Out-of-Distribution Identification: Let Detector Tell Which I Am Not Sure

Ruoqi Li[1], Chongyang Zhang[1,2⋆], Hao Zhou[1], Chao Shi[1], and Yan Luo[1]

[1]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[2]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, Shanghai 200240, China
{nilponi, sunny_zhang, zhouhao_0039, shichaostone, luoyan_bb}@sjtu.edu.cn

Abstract. The superior performance of object detectors is often established under the condition that the test samples are in the same distribution as the training data. However, in most practical applications, out-of-distribution (OOD) instances are inevitable and usually lead to detection uncertainty. In this work, the Feature structured OOD-IDentification (FOOD-ID) model is proposed to reduce the uncertainty of detection results by identifying the OOD instances. Instead of outputting each detection result directly, FOOD-ID uses a likelihood-based measuring mechanism to identify whether the feature satisfies the corresponding class distribution and outputs the OOD results separately. Specifically, the clustering-oriented feature structuration is firstly developed using class-specified prototypes and Attractive-Repulsive loss for more discriminative feature representation and more compact distribution. With the structured features space, the density distribution of all training categories is estimated based on a class-conditional normalizing flow, which is then used for the OOD identification in the test stage. The proposed FOOD-ID can be easily applied to various object detectors including anchor-based frameworks and anchor-free frameworks. Extensive experiments on the PASCAL VOC-IO dataset and an industrial defect dataset demonstrate that FOOD-ID achieves satisfactory OOD identification performance, with which the certainty of detection results is improved significantly.

Keywords: Out-of-Distribution, Identification, Object Detection

1 Introduction

Over the last decade, the success of object detection has boosted widespread applications in various fields. However, their superior performance often relies on the assumption that test instances and training instances are in the same distribution [10][33]. When encountering out-of-distribution (OOD) inputs, the detector may make some seemingly stupid mistakes as shown in Fig. 1. The

⋆ Corresponding Author.

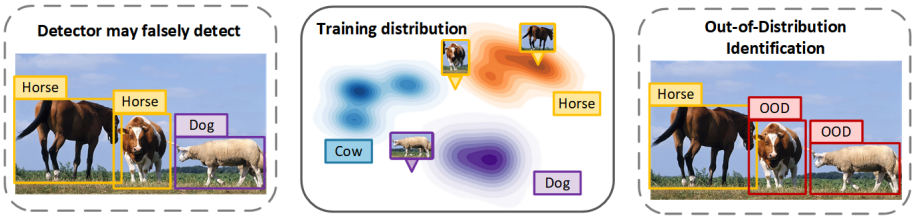

Fig. 1. The object detectors may falsely detect two types of OOD objects, including unknown-category objects and confusing-category objects. Our proposed FOOD-ID can perform OOD identification to improve detection certainty.

sheep is mistakenly classified as a dog and the cow is mistakenly classified as a horse. The former belongs to the unknown-category objects, which do not appear in the training set but are mistakenly detected and classified as a known class. The latter belongs to confusing-category objects, although it is a known class object, it is misclassified into another known class since it is located in low-probability distribution regions. The above drawback limits the deployment of object detectors in safety-critical applications, in which the detection results should be with high certainty to avoid high risks.

The phenomenon is more obvious in industrial automatic production. To avoid immeasurable risks, each unqualified product needs to be detected by a defect detector. However, it is difficult to balance the miss rate and false rate well because few or zero misses tend to bring more false positives. We observe that most of the false detections are made by the reason that their features are out of training distribution, which makes the detection results high uncertainty. Thus, identifying whether each detected object is OOD or not, can reduce the detection uncertainty, and then the false rate can be decreased effectively. At the same time, rather than simply suppressing such OOD false detections, they need to be output separately for further confirmation by humans.

Most existing object detectors follow the in-distribution (IND) assumption and can not identify OOD objects. Recently, some methods [24, 21, 27, 43, 49, 38] leverage an independent classifier to recognize OOD image input, preventing them from entering downstream tasks. However, OOD objects and their combination with IND object detection are rarely explored. Besides, Open set detection methods [4, 29, 31] are dedicated to simultaneously performing known-category object detection and new unknown-category object recognition. However, apart from unknown-category instances, OOD objects also include confusing-category objects such as the cow that is mistakenly detected as a horse in Fig. 1.

To address the above challenges, we propose FOOD-ID, a unified model capable of object detection and OOD identification. Specifically, FOOD-ID adds a dynamic prototype branch to the detector head, which can dynamically store and update the multi-scale feature prototypes of training categories. The clustering-oriented feature structuration is developed by class-specified prototypes and Attractive-Repulsive loss for discriminative feature representation and compact

distribution. With the structured feature space, the density distribution of all categories is estimated based on a class-conditional normalizing flow (CCNF). At test time, the log-likelihood of each detected object feature in the distribution of all categories will be predicted by the trained CCNF, based on which OOD identification is performed. With OOD identification, the detector is able to express uncertainty by picking out the OOD objects which have a higher error probability, and then the certainty of detection can be improved significantly.

Our contributions can be summarized as follows:

1. We propose FOOD-ID, a unified model capable of object detection and out-of-distribution identification.
2. We propose the clustering-oriented feature structuration developed by class-specific prototypes and Attractive-Repulsive loss. Furthermore, we adopt a class-conditional normalizing flow for feature distribution modeling, which can be used to estimate the likelihood and identify OOD objects.
3. We conduct experiments on various object detection frameworks and datasets. FOOD-ID achieves satisfactory OOD identification performance and improves the certainty of detection results.

2 Related Work

2.1 In-Distribution Object Detection

In recent years, object detectors have achieved rapid development from anchor-based frameworks [13, 12, 39, 26, 25, 37] to recent anchor-free frameworks [42, 7, 19, 48]. Continuous breakthroughs in accuracy and speed have allowed them to be widely deployed in various applications. Object detectors are trained to detect objects of known classes labeled in the training set, and perform detection based on the assumption that test inputs satisfy the training distribution. The ideal in-distribution setting ignores the existence of OOD inputs, which will often encounter in practical applications. In this case, it is difficult for the detector to make confident and correct predictions of these OOD objects due to a lack of knowledge, resulting in large uncertainty in detection results. The above flaws limit their deployment in safety-critical real-world applications.

2.2 Out-of-Distribution Detection

OOD detection aims to detect OOD inputs in advance and prevent them from entering downstream tasks, e.g. localization or segmentation. Recent works improve OOD detection by using the ODIN score [24, 16], Mahalanobis distance [21], energy score [27], ensemble [43, 47], flows [49, 1] and generative models [38]. However, the above studies usually focus on image classification, simply classifying the input as IND or OOD image, and rarely explore OOD objects in more complex object detection tasks. Recently, the work VOS [6] has begun to investigate the integration of OOD detection into object detection, but their focus on OOD objects is still on all detections on unknown categories from other

datasets. VOS synthesizes outliers by sampling and trains the head to classify OOD in a supervised way. While our FOOD-ID is only based on IND samples and does not rely on OOD supervision. Open set detection requires correctly classifying known classes and labeling other classes that are not in the training set as unknown. Various discriminative [2, 46] and generative models [11, 32, 34, 36] have been proposed to improve open set recognition. Although most studies still focus on image classification, some studies have paid attention to suppressing open set false positives in object detection. MC Dropout-based [29], Gaussian Mixture Model[31] and distance-based methods [30] are used to extract model uncertainty and reject open-set error. Unlike the above studies, the OOD objects we target do not necessarily belong to an independent new unknown-category, but can also be confusing-category objects that are located in low-probability regions of known class distribution. In addition, we do not wish to simply suppress OOD detections, but rather perform OOD identification across all detections, and output OOD separately for human experts to utilize.

2.3 Uncertainty Estimation in Object Detection

The key to performing OOD identification is that model needs the ability to estimate its epistemic uncertainty, which is the uncertainty caused by the model's lack of knowledge [17]. Sampling-based methods (such as Bayesian-based [9], ensemble-based [20] and test-time augmentation [44]) are often used in regression, but they are difficult to apply in object detection due to time consumption. Recently proposed non-sampling uncertainty estimation methods (such as Gaussian yolov3 [3], Gaussian FCOS [22], GFLv2 [23]) measure uncertainty in terms of variance by modeling the localization output as a distribution rather than a deterministic value. The above methods pay more attention to the uncertainty of localization to achieve accurate bounding box regression to improve detection performance. However, they are still based on the IND assumption and do not consider OOD inputs. Instead, we focus on the uncertainty in the detection results of OOD objects due to the model's lack of knowledge. Our proposed method adopts a sampling-free manner, and can be used as a plug-in to enhance object detection frameworks.

3 Method

3.1 Problem Statement

We consider a object detection model M_C which is trained with the training set D, which contains C known categories $\mathcal{K} = \{1, 2, ..., C\} \subset N_+$. On the one hand, the detection model M_C needs to perform the correct classification and localization of IND objects x_{IND}, which include known objects that satisfy a high-probability distribution of known classes $\mathcal{K}$ in the training set D. On the other hand, the M_C needs to identify OOD objects x_{OOD}, including confusing-category and unknown-category objects. The former are located in low-probability regions of known class distribution. The latter do not belong to a known class $\mathcal{K}$ in the training set D.

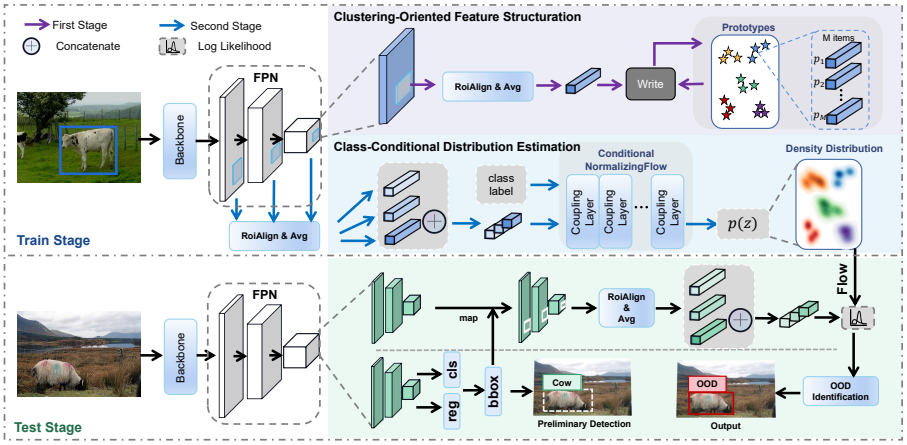

Fig. 2. Overview of the proposed FOOD-ID.

3.2 Overview

We propose Feature structured OOD-IDentification (FOOD-ID), a unified model for object detection and OOD identification. Fig.2 overviews the training and testing procedures for FOOD-ID. In the first training stage, all labeled ground-truth bounding boxes are mapped to Feature Pyramid Networks (FPN) and their multi-scale features are extracted by the RoIAlign layer and Average Pooling layer. Then the features are used to dynamically update the prototypes of the corresponding class. The Attractive-Repulsive loss is added to the detector to form a more discriminative feature space (detailed in Section 3.3). In the second training stage, both class labels and ground-truth features extracted by the trained detector are input to the class-conditional normalizing flow to model the density distribution of the feature space (detailed in Section 3.4). In the testing stage, the detector firstly obtains preliminary detection results and maps them to FPN for feature extraction. Each object feature is then fed into the trained normalizing flow to estimate the log-likelihood of satisfying each class distribution. Objects with a high likelihood class that is different from the original detected class or objects with high entropy are identified as OOD objects, while for other IND objects, the original detection results are output (detailed in Section 3.5).

3.3 Clustering-Oriented Feature Structuration

The clustering-oriented feature structuration is developed for more discriminative feature representation and more compact distribution. The feature structuration requires centralization and compactification. For centralization, we introduce a dynamic prototype branch to dynamically update class prototypes to form prototype-centric feature clusters. For compactification, the Attractive-Repulsive loss are proposed to attract features to corresponding prototypes and encourage different class prototypes mutually repulsive.

The dynamic prototype branch is introduced to dynamically store and update class prototypes as cluster centers for feature centralization. Suppose there are a total of C classes in the training set, and feature maps of S scales are extracted through FPN, and M prototype features are stored for each class on each scale, that is, a total of $C \times S \times M$ prototypes are stored in the branch. During training, the ground-truth bounding boxes in the training set are respectively mapped to the S scale feature maps extracted by FPN, and then the ground-truth object features are extracted by the RoIAlign layer and Average Pooling layer. Inspired by the memory network [45, 41, 28, 35], we adopt a similar memory update strategy for prototypes. When K object features q^K of class c on the scale j are extracted, they are used to update the corresponding M prototypes p^M.

First, calculate the cosine similarity of K object features and M prototype features as update weight $w^{k,m}$, as follows:

$$w^{k,m} = \text{softmax}((p^m)^T q^k) \tag{1}$$

Then for each prototype p^m, select the most similar object features U_m to update p^m using the following formula, where $f(\cdot)$ is the L_2 normalization.

$$p^m = f(p^m + \sum_{k \in U_m} w^{'k,m} q^k) \tag{2}$$

$$w^{'k,m} = \frac{w^{k,m}}{\max_{k' \in U_m} w^{k',m}} \tag{3}$$

Then, the Attractive-Repulsive loss is introduced to facilitate feature compactification. The attractive loss encourages the object feature to be close to its most similar prototype of the corresponding class, which is calculated as the L_2 distance between the object feature q^k and its closest positive class prototype p^{pos}.

$$\mathcal{L}_{attractive} = \sum_k^K \left\| q^k - p^{pos} \right\|_2 \tag{4}$$

$$pos = \underset{m \in M_c}{\text{argmax}}\, w^{k,m} \tag{5}$$

While the repulsive loss encourages prototypes of different classes to be more dispersed and is calculated as formula (6). The first term is the L_2 distance between the object feature q^k and its closest p^{pos} prototype in class c. While the second term is the distance between the object feature q^k and its closest negative prototype p^{neg} in other classes, where α is a parameter used to control the gap between the two distances, which is set to 1 in the experiment.

$$\mathcal{L}_{repulsive} = \sum_k^K [\left\| q^k - p^{pos} \right\|_2 - \left\| q^k - p^{neg} \right\|_2 + \alpha]_+ \tag{6}$$

$$neg = \underset{m \in M_{c'}, c' \neq c}{\text{argmax}}\, w^{k,m} \tag{7}$$

So the loss function of the detector includes the classification loss $\mathcal{L}_{cls}$ and the localization loss $\mathcal{L}_{loc}$ of the original head, as well as the Attractive-Repulsive loss $\mathcal{L}_{AR}$ with the balance parameters η and λ, which are empirically set to 0.5.

$$\mathcal{L}_{AR} = \eta \mathcal{L}_{attractive} + (1 - \eta) \mathcal{L}_{repulsive} \tag{8}$$

$$\mathcal{L}_{detector} = \mathcal{L}_{cls} + \mathcal{L}_{loc} + \lambda \mathcal{L}_{AR} \tag{9}$$

Note that to ensure high-quality prototypes, the dynamic prototype branch will start updating only when the model has sufficient detection capabilities (i.e. after training for long enough epochs). After at least one epoch of storage of all ground-truth object features in the training set, the AR loss is added.

3.4 Class-Conditional Distribution Estimation

Then we model the distribution of the structured feature space based on class-conditional normalizing flow (CCNF). Normalizing flows are a class of generative probabilistic models, first proposed by Dinh et al. [5]. These models can fit arbitrary density distributions $q(x)$ by a simple base density $q(z)$ and a invertible mapping $g : X \to Z$. Then, the log-likelihood of any $x \in X$ can be estimated as follows:

$$\log q(x, \theta) = \log q(z) + \log |det J_x| \tag{10}$$

where the latent variable z is usually assumed to satisfy a standard multivariate Gaussian prior ($z \sim \mathcal{N}(0, \mathbb{I})$)[5] and $J_x = \nabla_x g(x,\ \theta)$ is the Jacobian matrix of a invertible flow model ($z = g(x,\ \theta)$) with trainable parameter θ.

In order to model the distribution of all categories in the structured feature space in a unified manner, we introduce class information to achieve better feature distinction. Similar to [1], we assume the latent variable z satisfy a Gaussian mixture model with class-dependent mean μ_y and a unit covariance matrix $\mathbb{I}$ as follows, where y is the class label.

$$q(Z|Y) = N(\mu_y, \mathbb{I}) \quad and \quad q(z) = \textstyle\sum_y p(y)\mathcal{N}(\mu_y, \mathbb{I}) \tag{11}$$

We utilize the Information Bottleneck(IB)-based loss function [1] to train the class-conditional normalizing flow, where the trade-off parameter β balances the two terms.

$$\mathcal{L}_{CCNF} = \mathcal{L}_X - \beta \mathcal{L}_Y \tag{12}$$

The $\mathcal{L}_X$ term represents the mutual information term $I(X, Z)$, which is approximated by the empirical mean of the negative log-likelihood of the unconditional normalizing flow over a training dataset. Note that the input is a noisy version $X' = X + \mathcal{E}$, where $\mathcal{E} \sim \mathcal{N}(0, \sigma^2\mathbb{I}) = p(\mathcal{E})$ to artificially introduce a minimal amount of information loss. The $\mathcal{L}_X$ term encourages the normalizing flow to ignore class information and become an accurate likelihood model.

$$\mathcal{L}_X = \mathbb{E}_{p(X),p(\mathcal{E})}[-\log q(x + \varepsilon)] \tag{13}$$

While the $\mathcal{L}_Y$ term represents the mutual information term $I(X, Y)$, which is defined as the empirical mean of the log-posterior in a training set $\{x_i, y_i, \varepsilon_i\}_{i=1}^N$ of size N as follows. This term encourages each pair $g_\theta(x + \varepsilon)$ to be drawn to the correct cluster center μ_y while the cluster centers ($\mu_{Y\neq y}$) of the other classes are repulsed, which ensures accurate classification.

$$\mathcal{L}_Y = \frac{1}{N} \sum_{i=1}^{N} \log \frac{\mathcal{N}(g_\theta(x_i + \varepsilon_i); \mu_{y_i}, \mathbb{I})p(y_i)}{\sum_{y'} \mathcal{N}(g_\theta(x_i + \varepsilon_i); \mu_{y'}, \mathbb{I})p(y')} \tag{14}$$

We construct the CCNF based on invertible neural networks composed of several affine coupling layers [5]. We first extract multi-scale features from all ground-truth objects in the training set via the trained dynamic prototype branch and concatenate them together. Then all object features and class labels are input to the CCNF for training based on Formula (12). As a result, the trained CCNF can perform effective density estimation and class distinction.

3.5 Likelihood-based Out-of-Distribution Identification

At test time, the classification branch and localization branch of the detector firstly obtain preliminary detection results, and the detection boxes are mapped to FPN and then object features are extracted through the dynamic prototype branch. Then the log-likelihood of each object feature in the distribution of all categories will be predicted by the trained CCNF as follows:

$$\log p(x) = \log \sum_y \exp(\frac{\|z-\mu_y\|_2^2}{2}) + \log |detJ_x| \tag{15}$$

Likewise, the entropy of the likelihoods is calculated as follows:

$$H(x) = -\sum_x p(x) \log p(x) \tag{16}$$

Compared with the original predicted class, objects with a different maximum likelihood class will be regarded as OOD directly. Otherwise, we further adopt entropy as the uncertainty score and identify objects with entropy greater than a threshold. The former case means that the object features are more in line with the distribution of another category rather than the detected category, which corresponds to the misclassification of confusing-category objects. The latter case means that the object feature achieve similar likelihood across all categories, which correspond to unknown-category objects. While for other IND objects, the original detection results are output.

4 Experiments

4.1 Experimental Setup

Datasets For a given dataset D containing a total of N categories, we first divide it into the training set D_{train} and test set D_{test} by stratified sampling. We select the instances in D_{train} that only contain $C(C < N)$ known classes objects as the model's training dataset D_C, which is achieved by deleting all images in D_{train} that contain unknown $N - C$ classes objects. During testing, the original test dataset D_{test} with N categories is used. To evaluate the performance of methods on both general object detection tasks and specific object detection tasks, we tested the following datasets:

PASCAL VOC-IO[8]: The dataset PASCAL VOC contains a total of 20 categories. We select the first $C = 15$ categories as known categories, pick out the data that only contain these C categories in the original training set as the

training set D_C, and use the original test set D_{test} containing 20 categories as the test to construct PASCAL VOC-IO.

Crack Defect: We build Crack Defect dataset to detect crack defects in the junction of reed tubes, and it contains four main categories, namely, *crack*, *deformation*, *bubble*, and *stain*. Only the categories of *crack* need to be detected for product screening. *Deformation*, *bubble*, and *stain* are all qualified products that are easily mistakenly detected as *crack* in practice. Crack Defect is constructed by selecting the instances that only contain the categories of crack as the training set D_C, and using the test set D_{test} containing all categories.

Metrics We categorize the raw detections outputs of the object detector into correct detections D_C, OOD false detections D_{OF}, and remaining false detections. A detection box is considered as D_C if it is located and classified correctly (has an IoU greater than 0.5 with a ground-truth box of the predicted class). A detection box is considered as D_{OF} if it is correctly located but misclassified (has an IoU greater than 0.5 with a ground-truth box of a class different from the predicted class). The remaining false detections are usually background detections or duplicate detections. We tested the following metrics to evaluate the distinction between D_C and D_{OF}:

ROC: Receiver Operating Characteristic (ROC) curve represents the trade-off between true positive rate (TPR) and false positive rate (FPR) when changing the uncertainty threshold θ in OOD identification. TPR represents the proportion of D_{OF} that are correctly identified and FPR represents the proportion of D_C that are misidentified as OOD. And the area under the ROC curve (AUC) is also calculated to represent the overall performance.

$$TPR(\theta) = \frac{|D_{OF} > \theta|}{|D_{OF}|} \quad FPR(\theta) = \frac{|D_C > \theta|}{|D_C|} \tag{17}$$

TPR@FPR: We report TPR at 5%, 10% and 20% FPR respectively. These operating points evaluate the identification rate of D_{OF} under a low misidentification of D_C, which corresponds to the ability to identify OOD with the lowest possible miss-rate in the application.

Precision: We calculated the precision before and after OOD identification respectively. The precision before OOD identification is calculated as the proportion of D_C in all raw detection results. After taking 10% of all detected results as OOD, all detection results are divided into IND set and OOD set. The precision of the former is the proportion of Dc in the IND set, and the precision of the latter is the proportion of D_{OF} in the OOD set.

4.2 Main Results

We test on three representative object detectors separately: FCOS [42], an anchor-free one-stage detector; RetinaNet [25], an anchor-based one-stage detector; and Faster RCNN [39], an anchor-based two-stage detector. For each object detector, we train the model and test it under a certain model detection

Table 1. The OOD identification performance measured by AUC and TPR@FPR metrics across both datasets and three detectors.

Method	PASCAL VOC-IO				Crack Defect			
	AUC	TPR@			AUC	TPR@		
		5%FPR	10%FPR	20%FPR		5%FPR	10%FPR	20%FPR
FCOS								
Baseline-Score	0.830	43.5	59.0	73.6	0.865	51.8	68.1	80.3
Baseline-Entropy	0.861	53.0	64.7	77.4	0.882	56.9	70.2	81.4
FOOD-ID-Proto	0.899	64.3	72.8	82.7	0.906	68.3	76.0	84.5
FOOD-ID-CCNF	**0.913**	**70.1**	**80.1**	**88.0**	**0.921**	**73.4**	**82.7**	**90.3**
RetinaNet								
Baseline-Score	0.900	53.5	72.7	84.9	0.923	62.7	79.9	90.2
Baseline-Entropy	0.903	60.4	72.3	83.2	0.900	59.4	71.1	82.4
FOOD-ID-Proto	0.919	69.5	79.0	87.7	0.936	75.2	83.4	90.8
FOOD-ID-CCNF	**0.928**	**76.5**	**85.3**	**91.2**	**0.942**	**80.5**	**88.5**	**93.6**
Faster RCNN								
Baseline-Score	0.840	35.5	54.7	71.6	0.872	49.1	64.3	78.4
Baseline-Entropy	0.843	45.4	61.4	74.7	0.881	50.2	65.9	81.0
FOOD-ID-Proto	0.861	50.2	67.8	78.1	0.887	56.7	72.2	82.7
FOOD-ID-CCNF	**0.866**	**57.7**	**68.1**	**79.7**	**0.908**	**63.4**	**75.6**	**85.2**

threshold. After obtaining the detection results, we compare the performance of OOD identification based on the following methods:

Baseline-Score: The confidence score of the detector is used as the criterion for uncertainty estimation, low score means high uncertainty [15, 29, 4].

Baseline-Entropy: The entropy of the confidence scores for all classes of the detector is used for uncertainty estimation, high entropy means high uncertainty [40, 18, 14].

FOOD-ID-Proto: Based on the dynamic prototype branch, the distance from the detected object feature to the nearest prototype of the predicted category is used as the uncertainty score, large distance means high uncertainty.

FOOD-ID-CCNF: Based on the trained CCNF, we leverage both likelihood and entropy to measure the uncertainty. As described in Section 3.5, compared to the original predicted class, objects with a different maximum likelihood class will be regarded as OOD directly. Otherwise, we further adopt entropy as the uncertainty score to identify OOD objects.

As shown in Table 1, the two methods we proposed achieve advanced performance on OOD identification, which is maintained across three object detectors and two datasets. Especially under a very low misidentify rate of D_C, FOOD-ID can achieve significantly better D_{OF} identification performance than baseline methods. It can be noted that FOOD-ID-CCNF outperforms FOOD-ID-Proto, because the former can accurately model the training feature distribution and estimate exact likelihood, while the latter only measures the distance to the class closest prototype. The complete ROC curve on PASCAL VOC-IO is shown in Fig. 3. Meanwhile, as shown in the Table 2, the raw detection results are divided into IND set and OOD set through ood identification. For the IND set,

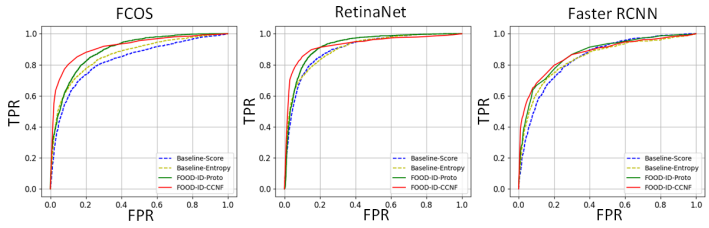

Fig. 3. The OOD identification performance measured by ROC metric on PASCAL VOC-IO dataset.

Table 2. The precision of detection results before and after excluding 10% of the detection results as OOD with OOD identification on PASCAL VOC-IO dataset.

Method	FCOS			RetinaNet			Faster RCNN		
	Before	After		Before	After		Before	After	
	IND Set	IND Set	OOD Set	IND Set	IND Set	OOD Set	IND Set	IND Set	OOD Set
Baseline-Score	0.797	0.827	0.234	0.738	0.772	0.268	0.759	0.785	0.167
Baseline-Entropy		0.830	0.358		0.773	0.369		0.782	0.273
FOOD-ID-Proto		0.831	0.367		0.789	0.437		0.789	0.260
FOOD-ID-CCNF		**0.836**	**0.414**		**0.791**	**0.540**		**0.792**	**0.335**

the precision of detection results is significantly improved, which means that OOD identification improves the certainty of detection. For the OOD set, it is a more challenging task with the interference of background false positives. FOOD-ID-CCNF can achieve advanced precision in both IND set and OOD set.

4.3 Ablation Study

FOOD-ID is mainly composed of clustering-oriented feature structuration and class-conditional distributed estimation. We investigate the impact of each component on the overall performance of the model.

The benefit of feature structuration To explore the need for feature structuration, we conduct extra experiments on detectors trained without feature structuration. We still use the dynamic prototype branch to extract ground-truth object features but do not train with Attractive-Repulsive loss. The same conditional normalizing flow model is then used to model the feature distribution and perform OOD identification at test time. The OOD identification results are shown in Table 3. It can be seen that in the absence of AR loss, the ability to identify OOD is severely degraded after passing the same distribution modeling model. Thus, feature structuration facilitates subsequent accurate distribution modeling and class distinction.

We visualize the ground-truth features of the PASCAL VOC-IO training set extracted by the dynamic prototype branch applied to FCOS with or without

Table 3. The OOD identification performance with or without Attractive-Repulsive(AR) loss on PASCAL VOC-IO dataset.

Method	FCOS				RetinaNet				Faster RCNN			
	AUC	TPR@			AUC	TPR@			AUC	TPR@		
		5%FPR	10%FPR	20%FPR		5%FPR	10%FPR	20%FPR		5%FPR	10%FPR	20%FPR
FOOD-ID-Proto w/o ARloss	0.705	16.1	26.9	43.1	0.704	18.8	32.3	47.2	0.687	16.1	25.7	41.1
FOOD-ID-Proto w ARloss	0.899	64.3	72.8	82.7	0.919	69.5	79.0	87.7	0.861	50.2	67.8	78.1
FOOD-ID-CCNF w/o ARloss	0.888	69.1	75.1	83.6	0.921	73.9	82.6	89.9	0.857	54.5	65.3	76.3
FOOD-ID-CCNF w ARloss	**0.913**	**70.1**	**80.1**	**88.0**	**0.928**	**76.5**	**85.3**	**91.2**	**0.866**	**57.7**	**68.1**	**79.7**

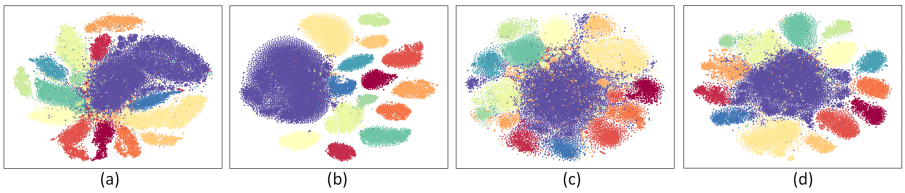

Fig. 4. Visualization of the detector feature distribution and the corresponding normalizing flow latent variable distribution with or without AR loss on PASCAL VOC-IO dataset. (a) Detector feature distribution without AR loss. (b) Detector feature distribution with AR loss. (c) Latent variable distribution without AR loss. (d) Latent variable distribution with AR loss.

AR loss by t-SNE in Fig. 4, where the same color dots represent the features of the same class. It can be seen intuitively that the feature distribution with AR loss is more structural and discriminative. It is achieved by utilizing dynamic prototypes as cluster centers for centralization and AR loss for compactification.

The benefit of class-conditional distribution estimation To explore the benefit of class-conditional distribution estimation, we compare the impact of distribution modeling using class-conditional normalizing flow and unconditional normalizing flow (UNF) on OOD identification performance. The former adopts the CCNF we describe in Subsection 3.4. The latter adopts a series of UNFs to individually model the distribution of each class of features. The UNFs are trained with the objective of minimizing the log-likelihood in formula (10), assuming that the distribution of the latent variables z satisfies a multivariate Gaussian distribution [5].

The performance of OOD identification on the same detection model using different flow models is shown in Table 4. It can be seen that class-conditional distribution estimation has a clear advantage over unconditional distribution estimation in terms of OOD identification. The latter case can only access the features of a single category with ignorance of other category information, resulting in a weak feature discrimination ability because the features of different categories are not conditionally independent.

Table 4. The performance of OOD identification under distribution modeling using class-conditional normalizing flow (CCNF) and unconditional normalizing flow (UNF) on PASCAL VOC-IO dataset.

Method	FCOS				RetinaNet				Faster RCNN			
	AUC	TPR@			AUC	TPR@			AUC	TPR@		
		5%FPR	10%FPR	20%FPR		5%FPR	10%FPR	20%FPR		5%FPR	10%FPR	20%FPR
UNF	0.881	65.6	71.4	78.9	0.885	60.3	70.9	82.9	0.830	39.2	54.8	72.6
CCNF	**0.913**	**70.1**	**80.1**	**88.0**	**0.928**	**76.5**	**85.3**	**91.2**	**0.866**	**57.7**	**68.1**	**79.7**

The trade-off between density estimation and class distinction To explore the trade-off between density estimation and class distinction in the class-conditional distribution estimation, we experiment with different values of the balance parameter β in the loss function of the CCNF. The OOD identification results of different β with FOOD-ID-CCNF applied to FCOS are shown in Table 5. We also visualize the distribution of latent variables for CCNF trained with different β on PASCAL VOC-IO training set by t-SNE, as shown in Fig. 5.

Table 5. The OOD identification results under different β on PASCAL VOC-IO dataset.

β	AUC	TPR@5%FPR	TPR@10%FPR	TPR@20%FPR
0.0	0.500	4.9	9.8	19.8
0.5	0.911	69.9	78.1	86.6
1.0	**0.913**	**70.1**	**80.1**	**88.0**
2.0	0.912	68.6	80.0	87.4
10.0	0.912	68.1	78.0	87.3

The β trades off density estimation and class distinction. Smaller β encourages more accurate density estimation, at the cost of losing class distinction. When β is equal to 0, it will degenerate into a density model that does not consider class conditions, resulting in a lack of OOD identification ability. When β increases, the flow can achieve more efficient classification, but it is more difficult to accurately estimate the density within the class. We take $\beta = 1$ to achieve a balance.

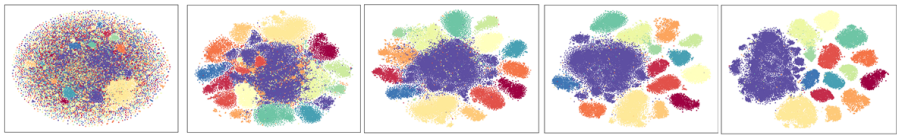

Fig. 5. Visualization of the distribution of latent variables with different β values on PASCAL VOC-IO dataset.

4.4 Visualization

Fig. 6 shows the comparison of the detection results of FOOD-ID and the original detector on two datasets. FOOD-ID demonstrates advanced OOD identification capabilities. It also can be noted that it is difficult to identify OOD by confidence score of original detector, because OOD usually has a non-low confidence score.

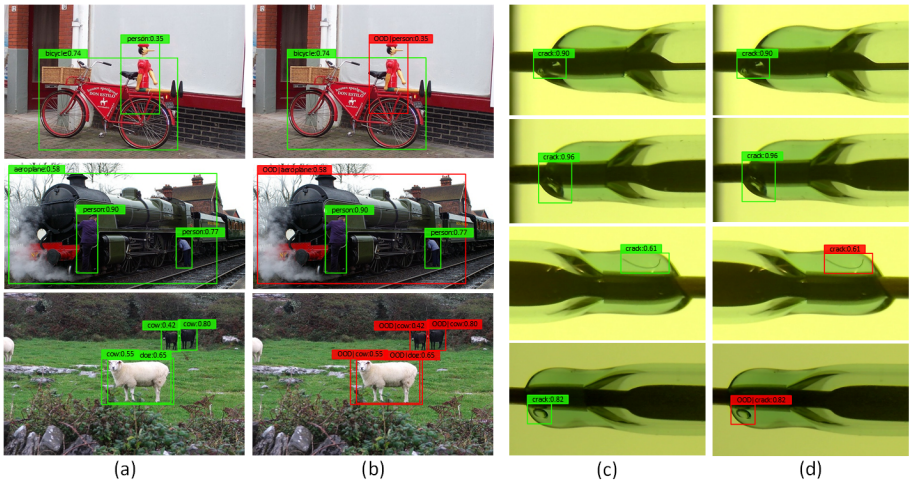

Fig. 6. Visualization of detection results. (a) Original detections on PASCAL VOC-IO dataset. (b) FOOD-ID detections on PASCAL VOC-IO dataset. (c) Original detections on Crack Defect dataset. (d) FOOD-ID detections on Crack Defect dataset.

5 Conclusions

Out-of-distribution inputs limit the deployment of in-distribution object detectors in safety-critical real-world applications. In this paper, we propose a unified model FOOD-ID capable of object detection and out-of-distribution identification. FOOD-ID develops the clustering-oriented feature structuration by class-specific prototypes and Attractive-Repulsive loss. Furthermore, a class-conditional normalizing flow is adopted to model the feature distribution and estimate likelihood at test time. FOOD-ID achieves satisfactory OOD identification performance and improves the certainty of detection results.

Acknowledgement. This work was supported in part by the National Natural Science Fund of China(61971281), the National Key R&D Program of China (2021YFD1400104), the Shanghai Municipal Science and Technology Major Project(2021SHZDZX0102), and the Science and Technology Commission of Shanghai Municipality(18DZ2270700).

References

1. Ardizzone, L., Mackowiak, R., Rother, C., Kothe, U.: Training normalizing flows with the information bottleneck for competitive generative classification. arXiv: Learning (2020)
2. Bendale, A., Boult, T.E.: Towards open set deep networks. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 1563–1572 (2016)
3. Choi, J., Chun, D., Kim, H., Lee, H.J.: Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 502–511 (2019)
4. Dhamija, A., Günther, M., Ventura, J., Boult, T.: The overlooked elephant of object detection: Open set. pp. 1010–1019 (03 2020). https://doi.org/10.1109/WACV45572.2020.9093355
5. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real nvp. ArXiv **abs/1605.08803** (2017)
6. Du, X., Wang, Z., Cai, M., Li, S.: Vos:learning what you don't know by virtual outlier synthesis. In: International Conference on Learning Representations (2022), https://openreview.net/forum?id=TW7d65uYu5M
7. Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., Tian, Q.: Centernet: Keypoint triplets for object detection. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 6568–6577 (2019)
8. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision **88**(2), 303–338 (2010)
9. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. ArXiv **abs/1506.02142** (2016)
10. Gawlikowski, J., Tassi, C.R.N., Ali, M., Lee, J., Humt, M., Feng, J., Kruspe, A.M., Triebel, R., Jung, P., Roscher, R., Shahzad, M., Yang, W., Bamler, R., Zhu, X.: A survey of uncertainty in deep neural networks. ArXiv **abs/2107.03342** (2021)
11. Ge, Z., Demyanov, S., Chen, Z., Garnavi, R.: Generative openmax for multi-class open set classification. ArXiv **abs/1707.07418** (2017)
12. Girshick, R.B.: Fast r-cnn. 2015 IEEE International Conference on Computer Vision (ICCV) pp. 1440–1448 (2015)
13. Girshick, R.B., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. 2014 IEEE Conference on Computer Vision and Pattern Recognition pp. 580–587 (2014)
14. Harakeh, A., Smart, M., Waslander, S.L.: Bayesod: A bayesian approach for uncertainty estimation in deep object detectors. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 87–93. IEEE (2020)
15. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv preprint arXiv:1610.02136 (2016)
16. Hsu, Y.C., Shen, Y., Jin, H., Kira, Z.: Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 10948–10957 (2020)
17. Hüllermeier, E., Waegeman, W.: Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. Mach. Learn. **110**, 457–506 (2021)

18. Kaur, R., Jha, S., Roy, A., Park, S., Sokolsky, O., Lee, I.: Detecting oods as datapoints with high uncertainty. arXiv preprint arXiv:2108.06380 (2021)
19. Kong, T., Sun, F., Liu, H., Jiang, Y., Li, L., Shi, J.: Foveabox: Beyound anchor-based object detection. IEEE Transactions on Image Processing **29**, 7389–7398 (2020)
20. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: NIPS (2017)
21. Lee, K., Lee, K., Lee, H., Shin, J.: A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In: NeurIPS (2018)
22. Lee, Y., won Hwang, J., Kim, H., Yun, K., Park, J.: Localization uncertainty estimation for anchor-free object detection. ArXiv **abs/2006.15607** (2020)
23. Li, X., Wang, W., Hu, X., Li, J., Tang, J., Yang, J.: Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 11627–11636 (2021)
24. Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. arXiv: Learning (2018)
25. Lin, T.Y., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**, 318–327 (2020)
26. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: ECCV (2016)
27. Liu, W., Wang, X., Owens, J.D., Li, Y.: Energy-based out-of-distribution detection. ArXiv **abs/2010.03759** (2020)
28. Miller, A., Fisch, A., Dodge, J., Karimi, A.H., Bordes, A., Weston, J.: Key-value memory networks for directly reading documents. arXiv preprint arXiv:1606.03126 (2016)
29. Miller, D., Nicholson, L., Dayoub, F., Sünderhauf, N.: Dropout sampling for robust object detection in open-set conditions. 2018 IEEE International Conference on Robotics and Automation (ICRA) pp. 1–7 (2018)
30. Miller, D., Sünderhauf, N., Milford, M., Dayoub, F.: Class anchor clustering: A loss for distance-based open set recognition. 2021 IEEE Winter Conference on Applications of Computer Vision (WACV) pp. 3569–3577 (2021)
31. Miller, D., Sunderhauf, N., Milford, M., Dayoub, F.: Uncertainty for identifying open-set errors in visual object detection. IEEE Robotics and Automation Letters **7**, 215–222 (2022)
32. Neal, L., Olson, M.L., Fern, X.Z., Wong, W.K., Li, F.: Open set learning with counterfactual images. In: ECCV (2018)
33. Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J.V., Lakshminarayanan, B., Snoek, J.: Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In: NeurIPS (2019)
34. Oza, P., Patel, V.M.: C2ae: Class conditioned auto-encoder for open-set recognition. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 2302–2311 (2019)
35. Park, H., Noh, J., Ham, B.: Learning memory-guided normality for anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14372–14381 (2020)
36. Perera, P., Morariu, V.I., Jain, R., Manjunatha, V., Wigington, C., Ordonez, V., Patel, V.M.: Generative-discriminative feature representations for open-set recognition. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 11811–11820 (2020)

37. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: Unified, real-time object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 779–788 (2016)
38. Ren, J., Liu, P.J., Fertig, E., Snoek, J., Poplin, R., DePristo, M.A., Dillon, J.V., Lakshminarayanan, B.: Likelihood ratios for out-of-distribution detection. In: NeurIPS (2019)
39. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**, 1137–1149 (2015)
40. Steinhardt, J., Liang, P.S.: Unsupervised risk estimation using only conditional independence structure. Advances in Neural Information Processing Systems **29** (2016)
41. Sukhbaatar, S., Weston, J., Fergus, R., et al.: End-to-end memory networks. Advances in neural information processing systems **28** (2015)
42. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 9626–9635 (2019)
43. Vyas, A., Jammalamadaka, N., Zhu, X., Das, D., Kaul, B., Willke, T.L.: Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. In: ECCV (2018)
44. Wang, G., Li, W., Aertsen, M., Deprest, J.A., Ourselin, S., Vercauteren, T.K.M.: Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. Neurocomputing **335**, 34 – 45 (2019)
45. Weston, J., Chopra, S., Bordes, A.: Memory networks. arXiv preprint arXiv:1410.3916 (2014)
46. Yoshihashi, R., Shao, W., Kawakami, R., You, S., Iida, M., Naemura, T.: Classification-reconstruction learning for open-set recognition. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 4011–4020 (2019)
47. Yu, Q., Aizawa, K.: Unsupervised out-of-distribution detection by maximum classifier discrepancy. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 9517–9525 (2019)
48. Zhu, C., He, Y., Savvides, M.: Feature selective anchor-free module for single-shot object detection. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 840–849 (2019)
49. Zisselman, E., Tamar, A.: Deep residual flow for out of distribution detection. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 13991–14000 (2020)