UC-OWOD: Unknown-Classified Open World Object Detection (Supplementary Materials)

Zhiheng Wu[1,2], Yue Lu[1,2], Xingyu Chen[3], Zhengxing Wu[1,2,⋆], Liwen Kang[1,2], and Junzhi Yu[1,4]

[1] Institute of Automation, Chinese Academy of Sciences
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences
[3] Xiaobing.AI
[4] Peking University

The full quantitative experiment and more visualizations are included to demonstrate the effectiveness of the method.

1 Quantitative Results

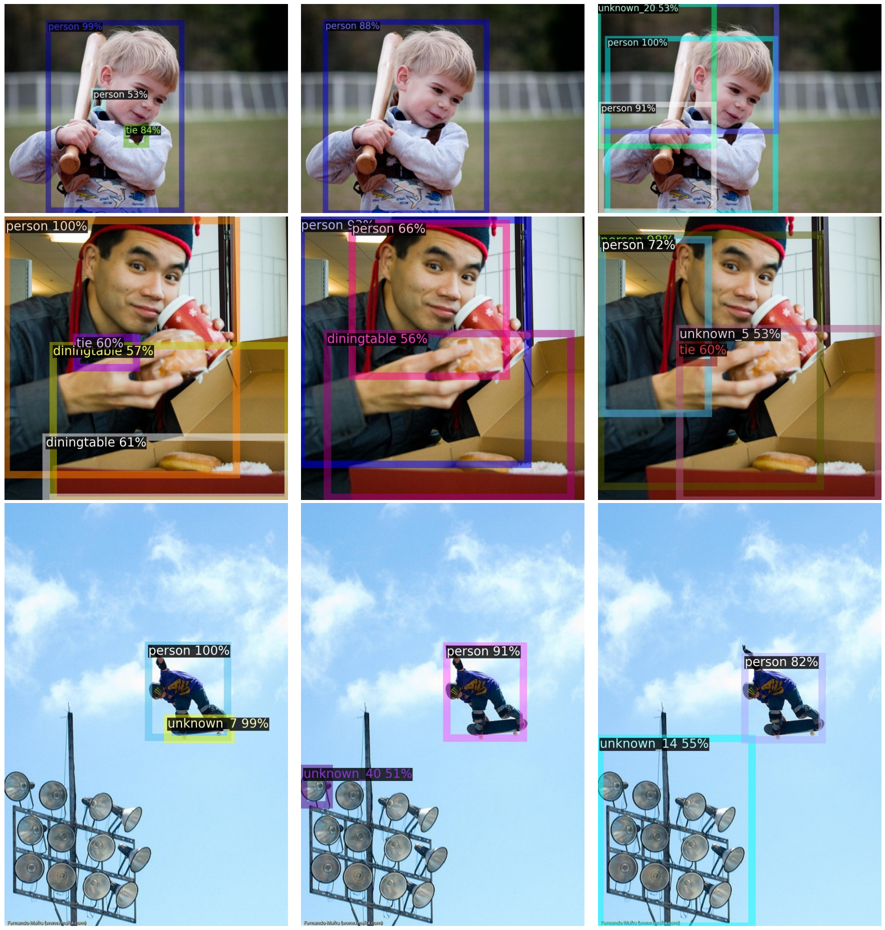
Table 1 shows the full results of the proposed UC-OWOD evaluation protocol. The detection performance of known classes is calculated by mAP. As mentioned above, *Oracle* is a detector trained with annotations of all known and unknown instances. Since the training set only has labels of known classes in task 1, the detection result of *Oracle* on unknown classes are not considered. Without finetuning, the model will completely forget previous classes, which results in significant mAP drop (55.38% vs. 0%). By finetuning, part of the detection ability of the preciously known classes can be restored (40.90% mAP), but WI/A-OSE performance does suffer. The finetuned detector is more inclined to classify an object into known classes. Regarding the scores about unknown classes, task 4 cannot be measured due to the lack of unknown ground-truth. On tasks with the previous known, our method learns better than ORE on the current known. However, due to incomplete annotations of the validation set, detection of unknown objects such as *house* are regarded as false detection. For this reason, Both mAP of our model is lower than the ORE. Therefore, mAP can only measure the detection performance of the model for known classes to a certain extent.

2 Qualitative Results

Since Faster-RCNN cannot detect any unknown objects, we only qualitatively compare *Oracle*, ORE and our model, as shown in Fig. 1. For each test image, columns from left to right are the detection results of *Oracle*, ORE, and our model. Both *Oracle* and ORE failed to detect *baseball bat* and *donut*, etc. This implies that our model is better at detecting unknown objects. In order to better analyze the performance of the model on the UC-OWOD problem, we use some images with multiple unknown instances to test, as shown in Fig. 2. The results

Table 1. The comparison of *Oracle*, ORE, and our model on UC-OWOD. WI, A-OSE, UC-mAP and UC-Recall reflect how the model handles unknown classes, and mAP measures the ability to detect known classes. It can be seen that our model far outperforms other models in handling unknown classes.

Task 1		Oracle	Faster-RCNN	Faster-RCNN +Finetuning	ORE	Ours	Ours+UCR
mAP (↑)	Current known	56.49	55.38	-	56.34	50.66	-
UC-mAP	(↑)	0	0	-	0.0133	**0.1344**	-
WI	(↓)	-	0.0188	-	0.0155	**0.0136**	-
A-OSE	(↓)	-	13300	-	10672	**9294**	-
UC-Recall	(↑)	-	0	-	0.7772	**2.3915**	-
Task 2		**Oracle**	**Faster-RCNN**	**Faster-RCNN +Finetuning**	**ORE**	**Ours**	**Ours+UCR**
mAP (↑)	Previously known	54.83	0	40.90	52.27	33.13	33.13
	Current known	37.92	36.15	31.60	25.49	30.54	30.54
	Both	46.37	18.07	36.25	38.88	31.84	31.84
UC-mAP	(↑)	15.50	0	0	0.0065	0.0862	**0.1694**
WI	(↓)	0.0022	0.0069	0.0140	0.0153	0.0116	0.0117
A-OSE	(↓)	6050	4582	7169	10376	5602	5602
UC-Recall	(↑)	40.45	0	0	0.0371	2.6926	**3.4431**
Task 3		**Oracle**	**Faster-RCNN**	**Faster-RCNN +Finetuning**	**ORE**	**Ours**	**Ours+UCR**
mAP (↑)	Previously known	30.77	0	30.55	38.45	28.80	28.80
	Current known	22.56	19.78	18.16	12.65	16.34	16.34
	Both	28.03	6.59	26.42	29.85	24.65	24.65
UC-mAP	(↑)	10.61	0	0	0.0070	0.0249	**0.0744**
WI	(↓)	0.0042	0.0241	0.0099	0.0086	0.0073	**0.0073**
A-OSE	(↓)	4857	4841	9181	7544	3801	**3801**
UC-Recall	(↑)	28.54	0	0	0.8833	4.8077	**8.7303**
Task 4		**Oracle**	**Faster-RCNN**	**Faster-RCNN +Finetuning**	**ORE**	**Ours**	**Ours+UCR**
mAP (↑)	Previously known	29.18	0	24.74	30.08	25.57	-
	Current known	19.04	17.18	16.51	13.10	15.88	-
	Both	26.64	4.30	22.68	25.83	23.14	-



Oracle **ORE** **Ours**

Fig. 1. Detection results of *Oracle*, ORE and our model. In the first row, *Oracle* and ORE fail to detect the *baseball bat* in the image. In the second row, our model is able to correctly detect the *donut*, while the other models mis-detect it as a *dining table*. In the third row, our model and ORE can detect *broadcast*, but the localization of our model is more accurate.

show that our model can correctly distinguish unknown objects, i.e., classifying *baseball* as *unknown*-34 and *cap* as *unknown*-17. In contrast, *Oracle* and ORE can only detect unknown objects as one class. Fig. 3 shows the detection results of the same-class unknown objects on different images. Our model is able to detect *tennis racket* as *unknown*-37 on different images, which both *Oracle* and ORE fail to do. Fig. 4 shows the qualitative results of incremental learning of our model on different tasks. Our model is able to detect unknown objects and classify them as known classes when their labels are introduced, such as *zebra*.

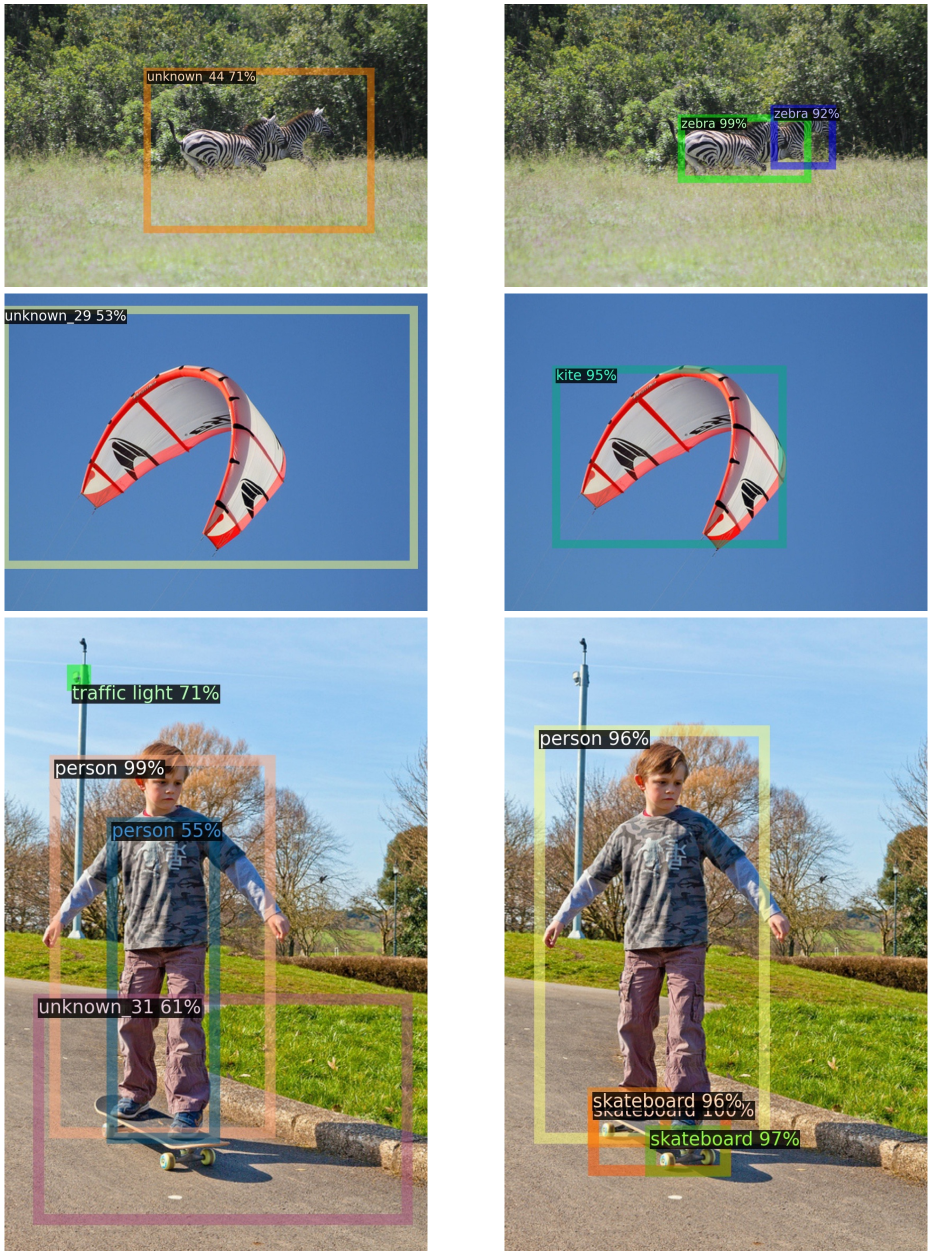
Fig. 2. Detection results of multiple unknown objects. Only our model can correctly distinguish different unknown classes in an image.

Oracle

ORE

Ours

Fig. 3. Detection results of unknown objects of the same class. Only our model can correctly locate unknown objects and classify them into the same unknown class.



Before Task 4 **Task 4**

Fig. 4. The detection results of our model before task 4 are shown on the left. The corresponding predictions after incremental training using task 4 are shown on the right. In the first row, the *unknown*-44 on the left is correctly predicted as *zebra* in task 4. In the second row, the *unknown*-29 is correctly detected as *kite*. In the third row, task 4 correctly detects *unknown*-31 as *skateboard*.