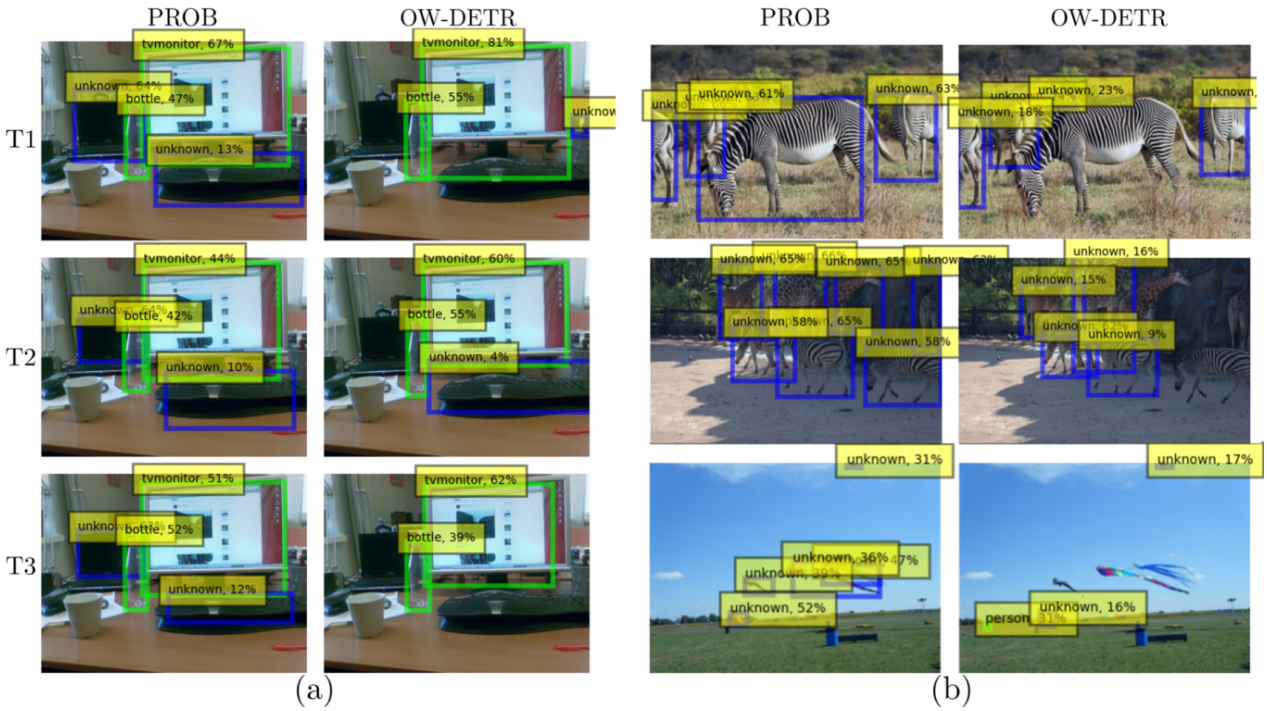

Figure 10. **Uknown object detection consistency on M-OWODB.** (a) depicts the predictions of PROB and OW-DETR on the same image across different tasks. PROB consistently detects the same unknown objects as such, while OW-DETR does not. (b) examples of PROB and OW-DETR predictions. Even in the same image, OW-DETR does not detect obvious examples of unknown objects (e.g., foreground zebra in the top image), while PROB consistently detects unknown objects from the same class.

points in the object area, it is possible that they represent unknown or unlabeled objects that were not annotated in the ground truth. Meanwhile, no instances of labeled objects can be found in the 'background' region of the embeddings.

D. Additional Implementation Details

We found that assuming that the channels are iid distributed did not result in a change in performance while reducing training time and improving model stability, i.e.,

$$\mathbf{\Sigma} = \boldsymbol{I} \cdot \boldsymbol{\sigma}.$$

This iid assumption makes inverting $\mathbf{\Sigma}$ trivial and easily computed during training with little effect on training time. When training with an unrestricted $\mathbf{\Sigma}$, heavy regularization of the matrix inversion calculation of the covariance, required for stable training, causes it to become diagonal. For objectness prediction (inference/evaluation **only**), we added an objectness temperature hyperparameter, τ,

$$f^t_{\text{obj}}(\boldsymbol{q}) = \exp\big(-\tau \cdot d_M(\boldsymbol{q})^2\big),$$

which was set to 1.3 in all of our experiments ($\tau = 1.3$), based on our temperature sweep experiments (see Sec. B.1). PROB is then trained end-to-end with the joint loss:

$$\mathcal{L} = \mathcal{L}_{\text{c}} + \mathcal{L}_{\text{b}} + \alpha\mathcal{L}_{\text{o}},$$

with four Nvidia A100 40GB GPUs, with a batch size of 5. The learning rate was taken to be 2×10^{-3}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay of 10^{-4}, and a learning rate drop after 35 epochs by a factor of 10. For finetuning during the incremental learning step, the learning rate is reduced by a factor of 10. All other hyperparameters were taken as reported in OW-DETR [9].

E. Limitations and Social Impacts

While the OWOD field has been rapidly progressing, much improvement is required to reach the more nuanced aspects of the OWOD objective. As known and unknown object detection rely on different forms of supervision, their predictions are imbalanced with respect to the relation of prediction score and confidence. Exploration of energy-based models [8] could be a solution to this problem while enabling better separation of known and unknown object classes in the embedded feature space. Additional work is still needed in better benchmark design. Current benchmarks expose an entire dataset of novel objects per task. As unknown object recall improves, OWOD algorithms should begin attempting to only discover unknown objects detected by the model. This essentially adds an additional active learning stage between incremental learning steps.

Open-world learning bridges the gap between benchmarks and the real world. In doing so, OWOD algorithms will encounter situations with social impact. To detect new objects, OWOD relay on unknown object detection, which may be biased given the initial training dataset. Future research should not only look at unknown object detection capabilities but also its possible biases. To do so, it would be useful to break down unknown object detection capabilities into the relevant subclasses. Future models should integrate 'forgetting' capabilities that can be applied to particular object classes out of legal and/or privacy concerns. Finally, saving actual images as exemplars may also constitute privacy violations in the open world. As OWOD methods are deployed in the real world, images selected as exemplars will inevitably contain not only the known but also other unknown object classes. These images will be stored as part of the algorithm's lifetime and may contain private or sensitive information. Future work should work on either replacing or censoring selected exemplars to avoid such situations.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers, 2020. 3

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 5

[3] Francisco M. Castro, Manuel J. Marin-Jimenez, Nicolas Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 5

[4] Yinong Chen and Gennaro De Luca. Technologies supporting artificial intelligence and robotics application development. *Journal of Artificial Intelligence and Technology*, 1(1):1–8, Jan. 2021. 1

[5] 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. 1

[6] R. Elakkiya, V. Subramaniyaswamy, V. Vijayakumar, and Aniket Mahanti. Cervical cancer diagnostics healthcare system using hybrid object detection adversarial networks. *IEEE Journal of Biomedical and Health Informatics*, 26(4):1464–1471, 2022. 1

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 5

[8] Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one, 2019. 17

[9] Akshita Gupta, Sanath Narayan, K J Joseph, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Ow-detr: Open-world detection transformer. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 9235–9244, June 2022. 1, 2, 3, 5, 6, 8, 11, 13, 14, 17

[10] Mohsen Jafarzadeh, Akshay Raj Dhamija, Steve Cruz, Chunchun Li, Touqeer Ahmad, and Terrance E. Boult. Open-world learning without labels. *CoRR*, abs/2011.12906, 2020. 1

[11] K J Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards open world object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5830–5840, June 2021. 1, 2, 3, 5, 6, 8, 12, 13, 14

[12] Dahun Kim, Tsung-Yi Lin, Anelia Angelova, In So Kweon, and Weicheng Kuo. Learning open-world object proposals without learning to classify, 2021. 2, 3, 6, 11

[13] Louis Lecrosnier, Redouane Khemmar, Nicolas Ragot, Benoit Decoux, Romain Rossi, Naceur Kefi, and Jean-Yves Ertaud. Deep learning-based object detection, localisation and tracking for smart wheelchair healthcare mobility. *International Journal of Environmental Research and Public Health*, 18(1), 2021. 1

[14] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 4

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

[16] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21464–21475. Curran Associates, Inc., 2020. 4

[17] Yang Liu, Idil Esen Zulfikar, Jonathon Luiten, Achal Dave, Deva Ramanan, Bastian Leibe, Aljoša Ošep, and Laura Leal-Taixé. Opening up open world tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19045–19055, June 2022. 1

[18] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu. Large-scale long-tailed recognition in an open world. *CoRR*, abs/1904.05160, 2019. 1

[19] Zeyu Ma, Yang Yang, Guoqing Wang, Xing Xu, Heng Tao Shen, and Mingxing Zhang. Rethinking open-world object detection in autonomous driving scenarios. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, page 1279–1288, New York, NY, USA, 2022. Association for Computing Machinery. 1, 2

[20] Muhammad Maaz, Hanoona Rasheed, Salman Khan, Fahad Shahbaz Khan, Rao Muhammad Anwer, and Ming-Hsuan Yang. Class-agnostic object detection with multi-modal transformer. In *17th European Conference on Computer Vision (ECCV)*. Springer, 2022. 2, 5

[21] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *ICRA*, 2018. 14

[22] Can Peng, Kun Zhao, and Brian C Lovell. Faster ilod: Incremental learning for object detectors based on faster rcnn. *PRL*, 2020. 8, 14

[23] Ameya Prabhu, Philip Torr, and Puneet Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *The European Conference on Computer Vision (ECCV)*, August 2020. 13

[24] Lu Qi, Jason Kuen, Yi Wang, Jiuxiang Gu, Hengshuang Zhao, Zhe Lin, Philip H. S. Torr, and Jiaya Jia. Open-world entity segmentation. *CoRR*, abs/2107.14228, 2021. 1

[25] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 5

[26] Kuniaki Saito, Ping Hu, Trevor Darrell, and Kate Saenko. Learning to detect every thing in an open world, 2021. 3, 11

[27] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. Incremental learning of object detectors without catastrophic forgetting. In *ICCV*, 2017. 8, 14

[28] Deepak Kumar Singh, Shyam Nandan Rai, KJ Joseph, Rohit Saluja, Vineeth N Balasubramanian, Chetan Arora, Anbumani Subramanian, and CV Jawahar. Order: Open world object detection on road scenes. In *Proc. NeurIPS Workshops*, 2021. 1, 2

[29] Kuan-Chieh Wang, Paul Vicol, Eleni Triantafillou, and Richard Zemel. Few-shot out-of-distribution detection. *International Conference on Machine Learning (ICML) Workshop on Uncertainty and Robustness in Deep Learning*, 2020. 4

[30] Weiyao Wang, Matt Feiszli, Heng Wang, Jitendra Malik, and Du Tran. Open-world instance segmentation: Exploiting pseudo ground truth from learned pairwise affinity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4422–4432, June 2022. 1

[31] Jiaxi Wu, Jiaxin Chen, and Di Huang. Entropy-based active learning for object detection with progressive diversity constraint. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9397–9406, June 2022. 5

[32] Yan Wu, Xiaowei Zhao, Yuqing Ma, Duorui Wang, and Xianglong Liu. Two-branch objectness-centric open world detection. In *Proceedings of the 3rd International Workshop on Human-Centric Multimedia Analysis*, HCMA '22, page 35–40, New York, NY, USA, 2022. Association for Computing Machinery. 1, 2, 3, 5, 6, 13

[33] Zhiheng Wu, Yue Lu, Xingyu Chen, Zhengxing Wu, Liwen Kang, and Junzhi Yu. Uc-owod: Unknown-classified open world object detection, 2022. 1, 2, 6

[34] Jinan Yu, Liyan Ma, Zhenglin Li, Yan Peng, and Shaorong Xie. Open-world object detection via discriminative class prototype learning. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 626–630. IEEE, 2022. 1, 2, 5, 6, 13

[35] Shengchang Zhang, Zheng Nie, and Jindong Tan. Novel objects detection for robotics grasp planning. In *2020 10th Institute of Electrical and Electronics Engineers International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 43–48, 2020. 1

[36] Hanbin Zhao, Hui Wang, Yongjian Fu, Fei Wu, and Xi Li. Memory-efficient class-incremental learning for image classification. *IEEE Transactions on Neural Networks and Learning Systems*, 33(10):5966–5977, 2022. 5

[37] Xiaowei Zhao, Xianglong Liu, Yifan Shen, Yixuan Qiao, Yuqing Ma, and Duorui Wang. Revisiting open world object detection, 2022. 1, 2, 5, 6

[38] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 3, 5, 6