TOWARDS UNKNOWN-AWARE LEARNING WITH VIRTUAL OUTLIER SYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

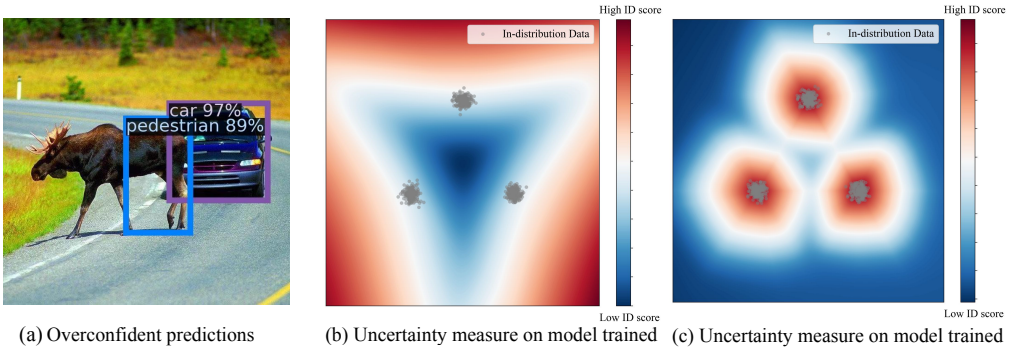
Out-of-distribution (OOD) detection has received much attention lately due to its importance in the safe deployment of neural networks. One of the key challenges is that models lack supervision signals from unknown data, and as a result, can produce overconfident predictions on OOD data. Previous approaches rely on real outlier datasets for model regularization, which can be costly and sometimes infeasible to obtain in practice. In this paper, we present **VOS**, a novel framework for OOD detection by adaptively synthesizing virtual outliers that can meaningfully regularize the model's decision boundary during training. Specifically, VOS samples virtual outliers from the low-likelihood region of the class-conditional distribution estimated in the feature space. Alongside, we introduce a novel unknown-aware training objective, which contrastively shapes the uncertainty space between the ID data and synthesized outlier data. VOS achieves *state-of-the-art* performance on both object detection and image classification models, reducing the FPR95 by up to 10.9% compared to the previous best method.

1 INTRODUCTION

Modern deep neural networks have achieved unprecedented success in known contexts for which they are trained, yet they do not necessarily know what they don't know (Nguyen et al., 2015). In particular, neural networks have been shown to produce high posterior probability for out-of-distribution (OOD) test inputs, which should not be predicted by the model. Taking self-driving car as an example, an object detection model trained to recognize in-distribution objects (*e.g.,* cars, stop signs) can produce a high-confidence prediction for an unseen object of a moose (see Figure 1(a)). Such a failure case raises concerns in model reliability, and worse, may lead to a catastrophic effect when deployed in safety-critical applications.

What makes OOD detection particularly challenging? To explain this, modern neural networks are commonly optimized on the in-distribution (ID) data only, and lack explicit knowledge of unknowns during training time. The resulting decision boundary, despite being useful on ID tasks such as classification, can undesirably cover OOD data. We illustrate this in Figure 1. The ID data (gray) consists of three class-conditional Gaussians, on which a three-way softmax classifier is trained. The resulting classifier is overconfident for regions far away from the ID data (see the red shade in Figure 1(b)). When such models are directly employed, the decision boundary can be ill-fated for OOD detection. Ideally, a model should learn a more compact decision boundary that produces low uncertainty for the ID data, with high OOD uncertainty elsewhere (*e.g.*, Figure 1(c)). However, achieving this goal is non-trivial due to the lack of supervision signal of unknowns. It is hard to comprehensively anticipate unknown data in advance under a large space of OOD uncertainty. This prompts the following question: *How can we enable unknown-aware deep neural networks without explicit knowledge of the unknowns in advance?*

Motivated by this, we propose a novel unknown-aware learning framework dubbed **VOS** (**V**irtual **O**utlier **S**ynthesis), which optimizes the dual objectives of both ID task and OOD detection performance. In a nutshell, VOS consists of three components tackling challenges of outlier synthesis and effective model regularization with synthesized outliers. First, we estimate the class-conditional distribution in the *feature space*, and propose to sample outliers from the low-likelihood region that is on the decision boundary of ID classes (Section 3.1). Key to our method, we show that sampling in the feature space is more tractable than synthesizing images in the high-dimensional pixel space (Lee



(a) Overconfident predictions on OOD data

(b) Uncertainty measure on model trained without outliers (not ideal)

(c) Uncertainty measure on model trained with virtual outliers (ours)

Figure 1: (a) A Faster-RCNN (Ren et al., 2015) model trained on BDD-100k dataset (Yu et al., 2020) produces overconfident predictions for OOD object (*e.g.*, moose). (b)-(c) The uncertainty measurement with and without virtual outlier training. The in-distribution data $\mathbf{x} \in \mathcal{X} = \mathbb{R}^2$ is sampled from a Gaussian mixture model). Regularizing the model with virtual outliers (c) better captures the out-of-distribution uncertainty than without (b). *Details of uncertainty measure is described in Section 3.4*.

et al., 2018a). Alongside, we propose a novel unknown-aware training objective, which contrastively shapes the uncertainty surface between the ID data and synthesized outliers (Section 3.2). During training, VOS simultaneously performs the ID task (*e.g.*, classification or object detection) as well as the OOD uncertainty regularization. During inference time, the uncertainty estimation branch produces a larger probabilistic score for ID data and vice versa, which enables effective OOD detection (Section 3.3).

VOS offers several compelling advantages compared to existing solutions. **(1)** VOS is a general learning framework that is effective for both object detection and image classification tasks, whereas previous methods were primarily driven by image classification. Image-level detection can be limiting as an image could be OOD in certain regions while being in-distribution elsewhere. Our work bridges a critical research gap since OOD detection for object detection is particularly underexplored in literature. **(2)** VOS enables *adaptive* outlier synthesis, which can be flexibly and conveniently used for any ID data without manual data collection or cleaning. In contrast, previous methods using outlier exposure (Hendrycks et al., 2019) require an auxiliary image dataset that is sufficiently diverse, which can be arguably prohibitive to obtain. Moreover, one needs to perform careful data cleaning to ensure the auxiliary outlier dataset does not overlap with ID data. **(3)** VOS synthesizes outliers that can meaningfully improve and tighten the decision boundary between ID and OOD data. In contrast, existing solutions use outliers that are either too trivial to regularize the OOD estimator, or too hard to be separated from ID data, resulting in sub-optimal performance. Our key contributions and results are summarized as follows:

- We propose a new framework VOS addressing a pressing issue—unknown-aware deep learning that optimizes for both ID and OOD performance. VOS establishes *state-of-the-art* results on both object detection and classification tasks. Compared to the best method, VOS reduces the FPR95 by up to 10.9% while preserving the accuracy on the ID task.
- We conduct extensive ablations and reveal important insights by contrasting different outlier synthesis approaches. We show that VOS is more advantageous than generating outliers directly in the high-dimensional pixel space (*e.g.*, using GAN (Lee et al., 2018a)) or using noise as outliers.
- We comprehensively evaluate our method on common OOD detection benchmarks, along with a more challenging yet underexplored task in the context of object detection. Our effort facilitates future research to evaluate OOD detection in a setting closer to real-world applications. Code and dataset will be released for reproducible research.

2 Problem Setup

We start by formulating the problem of OOD detection in the setting of object detection. Our framework can be easily generalized to image classification when the bounding box is the entire image (see Section 4.2). Most previous formulations of OOD detection treat entire images as anomalies, which can lead to ambiguity shown in Figure 1. In particular, natural images are not monolithic entities, but instead are composed of numerous objects and components. Knowing which regions of an image are anomalous could allow for safer handling of unfamiliar objects. This setting is

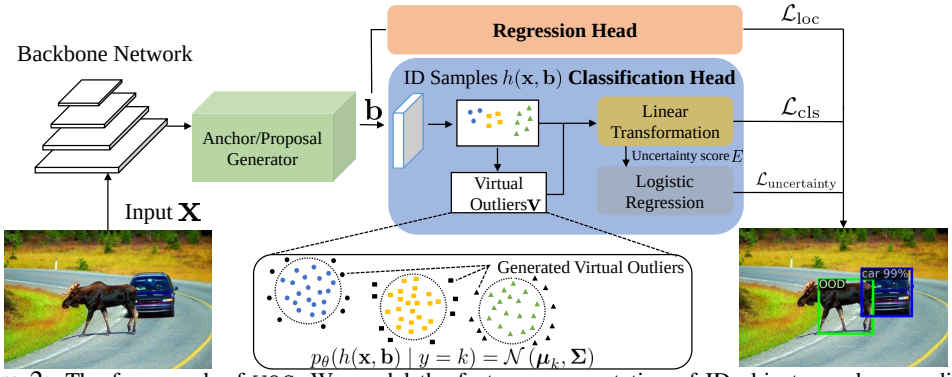


Figure 2: The framework of VOS. We model the feature representation of ID objects as class-conditional Gaussians, and sample virtual outliers $\mathbf{v}$ from the low-likelihood region. The virtual outliers, along with the ID objects, are used to produce the uncertainty loss for regularization. The uncertainty estimation branch ($\mathcal{L}_{\text{uncertainty}}$) is jointly trained with the object detection loss ($\mathcal{L}_{\text{loc}}, \mathcal{L}_{\text{cls}}$).

more realistic in practice, yet also more challenging as it requires reasoning OOD uncertainty at the fine-grained object level.

Specifically, we denote the input and label space by $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \{1, 2, ..., K\}$, respectively. Let $\mathbf{x} \in \mathcal{X}$ be the input image, $\mathbf{b} \in \mathbb{R}^4$ be the bounding box coordinates associated with object instances in the image, and $y \in \mathcal{Y}$ be the semantic label for K-way classification. An object detection model is trained on in-distribution data $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{b}_i, y_i)\}_{i=1}^{N}$ drawn from an unknown joint distribution $\mathcal{P}$. We use neural networks with parameters θ to model the bounding box regression $p_\theta(\mathbf{b}|\mathbf{x}, \mathbf{y})$ and the classification $p_\theta(\mathbf{y}|\mathbf{x})$.

The OOD detection can be formulated as a binary classification problem, which distinguishes between the in- vs. out-of-distribution objects. Let $P_{\mathcal{X}}$ denote the marginal probability distribution on $\mathcal{X}$. Given a test input $\mathbf{x}^* \sim P_{\mathcal{X}}$, as well as an object instance $\mathbf{b}^*$ predicted by the object detector, the goal is to predict $p_\theta(g|\mathbf{x}^*, \mathbf{b}^*)$. We use $g = 1$ to indicate a detected object being in-distribution, and $g = 0$ being out-of-distribution, with semantics outside the support of $\mathcal{Y}$.

3 METHOD

Our novel unknown-aware learning framework is illustrated in Figure 2. Our framework encompasses three novel components and addresses: (1) how to synthesize the virtual outliers (Section 3.1), (2) how to leverage the synthesized outliers for effective model regularization (Section 3.2), and (3) how to perform OOD detection during inference time (Section 3.3)?

3.1 VOS: VIRTUAL OUTLIER SYNTHESIS

Our framework VOS generates virtual outliers for model regularization, without relying on external data. While a straightforward idea is to train generative models such as GANs (Goodfellow et al., 2014; Lee et al., 2018a), synthesizing images in the high-dimensional *pixel space* can be difficult to optimize. Instead, our key idea is to synthesize virtual outliers in the *feature space*, which is more tractable given lower dimensionality. Moreover, our method is based on a discriminatively trained classifier in the object detector, which circumvents the difficult optimization process in training generative models.

Specifically, we assume the feature representation of object instances forms a class-conditional multivariate Gaussian distribution (which is verified using embeddings of a real network, see Figure 3):

$$p_\theta(h(\mathbf{x}, \mathbf{b})|y = k) = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}),$$

where $\boldsymbol{\mu}_k$ is the Gaussian mean of class $k \in \{1, 2, .., \mathrm{K}\}$, $\boldsymbol{\Sigma}$ is the tied covariance matrix, and $h(\mathbf{x}, \mathbf{b}) \in \mathbb{R}^m$ is the latent representation of an object instance $(\mathbf{x}, \mathbf{b})$. To extract the latent representation, we use the penultimate layer of the neural network. The dimensionality m is significantly smaller than the input dimension d.

To estimate the parameters of the class-conditional Gaussian, we compute empirical class mean $\widehat{\boldsymbol{\mu}}_k$ and covariance $\widehat{\boldsymbol{\Sigma}}$ of training samples $\{(\mathbf{x}_i, \mathbf{b}_i, y_i)\}_{i=1}^{N}$:

$$\widehat{\boldsymbol{\mu}}_k = \frac{1}{N_k} \sum_{i:y_i=k} h(\mathbf{x}_i, \mathbf{b}_i) \tag{1}$$

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{N} \sum_{k} \sum_{i:y_i=k} \left(h(\mathbf{x}_i, \mathbf{b}_i) - \widehat{\boldsymbol{\mu}}_k\right) \left(h(\mathbf{x}_i, \mathbf{b}_i) - \widehat{\boldsymbol{\mu}}_k\right)^\top, \tag{2}$$

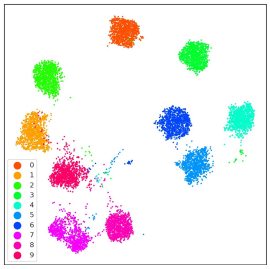


Figure 3: UMAP visualization of feature embeddings of PASCAL-VOC (on a subset of 10 classes).

where N_k is the number of objects in class k, and N is the total number of objects. We use online estimation during training, where we construct a class-conditional queue with $|Q_k|$ object instances from each class. We enqueue the embeddings of objects in the each minibatch (to each class-conditional queue) and accordingly dequeue the same number of object embeddings to update the queue.

Sampling from the feature representation space. We propose sampling the virtual outliers from the feature representation space, using the multivariate distributions estimated above. Ideally, these virtual outliers should help estimate a more compact decision boundary between ID and OOD data. To achieve this, we propose sampling the virtual outliers $\mathcal{V}_k$ from the ϵ-likelihood region of the estimated class-conditional distribution:

$$\mathcal{V}_k = \{\mathbf{v}_k | \frac{1}{(2\pi)^{m/2} |\widehat{\boldsymbol{\Sigma}}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{v}_k - \widehat{\boldsymbol{\mu}}_k)^\top \widehat{\boldsymbol{\Sigma}}^{-1} (\mathbf{v}_k - \widehat{\boldsymbol{\mu}}_k)\right) < \epsilon\}, \tag{3}$$

where $\mathbf{v}_k \sim \mathcal{N}(\widehat{\boldsymbol{\mu}}_k, \widehat{\boldsymbol{\Sigma}})$ denotes the sampled virtual outliers for class k, which are in the sublevel set based on the likelihood. ϵ is sufficiently small so that the sampled outliers are near class boundary. Ablation study on the effect of ϵ is provided in Appendix C.

Classification outputs for virtual outliers. For a given sampled virtual outlier $\mathbf{v} \in \mathbb{R}^m$, the output in the classification branch can be derived through a linear transformation:

$$f_k(\mathbf{v}_k; \boldsymbol{\theta}_{\text{cls}}) = \mathbf{v}_k \cdot \boldsymbol{\theta}_{\text{cls}}^k, \tag{4}$$

where $\boldsymbol{\theta}_{\text{cls}}^k$ is the weight vector of the fully connected layer corresponding to class k. We proceed with describing how to regularize the output of virtual outliers for improved OOD detection.

3.2 Unknown-aware Training Objective

We now introduce a new training objective for unknown-aware learning, leveraging the virtual outliers in Section 3.1. The key idea is to perform visual recognition task while regularizing the model to produce a low uncertainty score for ID data, and a high uncertainty score for the synthesized outlier. In the case of object detection, the overall training objective combines the standard object detection loss, along with a regularization loss in terms of uncertainty:

$$\min_{\theta} \mathbb{E}_{(\mathbf{x},\mathbf{b},y)\sim\mathcal{D}} \left[\mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{loc}}\right] + \beta \cdot \mathcal{L}_{\text{uncertainty}}, \tag{5}$$

where $\mathcal{D}$ is the training data. $\mathcal{L}_{\text{cls}}$ and $\mathcal{L}_{\text{loc}}$ are losses for classification and bounding box regression, respectively. This can be simplified to classification task without considering $\mathcal{L}_{\text{loc}}$.

Uncertainty score. To measure the uncertainty, we propose a *generalized energy score*, which is derived from the output of the classification branch. For ID object $(\mathbf{x}, \mathbf{b})$, the energy is defined as:

$$E(\mathbf{x}, \mathbf{b}) = -\log \sum_{k=1}^{K} w_k \cdot \exp^{f_k(h(\mathbf{x},\mathbf{b});\boldsymbol{\theta}_{\text{cls}})}, \tag{6}$$

where w_k are learnable weight coefficients, and $f_k(h(\mathbf{x}, \mathbf{b}); \boldsymbol{\theta}_{\text{cls}})$ is the logit output for class k in the classification branch. Similarly, the energy score for the virtual outliers is:

$$E(\mathbf{v}) = -\log \sum_{k=1}^{K} w_k \cdot \exp^{f_k(\mathbf{v};\boldsymbol{\theta}_{\text{cls}})}. \tag{7}$$

In particular, we show in Section 4 that a learnable $\mathbf{w}$ is more flexible than a constant $\mathbf{w}$, given the inherent class imbalance in object detection datasets. Additional visualization and analysis of the learned weight coefficient w_k is shown in Appendix G.

Uncertainty regularization. The uncertainty score is then passed into a logistic regression classifier with weight coefficient θ_{u}, which predicts high probability for ID object $(\mathbf{x}, \mathbf{b})$ and low probability for virtual outliers $\mathbf{v}$:

$$\mathcal{L}_{\text{uncertainty}} = \mathbb{E}_{\mathbf{v}\sim\mathcal{V}}\left[-\log\frac{1}{1+\exp^{-\theta_{\text{u}}\cdot E(\mathbf{v})}}\right] + \mathbb{E}_{(\mathbf{x},\mathbf{b})\sim\mathcal{D}}\left[-\log\frac{\exp^{-\theta_{\text{u}}\cdot E(\mathbf{x},\mathbf{b})}}{1+\exp^{-\theta_{\text{u}}\cdot E(\mathbf{x},\mathbf{b})}}\right]. \quad (8)$$

The learning process allows greater flexibility in contrastively shaping the uncertainty surface, resulting in more distinguishable in- and out-of-distribution objects. Liu et al. (2020a) employed energy for model uncertainty regularization, however, the loss function is based on the squared hinge loss and requires tuning two margin hyperparameters. In contrast, our uncertainty regularization loss is completely *hyperparameter-free* and is much easier to use in practice. Moreover, VOS produces probabilistic score for OOD detection, whereas Liu et al. (2020a) relies on non-probabilistic energy score. We provide ablation studies in Section 4.1 demonstrating the superiority of our loss function.

3.3 Inference-time OOD Detection

During inference, we use the output of the logistic regression uncertainty branch for OOD detection. In particular, given a test input $\mathbf{x}^*$, the object detector produces a bounding box prediction $\mathbf{b}^*$. The OOD uncertainty score for the predicted object $(\mathbf{x}^*, \mathbf{b}^*)$ is given by:

$$p_\theta(g \mid \mathbf{x}^*, \mathbf{b}^*) = \frac{\exp^{-\theta_{\text{u}}\cdot E(\mathbf{x}^*,\mathbf{b}^*)}}{1+\exp^{-\theta_{\text{u}}\cdot E(\mathbf{x}^*,\mathbf{b}^*)}}. \quad (9)$$

For OOD detection, one can exercise the common thresholding mechanism to distinguish between ID and OOD objects:

$$G(\mathbf{x}^*, \mathbf{b}^*) = \begin{cases} 1 & \text{if } p_\theta(g \mid \mathbf{x}^*, \mathbf{b}^*) \geq \gamma, \\ 0 & \text{if } p_\theta(g \mid \mathbf{x}^*, \mathbf{b}^*) < \gamma. \end{cases} \quad (10)$$

The threshold γ is typically chosen so that a high fraction of ID data (e.g., 95%) is correctly classified. Our framework VOS is summarized in Algorithm 1.

Algorithm 1 VOS: Virtual Outlier Synthesis for OOD detection

Input: ID data $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{b}_i, y_i)\}_{i=1}^N$, randomly initialized detector with parameter θ, queue size $|Q_k|$ for Gaussian density estimation, weight for uncertainty regularization β and likelihood ϵ.
Output: Object detector with parameter θ^*, and OOD detector G.
while *train* **do**
  Update the ID queue Q_k with the training objects $\{(\mathbf{x}, \mathbf{b}, y)\}$.
  Estimate the multivariate distributions based on ID training objects using Equation 1 and 2.
  Sample virtual outliers $\mathbf{v}$ using Equation 3.
  Calculate the regularization loss using Equation 8, update the parameters θ based on Equation 5.
end
while *eval* **do**
  Calculate the OOD uncertainty score using Equation 9.
  Perform thresholding comparison using Equation 10.
end

3.4 Effect of Virtual Outliers: An Illustrative Example

We illustrate the effect of training with and without virtual outliers in Figure 1. In particular, we consider a toy example of Gaussian mixtures, where $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{1, 2, 3\}$. The simplicity of the example allows us to see the changes of uncertainty surface straightforwardly. The tied covariance matrix is set to $\mathbf{\Sigma} = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.25 \end{bmatrix}$, $\|\boldsymbol{\mu_i} - \boldsymbol{\mu_j}\| = 12$ for $i \neq j$. The data distribution with three classes is shown in gray. Specifically, the model is trained with ID data only, on a 3-layer perceptron using the standard cross-entropy loss. Figure 1(b) shows the uncertainty in $\mathbb{R}^2$, measured

In-distribution $\mathcal{D}$	**Method**	**FPR95** $\downarrow$	**AUROC** $\uparrow$	**mAP (ID)**$\uparrow$
		OOD: MS-COCO / OpenImages		
PASCAL-VOC	MSP (Hendrycks & Gimpel, 2017)	71.41 / 74.79	82.62 / 79.51	48.7
	ODIN (Liang et al., 2018)	57.50 / 58.07	82.55 / 80.91	48.7
	Mahalanobis (Lee et al., 2018b)	65.77 / 62.72	81.31 / 80.63	48.7
	Gram matrices (Sastry & Oore, 2020)	63.28 / 68.28	79.67 / 77.28	48.7
	Energy score (Liu et al., 2020a)	59.82 / 61.01	82.01 / 81.12	48.7
	Generalized ODIN (Hsu et al., 2020)	75.87 / 75.89	76.35 / 77.12	48.1
	CSI (Tack et al., 2020)	60.73 / 58.72	81.35 / 81.44	48.1
	GAN-synthesis (Lee et al., 2018a)	62.42 / 62.22	82.91 / 80.50	48.5
	VOS-ResNet50 (ours)	$\mathbf{48.92}^{\pm 1.5}/\mathbf{54.41}^{\pm 1.3}$	$88.44^{\pm 0.8}/83.60^{\pm 0.7}$	$48.6^{\pm 0.1}$
	VOS-RegX4.0 (ours)	$49.61^{\pm 2.0}/56.07^{\pm 0.3}$	$\mathbf{89.00}^{\pm 0.4}/\mathbf{83.92}^{\pm 1.0}$	$\mathbf{51.3}^{\pm 0.0}$
Berkeley DeepDrive-100k	MSP (Hendrycks & Gimpel, 2017)	81.20 / 78.58	75.48 / 77.36	31.2
	ODIN (Liang et al., 2018)	63.22 / 57.20	73.91 / 77.10	31.2
	Mahalanobis (Lee et al., 2018b)	71.60 / 50.59	78.68 / 87.78	31.2
	Gram matrices (Sastry & Oore, 2020)	61.24 / 83.18	72.73 / 59.05	31.2
	Energy score (Liu et al., 2020a)	60.48 / 55.85	76.91 / 78.59	31.2
	Generalized ODIN (Hsu et al., 2020)	59.20 / 51.16	75.16 / 87.15	31.8
	CSI (Tack et al., 2020)	53.80 / 38.42	82.70 / 87.97	30.6
	GAN-synthesis (Lee et al., 2018a)	57.47 / 51.60	78.46 / 80.59	31.4
	VOS-ResNet50 (ours)	$46.06^{\pm 0.6}/36.82^{\pm 0.8}$	$86.06^{\pm 0.8}/88.84^{\pm 0.8}$	$31.3^{\pm 0.0}$
	VOS-RegX4.0 (ours)	$\mathbf{38.79}^{\pm 1.3}/\mathbf{28.67}^{\pm 1.7}$	$\mathbf{88.21}^{\pm 0.9}/\mathbf{91.05}^{\pm 0.4}$	$\mathbf{32.5}^{\pm 0.0}$

Table 1: **Main results.** Comparison with competitive out-of-distribution detection methods. All baseline methods are based on a model trained on **ID data only** using ResNet-50 as the backbone, without using any real outlier data. $\uparrow$ indicates larger values are better and $\downarrow$ indicates smaller values are better. All values are percentages. **Bold** numbers are superior results. We report standard deviations estimated across 3 runs. RegX4.0 denotes the backbone of RegNetX-4.0GF (Radosavovic et al., 2020) for the object detector.

by the negative energy (Liu et al., 2020a). The model trained without outliers is unable to capture the uncertainty that is far away from the ID data (red). In contrast, training with virtual outliers significantly improves the OOD uncertainty estimation, as shown in Figure 1(c).

4 Experimental Results

In this section, we present empirical evidence to validate the effectiveness of `VOS` on several real-world tasks, including both object detection (Section 4.1) and image classification (Section 4.2).

4.1 Evaluation on Object Detection

Experimental details. We use **PASCAL VOC**[1] (Everingham et al., 2010) and Berkeley DeepDrive (**BDD-100k**[2]) (Yu et al., 2020) datasets as the ID training data. For both tasks, we evaluate on two OOD datasets containing diverse visual categories: **MS-COCO** (Lin et al., 2014) and **OpenImages** (validation set) (Kuznetsova et al., 2020). We use the Detectron2 library (Girshick et al., 2018) and train on two backbone architectures: ResNet-50 (He et al., 2016) and RegNetX-4.0GF (Radosavovic et al., 2020). For each in-distribution class, we use 1,000 samples to estimate the class-conditional Gaussians. Since the threshold ϵ can be infinitesimally small, we instead choose ϵ based on the t-th smallest likelihood in a pool of 10,000 samples (per-class), generated from the class-conditional Gaussian distribution. A larger t corresponds to a larger threshold ϵ. As shown in Table 6, a smaller t yields good performance. We set $t = 1$ for all our experiments. *Extensive details on the datasets are described in Appendix A, along with a comprehensive sensitivity analysis of each hyperparameter (including the queue size $|Q_k|$, coefficient β, and threshold ϵ) in Appendix C.*

Metrics. For evaluating the OOD detection performance, we report: (1) the false positive rate (FPR95) of OOD samples when the true positive rate of ID samples is at 95%; (2) the area under the receiver operating characteristic curve (AUROC). For evaluating the object detection performance on the ID task, we report the common metric of mAP.

Comparison with existing approaches. In Table 1, we compare `VOS` with competitive OOD detection methods in literature. For a fair comparison, all the methods only use ID data without using

[1]PASCAL-VOC consists of the following ID labels: `Person, Car, Bicycle, Boat, Bus, Motorbike, Train, Airplane, Chair, Bottle, Dining Table, Potted Plant, TV, Sofa, Bird, Cat, Cow, Dog, Horse, Sheep`.

[2]BDD-100k consists of ID labels: `Pedestrian, Rider, Car, Truck, Bus, Train, Motorcycle, Bicycle, Traffic light, Traffic sign`.

	Method	AUROC ↑	mAP ↑
Image synthesis	$^{\diamond}$GAN (Lee et al., 2018a)	83.07	48.5
	$^{\diamond}$Mixup (Zhang et al., 2018) (mixing ratio 0.4)	60.97	44.3
	$^{\diamond}$Mixup (Zhang et al., 2018) (mixing ratio 1)	63.34	46.9
Noise as outliers	$^{\natural}$Additive Gaussian noise to ID features	67.98	48.7
	$^{\natural}$Trainable noise added to the ID features	65.63	48.6
	$^{\natural}$Gaussian noise	85.47	48.5
Negative proposals	$^{\clubsuit}$All negative proposals	62.36	48.1
	$^{\clubsuit}$Random negative proposals	65.89	48.5
	$^{\clubsuit}$Proposals with large background prob (Joseph et al., 2021)	76.43	48.5
	VOS (ours)	**88.44**	48.6

Table 2: Ablation on outlier synthesis approaches (on backbone of ResNet-50, COCO is the OOD data).

auxiliary outlier dataset. Our proposed method, VOS, outperforms competitive baselines, including Maximum Softmax Probability (Hendrycks & Gimpel, 2017), ODIN (Liang et al., 2018), Mahalanobis distance (Lee et al., 2018b), Generalized ODIN (Hsu et al., 2020), energy score (Liu et al., 2020a), CSI (Tack et al., 2020) and Gram matrices (Sastry & Oore, 2020). These baselines can be naturally extended to the object detection model due to the existence of classification head. We also compare with GAN-based approach for synthesizing outliers (Lee et al., 2018a), where VOS effectively improves the OOD detection performance (FPR95) by **11.41**% on BDD-100k and **13.5**% on Pascal VOC (COCO as OOD). Moreover, we show in Table 1 that VOS achieves stronger OOD detection performance while preserving a high accuracy on the original in-distribution task (measured by mAP). This is in contrast with CSI, which displays significant degradation, with mAP decreased by 0.7% on BDD-100k. *Details of reproducing baselines are in Appendix D. Additional results using a different OOD detection metric are in Appendix H.*

Ablation on outlier synthesis approaches. We compare VOS with different synthesis approaches in Table 2. Specifically, we consider three types of synthesis approach: (i$^{\diamond}$) synthesizing outliers in the pixel space, (ii$^{\natural}$) using noise as outliers, and (iii$^{\clubsuit}$) using negative proposals from RPN as outliers. For type I, we consider GAN-based (Lee et al., 2018a) and mixup (Zhang et al., 2018) methods. The outputs of the classification branch for outliers are forced to be closer to a uniform distribution. For mixup, we consider two different beta distributions $\mathrm{Beta}(0.4)$ and $\mathrm{Beta}(1)$, and interpolate ID objects in the pixel space. For Type II, we use noise perturbation to create virtual outliers. We consider adding fixed Gaussian noise to the ID features, adding trainable noise to the ID features where the noise is trained to push the outliers away from ID features, and using fixed Gaussian noise as outliers. Lastly, for type III, we directly use the negative proposals in the ROI head as the outliers for Equation 8, similar to Joseph et al. (2021). We consider three variants: randomly sampling n negative proposals (n is the number of positive proposals), sampling n negative proposals with a larger probability, and using all the negative proposals. All methods are trained under the same setup, with PASCAL-VOC as in-distribution data and ResNet-50 as the backbone. The loss function is the same as Equation 5 for all variants, with only difference being synthesis method.

The results are summarized in Table 2, where VOS outperforms alternative synthesis approaches both in the feature space ($\clubsuit$, $\natural$) or the pixel space ($\diamond$). Generating outliers in the pixel space ($\diamond$) is either unstable (GAN) or harmful for the object detection performance (mixup). Introducing noise ($\natural$), especially using Gaussian noise as outliers is promising. However, Gaussian noise outliers are relatively simple, and may not effectively regularize the decision boundary between ID and OOD as VOS does. Exploiting the negative proposals ($\clubsuit$) is not effective, because they are distributionally close to the ID data.

Ablation on different architecture and different feature layers. Lastly, we demonstrate that VOS is effective on alternative neural network architectures. In particular, using RegNet (Radosavovic et al., 2020) as backbone yields both better ID accuracy and OOD detection performance. We also explore using intermediate layers for outlier synthesis, where we show using VOS on the penultimate layer is the most effective. This is expected since the feature representations are the most discriminative at deeper layers. *We provide detailed results in Appendix E.*

Ablation on the uncertainty loss. We perform ablation on several variants of VOS, trained with different uncertainty loss $\mathcal{L}_{\text{uncertainty}}$. Particularly, we consider: (1) using the squared hinge loss for regularization as in Liu et al., (2) using constant weight $\mathbf{w} = [1, 1, ..., 1]^{\top}$ for energy score in Equation 6, and (3) classifying the virtual outliers as an additional $K + 1$ class in the classification

$\mathcal{D}$	Method	FPR95 ↓	AUROC ↑	object detection mAP (ID) ↑
PASCAL-VOC	VOS w/ hinge loss	50.04	88.30	46.5
	VOS w/ constant $\mathbf{w}$	53.16	88.10	48.9
	VOS w/ $K+1$ class	66.08	84.72	47.0
	VOS (ours)	**48.92**	**88.44**	48.6
Berkeley DeepDrive-100k	VOS w/ hinge loss	54.48	83.26	29.5
	VOS w/ constant $\mathbf{w}$	50.13	84.91	30.9
	VOS w/ $K+1$ class	59.11	83.29	30.1
	VOS (ours)	**46.06**	**86.06**	31.3

Table 3: **Ablation study.** Comparison with different regularization loss functions (on backbone of ResNet-50, COCO is the OOD data).

branch. The performance comparison is summarized in Table 3. Compared to the hinge loss, using our proposed logistic loss for regularizing the energy reduces the FPR95 by 8.42% on BDD-100k. While the squared hinge loss in Liu et al. requires tuning the hyperparameters, our uncertainty loss is completely *hyperparameter free*. In addition, we find that a learnable $\mathbf{w}$ for energy score is more desirable than a constant $\mathbf{w}$, given the inherent class imbalance in object detection datasets. Additional visualization of the learned weight coefficient w_k is shown in Appendix G, which aligns with the class frequency. Finally, classifying the virtual outliers as an additional class increases the difficulty of object classification, which does not outperform either. This ablation demonstrates the superiority of the uncertainty loss employed by `VOS`.

Comparison with training on real outlier data. We also compare with Outlier Exposure (Hendrycks et al., 2019) (OE), an approach that relies on the *real* outlier data for model regularization. For a fair comparison, we train the object detector on PASCAL-VOC using the same architecture ResNet-50, and use the OE objective for the classification branch. The real outliers for OE training are sampled from the OpenImages dataset (Kuznetsova et al., 2020). We perform careful deduplication to ensure there is no overlap between the outlier training data and PASCAL-VOC. Our method achieves OOD detection performance on COCO (AUROC: 88.44%) that favorably matches OE (AUROC: 89.41%), and does not impose strong data assumption. Note that OE takes 274 minutes on 4 GPUs while `VOS` takes 240 minutes, which is faster to train.

4.2 Evaluation on Image Classification

Going beyond object detection, we show that `VOS` is also suitable and effective on common image classification benchmark. We use CIFAR-10 (Krizhevsky & Hinton, 2009) as the ID training data, with standard train/val splits. We train on WideResNet-40 (Zagoruyko & Komodakis, 2016) and DenseNet-101 (Huang et al., 2017), where we substitute the object detection loss in Equation 5 with the cross-entropy loss. Following the common benchmark, we evaluate on six OOD datasets: `Textures` (Cimpoi et al., 2014), `SVHN` (Netzer et al., 2011), `Places365` (Zhou et al., 2018), `LSUN-C` (Yu et al., 2015), `LSUN-Resize` (Yu et al., 2015), and `iSUN` (Xu et al., 2015). The comparisons are shown in Table 4, with results averaged over six test datasets. `VOS` demonstrates competitive OOD detection results on both architectures without sacrificing the ID test classification accuracy (94.84% on pre-trained WideResNet vs. 94.68% using `VOS`). Additionally, *we provide comparative results with generative-based methods in Appendix F*.

Method	FPR95 ↓	AUROC ↑
	WideResNet / DenseNet	
MSP	51.05 / 48.73	90.90 / 92.46
ODIN	35.71 / 24.57	91.09 / 93.71
Mahalanobis	37.08 / 36.26	93.27 / 87.12
Gram Matrices	27.33 / 23.13	93.00 / 89.83
Energy	33.01 / 27.44	91.88 / 94.51
Generalized ODIN	39.94 / 26.97	92.44 / 93.76
CSI	35.66 / 47.83	92.45 / 85.31
GAN-synthesis	37.30 / 83.71	89.60 / 54.14
VOS (ours)	**24.87 / 22.47**	**94.06 / 95.33**

Table 4: OOD detection results of `VOS` and comparison with competitive baselines on two architectures: WideResNet-40 and DenseNet-101.

4.3 Qualitative Analysis

In Figure 4, we visualize the prediction on several OOD images, using object detection models trained without virtual outliers (top) and with `VOS` (bottom), respectively. The in-distribution data is BDD-100k. `VOS` performs better in identifying OOD objects (in green) than a vanilla object detector, and reduces false positives among detected objects. Moreover, the confidence score of the false-positive objects of `VOS` is lower than that of the vanilla model (see the truck in the 3rd column). *We visualize the synthesized virtual outliers in Appendix Section I.*

5 Related Work

OOD detection for classification can be broadly categorized into post hoc, generative-based and outlier exposure (OE)-based approaches. For post hoc methods, the softmax confidence score is a

Figure 4: Visualization of detected objects on the OOD images (from MS-COCO) by a vanilla Faster-RCNN (*top*) and VOS (*bottom*). The in-distribution is BDD-100k dataset. **Blue**: Objects detected and classified as one of the ID classes. **Green**: OOD objects detected by VOS, which reduce false positives among detected objects.

common baseline (Hendrycks & Gimpel, 2017), which can be arbitrarily high for OOD inputs (Hein et al., 2019). Several improvements have been proposed, such as ODIN (Liang et al., 2018; Hsu et al., 2020), Mahalanobis distance (Lee et al., 2018b), energy score (Liu et al., 2020a), and Gram matrices based score (Sastry & Oore, 2020). Different from Lee et al. (2018b), VOS performs dynamic estimation of class-conditional Gaussian during training, which shapes the uncertainty surface over time using a novel uncertainty regularization loss. Outlier Exposure-based methods exploited model regularization using natural images (Hendrycks et al., 2019; Mohseni et al., 2020; Zhang et al., 2021) or images synthesized by GANs (Lee et al., 2018a). However, real outlier data is often infeasible to obtain. In contrast, VOS automatically synthesizes virtual outliers which allows greater flexibility and generality. Generative models directly estimate the in-distribution density (Kingma & Dhariwal, 2018; Schirrmeister et al., 2020), which makes them natural alternatives for OOD detection. However, they are in general less competitive compared to discriminative-based methods and typically harder to optimize (Hinz et al., 2019; Kirichenko et al., 2020; Nalisnick et al., 2019). Tack et al. (2020) applied self-supervised learning for OOD detection, which we compare in Section 4. Jung et al. (2021) proposed a standardized max-logit approach for detecting outliers in semantic segmentation, which is a post-hoc approach. Zhao et al. (2021); Grcic et al. (2021) trained a generative model and synthesize outliers in the pixel space, which cannot be applied to object detection where a scene consists of both known and unknown objects. Their regularization terms are based on entropy maximization, which is different from VOS.

OOD detection for object detection is currently underexplored. Joseph et al. (2021) used energy score (Liu et al., 2020a) to identify the OOD data and then labeled them for incremental object detection. In contrast, VOS focuses on OOD detection and adopts a new unknown-aware training objective with a new test-time detection score, which is applicable to both object detectors and classifiers. Moreover, Joseph et al. (2021) used the negative proposals as unknown samples for model regularization, which is suboptimal as we show in Table 2. Harakeh & Waslander (2021) focused on uncertainty estimation for the localization regression, rather than OOD detection for classification problems. Several works (Dhamija et al., 2020; Miller et al., 2019; 2018; Hall et al., 2020; Deepshikha et al., 2021) used approximate Bayesian methods, such as MC-Dropout (Gal & Ghahramani, 2016) for OOD detection. They require multiple inference passes to generate the uncertainty score, which are computationally expensive on larger datasets and models.

Open-world object detection includes out-of-domain generalization (Kim et al., 2021; Wang et al., 2021), zero-shot object detection (Gu et al., 2021; Rahman et al., 2020) and incremental object detection (Liu et al., 2020b; Pérez-Rúa et al., 2020). Most of them either developed measures to mitigate catastrophic forgetting (Joseph et al., 2020) or used auxiliary information (Rahman et al., 2020), such as class attributes to perform object detection on unseen data, which is different from our focus of OOD detection.

6 Conclusion

In this paper, we proposed VOS, a novel unknown-aware training framework for OOD detection. Different from methods that require real outlier data, VOS adaptively synthesizes outliers during training by sampling virtual outliers from the low-likelihood region of the class-conditional distributions. The synthesized outliers meaningfully improve the decision boundary between the ID data and OOD data, resulting in state-of-the-art OOD detection performance while preserving the performance of the ID task. VOS is effective and suitable for both object detectors and classifiers. We hope our work will inspire future research towards unknown-aware deep learning in real-world settings.

REPRODUCIBILITY STATEMENT

Authors of the paper recognizes the importance and value of reproducible research. We summarize our efforts below to facilitate reproducible results:

1. **Datasets.** We use publicly available datasets, which are described in detail in *Section 4.1*, *Section 4.2*, and *Appendix A*.
2. **Baselines.** The description and hyperparameters of the OOD detection baselines are explained in *Appendix D*.
3. **Model training.** Our model training on object detection is based on the publicly available Detectron2 codebase: `https://github.com/facebookresearch/detectron2`. Hyperparamters are specified in *Section 4.1*, with a thorough ablation study provided in *Appendix C*.
4. **Methodology.** Our method is fully documented in Section 3, with the pseudo algorithm detailed in *Algorithm 1*.
5. **Open Source.** The codebase and the dataset will be released for reproducible research.

ETHICS STATEMENT

Our project aims to improve the reliability and safety of modern machine learning models. Our study can lead to direct benefits and societal impacts, particularly for safety-critical applications such as autonomous driving. Our study does not involve any human subjects or violation of legal compliance. We do not anticipate any potentially harmful consequences to our work. Through our study and releasing our code, we hope to raise stronger research and societal awareness towards the problem of out-of-distribution detection in real-world settings.

REFERENCES

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, pp. 3606–3613, 2014.

Kumari Deepshikha, Sai Harsha Yelleni, P. K. Srijith, and C. Krishna Mohan. Monte carlo dropblock for modelling uncertainty in object detection. *CoRR*, abs/2108.03614, 2021.

Akshay Raj Dhamija, Manuel Günther, Jonathan Ventura, and Terrance E. Boult. The overlooked elephant of object detection: Open set. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2020*, pp. 1010–1019, 2020.

Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. In *Advances in Neural Information Processing Systems 2019, NeurIPS 2019*, pp. 3603–3613, 2019.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. `https://github.com/facebookresearch/detectron`, 2018.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations, ICLR 2020*, 2020.

Matej Grcic, Petra Bevandic, and Sinisa Segvic. Dense open-set recognition with synthetic outliers generated by real NVP. In *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISIGRAPP 2021, Volume 4: VISAPP, Online Streaming, February 8-10, 2021*, pp. 133–143. SCITEPRESS, 2021. doi: 10.5220/0010260701330143. URL `https://doi.org/10.5220/0010260701330143`.

Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Zero-shot detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021.

David Hall, Feras Dayoub, John Skinner, Haoyang Zhang, Dimity Miller, Peter Corke, Gustavo Carneiro, Anelia Angelova, and Niko Sünderhauf. Probabilistic object detection: Definition and evaluation. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2020*, pp. 1020–1029, 2020.

Ali Harakeh and Steven L. Waslander. Estimating and evaluating regression predictive uncertainty in deep object detectors. In *International Conference on Learning Representations*, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pp. 630–645. Springer, 2016.

Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 41–50, 2019.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations, ICLR 2017*, 2017.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019.

Tobias Hinz, Stefan Heinrich, and Stefan Wermter. Generating multiple objects at spatially distinct locations. In *International Conference on Learning Representations, ICLR 2019*, 2019.

Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10951–10960, 2020.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

K. J. Joseph, Jathushan Rajasegaran, Salman H. Khan, Fahad Shahbaz Khan, Vineeth Balasubramanian, and Ling Shao. Incremental object detection via meta-learning. *arXiv preprint arXiv:2003.08798*, 2020.

K. J. Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N. Balasubramanian. Towards open world object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2021*, 2021.

Sanghun Jung, Jungsoo Lee, Daehoon Gwak, Sungha Choi, and Jaegul Choo. Standardized max logits: A simple yet effective approach for identifying unexpected road obstacles in urban-scene segmentation. In *IEEE International Conference on Computer Vision, ICCV*, 2021.

Dahun Kim, Tsung-Yi Lin, Anelia Angelova, In So Kweon, and Weicheng Kuo. Learning open-world object proposals without learning to classify. *CoRR*, abs/2108.06753, 2021.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems 2018, NeurIPS 2018*, pp. 10236–10245, 2018.

Polina Kirichenko, Pavel Izmailov, and Andrew G Wilson. Why normalizing flows fail to detect out-of-distribution data. *Advances in Neural Information Processing Systems*, 33, 2020.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report, 2009.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision*, pp. 1–26, 2020.

Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018a.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, pp. 7167–7177, 2018b.

Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations, ICLR 2018*, 2018.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755, 2014.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 2020a.

Xialei Liu, Hao Yang, Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Continual universal object detection. *CoRR*, abs/2002.05347, 2020b.

Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *IEEE International Conference on Robotics and Automation, ICRA 2018*, pp. 1–7, 2018.

Dimity Miller, Feras Dayoub, Michael Milford, and Niko Sünderhauf. Evaluating merging strategies for sampling-based uncertainty techniques in object detection. In *International Conference on Robotics and Automation, ICRA 2019*, pp. 2348–2354, 2019.

Sina Mohseni, Mandar Pitale, JBS Yadawa, and Zhangyang Wang. Self-supervised learning for generalizable out-of-distribution detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 5216–5223, 2020.

Eric T. Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Görür, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *International Conference on Learning Representations, ICLR 2019*, 2019.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 427–436, 2015.

Juan-Manuel Pérez-Rúa, Xiatian Zhu, Timothy M. Hospedales, and Tao Xiang. Incremental few-shot object detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020*, pp. 13843–13852, 2020.

Ilija Radosavovic, Raj Prateek Kosaraju, Ross B. Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020*, pp. 10425–10433, 2020.

Shafin Rahman, Salman H. Khan, and Fatih Porikli. Zero-shot object detection: Joint recognition and localization of novel concepts. *International Journal of Computer Vision*, 128(12):2979–2999, 2020.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pp. 91–99, 2015.

Chandramouli Shama Sastry and Sageev Oore. Detecting out-of-distribution examples with gram matrices. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, volume 119, pp. 8491–8501, 2020.

Robin Schirrmeister, Yuxuan Zhou, Tonio Ball, and Dan Zhang. Understanding anomaly detection with deep invertible networks through hierarchies of distributions and features. 2020.

Joan Serrà, David Álvarez, Vicenç Gómez, Olga Slizovskaia, José F. Núñez, and Jordi Luque. Input complexity and out-of-distribution detection with likelihood-based generative models. In *International Conference on Learning Representations, ICLR 2020*, 2020.

Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. In *Advances in Neural Information Processing Systems*, 2020.

Xin Wang, Thomas E. Huang, Benlin Liu, Fisher Yu, Xiaolong Wang, Joseph E. Gonzalez, and Trevor Darrell. Robust object detection via instance-level temporal cycle confusion. *CoRR*, abs/2104.08381, 2021.

Pingmei Xu, Krista A Ehinger, Yinda Zhang, Adam Finkelstein, Sanjeev R Kulkarni, and Jianxiong Xiao. Turkergaze: Crowdsourcing saliency with webcam based eye tracking. *arXiv preprint arXiv:1504.06755*, 2015.

Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving dataset for heterogeneous multitask learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020*, pp. 2633–2642, 2020.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018*, 2018.

Jingyang Zhang, Nathan Inkawhich, Yiran Chen, and Hai Li. Fine-grained out-of-distribution detection with mixup outlier exposure. *CoRR*, abs/2106.03917, 2021.

Zhilin Zhao, Longbing Cao, and Kun-Yu Lin. Revealing distributional vulnerability of explicit discriminators by implicit generators. *CoRR*, abs/2108.09976, 2021. URL `https://arxiv.org/abs/2108.09976`.

Bolei Zhou, Àgata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions Pattern Analysis and Machine Intelligence.*, 40(6):1452–1464, 2018.

Supplementary Material

A EXPERIMENTAL DETAILS

We summarize the OOD detection evaluation task in Table 5. The OOD test dataset is selected from MS-COCO and OpenImages dataset, which contains disjoint labels from the respective ID dataset. The PASCAL model is trained for a total of 18,000 iterations, and the BDD-100k model is trained for 90,000 iterations. We add the uncertainty regularizer (Equation 8) starting from 2/3 of the training. The weight β is set to 0.1. See *detailed ablations on the hyperparameters in Appendix C*.

	Task 1	**Task 2**
ID train dataset	VOC train	BDD train
ID val dataset	VOC val	BDD val
OOD dataset	COCO and OpenImages val	COCO and OpenImages val
#ID train images	16,551	69,853
#ID val images	4,952	10,000
#OOD images for COCO	969	1,914
#OOD images for OpenImages	1,852	1,852

Table 5: OOD detection evaluation tasks.

B SOFTWARE AND HARDWARE

We run all experiments with Python 3.8.5 and PyTorch 1.7.0, using NVIDIA GeForce RTX 2080Ti GPUs.

C EFFECT OF HYPERPARAMETERS

Below we perform sensitivity analysis for each important hyperparameter[1]. We use ResNet-50 as the backbone, trained on in-distribution dataset PASCAL-VOC.

Effect of ϵ. Since the threshold ϵ can be infinitesimally small, we instead choose ϵ based on the t-th smallest likelihood in a pool of 10,000 samples (per-class), generated from the class-conditional Gaussian distribution. A larger t corresponds to a larger threshold ϵ. As shown in Table 6, a smaller t yields good performance. We set $t = 1$ for all our experiments.

t	mAP↑	FPR95 ↓	AUROC↑	AUPR↑
1	48.7	**54.69**	**83.41**	**92.56**
2	48.2	57.96	82.31	88.52
3	48.3	62.39	82.20	88.05
4	48.8	69.72	80.86	89.54
5	48.7	57.57	78.66	88.20
6	48.7	74.03	78.06	91.17
8	48.8	60.12	79.53	92.53
10	47.2	76.25	74.33	90.42

Table 6: Ablation study on the number of selected outliers t (per class).

Effect of queue size $|Q_k|$. We investigate the effect of ID queue size $|Q_k|$ in Table 7, where we vary $|Q_k| = \{50, 100, 200, 400, 600, 800, 1000\}$. Overall, a larger $|Q_k|$ is more beneficial since the estimation of Gaussian distribution parameters can be more precise. In our experiments, we set the queue size $|Q_k|$ to $1,000$ for PASCAL and 300 for BDD-100k. The queue size is smaller for BDD because some classes have a limited number of object boxes.

Effect of β. As shown in Table 8, a mild value of β generally works well. As expected, a large value (e.g., $\beta = 0.5$) will over-regularize the model and harm the performance.

[1]Note that our sensitivity analysis uses the speckle noised PASCAL VOC validation dataset as OOD data, which is different from the actual OOD test datasets in use.

$\lvert Q_k \rvert$	mAP↑	FPR95 ↓	AUROC↑	AUPR↑
50	48.6	68.42	77.04	92.30
100	48.9	59.77	79.96	89.18
200	48.8	57.80	80.20	89.92
400	48.9	66.85	77.68	89.83
600	48.5	57.32	81.99	91.07
800	48.7	**51.43**	82.26	91.80
1000	48.7	54.69	**83.41**	**92.56**

Table 7: Ablation study on the ID queue size $|Q_k|$.

β	mAP↑	FPR95 ↓	AUROC↑	AUPR↑
0.01	48.8	59.20	82.64	90.08
0.05	48.9	57.21	83.27	91.00
0.1	48.7	**54.69**	**83.41**	**92.56**
0.15	48.5	59.32	77.47	89.06
0.5	36.4	99.33	57.46	85.25

Table 8: Ablation study on regularization weight β.

Effect of starting iteration for the regularizer. Importantly, we show that uncertainty regularization should be added in the middle of the training. If it is added too early, the feature space is not sufficiently discriminative for Gaussian distribution estimation. See table 9 for the effect of starting iteration Z. We use $Z = 12,000$ for the PASCAL-VOC model, which is trained for a total of 18,000 iterations.

Z	mAP↑	FPR95 ↓	AUROC↑	AUPR↑
2000	48.5	60.01	78.55	87.62
4000	48.4	61.47	79.85	89.41
6000	48.5	59.62	79.97	89.74
8000	48.7	56.85	80.64	90.71
10000	48.6	49.55	83.22	92.49
12000	48.7	**54.69**	**83.41**	92.56
14000	49.0	55.39	81.37	**93.00**
16000	48.9	59.36	82.70	92.62

Table 9: Ablation study on the starting iteration Z. Model is trained for a total of 18,000 iterations.

D BASELINES

To evaluate the baselines, we follow the original methods in MSP (Hendrycks & Gimpel, 2017), ODIN (Liang et al., 2018), Generalized ODIN (Hsu et al., 2020), Mahalanobis distance (Lee et al., 2018b), CSI (Tack et al., 2020), energy score (Liu et al., 2020a) and gram matrices (Sastry & Oore, 2020) and apply them accordingly on the classification branch of the object detectors. For ODIN, the temperature is set to be $T = 1000$ following the original work. For both ODIN and Mahalanobis distance, the noise magnitude is 0.0005. For GAN (Lee et al., 2018a), we follow the original paper and use a GAN to generate OOD images. The prediction of the OOD images/objects is regularized to be close to a uniform distribution, through a KL divergence loss with a weight of 0.1. We set the shape of the generated images to be 100×100 and resize them to have the same shape as the real images. We optimize the generator and discriminator using Adam (Kingma & Ba, 2015), with a learning rate of 0.001. For CSI (Tack et al., 2020), we use the rotations (0°, 90°, 180°, 270°) as the self-supervision task. We set the temperature in the contrastive loss to 0.5. We use the features right before the classification branch (with the dimension to be 1024) to perform contrastive learning. The weights of the losses that are used for classifying shifted instances and instance discrimination are both set to 0.1 to prevent training collapse.

E VIRTUAL OUTLIER SYNTHESIS USING EARLIER LAYER

In this section, we investigate the effect of using `VOS` on an earlier layer within the network. Our main results in Table 1 are based on the penultimate layer of the network. Here, we additionally evaluate the performance using the layer before the penultimate layer, with a feature dimension of $1,024$. The results are summarized in Table 10. As observed, synthesizing virtual outliers in the penultimate layer achieves better OOD detection performance than the earlier layer, since the feature representations are more discriminative at deeper layers.

Models	FPR95↓	AUROC↑	mAP↑
PASCAL VOC			
VOS-final	**48.92**	**88.44**	**48.6**
VOS-earlier	50.05	87.50	**48.6**
BDD-100k			
VOS-final	**46.06**	**86.06**	**31.3**
VOS-earlier	52.14	85.00	30.6

Table 10: Performance comparison of employing VOS on different layers. COCO is the OOD data.

F COMPARISON WITH GENERATIVE-BASED METHODS

Method	SVHN	C100	CelebA
Glow (Kingma & Dhariwal, 2018)	0.64	0.65	0.54
IGEBM (Du & Mordatch, 2019)	0.43	0.54	0.69
JEM-softmax (Grathwohl et al., 2020)	0.89	**0.87**	0.79
JEM-likelihood (Grathwohl et al., 2020)	0.67	0.67	0.75
Input Complexity (Serrà et al., 2020)	0.95	0.74	0.86
VOS (ours)	**0.96**	**0.87**	**0.87**

Table 11: Comparison with *generative-based* methods using density estimation. Models are trained on CIFAR-10. Values are AUROC.

In Table 11, we also compare with state-of-the-art methods using generative modeling (Du & Mordatch, 2019; Grathwohl et al., 2020; Kingma & Dhariwal, 2018; Serrà et al., 2020). We follow the same evaluation protocols as in prior literature. Using CIFAR-10 as ID data, `VOS` demonstrates competitive performance compared to the best method (Serrà et al., 2020) on different OOD datasets.

Note that generative modeling and density estimation are less feasible for object detection with multiple objects in the scene. In contrast, VOS is suitable for both object detection (Section 4.1) and image classification (Section 4.2) tasks.

G VISUALIZATION OF THE LEARNABLE WEIGHT COEFFICIENT w IN GENERALIZED ENERGY SCORE

To observe whether the learnable weight coefficient w_k in Equation 6 and 7 captures dataset-specific statistics during uncertainty regularization, we visualize w_k w.r.t each in-distribution class and the number of training objects of that class in Figure 5. We use the BDD-100k dataset (Yu et al., 2020) as the in-distribution dataset and the RegNetX-4.0GF (Radosavovic et al., 2020) as the backbone network. As can be observed, the learned weight coefficient displays a consistent trend with the number of training objects per class, which indicates the advantage of using learnable weights rather than constant weight vector with all 1s.

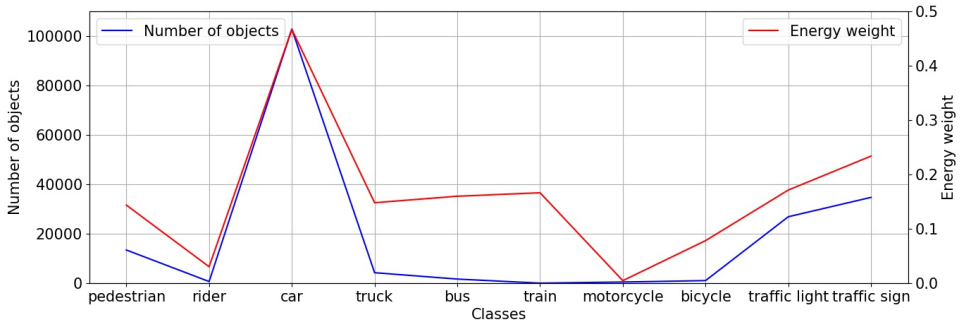


Figure 5: Visualization of learnable weight coefficient in the generalized energy score and the number of training objects per in-distribution class. The value of the weight coefficient is averaged over three different runs.

H OOD DETECTION RESULTS UNDER A DIFFERENT METRIC

There are two average precision metrics in literature: AUPR-In (by treating in-distribution data as positives) vs. AUPR-Out (by treating OOD as positives). From a practical perspective, the score threshold needs to be selected based on the in-distribution data (due to the impossibility to anticipate unknown data ahead of time). For this reason, metrics such as AUPR-Out are less applicable.

We have reported the average precision (AUPR-In) of the baselines and our approach in Table 12. The in-distribution data is the BDD100k dataset and the OOD data are COCO and OpenImages dataset. We use a backbone of ResNet-50. The reason for high AP scores is the sample imbalance between the in-distribution and OOD objects. Our evaluation contains significantly more in-distribution objects. For this reason, we believe FPR95 and AUROC are more meaningful metrics to report, which are shown in Table 1 of the main paper.

I VISUALIZATION OF THE VIRTUAL OUTLIERS

In this section, we visualize the synthesized virtual outliers by VOS using UMAP in Figure 6. The in-distribution dataset is the Pascal VOC dataset with the backbone of ResNet-50. Note that we cannot visualize virtual outliers in the pixel space since they are synthesized in low-dimensional feature space.

From Figure 6, the virtual outliers reside in the near-boundary region of the in-distribution feature cluster, which helps the model to learn a compact decision boundary between ID and OOD objects.

Method	COCO	OpenImages
MSP (Hendrycks & Gimpel, 2017)	99.53	99.21
ODIN (Liang et al., 2018)	99.38	99.52
Mahalanobis (Lee et al., 2018b)	96.66	97.26
Gram matrices (Sastry & Oore, 2020)	94.22	95.95
Energy score (Liu et al., 2020a)	99.53	99.79
Generalized ODIN (Hsu et al., 2020)	99.70	99.78
CSI (Tack et al., 2020)	99.69	99.65
GAN-synthesis (Lee et al., 2018a)	99.49	99.20
VOS	**99.71**	**99.81**

Table 12: OOD detection results under the metric of AUPR.

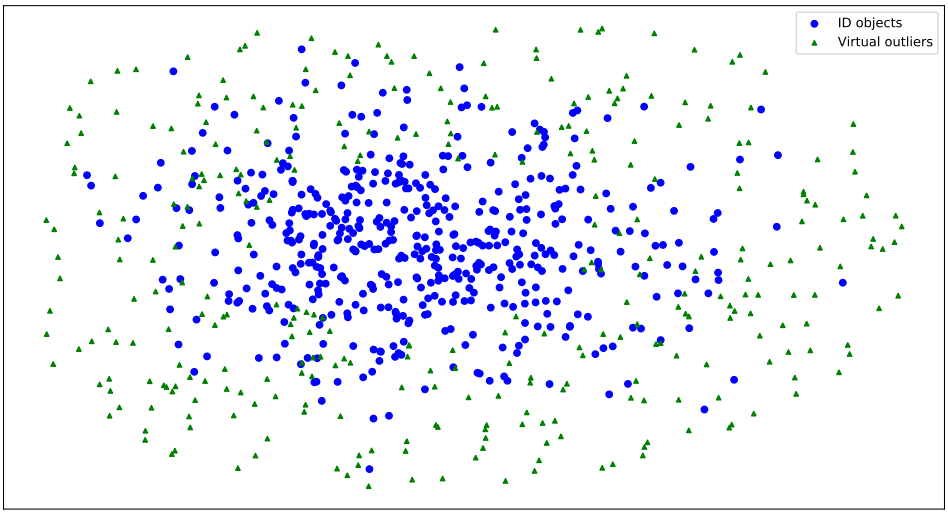


Figure 6: UMAP visualization of the synthesized virtual outliers. The blue points denote the object features from the in-distribution class of `Person`. The green points denote the synthesized virtual outliers from the low-density space *w.r.t* the features from that class.

J DISCUSSION ON THE DETECTED, REJECTED AND IGNORED OOD OBJECTS DURING INFERENCE

The focus of `VOS` is to mitigate the undesirable cases when an OOD object is detected and classified as in-distribution with high confidence. In other words, our goal is to ensure that "if the box is detected, it should be faithfully an in-distribution object rather than OOD". Although generating the bounding box for OOD data is not the focus of this paper, we do notice that VOS can improve the number of boxes detected for OOD data (+25% on BDD trained model compared to the vanilla Faster-RCNN).

The number of OOD objects ignored by RPN can largely depend on the confidence score threshold and the NMS threshold. Hence, we found it more meaningful to compare relatively with the vanilla Faster-RCNN under the same default thresholds. Using BDD100K as the in-distribution dataset and the ResNet as the backbone, `VOS` can improve the number of detected OOD boxes by 25% (compared to vanilla object detector). `VOS` also improves the number of rejected OOD samples by 63%.