Exploring Orthogonality in Open World Object Detection

Anonymous CVPR submission

Paper ID 8114

Abstract

Open world object detection aims to identify objects of unseen categories and incrementally recognize them once their annotations are provided. In distinction to the traditional paradigm that is limited to predefined categories, this setting promises a continual and generalizable way of estimating objectness using class-agnostic information. However, achieving such decorrelation between objectness and class information proves challenging. Without explicit consideration, existing methods usually exhibit low recall on unknown objects and can misclassify them into known classes. To address this problem, we exploit three levels of orthogonality in the detection process: First, the objectness and classification heads are disentangled by operating on separate sets of features that are orthogonal to each other in a devised polar coordinate system. Secondly, a prediction decorrelation loss is introduced to guide the detector towards more general and class-independent prediction. Furthermore, we propose a calibration scheme that helps maintain orthogonality throughout the training process to mitigate catastrophic interference and facilitate incremental learning of previously unseen objects. Our method is comprehensively evaluated on open world and incremental object detection benchmarks, demonstrating its effectiveness in detecting both known and unknown objects. Code is available at `https://github.com/anonymous-8114/OrthogonalDet`.

1. Introduction

Object detection, a fundamental task in computer vision, has conventionally followed a closed-world paradigm. Despite remarkable advances in this approach [4, 17, 58], it is limited by the assumption that all object classes to be detected are predefined and known during training. This inherently hinders the detector from identifying unseen objects and learning about newfound objects in the evolving world. To address these limitations, Joseph *et al.* [29] recently drew inspiration from open world recognition [2] and proposed a new setup called open world object detection, which tackles object detection in a more adaptable way by autonomously discovering unknown objects and incrementally recognizing them after oracle annotation, thus enabling the detector to operate continuously in an open world.

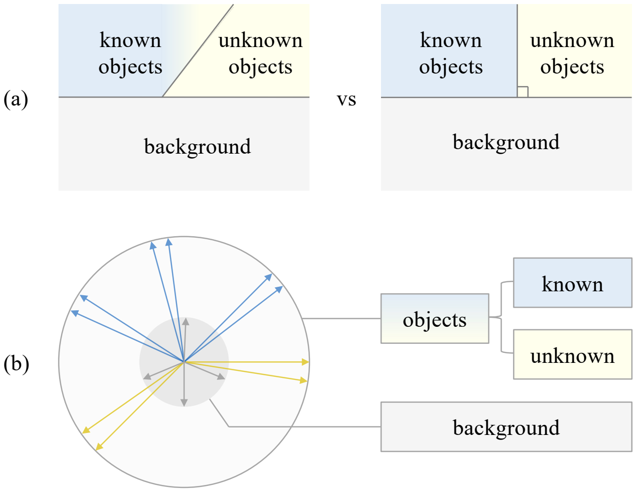


Figure 1. **Motivation for enforcing orthogonality in open world object detection.** (a) In the presence of evolving unknown classes, off-the-shelf detectors can easily neglect or misclassify unknown objects. This is largely due to the model's overreliance on class information to predict objectness, which we resolve by orthogonalizing them. (b) For example, by projecting features onto a polar coordinate system and having the decision boundary for objectness (by radius) to be orthogonal to that for categories (by angle), such interference could be mitigated. See the text for more approaches.

However, embracing this new open-world setting raises several key challenges due to the open-set [8, 52] and incremental [64] nature of the problem. Pivotally, the model should generalize to detect unknown objects and correctly assign them to a special unknown category. It also needs to adaptively incorporate new object knowledge without interfering with known classes. To these ends, a number of new methods [20, 29, 48, 49, 70, 86] have been developed. Unfortunately, their performance in this regard is usually subpar, as evidenced by the low recall for unknown classes and a tendency to confuse unknown objects with known ones.

Our key insight is to explain the deficiencies of previous methods by the dependency between their objectness and class predictions. The underlying rationale is twofold: First, because existing models [20, 49] often detect unknown objects by transferring class-specific information from known classes, some general objectness cues (*e.g*., shape) that are less prominent in previous distributions may be overlooked. This compromises the recall for unknown objects, especially for those that do not share a similar visual appearance. Secondly, the reliance on class-related information for objectness prediction leads to correlated decision boundaries in their feature spaces, as depicted in Fig. 1a. As a result, the unseen objects detected tend to resemble the known objects and are therefore more likely to be confused with them, which also introduces extra interference across incremental learning phases. More evidence for this newly considered correlation is included in subsequent sections.

To mitigate the above interference between objectness and class information, we are inspired by the literature on disentangling via orthogonalization [5, 40, 69] and propose to enforce orthogonality in both the feature space and the prediction space of our detector. Specifically, for the feature space, we adopt a polar coordinate system that decomposes each object feature into two orthogonal components, *i.e*., magnitude and direction, as illustrated in Fig. 1b. The two components are then used separately for objectness and class prediction, where magnitude corresponds to objectness as larger magnitudes indicate more significant objects, while different directions encode information about both known and unknown classes. Here, the unknown class is discriminated by a heuristic strategy that identifies out-of-distribution proposals with low prediction confidence [24]. Moreover, to enhance the orthogonality between objectness and class information, we introduce a decorrelation loss in the prediction space that penalizes the statistical correlation between objectness and class predictions. Together, these two designs promote more class-independent detection results, which is shown to substantially improve model performance for known and unknown objects, enabling better object discovery in open world object detection.

We further notice that the effectiveness of our proposed orthogonalization scheme could be undermined during incremental learning of newly presented objects, as feature orthogonality no longer holds under dramatic shifts in the class distribution. To address this issue, we develop a cross-task calibration layer that aligns the feature spaces of different tasks for joint orthogonalization (the term “calibration” is adopted from [65]). Concretely, the calibration layer is interleaved before the objectness and class prediction heads, and learns a set of task-specific affine transformations on object features to rectify the potential representation shift due to task changes. And during inference, we employ a routing algorithm to estimate the task for each object proposal and perform calibration transformation accordingly. This allows the model to continuously maintain orthogonality and thus produce more general and class-independent prediction under evolving object distributions.

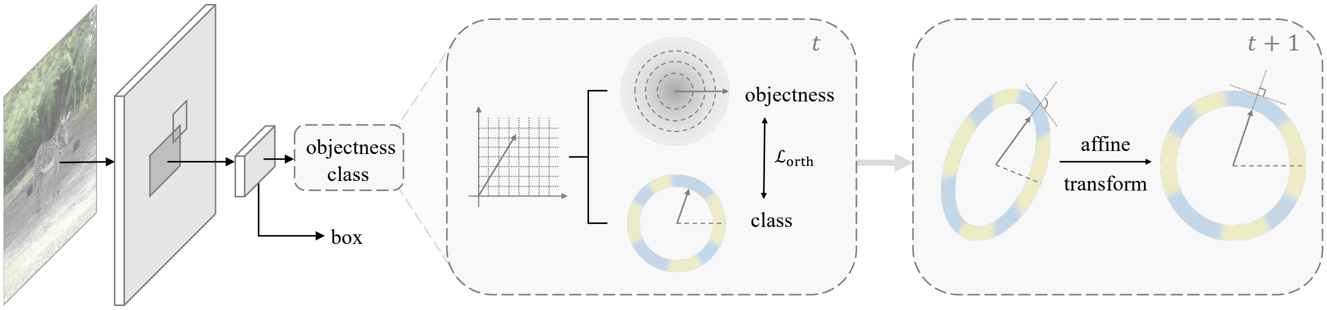
To summarize, the main contributions of this paper are as follows:

- We investigate the previously-neglected correlation between objectness and class information in open world object detection, and address it with a feature orthogonalization scheme and a decorrelation loss that jointly promote class-independent prediction.
- A cross-task calibration layer is devised to maintain feature orthogonality during incremental learning, which also helps to mitigate catastrophic forgetting on previously encountered object classes.
- Comprehensive experiments on two open world benchmarks (M-OWODB and S-OWODB) and an incremental benchmark (PASCAL VOC) clearly illustrate that our method achieves new state-of-the-art performances.

2. Related work

Object detection has witnessed significant progress in the model families of R-CNN [16, 17, 23, 59], YOLO [58], and DETR [4, 85]. To enhance their real-world capabilities, many new research settings have been proposed. Shmelkov *et al*. [64] introduced incremental object detection [9, 14, 30, 31, 41, 42, 56, 68, 79], which studies continual learning [7, 60] of new objects without catastrophic forgetting [51]. More recently, open-set object detection [8, 21, 35, 36, 43, 52, 62] and open-vocabulary object detection [10, 13, 18, 33, 44, 53, 71, 82–84] go beyond predefined categories by considering unknown objects and broader vocabularies, respectively. These research directions underscore the intricacies of open-world scenarios, calling for a more comprehensive approach.

Open world object detection, pioneered by Joseph *et al*. [29], investigates the detection of unknown objects and their incremental learning provided with oracle annotations. Subsequent studies [20, 49] explored contextual information to promote knowledge transfer from known classes to unknown classes. Conversely, another line of work aimed to reduce interference between known and unknown objects. For example, Zohar *et al*. [86] proposed to estimate general objectness with a class-agnostic Gaussian distribution. Wang *et al*. [70] mitigated the confounding effect of limited known object labels by using random region proposals. Most relevant to our work, Ma *et al*. [48] decoupled object localization and classification via a cascaded decoder, but they did not consider the correlation between objectness and class information, which is a critical aspect of the problem. Our method systematically addresses such interference in open world object detection via orthogonalization.



(a) Detection pipeline (b) Orthogonal representations (c) Calibrating orthogonality

Figure 2. **Overview of the proposed method for open world object detection.** Building on a Fast R-CNN [16] based architecture, we employ three levels of orthogonality to mitigate interference between objectness and class predictions. First, the feature representation of each object proposal is decomposed under a polar coordinate system for separate prediction. Secondly, these predictions are decorrelated using a regularization loss. Thirdly, we incorporate affine transformations to continually calibrate orthogonality during the learning process.

Orthogonality has been widely exploited in deep learning. Early literature enforced parameter orthogonality at network initialization [63] or during training [1, 34, 77] to improve training stability and generalization. It has since been extended to feature space and shown to be beneficial for enhancing discriminative representations [32, 57, 66] and disentangling different sets of features [5, 40, 69, 72]. Moreover, when used in prediction space, orthogonalizing a classifier w.r.t. selected attributes improves its fairness [78]. From a more technical perspective, there are two prevalent methods for orthogonalization, the first being hard orthogonalization on the Stiefel manifold [11], a special instance of which is the Gram-Schmidt process. Whereas the second implements a soft regularizer to enforce orthogonality [77]. Inspired by these studies, our work leverages multiple levels of orthogonality to facilitate open world object detection.

3. Method

3.1. Problem formulation

Open world object detection concerns object detection in an evolving world that continuously adds new unknown classes to the training data [29]. At any task t, the model receives labeled data from known classes $\mathcal{K}^t = \{1, 2, \ldots, C\}$ consisting of N images containing multiple object instances with bounding boxes $[x_k, y_k, w_k, h_k]$ and class labels c_k. Importantly, the trained model is required to identify objects from both known and unknown classes $\mathcal{U}^t = \{C+1, \ldots\}$, the latter of which can then be passed to an annotator to produce new training data from extended classes $\mathcal{K}^{t+1} = \mathcal{K}^t \cup \{C+1, \ldots, C+n\}$. By iterating this process, the object detector is able to continually discover and incorporate new object classes, thereby adapting to the open world. However, it poses significant challenges in generalizing detection to unknown objects and incrementally recognizing them without severely interfering with known classes.

We propose to tackle these challenges with the notion of orthogonality. In the following, we start with a base detector (Sec. 3.2) and enforce orthogonality in its feature space (Sec. 3.3) and prediction space (Sec. 3.4). Lastly, the incremental learning techniques are discussed in Sec. 3.5.

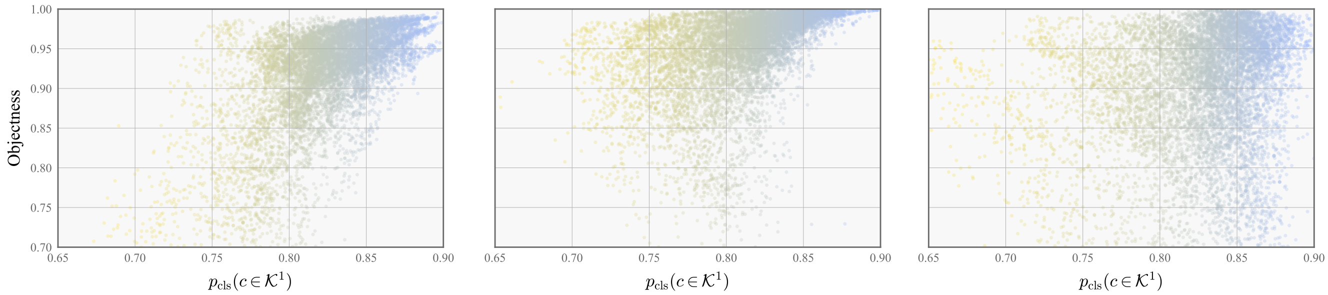
3.2. Base model

We take RandBox [70] as the base model because of its state-of-the-art performance on known classes. It features the removal of the region proposal network to mitigate the confounding effect by limited known objects, resulting in a Fast R-CNN [16] like architecture, as illustrated in Fig. 2a. Specifically, the model first extracts a feature map from the input image and uses RoI pooling to obtain object proposal features $\boldsymbol{f}$ based on randomly sampled bounding boxes. These features are then forwarded to a detection head h to generate objectness, category, and localization predictions. Since our work focuses on the first two parts (objectness and category), we simply represent the detection head h by two separate heads h_{obj} and h_{cls}, as in [86]. The objectness and class probabilities p_{obj} and p_{cls} can be inferred as:

$$p_{\text{obj}} = h_{\text{obj}}(\boldsymbol{f}), \quad p_{\text{cls}} = h_{\text{cls}}(\boldsymbol{f}), \tag{1}$$

where p_{cls} is distributed over $C+1$ classes, including C known classes and a special unknown category. From this, we can further derive the joint probability $p_{\text{joint}} = p_{\text{obj}} \cdot p_{\text{cls}}$ for each proposal and use it for model training.

The training of the base model is outlined below. First, a dynamic matcher [6] is applied to associate object proposals with the ground truth. The remaining proposals with top matching scores are pseudo-labeled as unknown objects. Both sets are then trained with the focal loss [38]. Additionally, when new classes are added, we adopt a replay buffer to retain knowledge of previous objects and mitigate forgetting, following the practice led by Joseph *et al.* [29].



(a) Base model, $r = 0.602$.

(b) Feature orthogonalization, $r = 0.433$.

(c) Our full method, $r = 0.162$.

Figure 3. **Correlation between objectness and class predictions.** (a) The base model shows a positive correlation between objectness and known class probability, resulting in lower recall for unknown classes. (b, c) Meanwhile, incorporating orthogonal designs progressively decorrelates the two predictions and leads to more class-independent detection results. Correlation is measured by the Pearson correlation coefficient (denoted r) [54] based on the top 100 scoring object proposals from each of 100 random images in the first task of M-OWODB.

3.3. Feature orthogonalization

Unfortunately, the base model has limited recall for unknown classes, likely due to its detections being correlated with class information, as evidenced in Fig. 3a. To remedy this problem and make the detection results more general and class-agnostic [27, 50], its objectness and class prediction heads are redesigned to operate on two separate sets of features that are orthogonal to each other.

Orthogonalizing objectness and class features. To efficiently decompose object features, we resort to a manually devised scheme that avoids the intensive computation and optimization errors of prevalent orthogonalization methods. Concretely, we draw inspiration from the literature on polar coordinate based decomposition [5, 40, 69] and project object features into two orthogonal components of magnitude and direction. The magnitude stands alone to signify objectness, while the direction is used for classification, so that the two prediction processes can be decoupled. The inference procedure is depicted in Fig. 2b and can be formulated as:

$$p_{\text{obj}} = h_{\text{obj}}(\|\boldsymbol{f}\|), \quad p_{\text{cls}} = h_{\text{cls}}\left(\frac{\boldsymbol{f}}{\|\boldsymbol{f}\|}\right). \tag{2}$$

Noteworthy, such a decomposition largely preserves feature expressiveness, as angular features have been widely used in many discriminative tasks [39, 47, 57] and learning settings [15, 25, 67]. Meanwhile, feature magnitude has also proven useful for some object-centric tasks [5, 46].

Unknown class discrimination. After encoding class information on a unit hypersphere, it becomes challenging to model unknown classes, as existing energy-based [29] or probabilistic [86] methods are designed for Euclidean feature spaces. We address this issue with a heuristic strategy based solely on model prediction. Intuitively, unknown objects can be viewed as outliers with lower prediction confidence and thus can be detected using a confidence-based criterion, *e.g.*, smaller maximum softmax probability [24]. Let p^o_{cls} denote the original class prediction and $c_{\max}$ be the class with the highest probability, then its complement is assigned to the unknown class probability as follows:

$$p_{\text{cls}}\big(c \in \mathcal{U}^t\big) = p^o_{\text{cls}}\big(c \in \mathcal{K}^t \setminus \{c_{\max}\}\big), \tag{3}$$

and the remaining class probabilities are rescaled to ensure their sum equals to one.

3.4. Prediction orthogonalization

To further disentangle objectness and class predictions and thereby enhance performance on unknown classes, we propose to directly enforce orthogonality in the output space of our object detector. This is accomplished by incorporating a new regularization loss that penalizes the statistical correlation between the two predictions.

Decorrelating objectness and class predictions. While the ideal goal would be to achieve complete independence between the two predictions, directly assessing dependence via mutual information is known to be difficult. Therefore, we opt for a more simplified approach that focuses on the linear correlation between the two variables. Let p_c denote the c-th class probability, then its linear correlation with the objectness p_{obj} can be effectively measured by the squared correlation coefficient [73], yielding the following loss:

$$\mathcal{L}_{\text{orth}} = \sum_c \frac{\big(\operatorname{cov}(p_c, p_{\text{obj}})\big)^2}{\operatorname{var}(p_c)\operatorname{var}(p_{\text{obj}})}, \tag{4}$$

where $\operatorname{var}(\cdot)$ and $\operatorname{cov}(\cdot,\cdot)$ calculate the variance and covariance across all proposals in a mini-batch. And it is directly added to the objective function for training.

The effectiveness of these orthogonal designs is verified in Figs. 3b and 3c. As can be seen, feature orthogonality enhances the objectness of unknown classes, while prediction orthogonality inhibits the bias toward known classes. It will be shown in later experiments that both designs contribute to higher recall and less confusion for unknown objects.

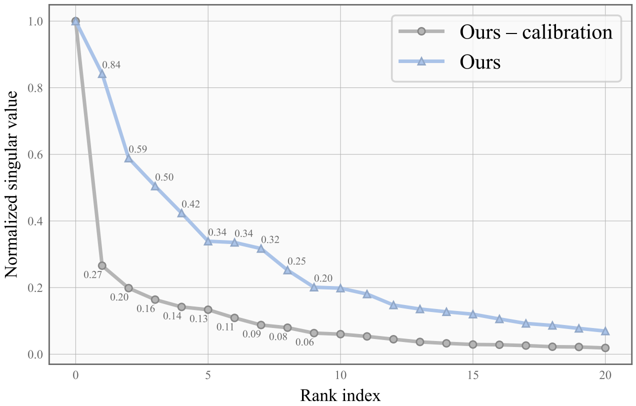


Figure 4. **Singular value spectrum of object features.** Calibrating orthogonality enhances the uniformity of singular values, thus facilitating object detection based on a spherical decision boundary. Object features are extracted from the top 500 scoring detections from 100 random images in the second task of M-OWODB.

3.5. Incremental learning

In addition to detecting open-set objects in a single stage, open world object detection embodies incremental learning of new objects, which poses a key challenge of catastrophic forgetting [51] about known classes due to distribution shift. To address this issue, we propose a calibration scheme (*i.e.*, a lightweight transformation on older representations [65]) that allows the model to remain class-independent and thus less prone to forgetting during learning.

Calibrating orthogonality under task change. A characteristic manifestation of catastrophic interference in our model is impaired orthogonality due to representation shift. As shown intuitively in Fig. 2c, the decision boundary for objectness can shift from a hypersphere to a hyperellipsoid and no longer be orthogonal to the angular class features. This is further supported by the spectrum analysis in Fig. 4, where the non-uniform singular values, akin to semi-axes of a hyperellipsoid, indicate the distorted representations. Building on these observations, we introduce a set of task-specific affine transformations $\boldsymbol{M}_i$ to calibrate the features of previous tasks for joint orthogonalization (with $\boldsymbol{M}_t = \boldsymbol{I}$ for the current task). Let $\boldsymbol{f}^o$ denote the original object proposal features, then the calibration process is as follows:

$$\boldsymbol{f} = \sum_{i=1}^{t} \boldsymbol{w}_i \boldsymbol{M}_i \boldsymbol{f}^o, \tag{5}$$

where $\boldsymbol{w}$ is a one-hot vector specifying the inference path within the calibration layer, while helping to mitigate forgetting through parameter isolation. The routing algorithm that determines $\boldsymbol{w}$ is presented below.

Routing within the calibration layer. In order to select the appropriate calibration transformation for each object proposal, an estimate of the associated task is required. Motivated by the practice in Sec. 3.3, we utilize class prediction confidence to assess task probabilities. Specifically, let p_c denote the c-th class probability before routing, then the i-th task probability $\boldsymbol{\pi}_i$ is set by the maximum softmax probability [24] in its corresponding class range $\mathcal{K}^i \setminus \mathcal{K}^{i-1}$. Thus, the routing vector $\boldsymbol{w}$ that controls task-specific calibration can be sampled from the following distribution:

$$\boldsymbol{w} \sim \mathrm{Cat}(\boldsymbol{\pi}), \quad \boldsymbol{\pi}_i = \frac{\max_{c \in \mathcal{K}^i \setminus \mathcal{K}^{i-1}} p_c}{Z}, \tag{6}$$

where Z is the normalization factor. Since directly choosing a maximum from the above distribution is not differentiable, we employ the Gumbel-Max trick [19] to draw a sample:

$$\boldsymbol{w} = \mathtt{one\_hot}\left(\arg\max_i \left(\log \boldsymbol{\pi} + \boldsymbol{G}\right)_i\right), \tag{7}$$

where $\boldsymbol{G}$ is the Gumbel noise. Then, a Straight-Through estimator [3, 28] is applied to compute the gradient, allowing for end-to-end training of the sampled routing vector $\boldsymbol{w}$ with the rest of the calibration parameters.

Lastly, we'd like to note that this calibration layer imposes a marginal computational and memory overhead (less than 0.1% per task in our model). Combined with the other orthogonal designs, it facilitates knowledge accumulation of evolving new classes in open world object detection.

4. Experiments

4.1. Experimental settings

Datasets. We follow the common setup in [48, 86] that evaluates each method in both open world and incremental object detection. For open world object detection, we consider the superclass-mixed benchmark (M-OWODB) [29] and the superclass-separated benchmark (S-OWODB) [20]. The former consists of COCO [37] and PASCAL VOC [12], while the latter uses only COCO, both grouped into four non-overlapping tasks. For incremental object detection, the class splits of VOC 2007 proposed in [64] are adopted, including three two-stage incremental settings.

Metrics. The evaluation process considers both known and unknown classes. For known classes, we employ mean average precision (mAP) as the only metric, which can be further divided into mAP for previously known classes and mAP for currently known classes, reflecting knowledge retention and adaptation, respectively. For unknown classes, unknown class recall (U-Recall) serves as the main metric, which measures the ability to retrieve unknown objects. In addition, we use wilderness impact (WI) [8] and absolute open-set error (A-OSE) [52] to evaluate the model's confusion of unknown objects with known classes.

Implementation details. We modify on RandBox [70] which adopts a Fast R-CNN [16] based architecture. It uses a ResNet-50 backbone [22] pretrained on ImageNet [61].

Task IDs (→)	**Task 1**		**Task 2**				**Task 3**				**Task 4**		
Method	U-Recall	mAP (↑)	U-Recall	mAP (↑)			U-Recall	mAP (↑)			mAP (↑)		
	(↑)	Current known	(↑)	Previously known	Current known	Both	(↑)	Previously known	Current known	Both	Previously known	Current known	Both
ORE [29]	4.9	56.0	2.9	52.7	26.0	39.4	3.9	38.2	12.7	29.7	29.6	12.4	25.3
OST [80]	-	56.2	-	53.4	26.5	39.9	-	38.0	12.8	29.6	30.1	13.3	25.9
OW-DETR [20]	7.5	59.2	6.2	53.6	33.5	42.9	5.7	38.3	15.8	30.8	31.4	17.1	27.8
OCPL [81]	8.3	56.6	7.7	50.6	27.5	39.1	11.9	38.7	14.7	30.7	30.7	14.4	26.7
2B-OCD [75]	12.1	56.4	9.4	51.6	25.3	38.5	11.6	37.2	13.2	29.2	30.0	13.3	25.8
UC-OWOD [76]	-	50.7	-	33.1	30.5	31.8	-	28.8	16.3	24.6	25.6	15.9	23.2
ALLOW [49]	13.6	59.3	10.0	53.2	34.0	45.6	14.3	42.6	26.7	38.0	33.5	21.8	30.6
PROB [86]	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
CAT [48]	23.7	60.0	19.1	55.5	32.7	44.1	24.4	42.8	18.7	34.8	34.4	16.6	29.9
RandBox [70]	10.6	**61.8**	6.3	-	-	45.3	7.8	-	-	39.4	-	-	35.4
Base model	8.4	59.8	6.4	54.7	36.7	45.7	8.0	45.8	28.6	40.1	40.7	22.2	36.1
Ours	**24.6**	61.3	**26.3**	55.5	**38.5**	**47.0**	**29.1**	**46.7**	**30.6**	**41.3**	**42.4**	**24.3**	**37.9**
ORE [29]	1.5	61.4	3.9	56.5	26.1	40.6	3.6	38.7	23.7	33.7	33.6	26.3	31.8
OW-DETR [20]	5.7	71.5	6.2	62.8	27.5	43.8	6.9	45.2	24.9	38.5	38.2	28.1	33.1
PROB [86]	17.6	73.4	22.3	66.3	36.0	50.4	24.8	47.8	30.4	42.0	42.6	31.7	39.9
CAT [48]	24.0	**74.2**	23.0	**67.6**	35.5	50.7	24.6	51.2	32.6	45.0	45.4	35.1	42.8
Base model	5.9	71.5	6.9	58.9	39.2	48.6	8.0	50.4	40.9	47.3	46.5	37.6	44.3
Ours	**24.6**	71.6	**27.9**	64.0	**39.9**	**51.3**	**31.9**	**52.1**	**42.2**	**48.8**	**48.7**	**38.8**	**46.2**

Table 1. **Open world object detection results on M-OWODB (top) and S-OWODB (bottom).** The comparison is presented in terms of unknown class recall (U-Recall) and mean average precision (mAP) for known objects. For a fair comparison, we compare with ORE [29] without energy based unknown identification and UC-OWOD [76] without unknown clustering refinement. Our method delivers leading performance on both known and unknown classes. Note that U-Recall is not calculated for Task 4 because all 80 classes are observed.

Task IDs (→)	**Task 1**			**Task 2**			**Task 3**		
Method	U-Recall (↑)	WI (↓)	A-OSE (↓)	U-Recall (↑)	WI (↓)	A-OSE (↓)	U-Recall (↑)	WI (↓)	A-OSE (↓)
ORE [29]	4.9	0.0621	10459	2.9	0.0282	10445	3.9	0.0211	7990
OST [80]	-	0.0417	4889	-	0.0213	2546	-	0.0146	2120
OW-DETR [20]	7.5	0.0571	10240	6.2	0.0278	8441	5.7	0.0156	6803
OCPL [81]	8.3	0.0413	5670	7.6	0.0220	5690	11.9	0.0162	5166
2B-OCD [75]	12.1	0.0481	-	9.4	0.0160	-	11.6	0.0137	-
PROB [86]	19.4	0.0569	5195	17.4	0.0344	6452	19.6	0.0151	2641
RandBox [70]	10.6	**0.0240**	4498	6.3	0.0078	1880	7.8	0.0054	1452
Base model	8.4	0.0244	5922	6.4	**0.0073**	2126	8.0	**0.0052**	1696
Ours	**24.6**	0.0299	**4148**	**26.3**	0.0099	**1791**	**29.1**	0.0077	**1345**

Table 2. **Unknown object confusion on M-OWODB.** The comparison is shown in terms of unknown class recall (U-Recall), wilderness impact (WI) and absolute open set error (A-OSE). Our method demonstrates state-of-the-art performance in U-Recall and A-OSE, while maintaining a competitive WI over most baselines. Note that these metrics are not calculated for Task 4 because all 80 classes are known.

The classification head in our model is a linear classifier, while the objectness head uses Batch Normalization [26]. Our model is trained with the AdamW optimizer [45] and a batch size of 12, following [70]. The training epochs for open world and incremental object detection follow [30, 70], taking about 36 and 8 hours on four NVIDIA 2080 Ti GPUs. The weight of the decorrelation loss is set to 1.0. A threshold of 0.15 is used to select the detection results. The code implementation is based on Detectron2 [74].

4.2. Main results

Open world object detection. The comparisons on M-OWODB and S-OWODB are summarized in Tab. 1. Our method shows consistent improvement over the base model, often achieving state-of-the-art results for both unknown class recall (U-Recall) and mean average precision (mAP). Specifically, for U-Recall, our method consistently outperforms the previous leading method CAT [48] by up to 7.3%. In terms of mAP, we surpass the most advanced baselines

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 [64]	69.9	70.4	69.4	54.3	48.0	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 [55]	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 [29]	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	77.0	67.7	64.5
Meta-ILOD [30]	76.0	74.6	67.5	55.9	57.6	75.1	85.4	77.0	43.7	70.8	60.1	66.4	76.0	72.6	74.6	39.7	64.0	60.2	68.5	60.5	66.3
ROSETTA [79]	74.2	76.2	64.9	54.4	57.4	76.1	84.4	68.8	52.4	67.0	62.9	63.3	79.8	72.8	78.1	40.1	62.3	61.2	72.4	66.8	66.8
OW-DETR [20]	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
PROB [86]	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
CAT [48]	76.5	75.7	67.0	51.0	62.4	73.2	82.3	83.7	42.7	64.4	56.8	74.1	75.8	79.2	78.1	39.9	65.1	59.6	78.4	67.4	67.7
Base model	80.8	74.1	77.1	58.0	63.0	79.1	88.8	82.6	48.9	49.8	62.2	77.2	79.2	79.1	81.5	40.9	49.2	67.6	78.9	77.5	69.8
Ours	82.4	77.3	78.2	59.7	61.2	84.3	90.1	80.2	49.8	81.7	58.2	74.0	82.9	81.0	81.2	38.3	70.8	68.0	77.4	70.2	**72.3**
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 [64]	70.5	79.2	68.8	59.1	53.2	75.4	79.4	78.8	46.6	59.4	59.0	75.8	71.8	78.6	69.6	33.7	61.5	63.1	71.7	62.2	65.8
Faster ILOD [55]	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 [29]	75.4	81.0	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
Meta-ILOD [30]	78.4	79.7	66.9	54.8	56.2	77.7	84.6	79.1	47.7	75.0	61.8	74.7	81.6	77.5	80.2	37.8	58.0	54.6	73.0	56.1	67.8
ROSETTA [79]	76.5	77.5	65.1	56.0	60.0	78.3	85.5	78.7	49.5	68.2	67.4	71.2	83.9	75.7	82.0	43.0	60.6	64.1	72.8	67.4	69.2
OW-DETR [20]	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
PROB [86]	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
CAT [48]	75.3	81.0	84.4	64.5	56.6	74.4	84.1	86.6	53.0	70.1	72.4	83.4	85.5	81.6	81.0	32.0	58.6	60.7	81.6	63.5	72.2
Base model	84.1	82.6	78.7	55.7	62.1	76.5	86.9	85.8	50.6	69.8	63.3	85.9	77.6	80.9	82.4	42.6	60.4	68.2	79.2	75.2	72.4
Ours	82.7	80.4	78.5	55.3	65.5	81.0	89.8	85.9	52.6	84.6	62.3	78.4	82.7	81.1	84.2	46.5	71.6	79.0	82.5	79.2	**74.7**
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 [64]	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 [55]	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 [29]	67.3	76.8	60.0	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
Meta-ILOD [30]	78.2	77.5	69.4	55.0	56.0	78.4	84.2	79.2	46.6	79.0	63.2	78.5	82.7	79.1	79.9	44.1	73.2	66.3	76.4	57.6	70.2
ROSETTA [79]	75.3	77.9	65.3	56.2	55.3	79.6	84.6	72.9	49.2	73.7	68.3	71.0	78.9	77.7	80.7	44.0	69.6	68.5	76.1	68.3	69.6
OW-DETR [20]	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
PROB [86]	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
CAT [48]	86.0	85.8	78.8	65.3	61.3	71.4	84.8	84.8	52.9	78.4	71.6	82.7	83.8	81.2	80.7	43.7	75.9	58.5	85.2	61.1	73.8
Base model	85.3	80.8	78.3	56.6	62.7	79.0	87.3	84.0	48.4	75.0	69.2	82.1	82.9	76.8	82.1	44.6	69.1	69.7	81.5	72.8	73.4
Ours	83.8	84.7	77.0	62.9	60.8	80.9	88.6	85.8	51.1	81.4	67.2	86.7	86.3	83.4	83.4	44.7	74.5	73.1	81.1	74.9	**75.6**

Table 3. **Incremental object detection results on PASCAL VOC.** The comparison is shown in terms of per-class mAP and overall mAP. Three incremental settings are considered, where the model is first trained on 10, 15, or 19 classes, and then incrementally updated on the remaining 10, 5, or 1 class(es). Our method outperforms existing baselines on both the current classes and the previously seen classes.

by 1.4–2.5% and 0.6–3.8% on the last three tasks of both benchmarks. The only metric where our performance lags is mAP on the first task, but this can be effectively addressed through incremental learning. It is worth mentioning that although the baselines PROB [86] and CAT also incorporate decoupling of class information, our orthogonality approach clearly outperforms them, even with a simpler base architecture and thus a lower computational cost.

We further examine unknown class confusion in Tab. 2. As can be seen, our method yields state-of-the-art absolute open-set error (A-OSE) and U-Recall while maintaining a very competitive wilderness impact (WI). In fact, our WI is lower than most baselines except RandBox [70], indicating effective discrimination of unknown classes.

Incremental object detection. As first discovered by Joseph *et al*. [30] and then verified in [20, 48, 86], explicit modeling of unknown objects can enhance the performance for incremental object detection. Therefore, we adapt our method to this setup and summarize the results in Tab. 3. While the base model built on RandBox is already very performant, our full method further improves the overall mAP, surpassing existing baselines by a remarkable 1.8–4.6%. In particular, the average mAP for previous classes is elevated by 1.4–4.3% over the base model, underscoring the efficacy of our designs in mitigating catastrophic forgetting.

4.3. Visualization

For a more intuitive illustration of our open-world performance, Fig. 5 presents a comparison with RandBox after the second task of M-OWODB. Our method manages to identify different types of objects, including some unknown objects in rare textures where RandBox falls short, such as the kite in the first image and the bed in the fourth image. Also, the confusion between known and unknown objects is mitigated, as showcased by our method accurately detecting the surfboard (which is an unseen category) in the

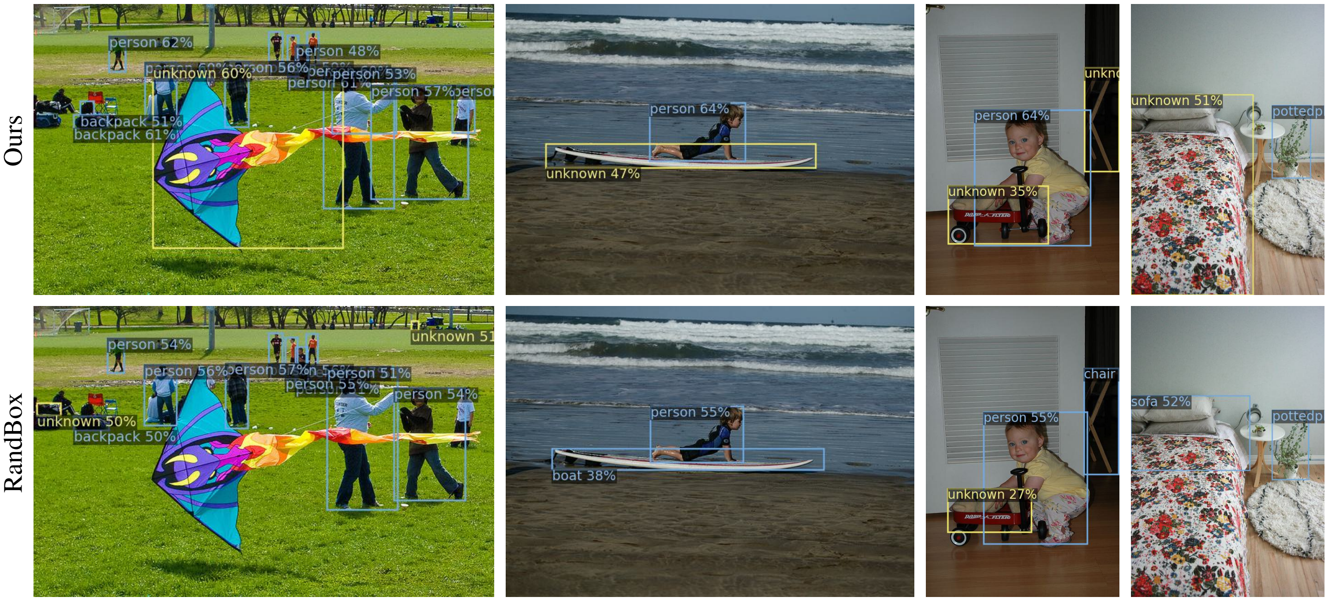


Figure 5. **Qualitative results on M-OWODB.** We compare with a leading open world object detector RandBox [70] in terms of known and unknown object detections after the second stage of M-OWODB. Each image pair uses the same score threshold for a fair comparison.

Task 1	U-Recall (↑)	K-mAP (↑)	WI (↓)	A-OSE (↓)
Base model	8.4	59.8	**0.0244**	5922
Feature	18.2	61.3	0.0276	4455
Prediction	15.2	59.9	0.0257	4645
Ours	**24.6**	**61.3**	0.0299	**4148**

Task 2	U-Recall	mAP (↑)		
	(↑)	Previously known	Current known	Both
Base model	6.4	54.7	36.7	45.7
– calibration	17.2	54.6	37.3	45.9
Ours	**26.3**	**55.5**	**38.5**	**47.0**

Table 4. **Ablation studies of orthogonal designs on M-OWODB.** Feature, prediction, and calibration denote the designs in Secs. 3.3, 3.4 and 3.5, respectively. K-mAP denotes mAP on known classes. It is clearly shown that all our designs contribute to the final performance.

second image without misclassifying it into known classes. Lastly, thanks to the better object representations learned by our approach, the localization of unknown objects significantly outperforms RandBox, as evidenced by the toy car in the third image. These examples clearly demonstrate the superiority of our method in detecting unknown objects and reducing confusion between unknown and known classes.

4.4. Ablation studies

Table 4 validates the effectiveness of our main orthogonal designs. For hyperparameter sensitivity and more detailed ablation experiments, please see the Appendix.

In the first task of M-OWODB, feature orthogonalization improves U-recall by 9.8% and K-mAP by 1.5%, while prediction orthogonality increases U-recall by an additional 6.4%, confirming their efficacy in detecting known and unknown objects. For unknown class confusion, both designs clearly reduce A-OSE, albeit with a slight increase in WI. Nevertheless, their confusions in terms of WI remain lower than most baseline methods, as shown earlier in Tab. 2.

The effectiveness of calibrating orthogonality is exemplified in the second task of M-OWODB. While our method without calibration yields a high U-Recall, its performance is saturated in terms of mAP, especially for the previously known classes, indicating reduced efficacy during incremental learning. In contrast, our calibration scheme delivers notable improvements in U-Recall (by 9.1%) and mAP (by 0.9–1.2%), effectively mitigating catastrophic inference.

5. Conclusion

This paper describes a simple and effective framework that exploits class-independent information for open world object detection. Specifically, we enforce orthogonality at multiple levels during object detection, including a feature orthogonalization scheme to disentangle different heads and a decorrelation loss that operates in the prediction space. In addition, a cross-task calibration layer is developed to maintain feature orthogonality when learning new classes. Finally, our method is extensively validated on both open world and incremental object detection benchmarks.

References

[1] El Mehdi Achour, François Malgouyres, and Franck Mamalet. Existence, stability and scalability of orthogonal convolutional neural networks. *JMLR*, 23(1):15743–15798, 2022. 3

[2] Abhijit Bendale and Terrance Boult. Towards open world recognition. In *CVPR*, pages 1893–1902, 2015. 1

[3] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 5

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020. 1, 2

[5] Di Chen, Shanshan Zhang, Jian Yang, and Bernt Schiele. Norm-aware embedding for efficient person search. In *CVPR*, pages 12615–12624, 2020. 2, 3, 4

[6] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. DiffusionDet: Diffusion model for object detection. In *ICCV*, pages 19830–19843, 2023. 3

[7] Zhiyuan Chen and Bing Liu. *Lifelong Machine Learning*. Morgan & Claypool Publishers, 2018. 2

[8] Akshay Dhamija, Manuel Gunther, Jonathan Ventura, and Terrance Boult. The overlooked elephant of object detection: Open set. In *WACV*, pages 1021–1030, 2020. 1, 2, 5

[9] Na Dong, Yongqiang Zhang, Mingli Ding, and Gim Hee Lee. Bridging non co-occurrence with unlabeled in-the-wild data for incremental object detection. In *NeurIPS*, pages 30492–30503, 2021. 2

[10] Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. Learning to prompt for open-vocabulary object detection with vision-language model. In *CVPR*, pages 14084–14093, 2022. 2

[11] Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1998. 3

[12] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 88:303–338, 2010. 5, 1

[13] Chengjian Feng, Yujie Zhong, Zequn Jie, Xiangxiang Chu, Haibing Ren, Xiaolin Wei, Weidi Xie, and Lin Ma. PromptDet: Towards open-vocabulary detection using uncurated images. In *ECCV*, pages 701–717, 2022. 2

[14] Tao Feng, Mang Wang, and Hangjie Yuan. Overcoming catastrophic forgetting in incremental object detection via elastic response distillation. In *CVPR*, pages 9427–9436, 2022. 2

[15] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, pages 4367–4375, 2018. 4

[16] Ross Girshick. Fast R-CNN. In *ICCV*, pages 1440–1448, 2015. 2, 3, 5, 1

[17] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014. 1, 2

[18] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *ICLR*, 2022. 2

[19] Emil Julius Gumbel. *Statistical Theory of Extreme Values and Some Practical Applications: A Series of Lectures*. US Government Printing Office, 1954. 5

[20] Akshita Gupta, Sanath Narayan, KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. OW-DETR: Open-world detection transformer. In *CVPR*, pages 9235–9244, 2022. 1, 2, 5, 6, 7

[21] Jiaming Han, Yuqiang Ren, Jian Ding, Xingjia Pan, Ke Yan, and Gui-Song Xia. Expanding low-density latent regions for open-set object detection. In *CVPR*, pages 9591–9600, 2022. 2

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5, 1

[23] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. *TPAMI*, 42(2):386–397, 2020. 2

[24] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*, 2017. 2, 4, 5

[25] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *CVPR*, pages 831–839, 2019. 4

[26] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015. 6, 1

[27] Ayush Jaiswal, Yue Wu, Pradeep Natarajan, and Premkumar Natarajan. Class-agnostic object detection. In *WACV*, pages 919–928, 2021. 4

[28] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-softmax. In *ICLR*, 2017. 5

[29] KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards open world object detection. In *CVPR*, pages 5830–5840, 2021. 1, 2, 3, 4, 5, 6, 7

[30] KJ Joseph, Jathushan Rajasegaran, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Incremental object detection via meta-learning. *TPAMI*, 44(12): 9209–9216, 2021. 2, 6, 7, 1

[31] Mengxue Kang, Jinpeng Zhang, Jinming Zhang, Xiashuang Wang, Yang Chen, Zhe Ma, and Xuhui Huang. Alleviating catastrophic forgetting of incremental object detection via within-class and between-class knowledge distillation. In *ICCV*, pages 18894–18904, 2023. 2

[32] José Lezama, Qiang Qiu, Pablo Musé, and Guillermo Sapiro. OLÉ: Orthogonal low-rank embedding, a plug and play geometric loss for deep learning. In *CVPR*, pages 8109–8118, 2018. 3

[33] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *CVPR*, pages 10965–10975, 2022. 2

[34] Shuai Li, Kui Jia, Yuxin Wen, Tongliang Liu, and Dacheng Tao. Orthogonal deep neural networks. *TPAMI*, 43(4):1352–1368, 2019. 3

[35] Wuyang Li, Xiaoqing Guo, and Yixuan Yuan. Novel scenes & classes: Towards adaptive open-set object detection. In *ICCV*, pages 15780–15790, 2023. 2

[36] Wenteng Liang, Feng Xue, Yihao Liu, Guofeng Zhong, and Anlong Ming. Unknown sniffer for object detection: Don't turn a blind eye to unknown objects. In *CVPR*, pages 3230–3239, 2023. 2

[37] 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*, pages 740–755, 2014. 5, 1

[38] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017. 3, 1

[39] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. SphereFace: Deep hypersphere embedding for face recognition. In *CVPR*, pages 212–220, 2017. 4

[40] Weiyang Liu, Zhen Liu, Zhiding Yu, Bo Dai, Rongmei Lin, Yisen Wang, James M Rehg, and Le Song. Decoupled networks. In *CVPR*, pages 2771–2779, 2018. 2, 3, 4

[41] Yuyang Liu, Yang Cong, Dipam Goswami, Xialei Liu, and Joost van de Weijer. Augmented box replay: Overcoming foreground shift for incremental object detection. In *ICCV*, pages 11367–11377, 2023. 2

[42] Yaoyao Liu, Bernt Schiele, Andrea Vedaldi, and Christian Rupprecht. Continual detection transformer for incremental object detection. In *CVPR*, pages 23799–23808, 2023. 2

[43] Yen-Cheng Liu, Chih-Yao Ma, Xiaoliang Dai, Junjiao Tian, Peter Vajda, Zijian He, and Zsolt Kira. Open-set semi-supervised object detection. In *ECCV*, pages 143–159, 2022. 2

[44] Yanxin Long, Youpeng Wen, Jianhua Han, Hang Xu, Pengzhen Ren, Wei Zhang, Shen Zhao, and Xiaodan Liang. CapDet: Unifying dense captioning and open-world detection pretraining. In *CVPR*, pages 15233–15243, 2023. 2

[45] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6, 1

[46] Sindy Löwe, Phillip Lippe, Francesco Locatello, and Max Welling. Rotating features for object discovery. In *NeurIPS*, 2023. 4

[47] Chunjie Luo, Jianfeng Zhan, Xiaohe Xue, Lei Wang, Rui Ren, and Qiang Yang. Cosine normalization: Using cosine similarity instead of dot product in neural networks. In *ICANN*, pages 382–391, 2018. 4

[48] Shuailei Ma, Yuefeng Wang, Ying Wei, Jiaqi Fan, Thomas H Li, Hongli Liu, and Fanbing Lv. CAT: Localization and identification cascade detection transformer for open-world object detection. In *CVPR*, pages 19681–19690, 2023. 1, 2, 5, 6, 7

[49] Yuqing Ma, Hainan Li, Zhange Zhang, Jinyang Guo, Shanghang Zhang, Ruihao Gong, and Xianglong Liu. Annealing-based label-transfer learning for open world object detection. In *CVPR*, pages 11454–11463, 2023. 1, 2, 6

[50] 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 *ECCV*, pages 512–531, 2022. 4

[51] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation*, 24:109–165, 1989. 2, 5

[52] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *ICRA*, pages 3243–3249, 2018. 1, 2, 5

[53] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection with vision transformers. In *ECCV*, pages 728–755, 2022. 2

[54] Karl Pearson. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58:240–242, 1895. 4

[55] Can Peng, Kun Zhao, and Brian C Lovell. Faster ILOD: Incremental learning for object detectors based on faster RCNN. *Pattern Recognition Letters*, 140:109–115, 2020. 7

[56] Juan-Manuel Perez-Rua, Xiatian Zhu, Timothy M Hospedales, and Tao Xiang. Incremental few-shot object detection. In *CVPR*, pages 13846–13855, 2020. 2

[57] Kanchana Ranasinghe, Muzammal Naseer, Munawar Hayat, Salman Khan, and Fahad Shahbaz Khan. Orthogonal projection loss. In *ICCV*, pages 12333–12343, 2021. 3, 4

[58] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788, 2016. 1, 2

[59] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *TPAMI*, 39(6):1137–1149, 2016. 2

[60] Mark B Ring. Continual learning in reinforcement environments. *PhD thesis, University of Texas at Austin*, 1994. 2

[61] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Fei-Fei Li. ImageNet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 5, 1

[62] Kuniaki Saito, Ping Hu, Trevor Darrell, and Kate Saenko. Learning to detect every thing in an open world. In *ECCV*, pages 268–284, 2022. 2

[63] Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013. 3

[64] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. Incremental learning of object detectors without catastrophic forgetting. In *ICCV*, pages 3400–3409, 2017. 1, 2, 5, 7

[65] Pravendra Singh, Vinay Kumar Verma, Pratik Mazumder, Lawrence Carin, and Piyush Rai. Calibrating CNNs for lifelong learning. In *NeurIPS*, pages 15579–15590, 2020. 2, 5

[66] Yifan Sun, Liang Zheng, Weijian Deng, and Shengjin Wang. SVDnet for pedestrian retrieval. In *ICCV*, pages 3800–3808, 2017. 3

[67] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, pages 3637–3645, 2016. 4

[68] Jianren Wang, Xin Wang, Yue Shang-Guan, and Abhinav Gupta. Wanderlust: Online continual object detection in the real world. In *ICCV*, pages 10829–10838, 2021. 2

[69] Yitong Wang, Dihong Gong, Zheng Zhou, Xing Ji, Hao Wang, Zhifeng Li, Wei Liu, and Tong Zhang. Orthogonal deep features decomposition for age-invariant face recognition. In *ECCV*, pages 738–753, 2018. 2, 3, 4

[70] Yanghao Wang, Zhongqi Yue, Xian-Sheng Hua, and Hanwang Zhang. Random boxes are open-world object detectors. In *ICCV*, pages 6233–6243, 2023. 1, 2, 3, 5, 6, 7, 8

[71] Zhenyu Wang, Yali Li, Xi Chen, Ser-Nam Lim, Antonio Torralba, Hengshuang Zhao, and Shengjin Wang. Detecting everything in the open world: Towards universal object detection. In *CVPR*, pages 11433–11443, 2023. 2

[72] Yuxiang Wei, Yupeng Shi, Xiao Liu, Zhilong Ji, Yuan Gao, Zhongqin Wu, and Wangmeng Zuo. Orthogonal Jacobian regularization for unsupervised disentanglement in image generation. In *ICCV*, pages 6721–6730, 2021. 3

[73] Sewall Wright. Correlation and causation. *Journal of Agricultural Research*, 20:557–585, 1921. 4

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

[75] 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*, pages 35–40, 2022. 6

[76] Zhiheng Wu, Yue Lu, Xingyu Chen, Zhengxing Wu, Liwen Kang, and Junzhi Yu. UC-OWOD: Unknown-classified open world object detection. In *ECCV*, pages 193–210, 2022. 6

[77] Di Xie, Jiang Xiong, and Shiliang Pu. All you need is beyond a good init: Exploring better solution for training extremely deep convolutional neural networks with orthonormality and modulation. In *CVPR*, pages 6176–6185, 2017. 3

[78] Yilun Xu, Hao He, Tianxiao Shen, and Tommi S Jaakkola. Controlling directions orthogonal to a classifier. In *ICLR*, 2022. 3

[79] Binbin Yang, Xinchi Deng, Han Shi, Changlin Li, Gengwei Zhang, Hang Xu, Shen Zhao, Liang Lin, and Xiaodan Liang. Continual object detection via prototypical task correlation guided gating mechanism. In *CVPR*, pages 9255–9264, 2022. 2, 7

[80] Shuo Yang, Peize Sun, Yi Jiang, Xiaobo Xia, Ruiheng Zhang, Zehuan Yuan, Changhu Wang, Ping Luo, and Min Xu. Objects in semantic topology. In *ICLR*, 2022. 6

[81] Jinan Yu, Liyan Ma, Zhenglin Li, Yan Peng, and Shaorong Xie. Open-world object detection via discriminative class prototype learning. In *ICIP*, pages 626–630, 2022. 6

[82] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *CVPR*, pages 14393–14402, 2021. 2

[83] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. RegionCLIP: Region-based language-image pretraining. In *CVPR*, pages 16793–16803, 2022.

[84] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, pages 350–368, 2022. 2

[85] 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. 2

[86] Orr Zohar, Kuan-Chieh Wang, and Serena Yeung. PROB: Probabilistic objectness for open world object detection. In *CVPR*, pages 11444–11453, 2023. 1, 2, 3, 4, 5, 6, 7

Exploring Orthogonality in Open World Object Detection

Supplementary Material

A. Experimental settings

A.1. Datasets

For open world object detection, we follow [48, 86] to adopt the superclass-mixed benchmark (M-OWODB) [29] and the superclass-separated benchmark (S-OWODB) [20], both consisting of 80 classes grouped into four sequential tasks, as summarized in Tab. 5. Specifically, M-OWODB is built on COCO [37] and PASCAL VOC [12], and it uses all VOC classes and data as the first task, and the remaining COCO classes as the three successive tasks. However, this can lead to data leakage across super-categories, *e.g.*, most vehicle-related classes belong to the first task, but the truck class is introduced in the second task. To address this, S-OWODB uses a stricter split of the COCO dataset that ensures a clear separation of super-categories across tasks, allowing for a fairer open-world evaluation.

For incremental object detection, which prioritizes the incremental learning capability, we adopt the class splits of PASCAL VOC 2007 proposed in [64]. It contains three two-stage incremental settings of 10 + 10, 15 + 5, and 19 + 1 classes. See the header of Tab. 3 for detailed class splits.

Task IDs (→)	**Task 1**	**Task 2**	**Task 3**	**Task 4**
M-OWODB split	VOC [12] Classes	Outdoor, Accessories, Appliances, Truck	Sports, Food	Electronic, Indoor, Kitchen, Furniture
# classes	20	20	20	20
# training images	16551	45520	39402	40260
# test images	4952	1914	1642	1738
# training instances	47223	113741	114452	138996
# test instances	14976	4966	4826	6039
S-OWODB split	Animals, Person, Vehicles	Outdoor, Accessories, Appliances, Furniture	Sports, Food	Electronic, Indoor, Kitchen
# classes	19	21	20	20
# training images	89490	55870	39402	38903
# test images	3793	2351	1642	1691
# training instances	421243	163512	114452	160794
# test instances	17786	7159	4826	7010

Table 5. **Task composition in M-OWODB (top) and S-OWODB (bottom).** The semantics of each task split and the number of associated training and test images and object instances are displayed.

A.2. Metrics

We use the common evaluation metrics in [20, 70, 86]. Among them, mean average precision (mAP) and unknown class recall (U-Recall) at IoU threshold of 0.5 serve as the two main metrics. In addition to these, wilderness impact (WI) [8] at IoU threshold of 0.8 and absolute open-set error (A-OSE) [52] at IoU threshold of 0.5 are employed to measure unknown class confusion. Specifically, WI shows the change in precision due to unknown misclassifications:

$$\mathrm{WI} = \frac{\mathrm{P}_{\mathcal{K}}}{\mathrm{P}_{\mathcal{K}\cup\mathcal{U}}} - 1, \tag{8}$$

where $\mathrm{P}_{\mathcal{K}}$ is the precision on known classes, and $\mathrm{P}_{\mathcal{K}\cup\mathcal{U}}$ is the precision when unknown classes are included. On the other hand, A-OSE measures the number of unknown object instances that are misclassified into known classes.

A.3. Implementation details

This section describes some key implementation details, organized by model architecture, training, and test protocols. Most of these are consistent with RandBox [70].

Model architecture. We adopt a Fast R-CNN [16] like architecture, which uses a ResNet-50 [22] pretrained on ImageNet [61] to extract a feature map for each input image, and then applies RoI pooling with 500 random proposals. The resulting proposal features are forwarded to a cascade of detection heads that iteratively refine the detection results. Each head contains a self-attention module, followed by a regression head, an objectness head, and a classification head. The classification head is a linear classifier, while the objectness head uses Batch Normalization [26].

Training scheme. The training process is supervised by ground truth annotations and pseudo-labels estimated from the matching scores. We incorporate standard training objectives including focal loss [38] and regression loss, along with a decorrelation loss with a weight of 1.0. The model is optimized by the AdamW optimizer [45] with a batch size of 12 and an initial learning rate of 2.5×10^{-5}. The training iterations and learning rate schedules in open world and incremental object detection follow [30, 70]. It is worth noting that our Fast R-CNN based model is efficient to train, taking about 36 hours to finish training on M-OWODB with four NVIDIA 2080 Ti GPUs. In comparison, a DETR [4] based model such as CAT [48] takes almost 48 hours on the same benchmark with eight NVIDIA 3090 GPUs.

Test protocol. During inference, we remove the prediction randomness using 10000 pre-defined object proposals covering various locations, shapes, and scales. These proposals are then pruned by non-maximum suppression at IoU threshold of 0.6. The final detection results are selected by a score threshold default to 0.15, following the code of [70].

Task 1	U-Recall (↑)	K-mAP (↑)	WI (↓)	A-OSE (↓)
Base model	8.4	59.8	**0.0244**	5922
Polar	13.2	61.1	0.0254	5026
Unknown	14.6	60.2	0.0265	4862
Feature	**18.2**	**61.3**	0.0276	**4455**

Task 2	U-Recall	mAP (↑)		
	(↑)	Previously known	Current known	Both
– calibration	17.2	54.6	37.3	45.9
Ours (sum)	23.5	55.3	**38.7**	47.0
Ours (max)	**26.3**	**55.5**	38.5	**47.0**

Table 6. **Ablation studies of detailed designs on M-OWODB.** Polar and unknown denote the two subdesigns in Sec. 3.3, namely polar coordinate based feature decomposition and unknown class discrimination. Whereas ours (sum) and ours (max) denote our full method with two different routing strategies related to Sec. 3.5.

B. Additional results

B.1. Ablation studies

This section presents additional ablation experiments on detailed designs (*e.g.*, the routing algorithm) and hyperparameter sensitivity, as a complement to Sec. 4.4.

Effectiveness of feature orthogonalization designs. To justify the two subdesigns in feature orthogonalization, we perform a set of ablation studies in Tab. 6. For polar coordinate based feature decomposition, it leads to clear improvements across three metrics (U-Recall, K-mAP, and A-OSE) with only a slight increase in WI. For the confidence-based unknown discrimination strategy, it significantly improves both U-Recall and A-OSE over the standard softmax-based approach (alternative methods like Gaussian modeling [86] are not compared because they do not directly apply to our spherical class feature space). In the meantime, it maintains competitive K-mAP and WI to existing baselines.

Effectiveness of the routing algorithm. To demonstrate the effectiveness of the maximum softmax probability [24] for routing calibration parameters, we compare it to a more straightforward alternative that sums all corresponding class probabilities as the task probability. Table 6 illustrates that while both methods significantly improve mAP, the maximum probability approach excels in U-Recall by incorporating prediction confidence. This can be understood intuitively with the following example: an object proposal with uniformly high probabilities, *i.e.*, high uncertainties on old tasks, should be considered new and unknown.

Hyperparameter sensitivity. Since most of our hyperparameters (*e.g.*, the score threshold for detection results) follow RandBox, we focus on the one newly introduced hyperparameter, the weight of the decorrelation loss $\mathcal{L}_{\text{orth}}$. Its sensitivity analysis is shown in Fig. 6. As can be seen, the inclusion of this new loss significantly improves U-Recall without affecting K-mAP, confirming its effectiveness. On the other hand, increasing the loss weight leads to a reduction in U-Recall, as the model may deviate from its main training objective. Nevertheless, the resulting U-Recall is still higher than the base one without the decorrelation loss.

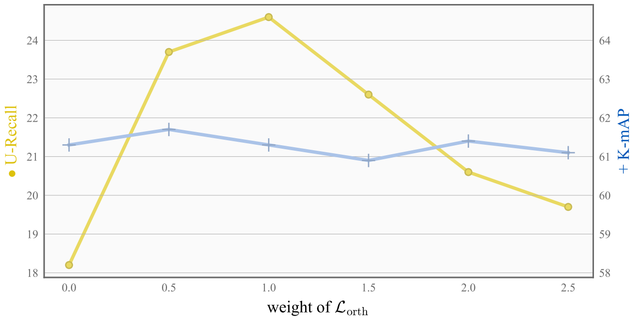


Figure 6. **Sensitivity to loss weight on M-OWODB.** We vary the weight of loss $\mathcal{L}_{\text{orth}}$ and report performance changes on Task 1.

B.2. Visualization

Figure 7 compares our method with RandBox [70] on all four tasks of M-OWODB. As can be observed, our model successfully detects various unknown objects, including the fire hydrant in the first image and the teddy bear in the third image. Meanwhile, its performance remains advantageous for known objects such as the dining table in the fourth image. Note that there are also some common failures between the two models (the skateboard in the second image and the knife in the fourth image) that could be improved.

Additionally, our incremental learning capability is illustrated in Fig. 8, where we test two checkpoints before and after incremental learning on the same image. The left two columns show that our model can continuously incorporate new class information (the backpack) to facilitate unknown object discovery (the surfboard). The last two columns suggest that, compared to RandBox, our model forgets less of the old class (dining table) and even exhibits some degree of knowledge consolidation as its object score increases.

C. Limitations

We would like to further discuss the limitations of our work: (1) This work is limited to existing open world object detection datasets [20, 29] to allow for more controllable experiments, while there is a growing need for research on scalability with larger models trained on more data. (2) The problem formulation assumes that the current task identifier is specified during training, which is not compatible with setups with blurry task boundaries or unknown task identifiers. (3) Our proposed method is scoped to the traditional supervised training scheme and needs adaptation for recent pre-training methods such as GLIP [33] and Detic [84].

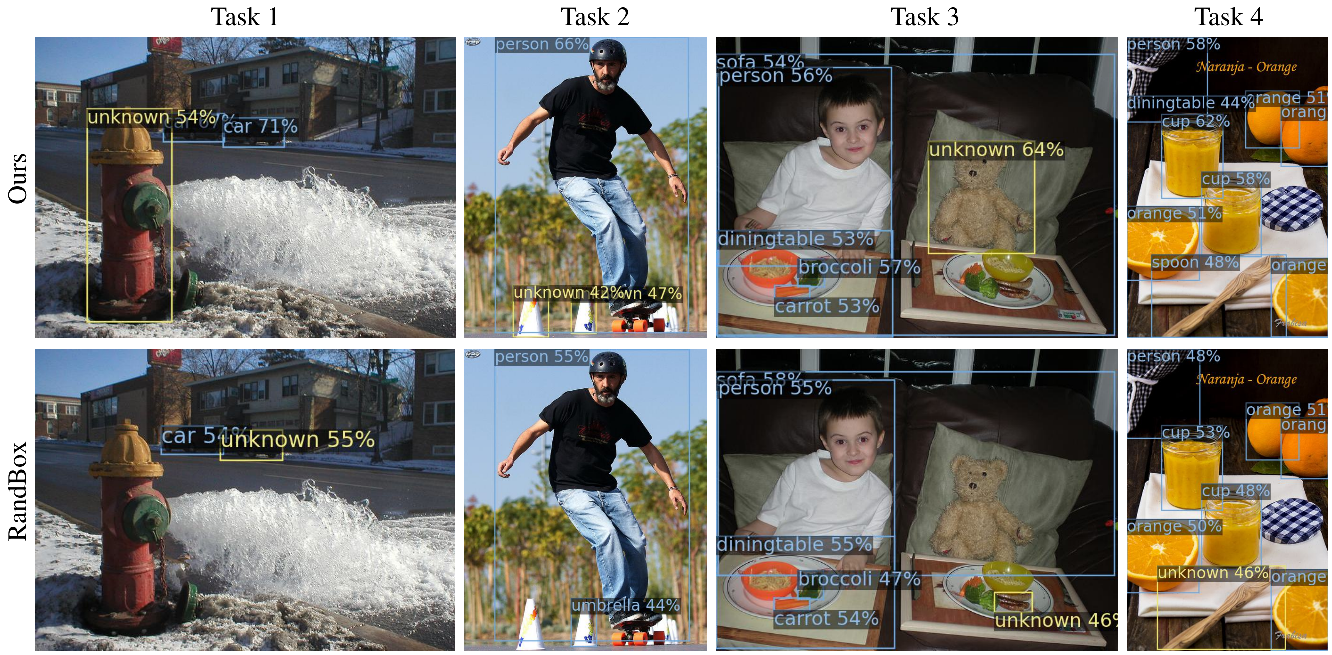


Figure 7. **Qualitative results after different tasks of M-OWODB.** Our method is compared with RandBox [70] in terms of known and unknown object detections after each stage of M-OWODB. Each image pair uses the same score threshold to ensure a fair comparison.

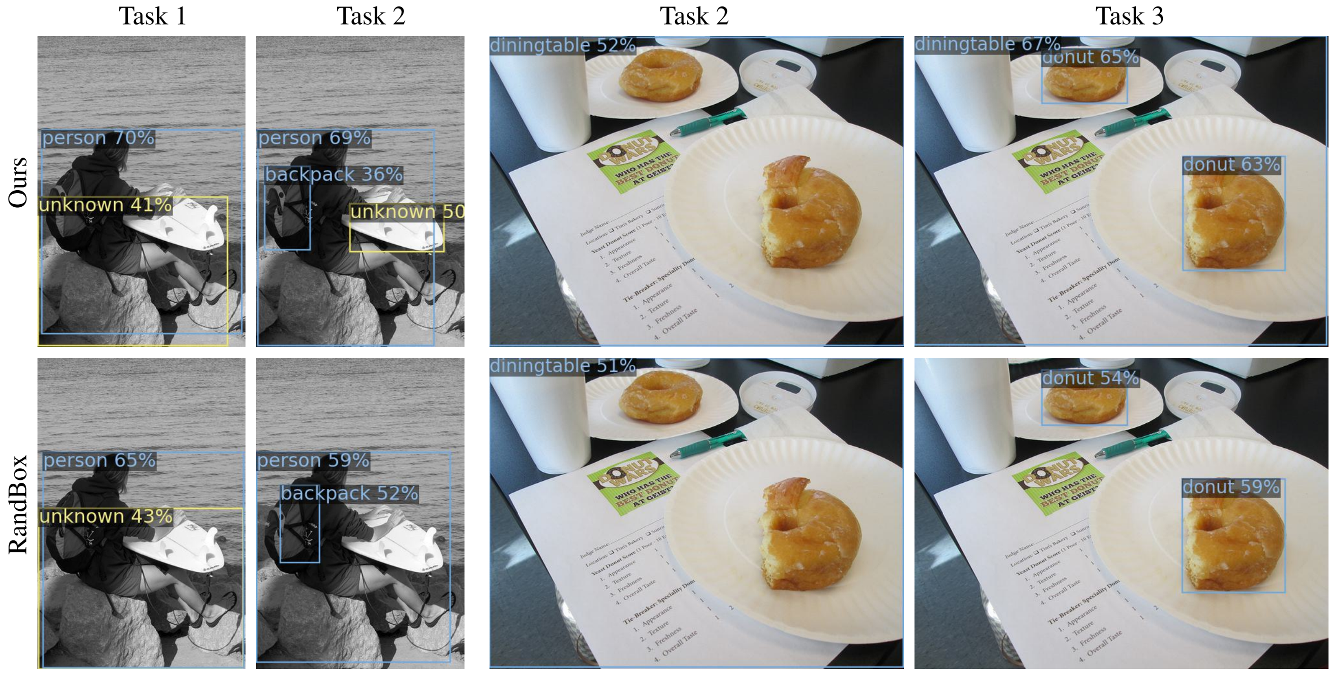


Figure 8. **Illustrations of the incremental learning capability on M-OWODB.** We compare with RandBox [70] in terms of known and unknown object detections across two successive stage of M-OWODB. Each image uses the same score threshold for a fair comparison.