Unknown Sniffer for Object Detection: Don't Turn a Blind Eye to Unknown Objects

Wenteng Liang[1,†], Feng Xue[1,†], Yihao Liu[1], Guofeng Zhong[2], Anlong Ming[1,*]
[1]Beijing University of Posts and Telecommunications, China
[2]Chongqing University of Posts and Telecommunications, China
{liangwenteng,xuefeng,l1h,mal}@bupt.edu.cn

Abstract

The recently proposed open-world object and open-set detection have achieved a breakthrough in finding never-seen-before objects and distinguishing them from known ones. However, their studies on knowledge transfer from known classes to unknown ones are not deep enough, resulting in the scanty capability for detecting unknowns hidden in the background. In this paper, we propose the unknown sniffer (UnSniffer) to find both unknown and known objects. Firstly, the generalized object confidence (GOC) score is introduced, which only uses known samples for supervision and avoids improper suppression of unknowns in the background. Significantly, such confidence score learned from known objects can be generalized to unknown ones. Additionally, we propose a negative energy suppression loss to further suppress the non-object samples in the background. Next, the best box of each unknown is hard to obtain during inference due to lacking their semantic information in training. To solve this issue, we introduce a graph-based determination scheme to replace hand-designed non-maximum suppression (NMS) post-processing. Finally, we present the Unknown Object Detection Benchmark, the first publicly benchmark that encompasses precision evaluation for unknown detection to our knowledge. Experiments show that our method is far better than the existing state-of-the-art methods. Code is available at: https://github.com/Went-Liang/UnSniffer.

1. Introduction

Detecting objects with a limited number of classes in the closed-world setting [2, 3, 14, 20, 21, 23, 31–33, 46] has been the norm for years. Recently, the popularity of autonomous driving [4, 7, 17, 25, 29, 30, 38, 39, 43–45] has raised the bar for object detection. That is, the detector should detect both known and unknown objects. '**Known Objects**' are those that belong to pre-defined categories, while '**Unknown Objects**' are those that the detector has never seen during training. Detecting unknown objects is crucial in coping with more challenging environments, such as autonomous driving scenes with potential hazards.

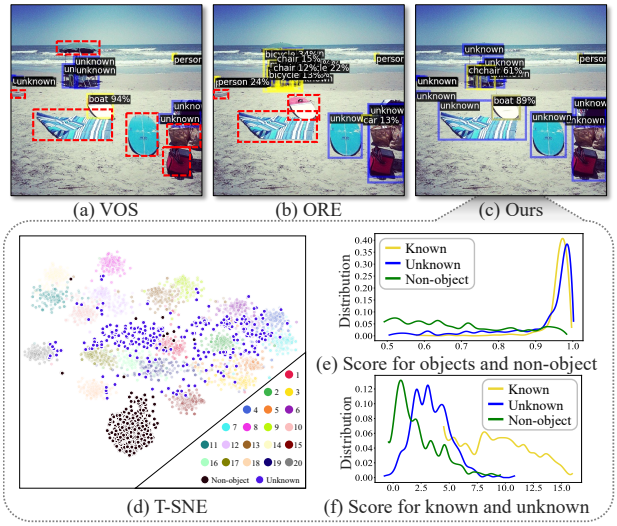


Figure 1. (a)-(c) the predicted unknown (blue), known (yellow), and missed (red) objects of VOS [9], ORE [18], and our model. (d) t-SNE visualization of various classes' hidden vectors. (e) score for objects and non-object (generalized object confidence). (f) score for unknown, known, and non-object (negative energy).

Since unknown objects do not have labels in the training set, how to learn knowledge that can be generalized to unknown classes from finite pre-defined categories is the key issue in detecting unknown objects. In recent years, a series of groundbreaking works have been impressive on open-set detection (OSD) [8, 9, 11, 28] and open-world object detection (OWOD) [12, 18, 37, 40]. Several OSD methods have used uncertainty measures to distinguish unknown

†Equal Contribution
*Corresponding Author

This work was supported by the national key R & D program intergovernmental international science and technology innovation cooperation project 2021YFE0101600.

objects from known ones. However, they primarily focus on improving the discriminatory power of uncertainty and tend to suppress non-objects along with many potential unknowns in the training phase. As a result, these methods miss many unknown objects. Fig. 1 (a) shows that VOS [9] misses many unknown objects, such as bags, stalls and surfboards. Furthermore, OWOD requires generating high-quality boxes for both known and unknown objects. ORE [18] and OW-DETR [12] collect the pseudo-unknown samples by an auto-labelling step for supervision and perform knowledge transfer from the known to the unknown by contrastive learning or foreground objectness. But the pseudo-unknown samples are unrepresentative of the unknown objects, thus limiting the model's ability to describe unknowns. Fig. 1 (b) shows that ORE [18] mis-detects many unknown objects, even though some are apparent.

In philosophy, there is a concept called '*Analogy*' [34], which describes unfamiliar things with familiar ones. We argue that *despite being ever-changing in appearance, the unknown objects are often visually similar to the objects of pre-defined classes*, as observed in Fig. 1 (d). The t-SNE visualization shows that the unknown objects tend to be among several pre-defined classes, while the non-objects are far away from them. This inspires us to express a unified concept of 'object' by the proposed generalized object confidence (GOC) score learned from the known objects only. To this end, we first discard the background bounding boxes and only collect the object-intersected boxes for training to prevent potential unknown objects from being classified as backgrounds. Then, a combined loss function is designed to enforce the detector to assign relatively higher scores to boxes tightly enclosing objects. Unlike 'objectness', non-object boxes are not used as the negative samples for supervision. Fig. 1 (e) shows that the GOC score distinctly separates non-objects and 'objects'. In addition, we design a negative energy suppression loss on top of VOS's energy calculation [9] to further widen the gap between the non-object and the 'object'. Fig. 1 (f) shows three distinct peaks for the knowns, unknowns and non-objects. Next, due to the absence of the unknown's semantic information in training, the detector hardly determines the best bounding box by a constant threshold when the number of objects cannot be predicted ahead of time. In our model, the best box determination is modelled as a graph partitioning problem, which adaptively clusters high-score proposals into several groups and selects one from each group as the best box.

As far as we know, the existing methods are evaluated on the COCO [22] and Pascal VOC benchmarks [10] that do not thoroughly label unknown objects. Therefore, the accuracy of unknown object detection cannot be evaluated. Motivated by this practical need, we propose the Unknown Object Detection Benchmark (UOD-Benchmark), which takes the VOC's training set as the training data and contains two test sets. (1) COCO-OOD containing objects with the unknown class only; (2) COCO-Mix with both unknown and known objects. They are collected from the original COCO dataset [22] and annotated according to the COCO's instance labeling standard. In addition, the Pascal VOC testing set is employed for evaluating known object detection.

Our key contributions can be summarized as follows:

- To better separate non-object and 'object', we propose the GOC score learned from known objects to express unknown objects and design the negative energy suppression to further limit non-object.
- The graph-based box determination is designed to adaptively select the best bounding box for each object during inference for higher unknown detection precision.
- We propose the UOD-Benchmark containing annotation of both known and unknown objects, enabling us to evaluate the precision of unknown detection. We comprehensively evaluate our method on this benchmark which facilitates future use of unknown detection in real-world settings.

2. Related Work

Open Set Classification and Detection aim to deal with unknown samples encountered in classification or detection tasks. Many uncertainties measuring the feature difference between unknown and known objects have been proposed, such as OpenMax [1], MSP [16], ODIN [19], Mahalanobis distance [5] and Energy [24]. For detection, some works [11, 27, 28] used Monte Carlo dropout to generate uncertainty scores. David *et al*. [13] proposed probabilistic detection quality to measure spatial and semantic uncertainty. Du *et al*. [8, 9] synthesized virtual outliers to shape the decision boundary of networks and used energy as an uncertainty measure. However, to ensure the accuracy of detecting known objects, they suppress both unknowns and non-objects in training, leading to a low recall of unknowns. In contrast, our method aims to detect all unknown objects.
Open-world object detection (OWOD) is proposed by ORE [18]. It detects both known and unknown objects by training pseudo-labeled unknown objects and incrementally learns updated annotations of new classes. OW-DETR [12] improves performance with multi-scale self-attention and deformable receptive fields. Yang *et al*. [40] introduced semantic topology to ensure that the feature representations are discriminative and consistent. UC-OWOD [37] also classifies unknown objects to achieve better results than ORE on measures about unknown classes. Zhao et al. [42] correct the auto-labeled proposals by Selective Search and calibrate the over-confident activation boundary by a class-specific expelling function. However, the auto-labeling step generates many pseudo-unknown samples that are unrepresentative of the unknowns in fact, limiting their ability to

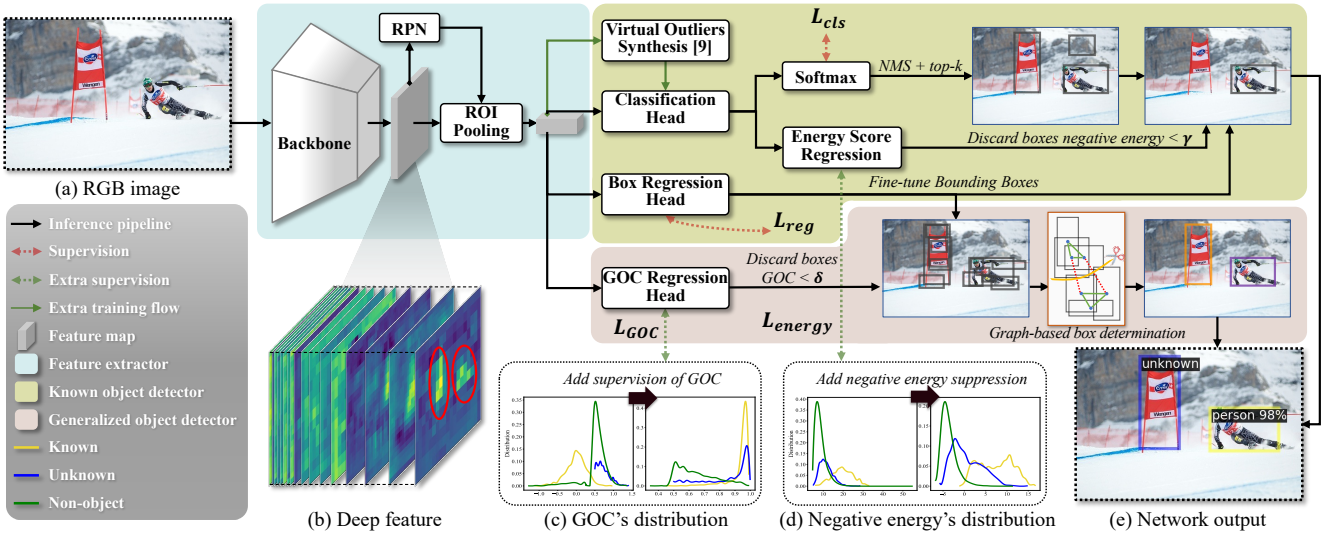


Figure 2. **The framework of UnSniffer** contains a feature extractor, a known object detector and a generalized object detector. (a) is the input RGB image. (b) visualizes several channels of deep features encoding the known and unknown objects at the same time, and the red circles mark the position of the objects. (c) shows the GOC score's distribution before and after training the GOC. (d) shows the negative energy's distribution before and after using negative energy suppression. (e) is the result.

transfer knowledge from the known to the unknown. Thus, during inference, many non-objects are mis-detected as unknown, leading to the low precision of unknown. This paper aims to reduce the false positives by the proposed GOC score and a graph-based box determination scheme.

3. Problem Formulation

Referring to [9], the problem of unknown detection in the setting of object detection is formulated as follows. We have a known class set $\mathcal{K} = \{1, 2, ..., C\}$ and an unknown class $C + 1$. The N input RGB images are denoted as $\{\mathbf{I}_1, ..., \mathbf{I}_N\}$, with corresponding labels $\{\mathbf{Y}_1, ..., \mathbf{Y}_N\}$. Each $\mathbf{Y}_i = \{\mathbf{y}_1, ..., \mathbf{y}_K\}$ contains a set of object instances with $\mathbf{y}_k = [l_k, x_k, y_k, w_k, h_k]$, where l_k is the class label for a bounding box represented by x_k, y_k, w_k, h_k. If $\mathbf{y}_k$ encloses a known object, $l_k \in \mathcal{K}$, otherwise $l_k = C + 1$.

The model is trained on the data containing known-class objects only $\{(\mathbf{I}_n, \mathbf{Y}_n)|l_k \in \mathcal{K}, \mathbf{y}_k \in \mathbf{Y}_n\}_{n=1}^{N^{\mathbf{train}}}$, but tested on the data including unknown objects $\{(\mathbf{I}_n, \mathbf{Y}_n)|l_k \in \mathcal{K} \cup \{C + 1\}, \mathbf{y}_k \in \mathbf{Y}_n\}_{n=1}^{N^{\mathbf{test}}}$, where $N^{\mathbf{train}}$ is the image number of the training set, $N^{\mathbf{test}}$ for that of the test set, and $N = N^{\mathbf{test}} + N^{\mathbf{train}}$.

4. Method

We propose the unknown sniffer (UnSniffer) to find both the known and unknown objects. The pipeline is shown in Fig. 2. The RGB image $\mathbf{I}_n$ is fed into a feature extractor [33] that captures numerous object proposals $\{b_i|i \in [1, M]\}$ and their feature vectors $\{f_i|f_i \in \mathbb{R}^{1024}, i \in [1, M]\}$. Taking the feature f_i as input, we use two detectors for known and unknown objects.

Firstly, the generalized object detector learns the proposed generalized object confidence (GOC) score to determine whether proposal b_i contains an object (See Sec. 4.1). Then, the graph-based box determination scheme is used to cluster the high-score proposals into several groups (See Sec. 4.2). We select the one with the highest GOC score in each group as a set of unknown predictions.

The second one, i.e., a known object detector, computes the class-specific probabilities and the negative energy score [9] for b_i. In addition to the classification head and box regression head commonly used in two-stage object detectors [2, 14, 20, 33], we employ the virtual outliers synthesis [9] to learn energy scores and remove the low-negative-energy proposals during inference. Unlike [9], we employ a negative energy suppression loss to enforce the negative energy scores of non-object boxes less than zero (See Sec. 4.3). It lowers the feature response inside non-object boxes and boosts the discriminative power of both detectors.

Finally, the first detector outputs the bounding box predictions of unknown class, and the second detector gives that of known class. We directly concatenate the two results and remove the unknown-class predictions whose IoU with any known-class prediction exceeds a constant threshold β. Fig. 2 (e) shows the merged result of image $\mathbf{I}_n$.

Note that the UnSniffer has two training stages, which are consistent with VOS [9]. In the first stage, we employ the training process of Faster-RCNN [33] (the red dot arrows in Fig. 2), where L_{cls} and L_{reg} are the losses for classification and bounding box regression, respectively. And the second stage additionally employs the losses proposed

in this paper (the green dot arrows in Fig. 2).

4.1. Generalized Object Confidence Score

Uncertainty Scores are usually modeled as either the maximum known-class confidence scores [16,19] or the entropy of the classification results [8,9,24]. It can be used to distinguish unknowns from known objects according to the high uncertainty scores of the unknown objects. However, the uncertainty's training phase suppresses both unknowns and non-objects, causing the inadequate detection of unknowns. **Objectness Scores** are usually used to judge whether a bounding box containing an object [26, 33, 47], which naturally meets the requirement of unknown object detection, such as the foreground objectness learning in OW-DETR [12] implemented by a binary classification. However, learning-based object proposal methods cannot avoid a misuse of unknown samples as negative samples, leading to low discriminative power between non-objects and unknowns.

Generalized Object Confidence Score and Losses. We propose the generalized object confidence (GOC) score. It can be used to judge whether a proposal contains an object (including unknown and known classes), while this capability stems from the fact that many unknowns are actually encoded by the pre-trained backbone, as the 'flag' shown in Fig. 2(b).

Different from uncertainty and objectness, the GOC score is trained using only known objects and can be generalized to unknown objects. Specifically, the GOC regression head that is composed of a linear transformation, denoted as Φ, is used to compute the GOC score $\Phi(f_i)$ for a given proposal's feature f_i. In the training phase, given an image-label pair $(\mathbf{I}_n, \mathbf{Y}_n)$, the region proposal network is firstly used to extract numerous proposals $B_n = \{b_i | i \in [1, M]\}$ from image $\mathbf{I}_n$. And we define the intersection over the predicted bounding box (IoP) and the intersection over the correct bounding box (IoC) for collecting training samples from B_n as follows:

$$IoP(b_i, \mathbf{y}_k) = \frac{|b_i \cap \mathbf{y}_k|}{|b_i|}, \quad IoC(b_i, \mathbf{y}_k) = \frac{|b_i \cap \mathbf{y}_k|}{|\mathbf{y}_k|} \quad (1)$$

where $\mathbf{y}_k$ is the k-th instance's bounding box in $\mathbf{Y}_n = \{\mathbf{y}_1, ..., \mathbf{y}_K\}$. Subsequently, for each proposal b_i in B_n, we find the object instance that has the maximum IoU with b_i. And the proposals enclosing the same object are assigned to the same group, obtaining K groups of proposals: $B_n^1, B_n^2, .., B_n^K$. Then, we divide the proposals of B_n^k into **c**omplete-object, **p**artial-object, **o**versized, and **n**on-object according to IoU, IoP, and IoC, as shown in Fig. 3:

$$\begin{aligned}
\mathbf{B}_n^{k,\mathbf{c}} &= \{b_i \in B_n^k | IoU(b_i, \mathbf{y}_k) \geq e_2\} \\
\mathbf{B}_n^{k,\mathbf{p}} &= \{b_i \in B_n^k | e_1 \leq IoU(b_i, \mathbf{y}_k) < e_2, IoP(b_i, \mathbf{y}_k) \geq \rho\} \\
\mathbf{B}_n^{k,\mathbf{o}} &= \{b_i \in B_n^k | e_1 \leq IoU(b_i, \mathbf{y}_k) < e_2, IoC(b_i, \mathbf{y}_k) \geq \rho\} \\
\mathbf{B}_n^{k,\mathbf{n}} &= \{b_i \in B_n^k | IoU(b_i, \mathbf{y}_k) < e_1\}
\end{aligned} \quad (2)$$

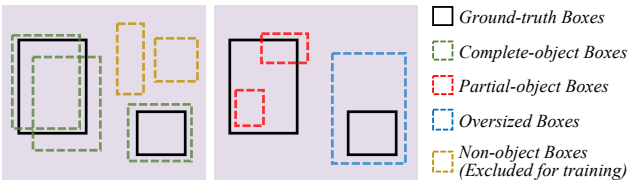


Figure 3. The sample definition in GOC supervision.

where e_1, e_2, ρ are the constant thresholds, as shown in Fig. 3. In order to prevent the potential unknown objects from being treated as background during training, we only use the first three groups in Eq. 2 to train the module Φ with three losses. In the first loss, the GOC scores of complete-object bounding boxes are pushed towards one:

$$L_{pos} = \frac{1}{K} \sum_{k \in [1,K]} \frac{1}{|B_n^{k,\mathbf{c}}|} \sum_{b_i \in \mathbf{B}_n^{k,\mathbf{c}}} \big(\Phi(f_i) - 1\big)^2 \quad (3)$$

Then, due to the lack of clear criteria measuring GOC scores of partial-object or oversized boxes, we suppress their GOC scores to below a constant δ:

$$L_{neg} = \frac{1}{K} \sum_{k \in [1,K]} \frac{1}{|B_n^{k,\mathbf{po}}|} \sum_{b_i \in \mathbf{B}_n^{k,\mathbf{po}}} \max\big(0, \Phi(f_i) - \delta\big) \quad (4)$$

where $B_n^{k,\mathbf{po}} = B_n^{k,\mathbf{p}} \cup B_n^{k,\mathbf{o}}$. Next, we improve the model's ability to capture a box enclosing an object more entirely by a contrastive loss, which compares two boxes in $\mathbf{B}_n^{k,\mathbf{c}}$:

$$L_{con} = \frac{1}{K} \sum_{k \in [1,K]} \left\lfloor \frac{2}{|\mathbf{B}_n^{k,\mathbf{c}}|} \right\rfloor \sum_{b_i, b_j \in \mathbf{B}_n^{k,\mathbf{c}}} \max\big(0, \frac{\Phi(f_i) - \Phi(f_j)}{\alpha} + \zeta\big) \quad (5)$$

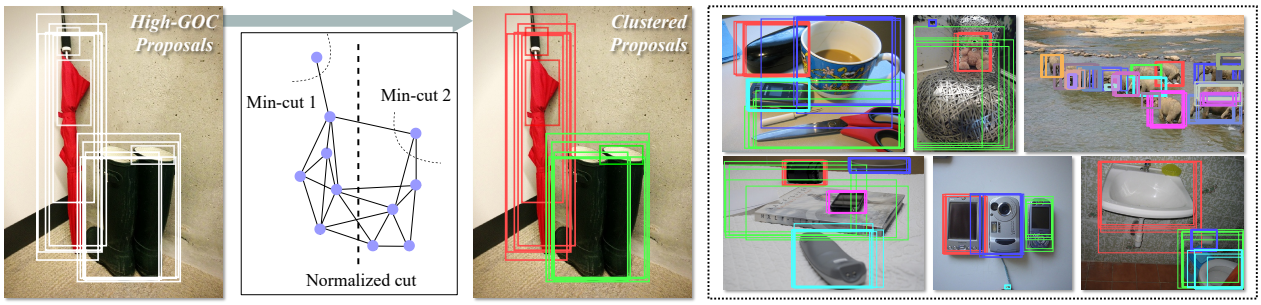
where $\alpha = 1$ when $IoU(b_j, \mathbf{y}_k) > IoU(b_i, \mathbf{y}_k)$, otherwise $\alpha = -1$. ζ is a tiny constant that is set to 0.01, and $i \neq j$. Finally, the total GOC loss is formulated as:

$$L_{GOC} = L_{neg} + L_{pos} + L_{con} \quad (6)$$

Since our training process does not utilize the sample of the background area, the GOC scores of non-object bounding boxes would not be affected greatly. On the contrary, the GOC scores of both unknown and known objects are pushed to a high score. As shown in Fig. 2 (c), when the GOC is not supervised, unknown boxes have the same output as non-object boxes, but it changes dramatically when the GOC regression head is supervised by L_{GOC}.

4.2. Graph-based Top-scoring Box Determination

By using the GOC score for ranking proposals during inference, we obtain the proposals where the objects are most likely to be. However, the traditional post-processing mechanism, i.e. using NMS and outputting top-k highest results, is inappropriate to determine the unknown prediction, as the number of objects cannot be prophesied at ahead of time.



(a) Graph-based top-scoring boxes determination (b) More exemplars

Figure 4. The illustration and more examples for the graph-based top-scoring box determination. The white rectangles denote the proposals with top GOC scores in the image. For other rectangles, each group of proposals is represented by the same color.

To address this issue, we perform the top-scoring box determination as a graph partitioning problem, which adaptively finds the best bounding box for each object, as shown in Fig. 4(a). Specifically, during the inference, given a set of proposals $\{b_i|i \in [1, M]\}$ and their GOC scores $\{\Phi(f_i)|i \in [1, M]\}$ for image $\mathbf{I}_n$, we construct a weighted undirected graph $G = (V, E)$. Each node in set V represents an object proposal b_i, and each edge in set E is formed by the IoU between both ends of this edge, i.e. $IoU(b_i, b_j)$. As shown in Fig. 4(a), considering that some of the proposals may only cover part of the objects, we employ the recursive two-way normalized cut algorithm [35] to decompose the entire graph G into several sub-graphs iteratively, which is terminated until the NCut value [35] of a sub-graph is lower than a threshold ε, where ε is determined by a threshold selection method in Sec. 6.1. Finally, the top-1 GOC proposal of each sub-graph is taken as the prediction.

It can be seen from Fig. 4(b), even if only IoU is used as the measurement of edge in graph G, our model still performs considerable proposal clustering and avoids outlier proposals as an independent group. Especially in the upper right prediction of Fig. 4(b), almost every elephant gets an independent group of proposals.

4.3. Negative Energy Suppression for Non-object

Referring to VOS [9], the energy score is employed to distinguish unknown objects from known ones, as shown in Fig. 2. For proposal b_i, the energy score is formulated as the negative of the weighted sum of this proposal's logit output in exponential space:

$$E(b_i) = -\log \sum\nolimits_{c\in[1,C]} \mathbf{w}_c \cdot \exp^{\mathbf{f}_c} \tag{7}$$

where $\mathbf{f}_c$ is the logit output for class c in the classification head, C is the number of the know classes, and $\mathbf{w}_c$ is the learnable parameter for alleviating the class imbalance. The proposals with higher negative energy scores are treated as known predictions, whereas others are unknown predictions. However, due to insufficient training, some non-object proposals gain such high negative energy scores that they are indistinguishable from objects, as shown in the left plot of Fig. 2(d). To address this issue, we propose negative energy suppression to further reduce the negative energy scores of non-object proposals. Specifically, we observe that most non-object boxes have lower negative energy scores than those of the known and unknown classes, which motivates us to design a suppression loss to constraint T proposals with the lowest negative energy scores:

$$L_{suppression} = \frac{1}{T} \sum\nolimits_{i\in[1,T]} \max(0, -E(b_i)) \tag{8}$$

The overall energy loss consists of our proposed $L_{suppression}$ and $L_{uncertainty}$ defined by VOS [9]:

$$L_{energy} = L_{suppression} + L_{uncertainty} \tag{9}$$

As shown in the right plot of Fig. 2(d), after training with loss L_{energy}, the negative energy distribution of the non-object is significantly different from that of the unknown, indicating that the non-objects are indeed suppressed. In addition, by using Eq. 8, the feature responses of non-object bounding boxes are reduced simultaneously, which further widens the GOC difference between non-object and object. It can be proved by the fact that the high GOC scores of unknowns in Fig. 1(e) (with $L_{suppression}$) are more than that in the right plot of Fig. 2(c) (without $L_{suppression}$). In addition, the ablation studies of Sec. 6.3 demonstrate that this approach improves the detection precision of the model.

5. Unknown Object Detection Benchmark

5.1. Datasets

In the proposed UOD-Benchmark, we refer to [9, 18] and use the Pascal VOC dataset [10] as the training data that contains annotations of 20 object categories. For testing, since the MS-COCO dataset [22] extends the PASCAL VOC categories to 80 object categories, we naturally employ MS-COCO to evaluate unknown objects. However, MS-COCO does not thoroughly label potential unknown objects in images. To address this issue, we propose two datasets, i.e., COCO-OOD and COCO-Mixed, which fully

Datasets	Images	Known	Unknown
VOC-Pretest	200	5.09	0
VOC-Test	4952	3.02	0
COCO-OOD♣	504	0	3.28
COCO-Mixed♣	897	2.96	2.82

Table 1. **The Statistics of datasets** that include the number of images, the average number of known and unknown instances per image. ♣ denotes the augmented datasets.

label the unknown objects. Firstly, according to the definition of objects in COCO [22], i.e. "objects are individual instances that can be easily labelled (person, chair, car)", we hand-pick more than a thousand images that have no area confused with this definition. Secondly, several master students are asked to mark the object regions they got at first glance by drawing polygons, referring to the object definition above. As shown in Fig. 5, we label almost every object in the selected images with fine-grained annotation. Finally, we have two datasets both for testing as follows:

COCO-OOD dataset contains only unknown categories, consisting of 504 images with fine-grained annotations of 1655 unknown objects. All annotations consist of original annotations in COCO and the augmented annotations on the basis of the COCO definition.

COCO-Mixed dataset includes 897 images with annotations of both known and unknown categories. It contains 2533 unknown objects and 2658 known objects, with original COCO annotations used as labels for known objects. Unambiguous unlabeled objects are also annotated. The dataset is more challenging to evaluate due to the images containing more object instances with complex categories and concentrated locations.

In addition, we employ the test set of the Pascal VOC dataset to evaluate the accuracy of known object detection. The statistics of these test datasets are placed in Table 1. Fig. 5 shows some fully annotated images in COCO-OOD and COCO-Mixed. Note that the VOC-Pretest is used to set a threshold, mentioned in Sec .6.1.

5.2. Evaluation Metrics

To evaluate the performance of known object detection, we employ a prevalent metric, i.e., mean Average Precision (mAP) [9]. As for the unknown detection performance, assuming that TP_u denotes the true positive proposals of unknown classes, FN_u for false negative proposals, and FP_u for false positive proposals, five metrics are employed:

- The Unknown Average Precision (U-AP) is used with reference to the conventional object detection [10].
- The Recall Rate (U-REC) and Precision Rate of Unknown (U-PRE) are defined as follows $U\text{-}REC = \frac{TP_u}{TP_u+FN_u}, U\text{-}PRE = \frac{TP_u}{TP_u+FP_u}$.

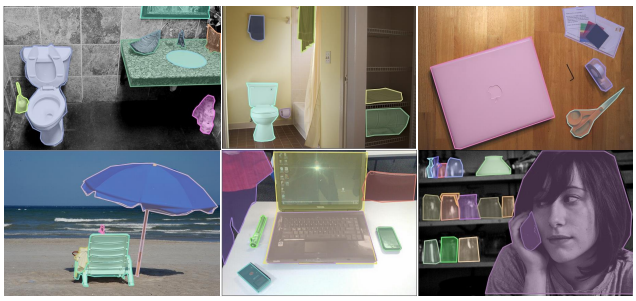


Figure 5. Annotated samples in COCO-OOD and COCO-Mix.

- For a comprehensive comparison, we report the Unknown F1-Score defined as the harmonic mean of U-PRE and U-REC: $U\text{-}F1 = \frac{2\times U\text{-}PRE\times U\text{-}REC}{U\text{-}PRE+U\text{-}REC}$.
- The Absolute Open-Set Error (A-OSE) [28] is also employed to report the number count of unknown objects that are wrongly classified as any known classes.
- Wilderness Impact (WI) [6] are defined as $WI = \frac{Precision\ in\ closed-set}{Precision\ in\ open-set} - 1$ to characterize the case that unknown objects are confused with the known.

Note that we measure mAP over different IoU thresholds from 0.5 to 0.95. Other metrics, such as U-REC, U-PRE, etc., are measured at the IoU threshold of 0.5. WI is measured at a recall rate of 0.8.

6. Experiment

6.1. Implementation Details

We use the Detectron2 [36] library and employ a ResNet-50 [15] backbone. δ in Eq. 4 is empirically set to 0.5. β is set to 0.98. We set the thresholds e_1 in Eq. 2 to 0.0, and e_2, ρ to 0.5 by parameter experiments. The proposal number T is set to 100 in Eq.8. We train the model for a total of 18,000 iterations. The starting iteration of our second stage is 12000, which is consistent with VOS [9].

Determining inference thresholds in pretest mode. Both VOS [9] and OWOD [18] determine the threshold before inference, but they bring in unknown data in an implicit or explicit way when computing the thresholds. Therefore, we introduce a pretest operation before inference, which selects 200 images from the training set that do not contain any potential unknown objects for threshold determination. The first row of Table 1 shows the statistics of the pretest dataset. In the pretest mode, we firstly obtain the negative energy score of the proposals predicted from the pretest dataset, and set the threshold γ such that 95% of predicted proposals have a negative energy score greater than it. Then, for the graph-based box determination, we set 10 thresholds of NCut value equally spaced from 0 to 1, and choose the threshold when the AP of known objects is the largest as ε.

*https://github.com/deeplearning-wisc/vos/issues/26
†https://github.com/deeplearning-wisc/vos/issues/13

Groups	Methods	VOC-Test	COCO-OOD				COCO-Mix						
		mAP	U-AP	U-F1	U-PRE	U-REC	mAP	U-AP	U-F1	U-PRE	U-REC	AOSE	WI
①	Faster-RCNN [33]	0.483	-	-	-	-	-	-	-	-	-	-	-
②	MSP [16]	0.470	0.213	0.314	0.279	0.359	0.364	0.055	0.169	0.190	0.153	588	0.135
	Mahalanobis [5]	0.447	0.129	0.271	0.309	0.241	0.351	0.051	0.149	0.207	0.116	604	0.165
	Energy score [24]	0.474	0.213	0.308	0.260	0.377	0.364	0.048	0.169	0.167	0.171	470	0.137
③	OW-DETR [12]	0.420	0.033	0.056	0.030	0.380	**0.414**	0.007	0.025	0.014	0.161	569	**0.086**
	ORE [18]	0.243	0.214	0.255	0.153	**0.782**	0.213	0.140	0.175	0.103	**0.592**	485	0.089
④	VOS^1 [9]	**0.485**	0.135	0.196	0.342	0.137	0.377	0.040	0.101	**0.262**	0.062	640	0.152
	VOS^2 [9]	0.469	0.205	0.317	0.291	0.348	0.364	0.051	0.172	0.184	0.163	409	0.124
⑤	Ours	0.464	**0.454**	**0.479**	**0.433**	0.535	0.359	**0.150**	**0.287**	0.222	0.409	**398**	0.175

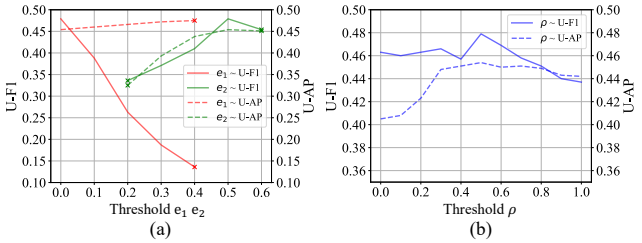
Table 2. Comparisons with the traditional detector ① and detectors using open-set classification ②, open-world object detection ③, and open-set detection ④ methods. VOS^1 means using the threshold in the official repository*, calculated on the BDD100K dataset [41]. VOS^2 means using the threshold computed on the COCO-OOD dataset by the official code†. Best results are in bold, second best are underlined.

Row	GOC	NES	GBD	U-AP	U-F1	U-PRE	U-REC
1	×	×	×	0.066	0.050	0.026	0.808
2	×	×	✓	0.250	0.434	0.395	0.481
3	×	✓	×	0.442	0.054	0.028	0.861
4	✓	×	×	0.479	0.323	0.215	0.646
5	×	✓	✓	0.409	0.467	**0.437**	0.502
6	✓	×	✓	0.455	0.454	0.399	0.528
7	✓	✓	×	**0.474**	0.342	0.234	**0.632**
8	✓	✓	✓	0.454	**0.479**	0.433	0.535

Table 3. **Ablation studies on COCO-OOD**. GOC, NES and GBD refer to 'generalized object confidence', 'negative energy suppression' and 'graph-based box determination', respectively. If 'GBD' is ×, we use NMS+top-k with the threshold of the known detector.

6.2. Results

Quantitative Analysis. In Table 2, we show UnSniffer's result on the UOD-Benchmark, along with the results of MSP [16], Mahalanobis [5], Energy score [24], ORE [18], OW-DETR [12] and VOS [9]. Note that OW-DETR is based on Deformable DETR [46] with a stronger discriminative power, while other methods use Faster-RCNN. Since U-PRE or U-REC cannot independently reflect the model's performance, we mainly employ U-AP, U-F1, AOSE and WI. Observably, on the COCO-OOD dataset, the U-AP of UnSniffer outperforms the 2^{nd} result by more than twice, and our U-F1 is 16.2% higher than the 2^{nd} result, at the cost of a 1.9% drop in mAP on VOC compared to Faster-RCNN [33]. On the COCO-Mix dataset, the UnSniffer still holds the lead in both U-AP and U-F1, which are 1% and 11.2% higher than the 2^{nd} results, respectively. Those comparisons demonstrate that UnSniffer outperforms the existing methods in unknown object detection, which is owed to to our GOC learning the overall confidence of objects from finite known objects. Furthermore, UnSniffer has the smallest AOSE (398) but the largest WI (0.175), which can be explained by the inverse relationship between WI and the count of known objects misclassified as an incorrect class.



(a) (b)

Figure 6. **Sensitivity analysis on (a) thresholds e_1, e_2, and (b) threshold ρ.** × indicates the failed training outside this threshold.

More details are illustrated in the supplementary material.

Qualitative Analysis. Fig. 7 visualizes the results of different methods on example images of the COCO-Mix (first two rows) and COCO-OOD dataset (last three rows). It can be seen that VOS [9], MSP [16], Mahalanobis distance [5], and Energy score [24] miss many objects of the unknown class, such as the surfboards in the 1^{st} image, the keyboard and water cup in the 3^{rd} image, the CD case in the 4^{th} image. Their failure is due to the suppression of unknown objects in training. For the OWOD methods, ORE and OW-DETR generate too many predicted object boxes on almost all images. Most predictions are false positives. In contrast, UnSniffer does not miss any unknown objects because we give reasonable GOC scores to all unknown objects. Moreover, using graph-based box determination, UnSniffer reliably predicts a single bounding box for each object even if two or more objects overlap (See the 4^{th} row). More results are available in the supplementary material.

6.3. Discussions and Analysis

Ablation Study. To investigate the contribution of each component in UnSniffer, we design ablation experiments in Table 3. Since the softmax in known object detector fails to provide reasonable confidences for unknowns, when the 'GOC' is ×, we employ the negative energy score for ranking the unknown proposals. Comparing the 1^{st} and 2^{nd} rows, it can be seen that the GBD outperforms NMS,

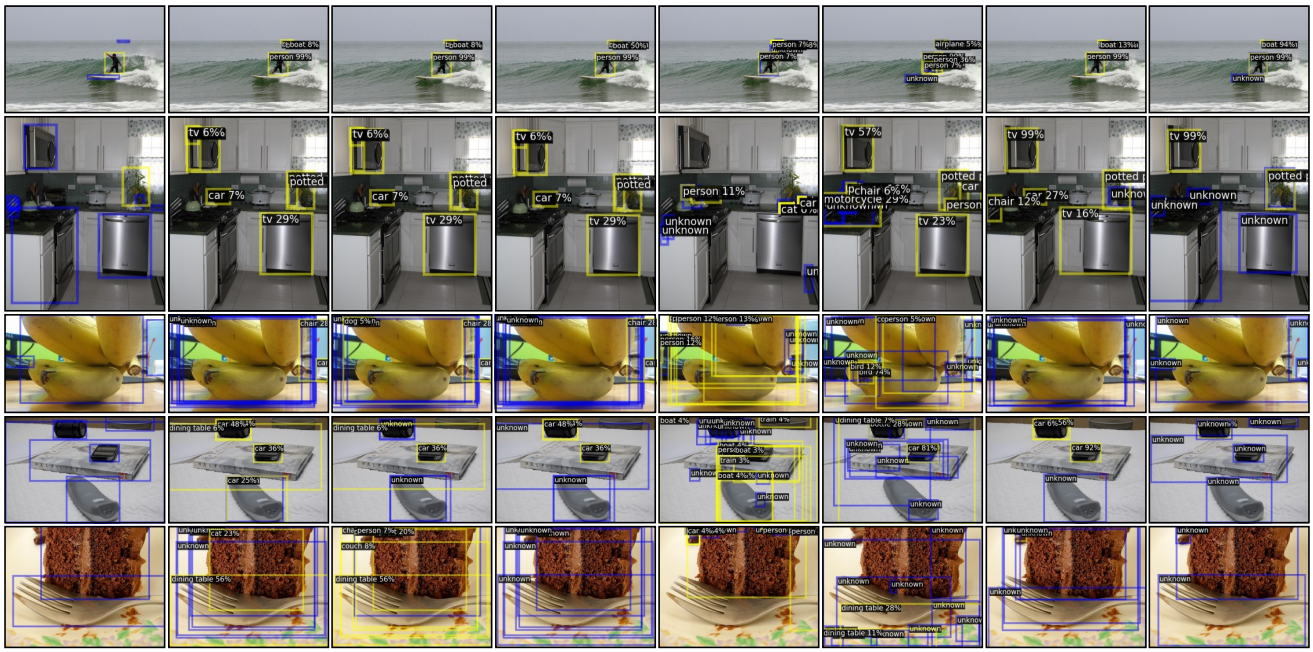


Figure 7. **Example results on COCO-Mix (first two rows) and COCO-OOD datasets (last three rows)**. 1^{st} column: ground truth; 2^{nd}-8^{th} columns: MSP [16], Mahalanobis [5], Energy score [24], OW-DETR [12], ORE [18], VOS [9] (with threshold computed on COCO-OOD dataset), and our method. The detections are overlaid on the known (yellow) and unknown (blue) class objects. Since ORE and OW-DETR generate too many results, we only draw the top-10 boxes for each image , and other methods draw all predicted boxes.

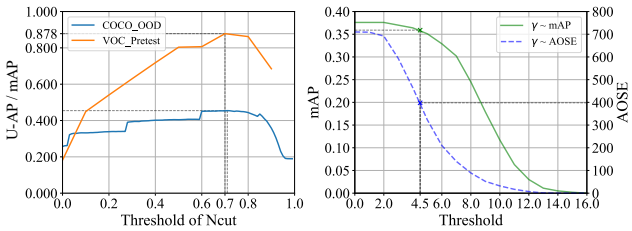


Figure 8. (a) Comparison between the thresholds ε determined in the pretest set and the COCO-OOD dataset. (b) Validation of the threshold γ computed in pretest mode on the COCO-Mix dataset.

and GBD plays a key role in improving detection precision. Comparing the 1^{st} and 4^{th} rows, obviously, GOC helps to increase U-AP by 41.3%, which shows that the GOC score well measures the probability that a proposal belongs to an object. Using both NES and GBD (5^{th} row) results in good U-PRE performance, indicating the effectiveness of NES in reducing false-positive unknown objects. Finally, using all modules achieves high U-PRE and U-REC simultaneously.

Parameter sensitivity analysis of GOC sampling. We adjust the thresholds e_1, e_2, ρ in Eq. 2, respectively. And the results on the COCO-OOD dataset are shown in Fig. 6. The best result is achieved when $e_1 = 0$, $e_2 = 0.5$, and $\rho = 0.5$. Note that, when e_1 exceeds 0.4, the network cannot be successfully trained. When e_2 is lower than 0.2, the training cannot converge, which means that the contrastive loss L_{con} is unsuitable for supervising too many samples.

Effectiveness of pretest-based threshold determination. As shown in Fig. 8 (a), the optimal threshold of ε determined by pretest data is almost the same as that determined by traversing the COCO-OOD dataset. Fig. 8 (b) shows the curve of mAP and AOSE when using different γ. Note that γ is determined to be 4.5 in the pretest data. When γ is equal to 4.5, the mAP loss of known objects is small, but the AOSE is largely reduced. It shows that UnSniffer largely retains the ability to detect known objects, meanwhile effectively alleviating the false detection of the unknown objects as a known class. These comparisons prove that the pretest mode is suitable for determining the threshold of an unknown object detector. And it only uses part of the training data without any risk of leaking test data.

7. Conclusion

To meet the real-world requirements for perceiving known and unknown objects, the UnSniffer is designed. Firstly, we design the GOC score that reliably measures the probability of a box that contains an object. Then, we model the top-scoring box determination as graph partitioning to obtain the best box for each object. Thirdly, the proposed negative energy suppression further limits the non-object boxes. Finally, we introduce the UOD-Benchmark to more comprehensively evaluate the real-world usability of the model. We hope our work inspires future research on unknown object detection in real-world settings.

References

[1] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[2] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 3

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, 2020. 1

[4] N. Deepika and V. V. Sajith Variyar. Obstacle classification and detection for vision based navigation for autonomous driving. In *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017. 1

[5] Taylor Denouden, Rick Salay, Krzysztof Czarnecki, Vahdat Abdelzad, Buu Phan, and Sachin Vernekar. Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance. *ArXiv*, abs/1812.02765, 2018. 2, 7, 8

[6] Akshay Dhamija, Manuel Gunther, Jonathan Ventura, and Terrance Boult. The overlooked elephant of object detection: Open set. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2020. 6

[7] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning (CORL)*, 2017. 1

[8] Xuefeng Du, Xin Wang, Gabriel Gozum, and Yixuan Li. Unknown-aware object detection: Learning what you don't know from videos in the wild. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 4

[9] Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. Vos: Learning what you don't know by virtual outlier synthesis. In *International Conference on Learning Representations (ICLR)*, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision (IJCV)*, 88(2):303–338, 2010. 2, 5, 6

[11] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning (ICML)*, 2016. 1, 2

[12] Akshita Gupta, Sanath Narayan, KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. OW-DETR: Open-world detection transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 4, 7, 8

[13] David Hall, Feras Dayoub, John Skinner, Haoyang Zhang, Dimity Miller, Peter Corke, Gustavo Carneiro, Anelia Angelova, and Niko Sünderhauf. Probabilistic object detection: Definition and evaluation. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2020. 2

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 3

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[16] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *ArXiv*, abs/1610.02136, 2016. 2, 4, 7, 8

[17] Joel Janai, Fatma Güney, Aseem Behl, Andreas Geiger, et al. Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends® in Computer Graphics and Vision (FTCGV)*, 12(1–3):1–308, 2020. 1

[18] KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards open world object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 2, 5, 6, 7, 8

[19] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *ArXiv*, abs/1706.02690, 2017. 2, 4

[20] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 3

[21] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 1

[22] 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 *European Conference on Computer Vision (ECCV)*, 2014. 2, 5, 6

[23] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision (ECCV)*, 2016. 1

[24] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems (NIPS)*, 2020. 2, 4, 7, 8

[25] Zhe Liu, Xin Zhao, Tengteng Huang, Ruolan Hu, Yu Zhou, and Xiang Bai. Tanet: Robust 3d object detection from point clouds with triple attention. In *AAAI Conference on Artificial Intelligence(AAAI)*, 2020. 1

[26] Jianxiang Ma, Anlong Ming, Zilong Huang, Xinggang Wang, and Yu Zhou. Object-level proposals. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 4

[27] Dimity Miller, Feras Dayoub, Michael Milford, and Niko Sünderhauf. Evaluating merging strategies for sampling-based uncertainty techniques in object detection. In *International Conference on Robotics and Automation (ICRA)*, 2019. 2

[28] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2018. 1, 2, 6

[29] Peter Pinggera, Sebastian Ramos, Stefan Gehrig, Uwe Franke, Carsten Rother, and Rudolf Mester. Lost and found: detecting small road hazards for self-driving vehicles. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016. 1

[30] Sebastian Ramos, Stefan Gehrig, Peter Pinggera, Uwe Franke, and Carsten Rother. Detecting unexpected obstacles for self-driving cars: Fusing deep learning and geometric modeling. In *IEEE Intelligent Vehicles Symposium (IV)*, 2017. 1

[31] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[32] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[33] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015. 1, 3, 4, 7

[34] Henrique Jales Ribeiro. *The Role of Analogy in Philosophical Discourse*. Springer International Publishing, 2014. 2

[35] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22(8):888–905, 2000. 5

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

[37] Zhiheng Wu, Yue Lu, Xingyu Chen, Zhengxing Wu, Liwen Kang, and Junzhi Yu. UC-OWOD: Unknown-classified open world object detection. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2

[38] Feng Xue, Anlong Ming, Menghan Zhou, and Yu Zhou. A novel multi-layer framework for tiny obstacle discovery. In *International Conference on Robotics and Automation (ICRA)*, 2019. 1

[39] Feng Xue, Anlong Ming, and Yu Zhou. Tiny obstacle discovery by occlusion-aware multilayer regression. *IEEE Transactions on Image Processing (TIP)*, 29:9373–9386, 2020. 1

[40] Shuo Yang, Peize Sun, Yi Jiang, Xiaobo Xia, Ruiheng Zhang, Zehuan Yuan, Changhu Wang, Ping Luo, and Min Xu. Objects in semantic topology. In *International Conference on Learning Representations (ICLR)*, 2022. 1, 2

[41] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 7

[42] Xiaowei Zhao, Xianglong Liu, Yifan Shen, Yixuan Qiao, Yuqing Ma, and Duorui Wang. Revisiting open world object detection. *ArXiv*, abs/2201.00471, 2022. 2

[43] Yu Zhou, Xiang Bai, Wenyu Liu, and Longin Latecki. Fusion with diffusion for robust visual tracking. In *Advances in Neural Information Processing Systems (NIPS)*, 2012. 1

[44] Yu Zhou, Xiang Bai, Wenyu Liu, and Longin Jan Latecki. Similarity fusion for visual tracking. *International Journal of Computer Vision (IJCV)*, 118(3):337–363, 2016. 1

[45] Yu Zhou., Yinfei Yang., Yi Meng., Xiang Bai, Wenyu Liu, and Longin Jan Latecki. Online multiple person detection and tracking from mobile robot in cluttered indoor environments with depth camera. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, 28(1):1455001.1–1455001.28, 2014. 1

[46] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations (ICLR)*, 2021. 1, 7

[47] C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *European Conference on Computer Vision (ECCV)*, 2014. 4