

# Topology-Matching Normalizing Flows for Out-of-Distribution Detection in Robot Learning

Jianxiang Feng<sup>\*,1</sup>, Jongseok Lee<sup>2,3</sup>, Simon Geisler<sup>1</sup>, Stephan Günnemann<sup>1</sup>, Rudolph Triebel<sup>2,3</sup>

<sup>1</sup> Department of Informatics, Technical University of Munich (TUM)

<sup>2</sup> Institute of Robotics and Mechatronics, German Aerospace Center (DLR)

<sup>3</sup> Department of Informatics, Karlsruhe Institute of Technology (KIT)

jianxiang.feng@tum.de, {jongseok.lee, rudolph.triebel}@dlr.de,  
{geisler, guennemann}@in.tum.de

**Abstract:** To facilitate reliable deployments of autonomous robots in the real world, Out-of-Distribution (OOD) detection capabilities are often required. A powerful approach for OOD detection is based on density estimation with Normalizing Flows (NFs). However, we find that prior work with NFs attempts to match the complex target distribution topologically with naïve base distributions leading to adverse implications. In this work, we circumvent this topological mismatch using an expressive class-conditional base distribution trained with an information-theoretic objective to match the required topology. The proposed method enjoys the merits of wide compatibility with existing learned models without any performance degradation and minimum computation overhead while enhancing OOD detection capabilities. We demonstrate superior results in density estimation and 2D object detection benchmarks in comparison with extensive baselines. Moreover, we showcase the applicability of the method with a real-robot deployment.

**Keywords:** Normalizing Flows, Out-of-Distribution, Robotic Introspection

## 1 Introduction

The reliable identification of **Out-of-Distribution (OOD)** data, which is not well represented in the training set, poses a pressing challenge on the path towards trustworthy open-world robotic systems such as self-driving cars [1], delivery drones [2] or healthcare robots [3]. For example, with widespread adoption in the perception pipeline, existing object detectors have been reported to overconfidently misclassify an **OOD** object into a known class, which might obfuscate the decision-making module and eventually cause catastrophic consequences in safety-critical scenarios [1, 4, 5].

**Normalizing Flows (NFs)** are a popular class of generative models [6, 7, 8, 9] that may be used for **OOD** detection. **NFs** represent complex probability distributions [10] with a learnable series of transformations from a simple base distribution to a complex target distribution. However, **NFs**' expressivity [11, 12, 13] and numerical stability [14, 15] is limited by a fundamental constraint: the supports of the base and target distribution should preserve *similar topological properties* (Definition 3.3.10 in Runde [16]). The topological properties subsume different geometrical characteristics of the target distribution, including its continuity, the number of connected components, or the number of modes. Increasing the capacity of the transformation may mitigate this constraint. Yet, this raises computation and memory demands [11, 17, 12]. An alternative to overcome the topological mismatch is to increase the flexibility of the base distribution, which is surprisingly under-explored in the **OOD** detection literature.

Therefore, we propose to equip **NFs** with efficient but flexible base distributions for **OOD** detection in robot learning. Concretely, we replace the frequently used uni-modal Gaussian base distribution

---

\*: work done when working at DLR.

code: <https://github.com/DLR-RM>

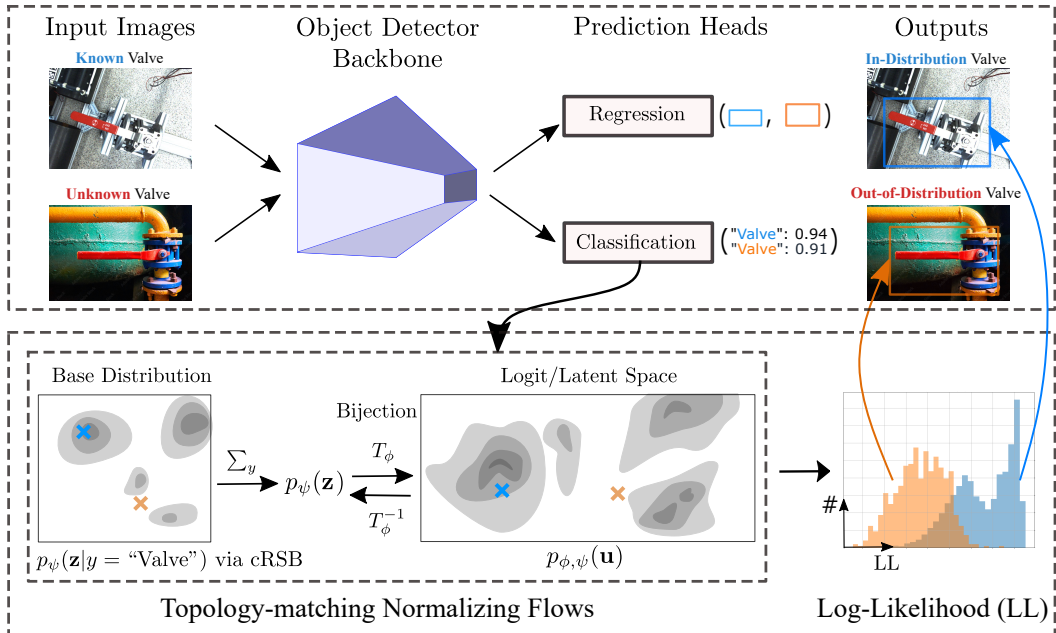


Figure 1: **The proposed architecture.** We overcome the topological mismatch problem in NFs to accurately model **In-Distribution (ID)** density. That is, the **Conditional Resampled Base Distributions (cRSB)** base distribution trained with **Information Bottleneck (IB)**  $p_\psi(\mathbf{z}|\mathbf{y})$  can, e.g., adapt the numbers of modes to match target distribution with complex topology. Then we can identify **OOD** objects by low predicted log-likelihoods more reliably (best viewed in color).

with the **cRSB**, a class-conditional version of a learnable base distribution for mitigating the topological problem in **NFs** – **Resampled Base Distributions (RSB)** [13]. **cRSB** can learn the required topological properties, like adapting the number of modes, to match the unknown topological structure of the latent class-specific target distribution (Figure 1). Moreover, we adapt our **cRSB** with an adapted **IB** objective [18] to balance fusing class-conditional information with the marginalized density estimation capabilities in **NFs**. **IB** [19] is an information-theoretic objective to incorporate task-specific details e.g. class conditions, which are commonly ignored in pure generative modeling. This delivers a topology in the base distribution that is more accurately aligned to the one in the target distribution (see Figure 3).

Our **OOD** detection approach using topology-matching **NFs** is powerful and yet resource-efficient for open-set object detection. It is applicable to diverse object detectors (e.g., Faster-RCNN [20] and Yolov7 [21] used in this work) with minor changes and no loss of prediction performance. Moreover, our approach is sampling-free, i.e., only a single forward pass is required for efficient test-time inference while keeping the space memory tractable. As a result, our method is suitable for robotic applications that require a fast and robust perception module. We empirically show the state-of-the-art performance of the proposed idea using synthetic density estimation and 2D object detection tasks against extensive baselines. To further validate the applicability in robotics, we examine an object detector equipped with the proposed method on an exemplary inspection and maintenance aerial robot, showing the practical benefits of negligible memory and run-time overhead.

**Contributions.** Our main contribution is a **NFs**-based **OOD** detection method that overcomes the topological constraints while taking class-conditional information into account. We show that training with **IB** yields effective representation with superior **OOD** detection capabilities. We conduct a comprehensive empirical evaluation using both synthetic density estimation and public object detection datasets followed by a real-world robot deployment, which overall shows the effectiveness of the proposed approach.

## 2 Methodology

**Problem Formulation** Given an image  $\mathbf{x} \in \mathcal{X}$  and a trained object detector  $F_\theta$  that localizes a set of objects with corresponding bounding box coordinates  $\mathbf{b}_i \in \mathcal{R}^4$  as well as class label  $y_i \in \mathcal{Y} = \{1, 2, \dots, C\}$ , the task is to distinguish if  $(\mathbf{x}, \mathbf{b}_i, y_i)$  is **ID**, i.e., drawn from  $\mathcal{P}_{id}$ , or **OOD**, i.e., belongs to the unknown distribution  $\mathcal{P}_{ood}$ . For conciseness, from now on we omit the suffix  $i$  and use  $y$  to denote the class label without further notice. As discussed, a powerful **OOD** detection can be obtained via density estimation using **NFs**. This density estimator identifies **OOD** objects with low likelihoods after being trained *only* on data drawn from  $\mathcal{P}_{id}$ . Following relevant prior [22, 23], we use the semantically rich logit space (pre-softmax layer) for density estimation. To note that, our method can be readily applied to other (high-dimensional) latent feature spaces.

**NFs** are known to be universal distribution approximators [10]. That is, they can model a complex target distribution  $p(\mathbf{u})$  on a space  $\mathcal{R}^d$  by defining  $\mathbf{u}$  as a transformation  $T_\phi : \mathcal{R}^d \rightarrow \mathcal{R}^d$  from a well-defined base distribution  $p_\psi(\mathbf{z})$ , where  $\phi$  and  $\psi$  are model parameters, respectively:

$$\mathbf{u} = T_\phi(\mathbf{z}) \text{ where } \mathbf{z} \sim p_\psi(\mathbf{z}) \quad (1)$$

where  $\mathbf{z} \in \mathcal{R}^d$  and  $p_\psi$  is commonly chosen as a uni-modal Gaussian. By designing  $T_\phi$

to be a *diffeomorphism*, that is, a bijection where both  $T_\phi$  and  $T_\phi^{-1}$  are differentiable, We can compute the likelihood of the input  $\mathbf{u}$  *exactly* based on the change-of-variables formula [24]:

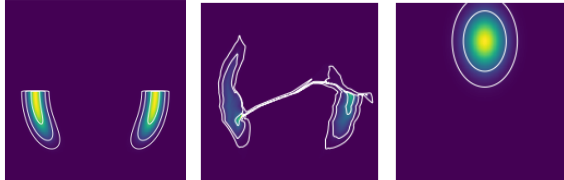
$$p_{\phi,\psi}(\mathbf{u}) = p_\psi(T_\phi^{-1}(\mathbf{u})) | \det(J_{T_\phi^{-1}}(\mathbf{u})) |, \quad (2)$$

where  $J_{T_\phi^{-1}}(\mathbf{u}) \in \mathcal{R}^{d \times d}$  is the Jacobian of the inverse  $T_\phi^{-1}$  with respect to  $\mathbf{u}$ . When the target distribution is unknown but samples thereof are available, we can estimate the parameter  $(\phi, \psi)$  by minimizing the forward **Kullback-Leibler Divergence (KLD)**, which is equivalent to maximizing the expected **Log-Likelihood (LL)**.

**Topological Mismatch** However, since the base distribution  $p_\psi(\mathbf{z})$  is usually a uni-modal Gaussian (e.g. Figure 2c) and  $T_\phi$  is a diffeomorphism, problems arise for modeling data distribution with different topological properties. These include well-separated multi-modal distributions or distributions with disconnected components (e.g., Figure 2a). For example, one can see that this leads to density filaments between the modes in Figure 2b. Cornish et al. [11] have shown that flows require a bijection with *infinite bi-Lipshitz constant* when modeling a target distribution with disconnected support using a unimodal base distribution. Besides the diminishing modeling performance, this renders the bijection to be numerically "non-invertible", thus, causing optimization instability during training and unreliability of likelihood calculation [14].

### 2.1 Conditional Resampled Base Distributions

One possible partial mitigation is by enriching the expressiveness of the flows. For example, by (a) increasing the number of layers or parameters, (b) using more complex base distributions, or (c) employing multiple **NFs**, e.g., mixtures of **NFs**. It is important to note that especially (a) and (c) may escalate the computational cost and memory burden. Moreover, scaling the normalizing flow's expressivity, (a) or (c), often does not increase the stability of the optimization [15] or the likelihood calculation. For these reasons, we pursue (b) and attempt to compensate for the complexity of the transformation with the elasticity of the base distribution. In other words, we use a more flexible but efficient base distribution to trade off a costly but sufficiently expressive bijection of the normalizing flow. This way we aim to capture desirable topological properties of the target distribution [17]. Following the prior work [25], to model the fidelitous distribution of data with task-specific conditions, e.g. class labels, we use a class-conditional base distribution. This way we get similar benefits like combining multiple conditional flows (c), however, without having to burden the computational



(a)  $p(\mathbf{u}|y=0)$  (b)  $p_{\phi,\psi}(\mathbf{u}|y=0)$  (c)  $p_\psi(\mathbf{z}|y=0)$   
 Figure 2: Filament connect modes in the modeled class-conditional distribution (b) if using (trainable) uni-modal base (c) for the multi-modal target (a).

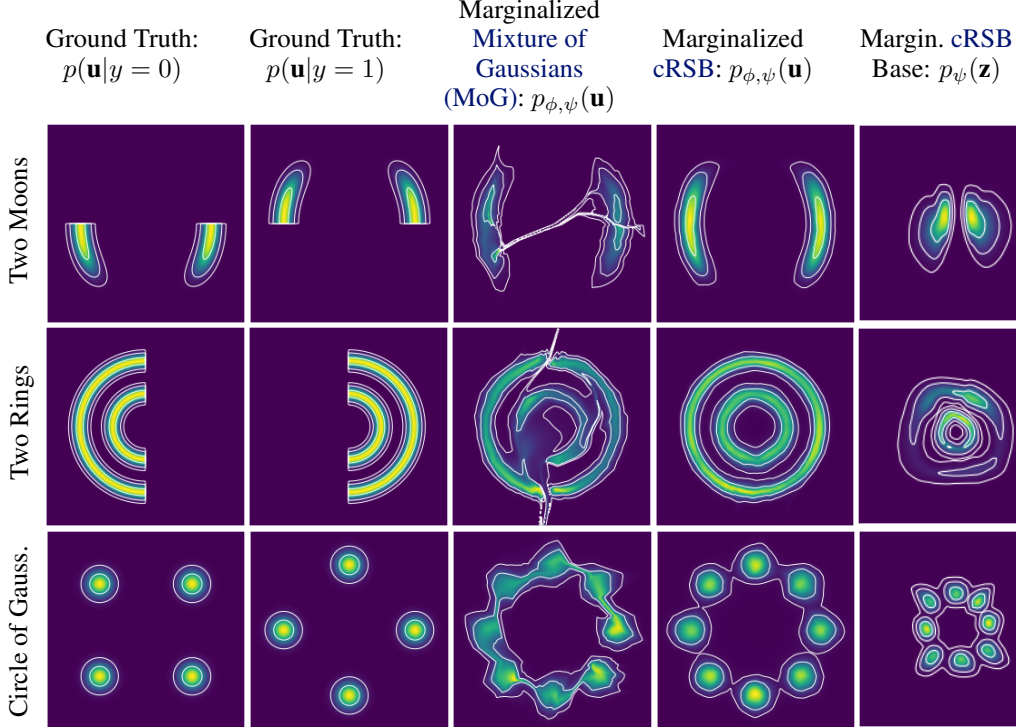


Figure 3: Visualization of density estimation using Real NVP with class conditional MoG, where each class is modeled by a uni-modal Gaussian, and cRSB as well as the class-marginalized density for the base distribution of cRSB.

cost on marginalization over classes. This is because, with (c), this operation requires repeated evaluation of the flows when each flow of the NFs mixture is class-conditional [26]. Even though a class-conditional distribution can specialize on a smaller fraction of the dataset containing similar instances, it will manifest in a multi-modal distribution.

Therefore, we propose to capture the complex topological properties in the target distribution with a more expressive base distribution instead of the uni-model Gaussian. To the end, we introduce cRSB by extending a powerful unconditional base distribution RSB [13] with class-conditional modeling. RSB deforms a uni-modal Gaussian in a learnable manner to obtain more complex distributions via Learned accept/reject sampling (LARS) [27]. LARS iteratively re-weighs samples drawn from a proposal distribution  $\pi(\mathbf{z})$ , e.g. a standard Gaussian, through a learned acceptance function  $a_{\psi} : \mathcal{R}^d \rightarrow [0, 1]$ . To reduce the computation cost in practice, this process is truncated by accepting the  $T$ -th samples if the previous  $T - 1$  samples get rejected. To take into account class-conditional information, we conditionalize the learnable acceptance function  $a_{\psi}(\mathbf{z}|y)$ . As a result, we have the conditional base distribution:

$$p_{\psi}(\mathbf{z}|y) = (1 - \alpha_T) \frac{a_{\psi}(\mathbf{z}|y)\pi(\mathbf{z})}{Z_y} + \alpha_T \pi(\mathbf{z}), \quad (3)$$

where  $a_{\psi} : \mathcal{R}^d \rightarrow [0, 1]^C$  and  $\alpha_T = (1 - Z_y)^{T-1}$ , where  $Z_y \in \mathcal{R}$  is the normalization factor for  $a_{\psi}(\mathbf{z}|y)\pi(\mathbf{z})$ . This factor can be estimated via Monte Carlo Sampling.

In Figure 3, we contrast the density estimation capabilities of NFs with the common MoG [8, 25] base distribution and our cRSB on three tasks with class-conditional structure using an appropriate learning objective (see next section). We find that our cRSB learns appropriate topology-matching base distributions (right outer column) and as a result, the respective NFs do not have adverse effects like filaments between the modes.

## 2.2 Training with Information Bottleneck

Unfortunately, directly training NFs with a conditional base distribution can lead to underperformance as observed in experiments (see Table 2 and appendix) and reported by Fetaya et al. [25].

We attribute this to the lack of explicit control for the balance between generative and discriminative modeling in the likelihood-based training objective of **NFs**. To alleviate this, we train the normalizing flow with a class-conditional base distribution using the **IB** objective [19]. To abuse the notations, we denote random variables by capital letters such as  $U, Z, Y$ , and their realizations by lowercase letters such as  $\mathbf{u}, \mathbf{z}, y$ . The **IB** minimizes the **Mutual Information (MI)**  $I(U, Z)$  between  $U$  and  $Z$ , while simultaneously maximizing the **MI**  $I(Z, Y)$  between  $Z$  and  $Y$ . Intuitively, the **IB** trades off between the objectives of modeling the class conditional information  $p(\mathbf{u}|y)$  with the marginalized density  $p(\mathbf{u})$ , thus allowing to leverage the class-conditional structure to facilitate more effective density estimation for data characterized with semantic classes.

However, the **IB** is not directly applicable to latent class-conditional distributions in **NFs** since the bijection  $T_\phi$  is lossless by design. Thus, for trading off the class-conditional information with density estimation capabilities, we adapt the approach proposed by Ardizzone et al. [18] for our **cRSB**. Specifically, we inject a small amount of noise  $\epsilon$  into the input  $U$  and hence  $Z_\epsilon = T_\phi^{-1}(U + \epsilon)$ . Further we define an asymptotically exact version of **MI**, namely the **Mutual Cross-Information (CI)** (more details in appendix):

$$\mathcal{L}_{\text{IBNF}} = \text{CI}(U, Z_\epsilon) - \beta \text{CI}(Z_\epsilon, Y) \quad (4)$$

$$\text{CI}(U, Z_\epsilon) = \mathbb{E}_{p(\mathbf{u}), p(\epsilon)} \left[ -\log \sum_{y'} p_\psi(\mathbf{z}_\epsilon | y') - \log |\det(J_{T_\phi^{-1}}(\mathbf{u} + \epsilon))| \right], \quad (5)$$

$$\text{CI}(Z_\epsilon, Y) = \mathbb{E}_{p(y)} \left[ \log \frac{p_\psi(\mathbf{z}_\epsilon | y)p(y)}{\sum_{y'} p_\psi(\mathbf{z}_\epsilon | y')p(y')} \right], \quad (6)$$

where  $\mathbf{z}_\epsilon = T_\phi^{-1}(\mathbf{u} + \epsilon)$ ,  $p(\epsilon) = \mathcal{N}(0, \sigma^2 \mathcal{I}_d)$  is a zero-meaned Gaussian with variance  $\sigma^2$ , and  $\beta$  trades off class information and generative density estimation. With flexible conditional base distributions defined in Eq. 3, we can train the *topology-matching* **NFs** with **IB** by substituting **cRSB** into the conditional base probability  $p_\psi(\mathbf{z}|y)$  in Eq. 5 and 6. More noteworthy, we observed that the **IB** is able to regularize the acceptance rate learning for **cRSB** to better assimilate the topological structure of the target distribution, leading to an overall improved performance on accurately approximating the complex target distribution (see Figure 4).

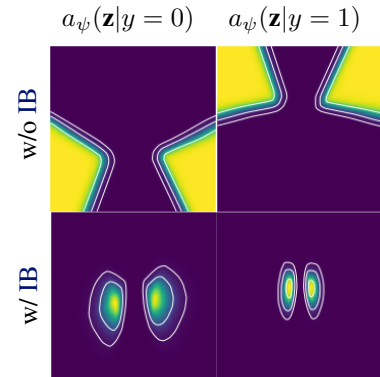


Figure 4: **cRSB** acceptance rate  $a_\psi(\mathbf{z})$  w/o and w/ **IB** training for Two Moons.

### 2.3 Detecting OOD Objects

During test time, we detect the **OOD** data based on the predicted **Log-Likelihood (LL)**. To note that, only one forward pass is required to evaluate the acceptance function in **cRSB**. Practically, we use Monte Carlo sampling to estimate the normalization factor  $Z$  offline so that no additional computation required for this during inference. We *marginalize* the density over classes for the base distribution defined in Eq. 3 and compute the final **LL** given the logits  $\mathbf{u}'$  from the test image:

$$\text{LL}_{\text{test}}(\mathbf{u}') = \log \sum_{y'} (p_\psi(T_\phi^{-1}(\mathbf{u}')|y')) + \log |\det(J_{T_\phi^{-1}}(\mathbf{u}'))|. \quad (7)$$

We then expect **LL** for **ID** objects to be higher than **OOD** ones.

## 3 Related Work

**Normalizing Flows NFs** [28] are a popular class of deep generative models. **NFs** have shown applicability in a variety of areas such as image generation [29, 30], uncertainty estimation [31, 32, 33] and **OOD** detection [6, 34, 35]. For **NFs**, one trend has been designing expressive flow-based architectures. Notable examples are affine coupling flows [29, 30], auto-regressive flows [36, 37], invertible ResNet blocks [38] and ODEs-based maps [39]. The major focus of these works is on reducing computing requirements for Jacobian computations while ensuring that each mapping is

invertible. Another research direction, currently emerging, is on addressing the topological mismatch [28, 10] of **NFs**. Targeting this problem, some existing works attempt to increase the learning capacity of the transformation via mixture models [26], latent variable models [11, 40] or injecting carefully specified randomness [41, 12]. These methods may be limited in their applicability to robotics because they either increase memory consumption by expanding the width of transformations or approximate the exact likelihood. Recently, these constraints have been addressed by improving the expressivity of the base distribution [13, 17]. In this paper, we build upon this class of methods since they only add slight computation overheads and thus are well suited for applications in robotics.

**Normalizing Flows for OOD Detection** **NFs** have been widely adapted for **OOD** detection due to its superior density estimation [42]. For example, though with some counter-intuitive observations on raw data space [34], **NFs** have demonstrated encouraging **OOD** detection results with additional refinements for raw data [43, 44, 45] or directly based on task-relevant feature embeddings [6, 7, 46, 47]. In this work, we directly apply **NFs** on the feature space. To note that, another principle direction is to estimate the error bound for this task [48]. Recently hybrid models [49, 7, 50] have shown remarkable performance gain on **OOD** detection by modeling the joint distribution of both data and its class labels. Such works suggest that class labels can provide useful information. However, directly performing class conditional modeling with **NFs** for **OOD** detection results in performance degradation. Tishby et al. [19], Ardizzone et al. [18] mitigate such performance degradation by utilizing **IB** for training **NFs**. This explicitly controls the trade-off between generative and discriminative modeling [9]. However, these works on **OOD** detection utilize **NFs** without much concern for the fundamental topological problem as the first citizen. Therefore, complementary to these approaches, we examine the problem of topological mismatch of **NFs** for **OOD** detection.

**OOD Detection in Object Detectors** **OOD** detection research has focused on image classification [42], which may be limited in relevance to robotic vision. In robotics, we may often need both categorization and localization of objects of interest. Therefore, we focus on object detection in open-set conditions here. In this domain, uncertainty estimation [51] has been considered propitious for **OOD** detection but suffered from computation burdens on runtime [52, 53, 54, 55] or memory costs [56]. To address this, instead of directly applying uncertainty estimation techniques for object detection [54, 2], another popular approach is to explicitly formulate the problem as **OOD** detection tasks [23, 57, 8, 58, 59]. Amongst them, **NFs** has been utilized as an expressive density estimator [8, 58]. However, despite the encouraging results, these approaches have not examined the problem of topological mismatch in **NFs**. As this might prevent additional performance improvements, this work examines the topology-matching **NFs** for **OOD** detection in object detectors.

## 4 Experiments

We next demonstrate the efficacy of our method. First, we evaluate on synthetic density estimation for distributions with distinct topological properties. We then evaluate the **OOD** detection performance on two object-detection data-sets adapted from their public counterparts [60, 61] for open-set (OS) experiments: Pascal-VOC-OS and MS-COCO-OS based on Glow [30] and a pre-trained Faster-RCNN [20] provided by Miller et al. [23] for a fair comparison. To showcase the practicality, we deploy the one-stage object detector Yolov7 [21] equipped with the proposed method on a real aerial manipulation robot along with the run-time and memory analysis. We empirically found that, to parameterize the acceptance function in **LARS**, a simple multi-layer perceptron (MLP) (2x128 for density estimation and 3x128 for object detection) is sufficient. We select the hyper-parameters (e.g.,  $T, \epsilon, \sigma, \beta$ ) based on the validation set. More details can be found in the supplementary materials.

**Datasets and Metrics** For density estimation, there are three synthetic datasets: two moons, two rings, and a circle of Gaussians. We employ the **KLD** between the target and the model distributions to measure the performance. For **OOD** detection, since existing object detection datasets are not ready for fair evaluation [4], we strictly follow the experimental protocol in [23]. For real robot deployment, we generate  $2k$  synthetic images of two objects (a valve and a crawler robot) rendered

based on their CAD models and additionally labeled  $2k$  real images.  $1k$  synthetic images are used for training and another  $1k$  for testing with all real images. We use the **Area Under Receiver Operation Curve (AUROC)** and the **True Positive Rate (TPR)** at different **False Positive Rate (FPR)** (5%, 10%, 20%) as metrics for this task, as they represent the performance of the potential operating points for safety-critical applications, which requires the **FPR** to be sufficiently low.

#### 4.1 Density Estimation

We compare the density estimation performance in Table 1 and provide qualitative results in Figure 3. We find that the **cRSB** base distribution consistently outperforms the class-conditional **Mixture of Gaussians (MoG)**. The performance improvement by **cRSB** can be generalized across two different **NFs** architectures, i.e. Real NVP and NSF.

Table 1: Performance on density estimation for different flow architectures w.r.t. **KLD**, i.e.,  $D_{KL}(p(\mathbf{u}, y) || p_{\phi, \psi}(\mathbf{u}, y))$ . Better base distribution is highlighted in bold.

Flow architecture Base distribution	Real NVP		NSFs	
	MoG_IB	cRSB_IB	MoG_IB	cRSB_IB
Two Moons	1.179	<b>1.066</b>	0.909	<b>0.906</b>
Two Rings	2.032	<b>1.704</b>	1.647	<b>1.602</b>
Circle of Gaussians	2.335	<b>1.667</b>	1.766	<b>1.653</b>

#### 4.2 OOD Detection in Object Detection

We compare our method (**cRSB\_IB**) with both flow-based and non-flow-based approaches. The latter consists of Mahalanobis Distance (**MD**) [62], Relative Mahalanobis Distance (**RMD**) [63], **GMMDet** [23], **Softmax**, **Entropy** and, their **Deep Ensemble** variants with five models [56]. Among flow-based approaches, we have six different base distributions, including unconditional ones (uni-modal Gaussian, **MoG**, **RSB**) and their conditional variants (**MoG\_CLS**, **cRSB\_CLS**) [25] and **MoG** trained with **IB** (**MoG\_IB**) [8, 18]. From Table 2, we can observe that flows with uni-modal Gaussian are able to provide satisfactory performance, i.e., better than most of non flow-based baselines, while flows with more expressive base distributions such as **MoG** and **RSB** can bring more benefits on Pascal-VOC-OS than MS-COCO-OS. When trained with **IB**, the more flexible conditional base distribution (**cRSB\_IB**) can mostly have greater performance gains (on both Pascal VOC and COCO) than its strong competitor (**MoG\_IB**) (only on COCO) in comparison with their counterparts without **IB** (**MoG\_CLS**). These results demonstrate the effectiveness of **cRSB** with **IB** for **OOD** detection in complicated 2D object detection tasks. We further provide the visualization from data before and after the flow transformation with different base distributions in Figure 5, evidencing the ability of matching complex topology of the target data distribution with **cRSB**.

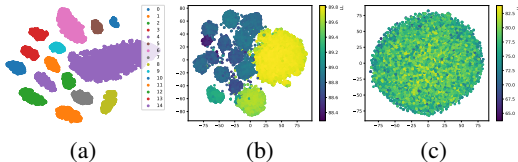


Figure 5: t-SNE visualization for (a) feature embeddings from the object detector (b) latents of the proposed learned base distribution **cRSB** and (c) the uni-modal Gaussian on the training set of Pascal-VOC-OS.

#### 4.3 Real Robot Deployment

Next, we validate the applicability in an application of robotic inspection and maintenance, where it is crucial to avoid false positives of **OOD** objects that appear routinely in outdoor environments. In this experiment, we train **Yolov7** with only synthetic images of two objects (a valve and a crawler robot) and deploy on the robot around only real objects. The task is to identify the falsely detected real objects as **OOD** since they are from a distribution different to the synthetic ones. Besides, the performance drop when compared with Table 2 is potentially attributed to the "closer" **OOD** data because the synthetic images are rendered in a highly photorealistic manner. However, our method still outperforms other baseline approaches in Figure 6c, where ours can notably achieve higher **TPR** around the low **FPR** region, which are commonly used as operating points for the robot. Computational efficiency is another important requirement. We compare the runtime and space memory consumption against a vanilla **Yolov7** using the **NVIDIA's** embedded GPU called **Jetson Orin** in Figure 6. The results indicate that the computational overhead of having an **OOD** detector is

Table 2: **OOD** detection performance on Pascal-VOC-OS and MS-COCO-OS datasets for different methods based on the Faster-RCNN from 3 random runs. The highest values are marked in **bold** and the second highest in *italics*.

	Pascal-VOC-OS				MS-COCO-OS			
	AUROC	TPR at			AUROC	TPR at		
		5%FPR	10%FPR	20%FPR		5%FPR	10%FPR	20%FPR
Softmax	0.901	60.1	72.8	83.1	0.882	61.3	70.6	78.1
Entropy	0.905	59.8	72.9	82.9	0.903	61.2	70.6	80.2
MD [62]	0.9	54.1	68.8	83.3	0.902	57.2	71.4	85.5
RMD [63]	0.838	15.2	28.4	77.4	0.531	1.7	2.6	7.1
Ensemble Softmax [56]	0.885	47.8	72.6	83.1	0.898	66.2	73.5	82.3
Ensemble Entropy [56]	0.887	47.8	72.5	83.1	0.906	66.2	73.5	82.3
GMMDet [23]	0.931	70.7	80.5	89.3	0.924	69.5	80.2	87.9
Flows Gaussian	0.915 ± 0.002	72.2 ± 0.75	77.8 ± 0.89	86.1 ± 0.67	0.924 ± 0.001	68.2 ± 0.73	81.2 ± 0.61	89.4 ± 0.04
Flows MoG	0.919 ± 0.002	69.0 ± 2.4	77.0 ± 2.5	86.5 ± 1.2	0.925 ± 0.001	68.3 ± 0.30	80.5 ± 0.50	89.6 ± 0.05
Flows RSB [13]	0.924 ± 0.003	72.8 ± 0.88	79.3 ± 1.0	87.1 ± 0.82	0.925 ± 0.001	68.6 ± 0.87	81.3 ± 0.31	89.5 ± 0.34
Flows MoG_CLS [25]	0.923 ± 0.001	69.2 ± 1.5	78.2 ± 1.3	88.5 ± 0.82	0.930 ± 0.001	68.5 ± 0.73	82.2 ± 0.31	89.7 ± 0.30
Flows MoG_IB [8]	0.934 ± 0.002	73.1 ± 1.3	79.6 ± 0.6	87.8 ± 0.2	0.924 ± 0.002	71.1 ± 0.9	79.6 ± 0.46	88.6 ± 0.63
Flows cRSB_CLS	0.919 ± 0.001	72.5 ± 0.37	78.8 ± 0.27	86.8 ± 0.42	0.924 ± 0.001	68.3 ± 0.14	81.1 ± 0.30	89.3 ± 0.18
Flows cRSB_IB (ours)	<b>0.946</b> ± 0.003	<b>78.5</b> ± 0.97	<b>84.0</b> ± 0.83	<b>90.8</b> ± 0.76	<b>0.934</b> ± 0.002	<b>73.3</b> ± 2.0	<b>84.3</b> ± 0.40	<b>91.3</b> ± 0.28

relatively small when compared to the vanilla Yolov7. Overall, these experiments validate our claim that our method features efficient runtime inference and cost-effective memory consumption.

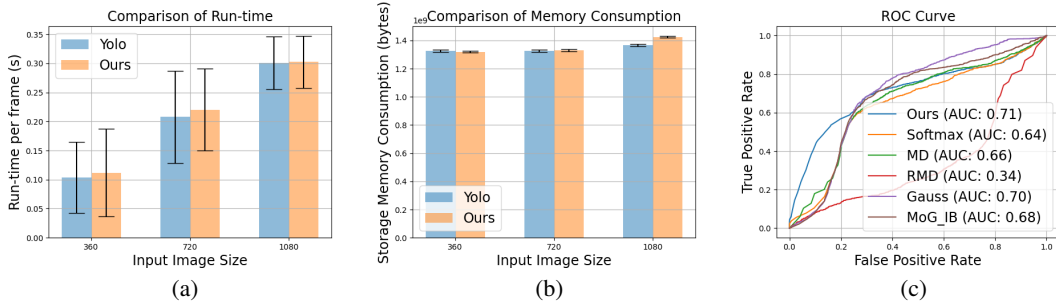


Figure 6: Results from experiments on a real robot. Run-time, memory consumption, and ROC curve are reported. Compared to the vanilla Yolov7, the proposed method does not yield significant computational costs, while providing performance gains in **OOD** detection.

## 5 Limitations

The proposed method is envisioned to work on feature embeddings instead of raw data to counteract the **NFs** artifacts of assigning higher likelihoods to **OOD** data [10]. This leads to two limitations. First, it's can't directly applied to the tasks/models that could not provide useful feature embeddings extracted from the raw data. Second, its performance is restricted to the quality of features. As reported by previous work [23, 8], learning more compact and centralized features can often lead to increased performance for **OOD** detection while feature collapse can be harmful to **OOD** detection. Besides, there are two limitations during deployment. The first is the prolonged initialization time for calculating the normalization factor in **LARS** based on Monte Carlo sampling. This might not be friendly for applications that require instant response at the beginning. Moreover, the current version of the proposed method does not consider the sequential nature of observations at deployment.

## 6 Conclusion

To endow robots with introspective awareness against **OOD** data, we propose the **NFs** equipped with effective yet lightweight **cRSB** and train with **IB** objective. Such **NFs** are able to mitigate the fundamental topological mismatch problem, facilitating more effective **OOD** detection capabilities. We present empirical evidence that the proposed method achieves superior performance both quantitatively and qualitatively. To demonstrate the run-time efficiency and minimum memory overheads, we deployed on a real-robot system. Overall, we hope that the results of our work stemming from an enriched base distribution can push forward the direction of **NFs**-based **OOD** detection in robot learning.



## Acknowledgments

We thank the anonymous reviewers for their thoughtful feedback. Jianxiang Feng and Simon Geisler are supported by the Munich School for Data Science (MUDS). Rudolph Triebel and Stephan Gunnemann are members of MUDS.

## References

- [1] J. Nitsch, M. Itkina, R. Senanayake, J. Nieto, M. Schmidt, R. Siegwart, M. J. Kochenderfer, and C. Cadena. Out-of-distribution detection for automotive perception. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 2938–2943. IEEE, 2021.
- [2] J. Lee, J. Feng, M. Humt, M. G. Müller, and R. Triebel. Trust your robots! predictive uncertainty estimation of neural networks with sparse gaussian processes. In *Conference on Robot Learning*, pages 1168–1179. PMLR, 2022.
- [3] J. Feng, J. Lee, M. Durner, and R. Triebel. Bayesian active learning for sim-to-real robotic perception. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10820–10827. IEEE, 2022.
- [4] A. Dhamija, M. Gunther, J. Ventura, and T. Boulton. The overlooked elephant of object detection: Open set. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.
- [5] R. Sinha, A. Sharma, S. Banerjee, T. Lew, R. Luo, S. M. Richards, Y. Sun, E. Schmerling, and M. Pavone. A system-level view on out-of-distribution data in robotics. *arXiv preprint arXiv:2212.14020*, 2022.
- [6] P. Kirichenko, P. Izmailov, and A. G. Wilson. Why normalizing flows fail to detect out-of-distribution data. *Advances in neural information processing systems*, 33:20578–20589, 2020.
- [7] H. Zhang, A. Li, J. Guo, and Y. Guo. Hybrid models for open set recognition. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 102–117. Springer, 2020.
- [8] R. Li, C. Zhang, H. Zhou, C. Shi, and Y. Luo. Out-of-distribution identification: Let detector tell which i am not sure. In *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part X*, pages 638–654. Springer, 2022.
- [9] R. Mackowiak, L. Ardizzone, U. Kothe, and C. Rother. Generative classifiers as a basis for trustworthy image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2971–2981, 2021.
- [10] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *The Journal of Machine Learning Research*, 22(1):2617–2680, 2021.
- [11] R. Cornish, A. Caterini, G. Deligiannidis, and A. Doucet. Relaxing bijectivity constraints with continuously indexed normalising flows. In *International conference on machine learning*, pages 2133–2143. PMLR, 2020.
- [12] H. Wu, J. Köhler, and F. Noé. Stochastic normalizing flows. *Advances in Neural Information Processing Systems*, 33:5933–5944, 2020.
- [13] V. Stimper, B. Schölkopf, and J. M. Hernández-Lobato. Resampling base distributions of normalizing flows. In *International Conference on Artificial Intelligence and Statistics*, pages 4915–4936. PMLR, 2022.

- [14] J. Behrmann, P. Vicol, K.-C. Wang, R. Grosse, and J.-H. Jacobsen. Understanding and mitigating exploding inverses in invertible neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 1792–1800. PMLR, 2021.
- [15] P. Hagemann and S. Neumayer. Stabilizing invertible neural networks using mixture models. *Inverse Problems*, 37(8):085002, 2021.
- [16] V. Runde. *A taste of topology*. Springer, New Delhi, 2005. Includes bibliographical references and index.
- [17] P. Jaini, I. Kobyzev, Y. Yu, and M. Brubaker. Tails of lipschitz triangular flows. In *International Conference on Machine Learning*, pages 4673–4681. PMLR, 2020.
- [18] L. Ardizzone, R. Mackowiak, C. Rother, and U. Köthe. Training normalizing flows with the information bottleneck for competitive generative classification. *Advances in Neural Information Processing Systems*, 33:7828–7840, 2020.
- [19] N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.
- [20] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.
- [21] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7464–7475, 2023.
- [22] H. Wei, R. Xie, H. Cheng, L. Feng, B. An, and Y. Li. Mitigating neural network overconfidence with logit normalization. *International Conference on Machine Learning (ICML)*, 2022.
- [23] D. Miller, N. Sünderhauf, M. Milford, and F. Dayoub. Uncertainty for identifying open-set errors in visual object detection. *IEEE Robotics and Automation Letters*, 7(1):215–222, 2021.
- [24] V. I. Bogachev and M. A. S. Ruas. *Measure theory*, volume 1. Springer, 2007.
- [25] E. Fetaya, J.-H. Jacobsen, W. Grathwohl, and R. Zemel. Understanding the limitations of conditional generative models. *arXiv preprint arXiv:1906.01171*, 2019.
- [26] J. Postels, M. Liu, R. Spezialetti, L. Van Gool, and F. Tombari. Go with the flows: Mixtures of normalizing flows for point cloud generation and reconstruction. In *2021 International Conference on 3D Vision (3DV)*, pages 1249–1258. IEEE, 2021.
- [27] M. Bauer and A. Mnih. Resampled priors for variational autoencoders. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 66–75. PMLR, 2019.
- [28] I. Kobyzev, S. J. Prince, and M. A. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE transactions on pattern analysis and machine intelligence*, 43(11): 3964–3979, 2020.
- [29] L. Dinh, J. Sohl-Dickstein, and S. Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.
- [30] D. P. Kingma and P. Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.
- [31] B. Charpentier, D. Zügner, and S. Günnemann. Posterior network: Uncertainty estimation without ood samples via density-based pseudo-counts. *Advances in Neural Information Processing Systems*, 33:1356–1367, 2020.

- [32] B. Charpentier, O. Borchert, D. Zügner, S. Geisler, and S. Günnemann. Natural posterior network: Deep bayesian uncertainty for exponential family distributions. *arXiv preprint arXiv:2105.04471*, 2021.
- [33] J. Postels, H. Blum, Y. Strümler, C. Cadena, R. Siegwart, L. Van Gool, and F. Tombari. The hidden uncertainty in a neural networks activations. *arXiv preprint arXiv:2012.03082*, 2020.
- [34] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan. Do deep generative models know what they don't know? *arXiv preprint arXiv:1810.09136*, 2018.
- [35] S. K. Lind, R. Triebel, L. Nardi, and V. Krueger. Out-of-distribution detection for adaptive computer vision. *arXiv preprint arXiv:2305.09293*, 2023.
- [36] C.-W. Huang, D. Krueger, A. Lacoste, and A. Courville. Neural autoregressive flows. In *International Conference on Machine Learning*, pages 2078–2087. PMLR, 2018.
- [37] C. Durkan, A. Bekasov, I. Murray, and G. Papamakarios. Neural spline flows. *Advances in neural information processing systems*, 32, 2019.
- [38] R. T. Chen, J. Behrmann, D. K. Duvenaud, and J.-H. Jacobsen. Residual flows for invertible generative modeling. *Advances in Neural Information Processing Systems*, 32, 2019.
- [39] W. Grathwohl, R. T. Chen, J. Bettencourt, I. Sutskever, and D. Duvenaud. Ffjord: Free-form continuous dynamics for scalable reversible generative models. *arXiv preprint arXiv:1810.01367*, 2018.
- [40] L. Dinh, J. Sohl-Dickstein, H. Larochelle, and R. Pascanu. A rad approach to deep mixture models. *arXiv preprint arXiv:1903.07714*, 2019.
- [41] D. Nielsen, P. Jains, E. Hoogeboom, O. Winther, and M. Welling. Survae flows: Surjections to bridge the gap between vaes and flows. *Advances in Neural Information Processing Systems*, 33:12685–12696, 2020.
- [42] J. Yang, K. Zhou, Y. Li, and Z. Liu. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*, 2021.
- [43] J. Ren, P. J. Liu, E. Fertig, J. Snoek, R. Poplin, M. Depristo, J. Dillon, and B. Lakshminarayanan. Likelihood ratios for out-of-distribution detection. *Advances in neural information processing systems*, 32, 2019.
- [44] E. Nalisnick, A. Matsukawa, Y. W. Teh, and B. Lakshminarayanan. Detecting out-of-distribution inputs to deep generative models using typicality. *arXiv preprint arXiv:1906.02994*, 2019.
- [45] D. Jiang, S. Sun, and Y. Yu. Revisiting flow generative models for out-of-distribution detection. In *International Conference on Learning Representations*, 2022. URL <https://openreview.net/forum?id=6y2KBh-0Fd9>.
- [46] B. Charpentier, C. Zhang, and S. Günnemann. Training, architecture, and prior for deterministic uncertainty methods. *arXiv preprint arXiv:2303.05796*, 2023.
- [47] J. Feng, M. Atad, I. V. Rodriguez Brena, M. Durner, and R. Triebel. Density-based feasibility learning with normalizing flows for introspective robotic assembly. In *18th Robotics: Science and System 2023 Workshops*, 2023. URL <https://elib.dlr.de/195846/>.
- [48] G. Chou, N. Ozay, and D. Berenson. Safe output feedback motion planning from images via learned perception modules and contraction theory. In *International Workshop on the Algorithmic Foundations of Robotics*, pages 349–367. Springer, 2022.

- [49] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan. Hybrid models with deep and invertible features. In *International Conference on Machine Learning*, pages 4723–4732. PMLR, 2019.
- [50] S. Cao and Z. Zhang. Deep hybrid models for out-of-distribution detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4723–4733, 2022. doi:10.1109/CVPR52688.2022.00469.
- [51] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, pages 1–77, 2023.
- [52] D. Miller, L. Nicholson, F. Dayoub, and N. Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3243–3249. IEEE, 2018.
- [53] J. Lee, M. Humt, J. Feng, and R. Triebel. Estimating model uncertainty of neural networks in sparse information form. In *International Conference on Machine Learning*, pages 5702–5713. PMLR, 2020.
- [54] A. Harakeh, M. Smart, and S. L. Waslander. Bayesod: A bayesian approach for uncertainty estimation in deep object detectors. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 87–93. IEEE, 2020.
- [55] J. Feng, M. Durner, Z.-C. Márton, F. Bálint-Benczédi, and R. Triebel. Introspective robot perception using smoothed predictions from bayesian neural networks. In *Robotics Research: The 19th International Symposium ISRR*, pages 660–675. Springer, 2022.
- [56] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.
- [57] X. Du, G. Gozum, Y. Ming, and Y. Li. SIREN: Shaping representations for detecting out-of-distribution objects. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL <https://openreview.net/forum?id=8E8tgnYlMn>.
- [58] N. Kumar, S. Šegvić, A. Eslami, and S. Gumhold. Normalizing flow based feature synthesis for outlier-aware object detection. *arXiv preprint arXiv:2302.07106*, 2023.
- [59] X. Du, X. Wang, G. Gozum, and Y. Li. Unknown-aware object detection: Learning what you don’t know from videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13678–13688, 2022.
- [60] M. Everingham, L. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.*, 88(2):303–338, 2010. URL <http://dblp.uni-trier.de/db/journals/ijcv/ijcv88.html#EveringhamGWZ10>.
- [61] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft coco: Common objects in context, 2014. URL <http://arxiv.org/abs/1405.0312>. cite arxiv:1405.0312Comment: 1) updated annotation pipeline description and figures; 2) added new section describing datasets splits; 3) updated author list.
- [62] K. Lee, K. Lee, H. Lee, and J. Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.
- [63] J. Ren, S. Fort, J. Liu, A. G. Roy, S. Padhy, and B. Lakshminarayanan. A simple fix to mahalanobis distance for improving near-ood detection. *arXiv preprint arXiv:2106.09022*, 2021.