HSIC-based Moving Weight Averaging for Few-Shot Open-Set Object Detection

Binyi Su
Subinyi@buaa.edu.cn
State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University

Hua Zhang
zhanghua@iie.ac.cn
Institute of Information Engineering, Chinese Academy of Sciences

Zhong Zhou*
zz@buaa.edu.cn
State Key Laboratory of Virtual Reality Technology and Systems, Beihang University & Zhongguancun Laboratory

ABSTRACT

We study the problem of few-shot open-set object detection (FOOD), whose goal is to quickly adapt a model to a small set of labeled samples and reject unknown class samples. Recent works usually use the weight sparsification for unknown rejection, but due to the lack of tailored considerations for data-scarce scenarios, the performance is not satisfactory. In this work, we solve the challenging few-shot open-set object detection problems from three aspects. First, different from previous pseudo-unknown sample mining methods, we employ the evidential uncertainty estimated by the Dirichlet distribution of probability to mine the pseudo-unknown samples from the foreground and background proposal space. Second, based on the statistical analysis between the number of pseudo-unknown samples and the Intersection over Union (IoU), we propose an IoU-aware unknown objective, which sharps the unknown decision boundary by considering the localization quality. Third, to suppress the over-fitting problem and improve the model's generalization ability for unknown rejection, we propose the HSIC-based (Hilbert-Schmidt Independence Criterion) moving weight averaging to update the weights of classification and regression heads, which considers the degree of independence between the current weights and previous weights stored in the long-term memory banks. We compare our method with several state-of-the-art methods and observe that our method improves the mean recall of unknown classes by 12.87% across all shots in the VOC-COCO dataset settings. Our code is available at https://github.com/binyisu/food.

CCS CONCEPTS

• **Computing methodologies** → **Scene anomaly detection**.

KEYWORDS

few-shot open-set object detection, evidential deep learning, HSIC-based moving weight averaging

*Corresponding author.

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3611850

ACM Reference Format:
Binyi Su, Hua Zhang, and Zhong Zhou. 2023. HSIC-based Moving Weight Averaging for Few-Shot Open-Set Object Detection. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3581783.3611850

1 INTRODUCTION

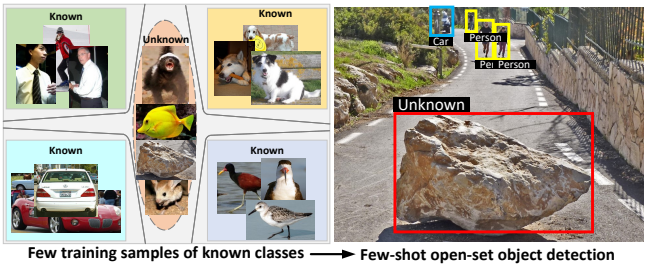

Figure 1: The few-shot open-set object detection aims to identify the known objects and reject the unknown objects based on limited training samples of known classes.

Object detection [11, 14, 26, 28, 34–36, 41–45, 47, 48, 50] is a fundamental computer vision task, which has achieved significant progress. However, modern object detectors often require a large number of annotated samples for training and develop with a close-set assumption, where the training and testing sets share the same classes. In real-world scenarios, these detectors quickly lose their efficiency when handling long-tail and unknown data. To tackle the above issues, few-shot open-set object detection (FOOD) [46] has been recently investigated, where the detector trained on few-shot close-set data is asked to detect all known objects and reject all unknown objects in open-set situations, as shown in Fig. 1.

Rejecting unknown objects with a limited number of training samples is indeed a challenging task, which needs to reject unknown objects without harming the detection accuracy of few-shot known classes. Recently, safe autonomous driving [12, 18, 53] has put forward higher requirements on FOOD, such as detecting hazards or anomalies in autonomous driving scenes. The FOOD problem refers to the challenge of training models with unbalanced datasets in real-world scenarios, where a subset of categories have fewer training samples than others. Furthermore, there are countless unknown classes that the model may encounter during inference that were not present in the training data. The challenge is to develop open-set detectors that can leverage the unbalanced training data to

accurately identify known classes while also rejecting unknown classes, which is an extremely difficult problem.

Previous method [46] leverages decoupling optimization and sparsification for few-shot unknown rejection, which lacks generalization and poses unsatisfactory performance for real-world applications. The few-shot training samples make the model easy to overfit the known classes and lose the generalization ability for unknown classes, and thus the model easily recognizes unknown objects as known classes. Therefore, obtaining a model with strong generalization is one of the keys to rejecting unknown classes. A well-known strategy to improve generalization is ensemble learning that averages the prediction results of many models. While it runs multiple models and increases the evaluation time, which cannot satisfy the requirement of real-time applications. Motivated by moving weight averaging (MWA) [2], which achieves state-of-the-art performance in out-of-distribution (OOD) generalization [32, 33], MWA averages the weights in a training trajectory and succeeds in robust prediction because it finds solutions with flatter loss landscapes [5], where it is defined as $\widehat{\theta} = (1-\alpha)\cdot\theta_{cur} + \alpha\cdot\overline{\theta}_{pre}$. Here, $\widehat{\theta}$, θ_{cur}, and $\overline{\theta}_{pre}$ express the updated weights, current weights, and previous average weights, respectively. α denotes a constant. However, MWA may not be optimal for all situations, as it assumes that the previous weights vary linearly over time (fixed constant α), which is not suitable for non-stationary data. One commonly observed issue is that the weight updates occur rapidly during the initial stages of training, while they become progressively slower towards the end. To solve this problem, we propose a new momentum-based weight averaging method to adaptively update the weight. By using Hilbert-Schmidt Independence Criterion (HSIC) to measure the independence between the current and previous weights, we can identify when the current weights are significantly different from the previous weights. This indicates that the model has encountered new and important data and needs to adapt its weights accordingly. HSIC provides a more flexible and adaptive approach to update the model's weights, allowing it to better adapt to weight distributions that change over time.

In this paper, we provide a new solution to solve the challenging FOOD problem in weight space. Specifically, we propose a moving weight averaging method based on Hilbert-Schmidt Independence Criterion (HSIC), which is used to average the weights obtained along a training trajectory. The HSIC function measures the degree of independence between the current and previous weights stored in the long-term memory bank, determining the direction of model updates in the form of momentum. Alongside, a challenging problem is that there is no real unknown data for training. Inspired by [46], we select pseudo-unknown samples with high uncertainty from the foreground and background proposals to regularize the predefined unknown branch. Instead of using energy score [13, 20, 46], we adopt the evidential uncertainty estimated by Dirichlet distribution of the output probability [3, 4] to select the pseudo-unknown samples in optimization. To improve the localization quality of unknown objects, we propose an innovative approach that involves an IoU-aware unknown training objective. This objective penalizes the model if there is a high Intersection over Union (IoU) between the predicted unknown object and the ground truth of known classes. In other words, if the model predicts an object as unknown, but it has a high IoU with the ground truth of known object, then the model is penalized. Experimental results show significant superiority of our method and indicate large room for improvement in this direction. Our main contribution is threefold:

- We propose a novel few-shot open-set object detector with the proposed HSIC-based moving weight averaging, which is verified to be effective for FOOD.
- We propose a new unknown sample mining approach based on evidential uncertainty estimation to mine the pseudo-unknown training data.
- We develop a novel IoU-aware unknown training objective, which effectively shapes the decision boundary between the known data and the mined pseudo-unknown data by considering the localization quality.

2 RELATED WORK

2.1 Few-Shot Open-Set Recognition

Few-Shot Open-Set Recognition (FSOSR) aims to quickly train a classifier based on a few examples while identifying all known classes and rejecting countless unknown classes in open-world scenes. Liu *et al.* [25] bench-marks the first FSOSR model, which modifies an existing meta-based few-shot learning framework for unknown recognition. On top of training the distance-based classifier, it adds an open-set loss term for pseudo-unknown samples, which are additionally sampled from the base data. Jeong *et al.* [19] proposes to solve FSOSR from the perspective of prototype transformation, which rejects samples by the distance from the transformed prototype. Pal *et al.* [31] utilizes a novel outlier calibration network to reject the unknown classes. Song *et al.* [39] selects the background region as the pseudo-unknown classes to train the classifier. Huang *et al.* [18] proposes task-adaptive negative class envisions for FSOSR to integrate threshold tuning into the learning process. However, few-shot open-set object detection is indeed a more challenging task than few-shot open-set recognition, because it involves not only identifying known and unknown object classes but also accurately localizing them in the image.

2.2 Few-Shot Open-Set Object Detection

Few-shot open-set object detection (FOOD) is an extension of FSOSR in object detection. Su *et al.* [46] bench-marks the first FOOD model, which is required not only to learn a discriminative detector to identify the pre-defined classes with few training samples but also to reject objects from unknown classes that never appear at training time. This method decouples training the known classes and unknown class, which assists the model to construct an unknown decision boundary and reject the unknown objects. Different from the previous method, we explore weight averaging in optimization to improve the model's generalization for unknown objects. By analyzing the degree of independence between the current weights and the previous weights, our method modifies the direction of model update, so that the model can better learn the generalization knowledge to reject unknown objects.

2.3 Moving Weight Averaging

Learning robust models that generalize well is critical for many real-world applications. Moving weight averaging (MWA) [2] averages the weights obtained along a training trajectory, which succeeds in out-of-distribution (OOD) detection [32], because it improves OOD generalization [5, 33, 38]. However, there is no work based on MWA to solve the challenging few-shot open-set detection task. We propose an HSIC-based moving weight averaging approach, which regularizes the model's generalization ability for unknown rejection in few-shot scenes and achieves a significant improvement.

2.4 Uncertainty Estimation

Estimating the uncertainty of model predictions is important for real-world applications. There are several uncertainty measurement methods, such as entropy, energy, and probability. Several works [8, 13, 20, 27, 46] adopt energy score to estimate the sample's uncertainty, where the energy is denoted as $-\log\sum\exp(logit)$. Several works [6, 13, 17] use the entropy score of the model predictions to choose the high-uncertainty samples, where the entropy is defined as $-\sum p\log(p)$. Some works [7, 16, 22, 49, 51] employ $\max(logit)$ or $\max(probability)$ to mine the high-uncertainty samples for unknown regularization. Differently, we are the first to use the Dirichlet distribution constructed by the output probability to estimate the proposal's uncertainty and select the high-uncertainty proposals as pseudo-unknown samples for optimization, which can assist the model to form a compact unknown decision boundary.

3 METHODOLOGY

3.1 Preliminary

The typical setup for FOOD follows the recent work [46]. We are given an object detection dataset $D = \{(x, y), x \in \mathbf{X}, y \in \mathbf{Y}\}$, where x denotes an input image and $y = \{(c_i, \widehat{b}_i)\}_{i=1}^{I}$ represents the objects with its class c and its box annotation $\widehat{b}$. The dataset D consists of the training set D_{tr} and the testing set D_{te}. D_{tr} contains K known classes $C_K = C_B \cup C_N = \{1, ..., K = B + N\}$, where $C_B = \{1, ..., B\}$ represents B base known classes, and $C_N = \{B + 1, ..., K\}$ expresses N novel known classes, each with M-shot support examples. In practice, the testing set D_{te} that includes $C_K = C_B \cup C_N$ known classes and C_U unknown classes is used to evaluate the detector. There is no overlap between the known labels C_K and unknown labels C_U. Due to the countless unknown categories, we merge all of them into one class $C_U = \{K + 1\}$ labeled as "unknown". Briefly speaking, we aim to employ the unbalanced or long-tail data split D_{tr} to train a detector in an open-set assumption, which can correctly classify $(K + 2)$ classes in total, including K known classes (base and novel classes), 1 unknown class, and 1 background class.

3.2 Baseline Setup

The proposed framework is illustrated in Fig. 2. We adopt Faster R-CNN [36] as the base detector consisting of a backbone, region proposal network (RPN), and R-CNN. Following the Evidential Deep Learning (EDL) [3, 4], we augment an unknown class of the classifier and mine the pseudo-unknown samples from the foreground and background proposals ranked by the evidential uncertainty, where a novel evidential deep learning loss L_{EDL} is used to optimize the model. Then the pseudo-unknown samples are used to regularize the proposed IoU-aware unknown loss L_U, which can assist the model to form a compact unknown decision boundary. During weight optimization, we adopt the proposed HSIC-based moving weight averaging to update the weights θ_{cls} and θ_{reg} of classification and regression heads in the form of momentum and develop a novel HSIC loss (L_{HSIC}) to regularize the model.

3.3 Pseudo-unknown sample mining

FOOD lacks real unknown data to train the model. Therefore, we need to generate the pseudo-unknown samples, which can help the model better generalize to unknown objects that it may encounter in real-world applications. The previous work [46] adopts the energy score, which ranks the proposal features and selects high-energy proposals as the unknown data. While the energy had its merits, it couldn't always capture the true essence of uncertainty and often suffered from limitations. We expect to develop a novel, efficient, and optimization-based pseudo-unknown sample mining method that can fit the true unknown distribution as closely as possible. Starting from this motivation, we innovatively redefine pseudo-unknown sample mining by the evidential uncertainty estimation, which employs the evidential uncertainty estimated by the Dirichlet distribution of probability to mine the pseudo-unknown samples from the proposal space. Our method is inspired by Evidential Deep Learning (EDL) [3], which has recently been introduced with the aim of utilizing the evidence framework of Dempster-Shafer theory [37] and subjective logic [21] to estimate uncertainty. It provides a structured and systematic means of formulating uncertainty modeling of input data, allowing for a more principled and reliable approach to uncertainty estimation. As shown in Fig. 2(a), we first assume the output probability P following the Dirichlet distribution $P \sim \mathrm{Dir}(P|\beta)$, and then estimate the evidential uncertainty (u) of each proposal b_i following the existing Evidential Deep Learning (EDL) [3]. The evidential uncertainty is formulated as:

$$u(b_i) = \frac{K+2}{\Delta(b_i)} = \frac{K+2}{\sum_{k=1}^{K+2}\beta_k(b_i)}, \tag{1}$$

where $K+2$ denotes the total number of classes including K known classes, 1 unknown class, and 1 background class. Based on Dempster-Shafer theory [37] and subjective logic [21], the class-wise strength β_k is linked to the learned evidence e_k by the equality $\beta_k = e_k + 1$, where e_k can be defined by $e_k = \exp(l_k)$, l_k represents the output logit of each class. For FOOD, high-uncertainty proposals can be regarded as pseudo-unknown samples, while low-uncertainty proposals are used to optimize the known classes. Here, we select top-k proposals ranked by the uncertainty criterion $u(b_i)$ from the foreground and background proposals as the pseudo-unknown samples, which can be used to regularize the virtual unknown class.

Following the previous Evidential Deep Learning, we directly predict β by deep neural networks. The model is trained by minimizing the following negative digamma function [1] of data. In particular, given a proposal b_i for $(K + 2)$-class classification, assuming that class probability follows a prior Dirichlet distribution, the FOOD model can be optimized for learning evidence:

$$L_{EDL} = \frac{1-\lambda_t}{N}\sum_{i=1}^{N}\sum_{k=1}^{K+2} c_{i,k}\left(\frac{d\ln(\Gamma(\Delta_i))}{d\Delta_i} - \frac{d\ln(\Gamma(\beta_k^i))}{d\beta_k^i}\right), \tag{2}$$

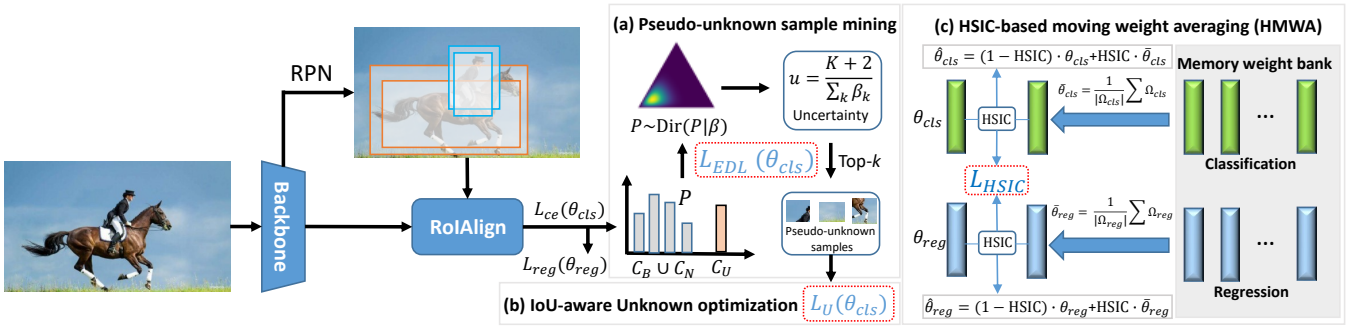

Figure 2: The framework of our proposed method. Our method is a two-stage detector with (a) pseudo-unknown sample mining, (b) IoU-aware unknown optimization, and (c) HSIC-based moving weight averaging. Pseudo-unknown sample mining selects unknown samples from the foreground and background proposals ranked by evidential uncertainty (u), which are formulated by a Dirichlet distribution of the prediction probability. The EDL loss L_{EDL} linking to evidential uncertainty (u) provides a principled and effective way to uncertainty modeling. The IoU-aware unknown optimization with the unknown loss L_U is proposed to regularize the unknown estimation by considering localization quality. The HSIC-based moving weight averaging with the HSIC loss L_{HSIC} is proposed to improve the model generalization ability for unknown rejection.

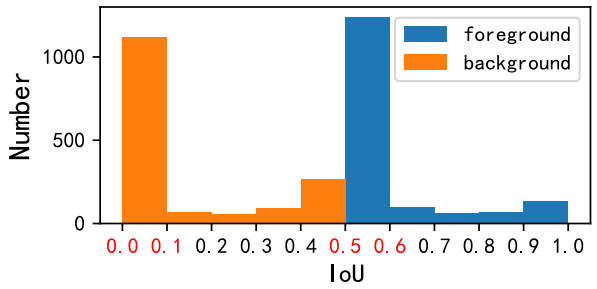

Figure 3: Mathematical statistic between the number of pseudo-unknown samples and IoU in training time. The IoU of pseudo-unknown samples from the background and foreground mainly falls in 0~0.1 and 0.5~0.6, respectively.

where N denotes the number of training samples, and $c_{i,k}$ is a binary element of the one-hot form of label c_i, and $\Gamma(\cdot)$ is the gamma function. Moreover, $\Delta_i = \sum_k \beta_k^i$ is the total Dirichlet strength over $K+2$ classes. $\lambda_t = n_c \exp\{-(\ln n_c/T)t\} \in [n_c, 1]$ denotes the annealing weighting factor, where $n_c \ll 1$ is a small positive constant, $t \leqslant T$ is the training iteration. The motivation of λ_t is that at the beginning of the training, the inaccurate uncertainty estimations of Eq. 1 are high-frequency cases so the EDL loss should be optimized more, while at the end of the training, the accurate estimations are dominant, thus the EDL loss should be tamer. The EDL loss function is a principled and effective way to model uncertainty in deep learning models. By linking to evidential uncertainty (u), it provides a measure of how uncertain the model is about a given prediction. The differences between our EDL loss and the previous [3] are that we add the annealing weighing factor and propose the digamma function for loss optimization instead of the logarithm function [3], because the digamma function makes the optimization of EDL loss smoother (Fig. 6(a)), where the digamma function is denoted as the logarithmic derivative of the gamma function [1].

3.4 IoU-aware unknown optimization

There are no explicit boundaries to separate known and unknown objects. Previous works [13, 30, 46, 55] lack the constraints of IoU, leading to the misidentification of known objects as unknown objects. Intuitively, a proposal that overlaps more regions with the ground truth location should have a lower unknown probability. This is because the greater the overlap, the more likely it is that the proposal actually contains the known object of interest, which should keep low uncertainty. Therefore, when optimizing a FOOD model, it is important to consider both the overlap between the pseudo-unknown proposal b_i and the ground truth location $\widehat{b}_i$, as well as the unknown probability P_U. Here, in order to accurately separate the unknown objects, we propose an IoU-aware unknown objective by considering localization quality:

$$L_U = -\frac{1}{N} \sum_{i=1}^{N} w_{b_i, \widehat{b}_i} \log P_U^i, \tag{3}$$

$$w_{b_i, \widehat{b}_i} = \begin{cases} 1 - \frac{|b_i \cap \widehat{b}_i|}{|b_i \cup \widehat{b}_i|}, & \text{unknown foreground} \\ 0.5 - \max\left(\frac{|b_i \cap \widehat{b}_i|}{|b_i \cup \widehat{b}_i|}, \lambda\right), & \text{unknwon background} \end{cases}, \tag{4}$$

where N represents the total number of pseudo-unknown training samples and $P_U = \exp(l_U)/(\sum_{k=1}^{K+2} \exp(l_k) - \exp(l_{gt}))$ is denoted as the predicted unknown probability, which is defined as a softmax probability without the logit of ground truth class l_{gt} [13]. Because there is no supervision for the unknown optimization, the above unknown probability can reduce the impact of optimizing unknown classes on known classes [13] and l_U expresses the unknown logit. λ is a small non-negative constant. The optimization process of L_U is that if the unknown proposal is selected from the foreground or background, a high unknown probability should correspond to a low IoU score. Note that for Faster R-CNN, the proposal with IoU≥0.5 is classified into the foreground, oppositely, the proposal with 0.5>IoU≥0 is classified into the background. The fact is that

when the IoU of the foreground and background is close or equal to 0.5 and 0, respectively, the proposal is considered to be more uncertain, as illustrated in Fig. 3. The mathematical statistic shows that the IoU of pseudo-unknown samples from the background and foreground mainly falls in 0∼0.1 and 0.5∼0.6, respectively. This is consistent with the optimization process of the model, as the model tries to find high-uncertainty samples with low IoU scores. For the unknown foreground, $w_{b_i,\widehat{b}_i}$ in Eq. 3 is pushed towards 0.5, and for the unknown background, $w_{b_i,\widehat{b}_i}$ is pushed towards λ (λ is a small constant). Therefore, it is essential to understand the impact of IoU on the proposal's uncertainty in the FOOD task. This knowledge can help us design better objectives by considering the IoU between the pseudo-unknown samples and the ground truths of known classes. Note that the constants 1 and 0.5 in $w_{b_i,\widehat{b}_i}$ are used to balance and positivize the IoU-aware weights of foreground and background.

3.5 HSIC-based moving weight averaging

Hilbert-Schmidt Independence Criterion (HSIC). HSIC is a widely used measure of independence between two high-dimensional random variables. It is particularly useful when dealing with data that has a large number of features or dimensions. In practice, we employ the unbiased HSIC estimator in [40] with n samples to measure the independence between variables U and V:

$$\mathrm{HSIC}^{k,l}(U,V) = \frac{1}{n(n-3)}\left[tr(\widetilde{U}\,\widetilde{V}^T) + \frac{\mathbf{1}^{\mathrm{T}}\widetilde{U}\mathbf{1}\mathbf{1}^{\mathrm{T}}\widetilde{V}\mathbf{1}}{(n-1)(n-2)} - \frac{2}{n-2}\mathbf{1}^T\widetilde{U}\,\widetilde{V}^T\mathbf{1}\right], \tag{5}$$

where $\widetilde{U}$ is the kernelized matrix of U with radial basis function (RBF) kernel k by $\widetilde{U}_{ij} = (1-\xi_{i,j})k(u_i,u_j)$, $\{u_i\} \sim U$ and the $(1-\xi_{ij})$ sets the diagonal of $\widetilde{U}$ to zeros. $\widetilde{V}$ is defined similarly with kernel l, tr expresses the trace of a matrix, and $\mathbf{1}$ is an all-one vector. Note that the HSIC value is equal to zero if and only if the two variables are independent.

In the context of FOOD tasks, there is a lack of real unknown data and only a limited number of training samples are available, it is highly likely that the trained model will suffer from overfitting issue and have poor generalization ability for unknown classes. This is because the model will likely memorize the few training data instead of learning generalizable patterns, leading to poor performance on new data. Recently, moving weight averaging (MWA) [2] has achieved state-of-the-art performance in out-of-distribution detection [32], which can improve the model generalization ability for unknown rejection. Our proposed method differs from the commonly used MWA method in that we use an HSIC-based approach to update the model's weights. Specifically, we measure the degree of independence between the current weights and the previous weights stored in the long-term memory bank and use this measure to update the model's weights. HSIC, or Hilbert-Schmidt Independence Criterion, is a statistical measure of the independence between two random variables. By using HSIC to measure the independence between the current and previous weights, we can identify when the current weights are significantly different from the previous weights. This indicates the model has encountered new and important data, prompting necessary weight updating. In contrast, MWA updates the model's weights by taking a weighted average of the current weights and the previous weights. This approach may not be optimal for all situations, as it assumes that the previous weights vary linearly over time, which is not suitable for non-stationary data. Differently, our HSIC-based moving weight averaging (HMWA) method provides a more flexible and adaptive approach to update the model's weights, allowing it to better adapt to the distribution of weight data over time.

As shown in Fig. 2(c), our HMWA includes two memory banks and an HSIC loss (L_{HSIC}). Specifically, we initialize the memory weight banks $\Omega_{cls,reg}$ of the classification and regression heads with size Q, respectively. In a training trajectory, we sample the classification and regression weights every S iterations and store them in the long-term memory banks $\Omega_{cls,reg}$. Then we calculate the averaging weights $\overline{\theta}_{cls,reg} = \sum \Omega_{cls,reg}/|\Omega_{cls,reg}|$ of the memory bank and adopt HSIC to measure the degree of independence between the current weights $\theta_{cls,reg}$ and the averaging weights $\overline{\theta}_{cls,reg}$, which can be expressed as $h_{cls,reg} = \mathrm{HSIC}(\theta_{cls,reg}, \overline{\theta}_{cls,reg})$. The weights of the current iteration are updated by considering the independence with the averaging weights $\overline{\theta}_{cls,reg}$ of the memory bank:

$$\widehat{\theta}_{cls} = (1-h_{cls})\cdot\theta_{cls} + h_{cls}\cdot\overline{\theta}_{cls}, \tag{6}$$

$$\widehat{\theta}_{reg} = (1-h_{reg})\cdot\theta_{reg} + h_{reg}\cdot\overline{\theta}_{reg}, \tag{7}$$

where $\widehat{\theta}_{cls,reg}$ denotes the updated weights. We repeat the above process every S iterations where the oldest head weights are out of the memory and the newest into the queue. In particular, important samples bring more model weight updates (small $h_{cls,reg}$), so the current weights account for a large in Eq. 6 and Eq. 7. HSIC encourages the model to smoothly focus on important samples, which play a positive role in weight updating. Simultaneously, we also propose an HSIC loss to learn more generalized weights of the prediction heads. HSIC loss can guide the model to find the flat minima, and flatten the loss landscapes, which makes the model converge well.

$$L_{HSIC} = \mathbb{E}_{\theta_{cls}}\left[1-\mathrm{HSIC}(\theta_{cls},\overline{\theta}_{cls})\right] + \mathbb{E}_{\theta_{reg}}\left[1-\mathrm{HSIC}(\theta_{reg},\overline{\theta}_{reg})\right]. \tag{8}$$

Note that we start optimizing the HSIC loss when the first set of sampled weights is stored in the memory banks. Alongside, a simple decay weight $\alpha = 1 - t/T$ is adopted during HSIC loss optimization.

3.6 Training and inference

Our method can be trained by minimizing the following weighted sum of losses in an end-to-end manner:

$$L = L_{rpn} + L_{ce} + L_{reg} + \lambda_1 L_{EDL} + \lambda_2 L_U + \lambda_3 L_{HSIC}, \tag{9}$$

where L_{rpn} denotes the objective function of RPN, L_{ce} and L_{reg} represent the classification and regression losses of R-CNN. λ_1, λ_2, and λ_3 are the weighting coefficients.

In inference, the input in an open-set assumption is fed into the FOOD model, and our method can produce locations b_i and directly predict classification labels $\widehat{y}_i = \arg\max_{j\in[1,\ldots,K+2]}(P_{ij})$ in a threshold-free way, which means that recognizing known objects and rejecting unknown objects depends entirely on the arguments of the maxima (arg max) of predicted probabilities P of all classes.

(a) VOC10-5-5	$mAP_K/mAP_N\uparrow$					$R_U\uparrow$					$AR_U\uparrow$				
Method	1	3	5	10	Mean	1	3	5	10	Mean	1	3	5	10	Mean
DS [30]	43.82/7.22	46.89/14.48	48.01/19.27	48.01/25.66	46.68/16.65	23.99	23.62	19.99	19.99	21.90	12.15	11.98	10.80	10.83	11.44
PROSER [55]	41.64/8.49	43.30/15.16	45.12/20.08	48.35/25.13	44.60/17.22	30.95	32.30	32.68	32.61	32.14	15.41	16.17	16.48	17.01	16.27
OpenDet [13]	43.45/8.27	46.47/14.09	47.56/17.90	50.95/25.14	47.11/16.35	33.64	30.62	32.13	36.30	33.17	17.28	15.89	16.72	18.89	17.20
FOOD [46]	43.97/8.95	48.48/16.83	50.18/23.10	53.23/28.60	48.97/19.37	43.72	44.52	45.65	45.84	44.93	23.51	23.58	23.61	23.86	23.64
Ours	**45.12/11.56**	**48.90/18.96**	**52.55/27.31**	**57.24/32.63**	**50.95/22.62**	**60.03**	**61.21**	**62.02**	**62.14**	**61.35**	**31.19**	**32.03**	**32.79**	**32.80**	**32.20**
(b) VOC-COCO	$mAP_K/mAP_N\uparrow$					$R_U\uparrow$					$AR_U\uparrow$				
Method	1	5	10	30	Mean	1	5	10	30	Mean	1	5	10	30	Mean
DS [30]	15.47/2.11	17.10/6.30	19.06/9.46	23.40/15.27	18.76/8.29	3.57	3.86	3.75	3.95	3.78	1.69	1.71	1.77	1.83	1.75
PROSER [55]	13.58/2.32	15.67/6.40	17.00/8.75	21.44/14.30	16.92/7.94	7.53	9.59	10.06	12.06	9.81	3.07	4.08	4.89	5.98	4.51
OpenDet [13]	16.01/2.51	17.16/7.19	18.53/8.62	22.93/14.02	18.66/8.09	7.24	11.49	13.89	18.07	12.67	3.14	5.21	6.32	8.76	5.86
FOOD [46]	15.83/2.26	18.08/6.69	20.17/9.35	23.9/14.47	19.5/8.19	15.76	20.02	21.48	23.17	20.11	7.20	9.45	9.56	11.45	9.42
Ours	**18.54/4.33**	**19.88/11.95**	**22.64/13.82**	**23.71/17.67**	**21.19/11.94**	**30.87**	**32.53**	**32.78**	**35.74**	**32.98**	**14.13**	**15.74**	**16.52**	**17.26**	**15.91**

Table 1: The few-shot open-set object detection results on (a) VOC10-5-5 and (b) VOC-COCO dataset settings. For a fair comparison, we report the average results of 10 random runs with the same backbone (Resnet50) for all comparison methods.

4 EXPERIMENT

4.1 Datasets

We follow the recent work [46] and use the same data split such as VOC10-5-5 and VOC-COCO to evaluate our method for a fair comparison. As for **VOC10-5-5**, we divide 20 classes into 10 base classes, 5 novel classes, and 5 unknown classes in PASCAL VOC [29]. Each novel class has 1, 3, 5, and 10 objects sampled from the train and validation sets of VOC07 and VOC12. The test set of VOC07 is selected as the testing data. As for **VOC-COCO**, we use the train and validation sets of PASCAL VOC as the known base training data. We select 20 categories disjoint with the 20 VOC classes as the novel classes in the train set of MS COCO2017 [23]. Each novel class has 1, 5, 10, and 30 objects sampled from the train sets of COCO2017. The remaining 40 classes are chosen as the unknown classes. The val2017 set is used as the testing data. More details are shown in the supplementary material.

4.2 Evaluation Metrics

The mean average precision (mAP) of known (mAP_K) and novel (mAP_N) classes is chosen to evaluate the known object detection performance. To evaluate the unknown detection performance, the recall (R_U) and average recall (AR_U) are reported. The unknown recall (R_U) is a popular metric, which is the ratio of well-found objects whose IoU with ground truth is higher than the threshold of 0.5. AR_U is the average recall at IoU thresholds from 0.5 to 0.95 with a 0.05 interval, which is a fairer metric for unknown evaluation.

4.3 Implementation Details

Similar to most open-set detection methods [13, 30, 46], ImageNet pre-trained Resnet50 [15] is used to initialize the backbone. We adopt a two-stage fine-tuning strategy [52] to train the few-shot open-set detector. In the base training stage, we employ the abundant samples of the base classes C_B to train the entire base detector from scratch, such as Faster R-CNN. Then, in the few-shot fine-tuning stage, a small balanced training set from base and novel classes (C_B+C_N) is used to fine-tune the model. Simultaneously, we scale the gradient from R-CNN and stop the gradient from RPN [36] to slowly update the parameters of the backbone network to get the few-shot open-set object detector. In the fine-tuning stage, L_{EDL}, L_U, and L_{HSIC} are optimized. Noting that the above three losses are not optimized in the base training phase. All models are trained using SGD optimizer with a mini-batch size of 16, a momentum of 0.9, and a weight decay of 1e-4. The learning rate of 0.02 is used in the first stage and 0.01 in the second stage. The coefficients n_c, λ, and τ are 0.01, 0.0001, and 0.05, respectively. The queue size Q of the memory bank is 32 and the sampling step S is 10 iterations.

4.4 Comparison Results

VOC10-5-5. As illustrated in Table 1**(a)**, we present the evaluation results on VOC10-5-5 and conduct a comparison of our performance with other state-of-the-art results obtained by DS [30], PROSER [55], OpenDet [13], and FOOD [46] with the same Resnet50 backbone. We choose 1, 3, 5, and 10 samples of each known class to train the open-set detectors in the two-stage fine-tuning way [52]. We can see that our method outperforms previous methods by a large margin for unknown rejection. For example, the unknown mean recall (R_U) reaches 61.35%, which outperforms the second best by 16.42%. The unknown average recall AR_U is a more fair metric, while our method outperforms the second best by 8.56%. The mAP of the known classes is also competitive. In particular, our mAP_K increases by 1.98% for the mean result of 1, 3, 5, and 10 shots. It demonstrates that our method not only has a strong unknown generalization ability with limited training samples but also performs well at identifying known classes.

VOC-COCO. As shown in Table 1**(b)**, we evaluate our method using the more challenging cross-dataset setting (VOC-COCO), where 40 classes of COCO are defined as the unknown classes. Our method achieves 12.87% (R_U) and 6.49% (AR_U) improvements over the second-best method (FOOD [46]), respectively. The highest recall of our method is 35.74% (30-shot), which means that one-third of unknown objects are recalled in the challenging cross-dataset setting, which demonstrates that our method exhibits a strong generalization for unknown classes. The knowledge learned by our method from a few samples and large categories is more generalized than the close-set detectors. Alongside, our method preserves a high accuracy on the original in-distribution task (measured by mAP_K and mAP_N) compared with other state-of-the-art methods.

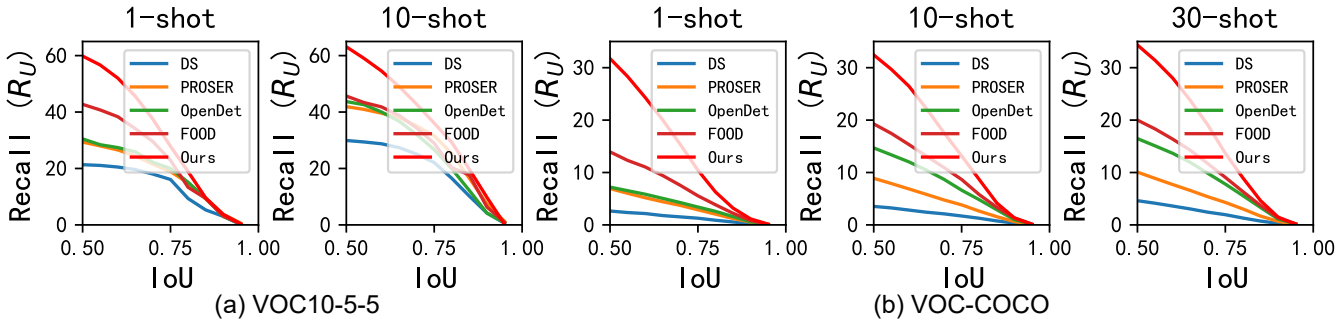

Figure 4: Unknown recall (R_U) versus IoU threshold on VOC10-5-5 (1 and 10-shot) and VOC-COCO (1, 10, and 30-shot). AR is twice the area enclosed by the recall-IoU curve.

Figure 5: Visualization of detected objects on the open-set images (from VOC, COCO, and RoadAnomaly [24]) by the FOOD [46] (*Top*) and our method (*bottom*). The inference model of RoadAnomaly is trained by a 1-shot VOC-COCO setting.

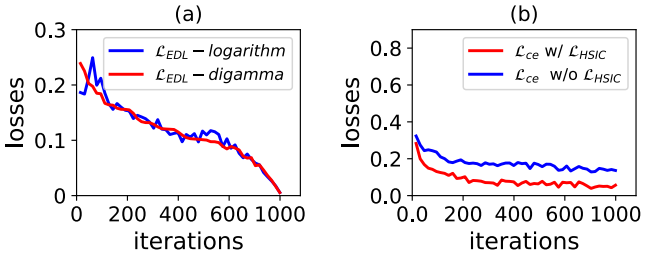

Figure 6: Visualization of the loss functions in 1-shot VOC-COCO setting. (a) Ablation studies of L_{EDL} with the logarithm and digamma functions. (b) Ablation studies of L_{ce} with/without L_{HSIC}.

Overall. For few-shot detection, our method achieves 3.25% (mAP_N) and 3.65% (mAP_N) improvements over the second-best method (mean results) in VOC10-5-5 and VOC-COCO dataset settings, respectively. The reason is that HMWA suppresses the overfitting issue caused by few training data, and then improves the general representation ability for few-shot classes. Focusing on suppressing overfitting problems and improving the unknown generalization is beneficial for dealing with the few-shot problems and open-set problems, simultaneously.

As shown in Fig. 4, we present the recall-IoU curves on VOC10-5-5 and VOC-COCO of different shot settings. They show that the recall-IoU performance varies significantly both across shots and IoU thresholds, and our proposed method could consistently outperform other state-of-the-art methods on all two benchmarks and ten thresholds. Fig. 5 visualizes the results of FOOD (top) [46] and our method (bottom) on VOC, COCO, and RoadAnomaly [24] datasets. Note that the inference model of RoadAnomaly is trained by a 1-shot VOC-COCO setting. It can be seen that FOOD misses many unknown objects such as sheep in VOC (the 2^{nd} column), orange in COCO (the 3^{rd} column), and stone in RoadAnomaly (the 7^{th} column). While our method does not miss any unknown objects and has high confidence scores because we learn more generalizable representations for unknown rejection. More qualitative and quantitative analyzes are shown in the supplementary material.

4.5 Ablation Study

4.5.1 Pseudo-unknown sample mining. As illustrated in Table 2(a), we carefully study the designs of different pseudo-unknown sample mining methods. A good sampling method can not only improve the performance of known classes but also boost the rejection ability of unknown classes. Our evidential uncertainty outperforms the known and unknown results of other methods including random selection, max logit [16], min score [13], maximum softmax probability (MSP) [49], entropy [17], energy [27], and conditional energy [46], which demonstrates its effectiveness. Table 2 (b) presents the results of only mining pseudo-unknown samples from foreground or background proposals, the performance of which is not as effective as mining samples from both of them. We also analyze the effect of logarithm-based, digamma-based, or without L_{EDL}, the

(a) Mining methods	mAP_K	R_U	AR_U
Random	17.32	21.73	10.14
Max logit [16]	17.79	28.77	12.62
Min score [13]	17.98	26.27	11.84
Maximum softmax probability [49]	18.09	26.18	11.51
Entropy [17]	17.39	25.83	11.80
Energy [27]	17.20	28.00	11.87
Conditional energy [46]	17.95	28.73	12.43
Our evidential uncertainty	**18.54**	**30.87**	**14.13**
(b) w/o foreground	17.01	27.89	12.03
(b) w/o background	17.88	28.95	13.48
(c) Logarithm-based L_{EDL}	18.51	30.29	13.97
(c) Digamma-based L_{EDL}	**18.54**	**30.87**	**14.13**
(c) w/o L_{EDL}	17.23	28.28	13.07

Table 2: Results of pseudo-unknown sample mining in 1-shot VOC-COCO settings. (a) Different mining methods. (b) Without foreground or background. (c) Logarithm-based, digamma-based, or without L_{EDL}.

(a) $w_{b_i,\widehat{b}_i}$	mAP_K	R_U	AR_U
$\lvert b_i \cap \widehat{b}_i\rvert / \lvert b_i \cup \widehat{b}_i\rvert$	18.01	27.96	11.46
$\max\left(\lvert b_i \cap \widehat{b}_i\rvert / \lvert b_i \cup \widehat{b}_i\rvert,\ \lambda\right)$	18.29	28.39	11.93
$1 - \lvert b_i \cap \widehat{b}_i\rvert / \lvert b_i \cup \widehat{b}_i\rvert$	18.08	28.47	11.99
$0.5 - \lvert b_i \cap \widehat{b}_i\rvert / \lvert b_i \cup \widehat{b}_i\rvert$	12.87	0	0
$1 - \max\left(\lvert b_i \cap \widehat{b}_i\rvert / \lvert b_i \cup \widehat{b}_i\rvert,\ \lambda\right)$	18.32	29.56	13.03
$0.5 - \max\left(\lvert b_i \cap \widehat{b}_i\rvert / \lvert b_i \cup \widehat{b}_i\rvert,\ \lambda\right)$	12.04	0	0
Ours (Eq. 4)	**18.54**	**30.87**	**14.13**
(b) w/o $w_{b_i,\widehat{b}_i}$	18.05	27.91	11.39
(c) w/o L_U	**19.17**	0	0

Table 3: Results of IoU-aware unknown optimization in 1-shot VOC-COCO settings. (a) Different IoU-based weights. (b) Without weight $w_{b_i,\widehat{b}_i}$. (c) Without L_U.

digamma-based L_{EDL} outperforms other settings by 0.03%∼1.31% (mAP_K) and 0.16%∼1.06% (AR_U). Simultaneously, the qualitative comparison between the logarithm and digamma is shown in Fig. 6(a). We can see that our digamma-based EDL loss is smoother than the logarithm, which illustrates its advantage.

4.5.2 IoU-aware unknown optimization. We first explore the different IoU-based weights ($w_{b_i,\widehat{b}_i}$). Table. 3(a) line 1-6 set the same $w_{b_i,\widehat{b}_i}$ of pseudo-unknown samples from the foreground and background. As illustrated in Table. 3(a), improper settings can lead to a significant drop in performance for both known and unknown classes, for example, $(0.5 - \lvert b_i \cap \widehat{b}_i\rvert / \lvert b_i \cup \widehat{b}_i\rvert)$, which causes the $w_{b_i,\widehat{b}_i}$ of foreground (IoU>0.5) is negative and the loss (L_U) swings between positive and negative, thus it is difficult to converge. Therefore, the unknown results are zero. According to the mathematical statistic between the number of pseudo-unknown samples and IoU (Fig. 3), our method (Eq. 4) designs more appropriate IoU-based weights, effectively improving the localization quality of unknown classes. Table. 3(b) and (c) present the ablation studies of without $w_{b_i,\widehat{b}_i}$ and L_U, respectively. It shows that $w_{b_i,\widehat{b}_i}$ can significantly boost the localization performance of unknown objects (AR_U: 11.39%⟶14.13%) and L_U is necessary for unknown rejection.

(a) Averaging methods	mAP_K	R_U	AR_U
Weight averaging [5]	**18.58**	27.87	11.96
Diverse weight averaging [33]	18.37	28.19	12.08
Moving weight averaging [2]	18.05	28.59	12.32
Our HMWA	18.54	**30.87**	**14.13**
(b) w/o HMWA	18.08	25.63	10.91
(c) w/o L_{HSIC}	18.28	30.10	13.78

Table 4: Results of WA in 1-shot VOC-COCO settings. (a) Different WA methods. (b) Without HMWA. (c) Without L_{HSIC}.

4.5.3 HSIC-based moving weight averaging. We improve the generalization of model to unknown classes by weight averaging [5]. As listed in Table 4(a), our HSIC-based moving weight averaging (HMWA) outperforms other averaging methods [2, 5, 33]) for unknown rejection, which verifies that considering the dependency between current and previous weights is helpful for unknown generalization. Our method provides a more flexible and adaptive way to update the model's weights of the prediction heads, making it to better adapt to the data distributions of the weights over time. When we remove HMWA (Table 4(b)), the unknown results drop a lot (AR_U: 14.13%⟶10.91%), which illustrates its necessity. Alongside, Table 4(c) presents the effectiveness of L_{HSIC}. We also show the ablation studies of cross-entropy-based classification loss (L_{ce}) with/without L_{HSIC}. As shown in Fig. 6(b), L_{ce} with L_{HSIC} is closer to zero as the iterations increase, which demonstrates that the HSIC loss guides the model to find flatter minima.

5 CONCLUSION

In this paper, we propose a new solution to solve the challenging FOOD problem. Specifically, we propose a few-shot open-set detector, which is a novel unknown-aware training framework for unknown rejection. Since there is no real unknown data, the evidential uncertainty estimated by the Dirichlet distribution of the output probability is used to mine the pseudo-unknown data in optimization. We also propose an IoU-aware unknown training objective, which meaningfully regularizes the unknown estimation by considering localization quality. Furthermore, the HSIC function measures the degree of independence between the current and previous weights, determining the direction of model updates. Extensive experiments on three dataset settings show that our method significantly outperforms the state-of-the-art methods.

6 ACKNOWLEDGMENTS

The National Key R&D Program of China under Grant (2022ZD0118102); in part by the National Natural Science Foundation of China under Grant 62272018 and 62072454.

REFERENCES

[1] Horst Alzer and Graham Jameson. 2017. A harmonic mean inequality for the digamma function and related results. *Rendiconti del Seminario Matematico della Università di Padova* 137 (2017), 203–209.

[2] Devansh Arpit, Huan Wang, Yingbo Zhou, and Caiming Xiong. 2022. Ensemble of Averages: Improving Model Selection and Boosting Performance in Domain Generalization. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.). Curran Associates, Inc., 8265–8277.

[3] Wentao Bao, Qi Yu, and Yu Kong. 2021. Evidential Deep Learning for Open Set Action Recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 13349–13358.

[4] Wentao Bao, Qi Yu, and Yu Kong. 2022. OpenTAL: Towards Open Set Temporal Action Localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2979–2989.

[5] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. 2021. SWAD: Domain Generalization by Seeking Flat Minima. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.). Curran Associates, Inc., 22405–22418.

[6] Robin Chan, Matthias Rottmann, and Hanno Gottschalk. 2021. Entropy Maximization and Meta Classification for Out-of-Distribution Detection in Semantic Segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 5128–5137.

[7] Hendrycks Dan and Gimpel Kevin. 2017. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In *Proceedings of the International Conference on Learning Representation*.

[8] Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. 2022. VOS: Learning What You Don't Know by Virtual Outlier Synthesis. In *Proc. Int. Conf. Learn. Represent. (ICLR)*.

[9] Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. 2022. Learning To Prompt for Open-Vocabulary Object Detection With Vision-Language Model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 14084–14093.

[10] Chengjian Feng, Yujie Zhong, Zequn Jie, Xiangxiang Chu, Haibing Ren, Xiaolin Wei, Weidi Xie, and Lin Ma. 2022. PromptDet: Towards Open-Vocabulary Detection Using Uncurated Images. In *Computer Vision – ECCV 2022*, Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (Eds.). Springer Nature Switzerland, Cham, 701–717.

[11] Ross Girshick. 2015. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[12] Torben Hagerup, Kurt Mehlhorn, and J. Ian Munro. 2023. Unknown Sniffer for Object Detection: Don't Turn a Blind Eye to Unknown Objects. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

[13] Jiaming Han, Yuqiang Ren, Jian Ding, Xingjia Pan, Ke Yan, and Guisong Xia. 2022. Expanding Low-Density Latent Regions for Open-Set Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[14] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. 2017. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[16] Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joseph Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. 2022. Scaling Out-of-Distribution Detection for Real-World Settings. In *Proceedings of the 39th International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (Eds.), Vol. 162. PMLR, 8759–8773.

[17] Alex Holub, Pietro Perona, and Michael C. Burl. 2008. Entropy-based active learning for object recognition. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 1–8. https://doi.org/10.1109/CVPRW.2008.4563068

[18] Shiyuan Huang, Jiawei Ma, Guangxing Han, and Shih-Fu Chang. 2022. Task-Adaptive Negative Envision for Few-Shot Open-Set Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7171–7180.

[19] Minki Jeong, Seokeon Choi, and Changick Kim. 2021. Few-shot Open-set Recognition by Transformation Consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 12561–12570.

[20] K J Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. 2021. Towards Open World Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5830–5840.

[21] Audun Jøsang. 2016. *Subjective Logic*. Springer.

[22] Shiyu Liang, Yixuan Li, and R. Srikant. 2018. Enhancing the reliability of out-of-distribution image detection in neural networks. In *Proceedings of the International Conference on Learning Representation*.

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft COCO: Common Objects in Context. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*.

[24] Krzysztof Lis, Krishna Nakka, Pascal Fua, and Mathieu Salzmann. 2019. Detecting the Unexpected via Image Resynthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[25] Bo Liu, Hao Kang, Haoxiang Li, Gang Hua, and Nuno Vasconcelos. 2020. Few-Shot Open-Set Recognition Using Meta-Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8795–8804. https://doi.org/10.1109/CVPR42600.2020.00882

[26] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. 2016. SSD: Single Shot MultiBox Detector. In *Computer Vision – ECCV 2016*, Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (Eds.). Springer International Publishing, Cham, 21–37.

[27] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. 2020. Energy-based Out-of-distribution Detection. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 21464–21475. https://proceedings.neurips.cc/paper_files/paper/2020/file/f5496252609c43eb8a3d147ab9b9c006-Paper.pdf

[28] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 10012–10022.

[29] Everingham Mark, Van Luc, and Williams Christopher. 2010. The PASCAL Visual Object Classes (VOC) Challenge. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*.

[30] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sunderhauf. 2018. Dropout sampling for robust object detection in open-set conditions. In *Proceedings of the IEEE Int. Conf. Robot. Autom. (ICRA)*. 3243–3249.

[31] Debabrata Pal, Valay Bundele, Renuka Sharma, Biplab Banerjee, and Yogananda Jeppu. 2022. Few-Shot Open-Set Recognition of Hyperspectral Images with Outlier Calibration Network. In *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*. 3801–3810.

[32] Alexandre Rame, Corentin Dancette, and Matthieu Cord. 2022. Fishr: Invariant Gradient Variances for Out-of-Distribution Generalization. In *Proceedings of the 39th International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (Eds.), Vol. 162. 18347–18377.

[33] Alexandre Ramé, Matthieu Kirchmeyer, Thibaud Rahier, Alain Rakotomamonjy, Patrick Gallinari, and Matthieu Cord. 2023. Diverse Weight Averaging for Out-of-Distribution Generalization. In *Proceedings of the Neural Information Processing Systems*.

[34] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You Only Look Once: Unified, Real-Time Object Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[35] Joseph Redmon and Ali Farhadi. 2017. YOLO9000: Better, Faster, Stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[36] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (Eds.), Vol. 28. Curran Associates, Inc.

[37] KARI SENTZ and SCOTT FERSON. [n.d.]. Combination of Evidence in Dempster-Shafer Theory. ([n. d.]). https://doi.org/10.2172/800792

[38] Ricardo Silva, Amir Globerson, and Amir Globerson. 2018. Averaging weights leads to wider optima and better generalization. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018 (34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018)*. Association For Uncertainty in Artificial Intelligence (AUAI), 876–885.

[39] Nan Song, Chi Zhang, and Guosheng Lin. 2022. Few-Shot Open-Set Recognition Using Background as Unknowns. In *Proceedings of the 30th ACM International Conference on Multimedia* (Lisboa, Portugal) *(MM '22)*. Association for Computing Machinery, New York, NY, USA, 5970–5979. https://doi.org/10.1145/3503161.3547933

[40] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. arXiv:1212.0402 [cs.CV]

[41] Binyi Su, Haiyong Chen, Peng Chen, Guibin Bian, Kun Liu, and Weipeng Liu. 2020. Deep learning-based solar-cell manufacturing defect detection with complementary attention network. *IEEE Transactions on Industrial Informatics* 17, 6 (2020), 4084–4095.

[42] Binyi Su, Haiyong Chen, Kun Liu, and Weipeng Liu. 2021. RCAG-Net: Residual Channelwise Attention Gate Network for Hot Spot Defect Detection of Photovoltaic Farms. *IEEE Transactions on Instrumentation and Measurement* 70 (2021), 1–14. https://doi.org/10.1109/TIM.2021.3054415

[43] Binyi Su, Haiyong Chen, and Zhong Zhou. 2021. BAF-detector: An efficient CNN-based detector for photovoltaic cell defect detection. *IEEE Transactions on Industrial Electronics* 69, 3 (2021), 3161–3171.

[44] Binyi Su, Haiyong Chen, Yifan Zhu, Weipeng Liu, and Kun Liu. 2019. Classification of Manufacturing Defects in Multicrystalline Solar Cells With Novel Feature Descriptor. *IEEE Transactions on Instrumentation and Measurement* 68, 12 (2019), 4675–4688. https://doi.org/10.1109/TIM.2019.2900961

[45] Binyi Su, Lei Yu, and Wen Yang. 2020. Event-Based High Frame-Rate Video Reconstruction With A Novel Cycle-Event Network. In *2020 IEEE International Conference on Image Processing (ICIP)*. 86–90. https://doi.org/10.1109/ICIP40778.2020.9191114

[46] Binyi Su, Hua Zhang, Jingzhi Li, and Zhong Zhou. 2022. Towards Few-Shot Open-Set Object Detection. arXiv:2210.15996 [cs.CV]

[47] Binyi Su, Hua Zhang, Zhaohui Wu, and Zhong Zhou. 2022. FSRDD: An Efficient Few-Shot Detector for Rare City Road Damage Detection. *IEEE Transactions on Intelligent Transportation Systems* 23, 12 (2022), 24379–24388. https://doi.org/10.1109/TITS.2022.3208188

[48] Binyi Su, Zhong Zhou, and Haiyong Chen. 2023. PVEL-AD: A Large-Scale Open-World Dataset for Photovoltaic Cell Anomaly Detection. *IEEE Transactions on Industrial Informatics* 19, 1 (2023), 404–413. https://doi.org/10.1109/TII.2022.3162846

[49] Yiyou Sun, Chuan Guo, and Yixuan Li. 2021. ReAct: Out-of-distribution Detection With Rectified Activations. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 144–157.

[50] Mingxing Tan, Ruoming Pang, and Quoc V. Le. 2020. EfficientDet: Scalable and Efficient Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[51] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. 2022. Open-Set Recognition: a Good Closed-Set Classifier is All You Need? arXiv:2110.06207 [cs.CV]

[52] Xin Wang, Thomas E. Huang, Trevor Darrell, Joseph E. Gonzalez, and Fisher Yu. 2020. Frustratingly simple few-shot object detection. In *Proc. Int. Conf. Mach. Learn. (ICML)*. 9861–9870.

[53] Zhenyu Wang, Yali Li, Xi Chen, Ser-Nam Lim, Antonio Torralba, Hengshuang Zhao, and Shengjin Wangk. 2023. Detecting Everything in the Open World: Towards Universal Object Detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

[54] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. 2021. Open-Vocabulary Object Detection Using Captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 14393–14402.

[55] Dawei Zhou, Hanjia Ye, and Dechuan Zhan. 2021. Learning placeholders for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4401–4410.

[56] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. 2022. Detecting Twenty-Thousand Classes Using Image-Level Supervision. In *Computer Vision – ECCV 2022*, Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (Eds.). Springer Nature Switzerland, Cham, 350–368.

A SUPPLEMENTARY MATERIAL

A.1 Dataset settings

VOC10-5-5 setting: C_B={aeroplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow}, C_N={diningtable, dog, horse, motorbike, person}, C_U={pottedplant, sheep, sofa, train, tvmonitor}={unknown}.

VOC-COCO setting: C_B={aeroplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, diningtable, dog, horse, motorbike, person, pottedplant, sheep, sofa, train, tvmonitor}, C_N={truck, traffic light, fire hydrant, stop sign, parking meter, bench, elephant, bear, zebra, giraffe, backpack, umbrella, handbag, tie, suitcase, microwave, oven, toaster, sink, refrigerator}, C_U={frisbee, skis, snowboard, sports ball, kite, baseball bat, baseball glove, skateboard, surfboard, tennis racket, banana, apple, sandwich, orange, broccoli, carrot, hot dog, pizza, donut, cake, bed, toilet, laptop, mouse, remote, keyboard, cell phone, book, clock, vase, scissors, teddy bear, hair drier, toothbrush, wine glass, cup, fork, knife, spoon, bowl}={unknown}.

RoadAnomaly setting: This dataset is mainly employed to test the generalization effect of our model in open-world road scenes. C_U={rhino, stone, deer, obstacle, road hole, wild boar, haystack, donkey, fox}={unknown}.

A.2 Preliminaries of Evidential Deep Learning and Weight Averaging

Details on evidential uncertainty and evidential deep learning can be found in the supplementary materials of [3]. For the theoretical proof that weight averaging can improve the model's generalization for out-of-distribution (or open-set) detection, please refer to [33].

A.3 Ablation Study of Pseudo-Unknown Sample Mining

We select top-k pseudo-unknown samples from foreground and background proposals ranked by the evidential uncertainty. Different choices of top-k and sampling ratio (fg:bg) are listed in Table 5. $k = 3$ and 1:1 perform better than other settings. Therefore, we adopt the above settings in all experiments.

Top-k	fg:bg	mAP_K	R_U	AR_U
1	1:1	18.15	28.75	12.45
3	1:1	**18.54**	**30.87**	**14.13**
5	1:1	18.26	30.62	13.97
10	1:1	18.43	30.09	13.48
3	1:2	18.31	29.27	13.16
3	1:3	18.50	29.42	13.33

Table 5: Pseudo-unknown sample mining. We list different choices of top-k and sampling ratio (fg:bg) in 1-shot VOC-COCO dataset setting.

A.4 Ablation Study of HMWA

We report different choices of the queue size and sampling step in HSIC-based moving weight averaging (HMWA). In Table 6, **(d)** and **(e)** perform better than other settings, which demonstrates that long-term memory (*i.e.*, larger $Q * S < T$, where T is the max iterations) is a good choice for HMWA.

	Queue size Q	Sampling step S	mAP_K	R_U	AR_U
(a)	32	1	18.18	30.04	13.98
(b)	32	2	18.39	30.18	14.02
(c)	32	3	18.47	29.89	13.97
(d)	32	5	**18.75**	30.24	14.08
(e)	32	10	18.54	**30.87**	**14.13**
(f)	4	10	18.43	29.74	13.97
(g)	8	10	18.21	30.58	14.09
(h)	16	10	18.09	30.77	14.11

Table 6: HSIC-based moving weight averaging. For the memory weight bank, we list different choices of queue size Q and sampling step S in 1-shot VOC-COCO dataset setting.

Backbone	mAP_K	R_U	AR_U
Resnet50	18.54	30.87	14.13
Resnet101	19.58	31.42	14.79
Swin-Tiny	21.80	33.24	15.89
Swin-Small	**23.52**	**34.40**	**16.71**

Table 7: We list different backbones in 1-shot VOC-COCO dataset setting.

A.5 Ablation Study of Swin Transformer

Swin Transformer [28] is a transformer-based architecture that uses hierarchical structures and local self-attention mechanisms to achieve state-of-the-art performance in many computer vision tasks. Swin Transformer achieves feature extraction by dividing the image into patches and applying local self-attention within these patches in a hierarchical manner. This allows Swin Transformer to capture both global and local features of an image effectively. Here, as illustrated in Table 7, we employ the Swin Transformer (Swin-Tiny and Swin-Small) as the backbone of Faster R-CNN and achieve an evident improvement in both known and unknown performance compared with Resnet (Resnet50 and Resnet101). For example, Swin-Tiny achieves 3.76% mAP_K and 1.76% AR_U improvements compared with Resnet50. Note that Swin-Tiny (29M) has roughly the same number of parameters as Resnet50 (26M), while Swin-Small (50M) has a similar number of parameters as Resnet101 (45M). It illustrates that in the case of almost the same parameter amount, Swin Transformer has a stronger feature generalization ability than Resnet.

A.6 Quantitative Results

As illustrated in Table 8, we report the unknown recall (R_U) of ten IoU thresholds (0.5 to 0.95 with 0.05 interval) with different shots on VOC10-5-5 and VOC-COCO dataset settings. Our method has demonstrated superior performance over the previous state-of-the-art results across various shot settings and IoU thresholds (VOC10-5-5: +7.27%∼+9.72% AR_U, VOC-COCO: +6.05%∼+7.41% AR_U). This highlights the effectiveness of our approach and its potential for real-world applications.

	VOC10-5-5										
	Method	IoU=0.5	IoU=0.55	IoU=0.6	IoU=0.65	IoU=0.7	IoU=0.75	IoU=0.8	IoU=0.85	IoU=0.9	IoU=0.95	Mean (AR_U)
1-shot	DS[30]	20.34	18.98	16.51	14.61	11.01	9.09	6.35	3.31	0.30	0.13	10.06
	PROSER[55]	29.35	27.98	26.56	24.44	21.47	18.89	14.89	9.22	3.74	0.39	17.69
	OpenDet[13]	30.48	28.42	27.41	25.89	22.17	20.01	15.12	9.38	3.03	0.23	18.21
	FOOD[46]	42.73	40.74	38.35	34.09	29.52	22.87	13.32	9.40	3.12	0.28	23.44
	Ours	**59.83**	**56.67**	**52.03**	**45.58**	**37.46**	**27.85**	**18.89**	**9.61**	**3.93**	**0.58**	**31.24**
3-shot	DS[30]	20.70	18.51	16.52	13.93	10.41	9.57	6.99	4.45	1.45	0.01	10.25
	PROSER[55]	30.08	28.37	27.30	24.82	22.50	19.18	15.15	9.12	3.06	0.19	17.98
	OpenDet[13]	33.81	29.14	28.75	26.24	22.50	20.18	16.54	8.13	4.01	0.13	18.94
	FOOD[46]	43.25	41.62	39.65	35.70	31.33	23.59	15.66	8.64	2.90	0.26	24.26
	Ours	**60.08**	**57.13**	**52.87**	**45.72**	**37.59**	**28.63**	**18.90**	**9.69**	**4.22**	**0.45**	**31.53**
5-shot	DS[30]	20.24	18.89	16.76	14.83	11.54	9.61	6.32	3.09	0.77	0.06	10.21
	PROSER[55]	30.09	28.73	27.32	24.90	22.31	19.68	15.41	9.73	3.12	0.13	18.14
	OpenDet[13]	33.09	28.33	27.27	26.01	23.08	20.83	16.82	8.73	3.13	0.12	18.74
	FOOD[46]	43.97	40.42	37.52	31.59	24.82	18.63	11.73	6.57	2.71	0.26	21.82
	Ours	**60.25**	**56.22**	**53.35**	**46.39**	**38.04**	**28.41**	**18.90**	**9.89**	**3.55**	**0.38**	**31.54**
10-shot	DS[30]	21.92	19.40	18.75	16.40	13.08	10.11	7.51	4.19	1.19	0.58	11.31
	PROSER[55]	31.91	28.94	27.58	25.10	22.07	18.59	15.27	11.51	4.47	0.29	18.57
	OpenDet[13]	33.71	30.61	28.03	26.75	23.04	20.69	18.86	10.19	4.19	0.77	19.68
	FOOD[46]	43.67	41.38	38.83	35.81	30.35	23.94	14.30	10.73	4.32	0.26	24.36
	Ours	**63.12**	**58.05**	**54.55**	**46.07**	**40.78**	**28.05**	**19.25**	**11.67**	**4.77**	**0.90**	**32.72**
	VOC-COCO											
	Method	IoU=0.5	IoU=0.55	IoU=0.6	IoU=0.65	IoU=0.7	IoU=0.75	IoU=0.8	IoU=0.85	IoU=0.9	IoU=0.95	Mean (AR_U)
1-shot	DS[30]	2.65	2.36	2.17	1.78	1.53	1.28	0.89	0.56	0.24	0.01	1.35
	PROSER[55]	6.93	6.03	5.18	4.42	3.72	2.88	2.05	1.21	0.53	0.11	3.31
	OpenDet[13]	7.21	6.53	5.86	5.06	4.26	3.41	2.58	1.59	0.58	0.05	3.71
	FOOD[46]	13.94	12.31	11.06	9.41	7.55	5.55	3.84	2.22	0.83	0.11	6.68
	Ours	**31.74**	**28.32**	**24.32**	**20.02**	**15.33**	**10.44**	**6.28**	**3.20**	**1.14**	**0.14**	**14.09**
5-shot	DS[30]	3.71	3.29	2.93	2.54	2.26	1.87	1.36	0.83	0.34	0.04	1.92
	PROSER[55]	9.31	8.36	7.31	6.11	5.09	4.05	3.04	1.73	0.69	0.08	4.58
	OpenDet[13]	10.05	9.00	7.95	6.72	5.76	4.66	3.48	2.12	0.79	0.06	5.06
	FOOD[46]	17.59	15.80	14.05	12.13	9.86	7.46	5.22	3.03	1.26	0.16	8.66
	Ours	**32.12**	**29.09**	**25.69**	**21.47**	**16.64**	**12.48**	**8.01**	**4.10**	**1.39**	**0.19**	**15.12**
10-shot	DS[30]	3.53	3.24	2.83	2.42	2.11	1.70	1.27	0.76	0.34	0.04	1.82
	PROSER[55]	8.89	7.90	6.87	5.82	4.76	3.82	2.63	1.57	0.63	0.08	4.30
	OpenDet[13]	14.69	13.37	12.03	10.45	8.69	6.62	4.81	2.89	1.00	0.10	7.47
	FOOD[46]	19.27	17.46	15.34	13.13	10.94	8.67	5.78	3.36	1.33	0.18	9.55
	Ours	**32.44**	**29.69**	**26.48**	**22.19**	**17.58**	**13.17**	**8.67**	**4.19**	**1.34**	**0.20**	**15.60**
30-shot	DS[30]	4.62	4.14	3.60	3.05	2.43	1.95	1.31	0.73	0.29	0.03	2.22
	PROSER[55]	10.07	8.88	7.71	6.58	5.38	4.29	3.04	1.68	0.67	0.10	4.84
	OpenDet[13]	16.47	15.09	13.67	11.94	9.85	7.79	5.58	3.33	1.12	0.20	8.50
	FOOD[46]	20.00	18.32	16.38	14.26	11.71	9.07	6.35	3.72	1.36	0.22	10.14
	Ours	**34.33**	**31.42**	**28.04**	**23.80**	**19.07**	**13.67**	**8.97**	**4.64**	**1.49**	**0.26**	**16.57**

Table 8: We report the unknown recall (R_U) of ten IoU thresholds (0.5 to 0.95 with 0.05 interval) with different shots.

A.7 Discussion

Here, we'd like to discuss the difference between the few-shot open-set object detection (FOOD) task and the prompt-based open-vocabulary (or zero-shot) object detection (OVOD) task [9, 10, 54, 56]. OVOD can leverage the long-tail dataset with the language [54] or class [56] prompt to train a detector (or large model), which can detect unknown classes without training samples. For OVOD, each unknown category needs auxiliary information (prompt), and the auxiliary information must be associated with the feature of the unknown category. The main differences are that 1) the auxiliary information or prompt of the unknown class is not provided in FOOD; 2) Due to the ambiguity of class definitions, it is impossible for the model to detect all classes in open-world scenes. Neither the language prompt nor the class prompt can contain all the category information in open-world scenes. In particular, there always exists unknown categories, where OVOD cannot tackle this situation; 3) FOOD aims to reject unknown objects and avoid identifying unknown classes as known classes with a high confidence score [46]. When a detected unknown object b_1 is not included in the prompt classes, OVOD will misrecognize it as a known class, however, FOOD will reject b_1 as an unknown class.

A.8 Limitations

Sometimes our algorithm makes a mistake and thinks that some low-quality proposals belong to an unknown category during testing, especially the novel categories with insufficient supervision. We currently have difficulty in removing these misclassified proposals from the final predictions. Although this doesn't affect the accuracy of our known categories, it is still important to reduce these false "unknown" predictions in the future. We are working on improving our method to address this issue.