Raising the Bar on the Evaluation of Out-of-Distribution Detection

Jishnu Mukhoti[*1,2], Tsung-Yu Lin[2], Bor-Chun Chen[2], Ashish Shah[2], Philip H.S. Torr[1], Puneet K. Dokania[1] [†], Ser-Nam Lim [2] [†]
[1]University of Oxford, [2]Meta AI

Abstract

*In image classification, a lot of development has happened in detecting out-of-distribution (OoD) data. However, most OoD detection methods are evaluated on a standard set of datasets, arbitrarily different from training data. There is no clear definition of what forms a "good" OoD dataset. Furthermore, the state-of-the-art OoD detection methods already achieve near perfect results on these standard benchmarks. In this paper, we define 2 categories of OoD data using the subtly different concepts of perceptual/visual and semantic similarity to in-distribution (iD) data. We define Near OoD samples as perceptually similar but semantically different from iD samples, and Shifted samples as points which are visually different but semantically akin to iD data. We then propose a GAN based framework for generating OoD samples from each of these 2 categories, given an iD dataset. Through extensive experiments on MNIST, CIFAR-10/100 and ImageNet, we show that **a)** state-of-the-art OoD detection methods which perform exceedingly well on conventional benchmarks are significantly less robust to our proposed benchmark. Moreover, we observe that **b)** models performing well on our setup also perform well on conventional real-world OoD detection benchmarks and vice versa, thereby indicating that one might not even need a separate OoD set, to reliably evaluate performance in OoD detection.*

1. Introduction

With the wide-spread deployment of deep learning models in real-life applications like autonomous driving [11] and medical diagnosis [56], it is imperative to ensure that in addition to being accurate, such models are also able to reliably quantify their uncertainty and identify inputs which they "don't know". One of the major applications of such uncertainty quantification methods is the detection of inputs sampled from a distribution different from the model's training distribution (i.e, Out-of-Distribution or OoD inputs). A lot of work has been done in this direction from the perspective of uncertainty quantification [41, 62, 46, 36], OoD Detection [22, 39, 12, 70, 40, 54], open-set recognition [44, 49] and the like.

(a) iD (b) Shifted (c) Near OoD

Figure 1: ***Shifted* and *Near OoD* samples obtained from ImageNet.** *Shifted samples* are visually different but semantically similar to iD. *Near OoD images* are perceptually similar to iD but are semantically dissimilar.

Since any point outside the training distribution can be considered OoD, the set of potential OoD inputs is infinite. This makes evaluating OoD detection a particularly challenging problem. The general evaluation practice involves using a proxy OoD dataset which is different from the training distribution (or in-distribution (iD) samples) to simulate an out-of-distribution scenario. The OoD detection algorithm is then evaluated on how well it can separate the iD samples from the OoD points. For the purposes of evaluation and benchmarking, it is natural to ask which proxy OoD dataset is best suited for measuring model performance. To answer this, we need to consider the different types of OoD inputs that can arise in a real-world scenario.

In image classification, we model the conditional categorical distribution $p(y|\mathbf{x})$ over classes, given an input image $\mathbf{x}$. Under the i.i.d assumption, both the training and test images are assumed to be sampled from the same continuous distribution in image space, i.e., $p_{\text{train}}(\mathbf{x}) = p_{\text{test}}(\mathbf{x})$. In case of OoD samples, this assumption is broken, i.e., $p_{\text{train}}(\mathbf{x}) \neq p_{\text{ood}}(\mathbf{x})$. Based on the conditional distribution $p(y|\mathbf{x})$, we can then define two kinds of OoD samples.

Distribution Shift: Although the distribution in image space is different, the conditional distribution over class labels remains the same, i.e., $p_{\text{train}}(y|\mathbf{x}) = p_{\text{ood}}(y|\mathbf{x})$, and $p_{\text{train}}(\mathbf{x}) \neq p_{\text{ood}}(\mathbf{x})$. Such samples are generally derived from the training set by applying transformations like cor-

[*]Corresponding author: jishnu.mukhoti@eng.ox.ac.uk.
[†]Primary mentors, alphabetical order.

ruptions [21] and semantic shifts [72, 32], where the transformed images have the same labels as the originals from training. For example, ImageNet-C/P [21] contain synthetic corruptions/perturbations applied to ImageNet [6]. Such datasets provide a controlled environment to study models under specific synthetic and real-world distribution shifts.

Unseen Categories: The second category of OoD comprise images of classes which the model has not been trained on, i.e., $p_{\text{train}}(y|\mathbf{x}) \neq p_{\text{ood}}(y|\mathbf{x})$, and $p_{\text{train}}(\mathbf{x}) \neq p_{\text{ood}}(\mathbf{x})$. For a given training set, any dataset having a disjoint set of labels qualifies as OoD with unseen categories. How do we then decide which OoD dataset is good for evaluation? The convention is to use a well-known set of (iD vs OoD) dataset pairs like MNIST [37] vs Fashion-MNIST [71], CIFAR-10 [34] vs SVHN [51] etc. However, **firstly,** the choice of these dataset pairs is relatively arbitrary and there is no guarantee that performance on these benchmarks will generalise to the real-world. **Secondly**, in recent literature [12, 70], the terms "Near OoD" and "Far OoD" have been used to indicate the difficulty of an OoD detection task with Near OoD datasets (CIFAR-10 vs CIFAR-100) being more difficult than Far OoD (CIFAR-10 vs SVHN). With no model-agnostic metric quantifying the "nearness" of an OoD dataset, these terms are also not well-defined. **Finally,** one of the current state-of-the-art OoD detection baselines, Vision Transformer [12], obtains around 96% AUROC on CIFAR-100 vs CIFAR-10 and over 99.5% AUROC on CIFAR-10 vs SVHN. Hence, the most popular OoD detection benchmarks are not necessarily the most effective. They can be saturated and might give us the impression that state-of-the-art baselines are robust to OoD. The near perfect AUROC scores also indicate that these benchmarks might be rendered redundant in future for evaluating the performance of even better OoD detection methods which outperform the Vision Transformer [12].

In this work, we thus aim to take a step towards improving the conventional evaluation process for OoD detection in image classification. We first look at the two types of OoD mentioned above through the lens of *perceptual/visual similarity* and *semantic similarity* [7, 3] between images. Perceptual similarity between two images denotes how visually similar they are and semantic similarity captures the similarity of concepts that they represent. With this in mind, we define:

1. ***Shifted sets*** *as perceptually dissimilar but semantically similar to the training distribution.*
2. ***Near OoD sets*** *as perceptually similar but semantically dissimilar to the training distribution.*
3. ***Far OoD sets*** *as both perceptually and semantically dissimilar to the training distribution.*

Clearly, images which are both perceptually and semantically similar to the training distribution would be iD. We show the hierarchy of OoD samples in fig. 2. In this work, we particularly focus on generating Shifted and Near OoD sets. Given the training set, it is difficult to define a single distance measure in the image space which can capture both perceptual and semantic similarity. Hence, we propose using a sampling based generative model, a Generative Adversarial Network (GAN) [15] and design regularisers for the GAN objective using the definitions above to generate OoD samples.

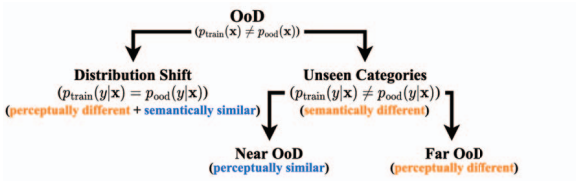


Figure 2: **Categories of OoD samples.** Samples obtained from a distribution shift retain their conditional distribution over classes, whereas samples obtained from an unseen category don't belong to any of the training classes.

More formally, for a training set $\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^N$, where $y_i \in \mathcal{S}$, $\forall i$, in order to generate *shifted samples*, we learn a transformation $t_{\text{shift}} : \mathbf{x} \rightarrow \hat{\mathbf{x}}$ in the image space, $\mathbf{x}, \hat{\mathbf{x}} \in \mathbb{R}^{H,W,C}$, such that $\mathbf{x}$ and $\hat{\mathbf{x}}$ are perceptually different and semantically similar, i.e., have the same label: $\arg\max_c p(y_c|\mathbf{x}) = \arg\max_c p(y_c|\hat{\mathbf{x}})$. This is an Image-to-Image translation problem and hence, we use a Pix-2-Pix [30] model to learn a distribution shift. In case of *Near OoD*, we want to learn a distribution in the close perceptual vicinity of $\mathcal{D}$. This can be seen as a transformation $t_{\text{near ood}} : \mathbf{z} \rightarrow \hat{\mathbf{x}}, \mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ where the generated image $\hat{\mathbf{x}}$ is perceptually similar to the training distribution but does not belong to any of the iD classes: $\arg\max_c p(y_c|\hat{\mathbf{x}}) \notin \mathcal{S}$. This is an Image generation problem and hence, we use a GAN for generating Near OoD samples. In fig. 1, we show examples from ImageNet of shifted as well as Near OoD samples. Through extensive experiments using several popular OoD detection baselines comparing our benchmarks with conventional ones, we make the following observations and contributions.

Firstly, the performance of state-of-the-art OoD detection baselines (Deep Ensembles [36] and Vision Transformers [12]), established to be relatively robust on standard benchmarks, is consistently worse under our proposed evaluation setup. This is true across datasets of all sizes: ImageNet, CIFAR-10/100 and MNIST, thereby showing that *there's still plenty of room for improvement in OoD detection research*. **Secondly,** we observe a consistent trend where models which perform better on our benchmarks also perform well on standard real-world benchmarks and vice versa. Assuming that standard benchmarks are indicative of real-world OoD detection performance, an assumption made by most existing works on OoD detection, the fact

then that our benchmarks have been created without the use of any external OoD dataset then indicates that *one might not need an OoD dataset to measure OoD detection performance*. **Finally,** to the best of our knowledge, our method is a novel way to generate benchmarks for the evaluation of any OoD detection method.

2. Related Work

State-of-the-art on OoD Detection: The problem of OoD detection has been tackled from different angles like uncertainty quantification [36] and open set recognition [44]. As mentioned in Section 1, a popular approach is uncertainty quantification, which naturally serves as a solution for OoD detection as OoD inputs should intuitively be assigned higher uncertainty. A well-known thread of work uses the softmax distribution from a neural network to capture uncertainty. This starts from [22] where the authors simply use the softmax probability and continues on to several methods including augmentations [40, 27, 38], calibration [17, 48], modified activation functions [59], logit normalization [67] and energy based models [43]. While these methods have a low computational overhead, the softmax distribution often fails to capture epistemic uncertainty [31] and is overconfident on incorrect predictions for OoD inputs [13]. A more principled approach uses the Bayesian formalism [50] and is applied to deep neural nets using approximate Bayesian inference methods [14, 1, 45]. Yet another popular uncertainty quantification method, often interpreted as a form of Bayesian Model Averaging [69], is deep ensembles [36] which uses an ensemble of neural networks and averages the softmax distributions to compute uncertainty. Deep ensembles and its modifications [68, 10] are often widely accepted as one of the the state-of-the-art methods for uncertainty quantification. Both approximate Bayesian inference and deep ensembles however, require either multiple forward passes at test time or multiple models to make predictions, thereby being computationally expensive. Attempting to achieve ensemble level performance from a single deterministic model, DUQ [62], SNGP [41, 42] and DDU [47], develop distance-aware deterministic models which can quantify uncertainty. In addition to the above, there are a set of works specifically targeted for OoD detection including using gram matrices [57], feature space singularity [29] and virtual logit matching [65]. Finally, [12] show that pre-trained Vision Transformers, when fine-tuned on a downstream dataset, achieve state-of-the-art AUROC scores on conventional OoD benchmarks. [73] contains a comprehensive set of OoD detection methods.

OoD evaluation procedure: OoD samples are generally one of two types: i) *distribution shifted* samples and ii) samples which belong to an *unseen category* which the model hasn't been trained on. For evaluation, the general practice is to use separate OoD datasets for testing. For shifted samples, some of the well-known datasets include ImageNet-C (corrupted) and ImageNet-P (perturbed) [21] which use synthetic corruptions and perturbations as well as stylised versions of ImageNet like ImageNet-R [20], ImageNet-Sketch [64] etc. There are also datasets containing specific real-world shifts like WILDS [32], Backgrounds [72], colored MNIST [16] etc. As shifted datasets retain the label information of the original dataset, models are evaluated on their calibration error [53], which compares the model confidence with its accuracy on the provided test set. On the other hand, to test a model's performance on unseen categories, the convention is to use pairs of (iD vs OoD) datasets and measure how well a model is able to tell apart OoD from iD data using scores like AUROC. Most works in the current literature use MNIST [37] vs Fashion-MNIST [71], CIFAR-10/100 vs SVHN & CIFAR-100/10 and ImageNet [6] vs ImageNet-O [24] as the standard benchmarks for unseen distributions. Recently, [19] released the Species dataset as an OoD dataset for ImageNet-21K. However, the choice of these datasets is relatively arbitrary and a lot of the current OoD benchmarks including CIFAR-10 vs SVHN/CIFAR-100 and MNIST vs Fashion-MNIST are saturated [12]. In this work, we thus show that generating OoD samples given a training set can produce significantly more challenging benchmarks for even some of the state-of-the-art OoD detection methods.

Generative models for OoD: Generative models have been previously used to generate samples on the boundary of image classes [38, 8, 75, 9, 74]. These include GAN based models [38, 8, 75], latent space sampling [9] and even text-to-image generators [74]. GANs in particular have often been employed to train their discriminators for anomaly detection [63, 52]. The purpose of these works however is to use generated samples during training to improve the performance on OoD detection [33, 4] or the general robustness of discriminative models to distribution shift. As mentioned before, with the lack of a clear definition of distance in image space, it is difficult to encode different types of OoD in a GAN and this is where one of our primary contributions lies. Secondly, our motivation is also orthogonal to these works. We use a GAN to improve the evaluation of OoD detection rather than improve the OoD detection methods themselves.

3. Method

In this section, we formalise our approach to generate shifted and near OoD samples given a training set. First, we encode perceptual and semantic similarity as quantifiable loss functions for generative models. Then we discuss the GAN architectures and objective functions to generate shifted and Near OoD samples respectively.

Perceptual similarity as a loss function As mentioned in section 1, perceptual similarity between images repre-

sents how visually similar they are. Since we have target images from the training set, we can use a Full-Reference Image Quality Assessment (FR-IQA) [2] metric to encode perceptual loss. There exist several FR-IQA metrics in the literature like SSIM [66], FSIM [76] and LPIPS [77] but we use the *Learned Perceptual Image Patch Similarity* (LPIPS) [77] as it is known to correlate with human judgement well. Let f_θ represent a pre-trained convolutional network. Given two images, $\mathbf{x}_1$ and $\mathbf{x}_2$, the LPIPS computes the cosine distance between feature space activations of $\mathbf{x}_1$ and $\mathbf{x}_2$ across different layers of the network f_θ as shown below:

$$\mathcal{L}_{\text{LPIPS}}(\mathbf{x}_1, \mathbf{x}_2) = \sum_l \left(\frac{1}{H_l W_l} ||f_\theta^l(\mathbf{x}_1) - f_\theta^l(\mathbf{x}_2)||_2^2 \right) \tag{1}$$

where $f_\theta^l(\mathbf{x}_1)$ and $f_\theta^l(\mathbf{x}_2) \in \mathbb{R}^{H_l, W_l, C_l}$ are the feature space representations from inputs $\mathbf{x}_1$ and $\mathbf{x}_2$ in layer l of the network. In this work, we use LPIPS with a VGG, although other architectures can be used as well. From here on out, we represent perceptual loss as $\mathcal{L}_{\text{LPIPS}}$. Minimizing $\mathcal{L}_{\text{LPIPS}}(\mathbf{x}_1, \mathbf{x}_2)$ encourages images $\mathbf{x}_1$ and $\mathbf{x}_2$ to be perceptually similar and vice versa.

Semantic similarity as a loss function In image classification, the semantic meaning of an image is encoded in its class label. To identify if an image is semantically similar to the training distribution, we need a model which understands the semantic meaning of the training distribution. However, a single classifier can make incorrect and confident predictions on inputs [17], especially when they are not from any of the training classes [53]. In this work, we take inspiration from Bayesian literature [50] and quantify semantic similarity, using the *mutual information* (MI) $\mathbb{I}[y, \theta | \mathcal{D}, \mathbf{x}]$, (also known as *information gain*) [13] between the posterior distribution over parameters of a Bayesian model $p(\theta|\mathcal{D})$ and its predicted distribution over classes $p(y|\mathbf{x}, \theta)$. Let $(\mathbf{x}_g, y_g)$ be an input and $\mathcal{S}$ be the set of training classes. If $y_g \in \mathcal{S}$, i.e., the sample belongs to one of the training set classes, seeing the sample $(\mathbf{x}_g, y_g)$ won't cause much information gain about the posterior $p(\theta|\mathcal{D})$. On the other hand, $y_g \notin \mathcal{S}$ will cause a high information gain about the posterior and $\mathbb{I}[y, \theta | \mathcal{D}, \mathbf{x}]$ will be high. Thus, in order to generate semantically similar/dissimilar images, we intuitively want $\mathbb{I}[y, \theta | \mathcal{D}, \mathbf{x}]$ to be low/high.

Due to the computational intractability of Bayesian inference in deep learning, we use a pre-trained deep ensemble which can be seen as a way to perform Bayesian Model Averaging [69] with each model in the ensemble being a sample from the posterior. Following [13], we approximate MI as the difference between the entropy of the average softmax distribution of the ensemble and the average entropy of the softmax distributions of each individual network in the ensemble. Let $p(y|\mathbf{x}, \theta_t)$ represent the softmax distribution produced by the t^{th} network in an ensemble of

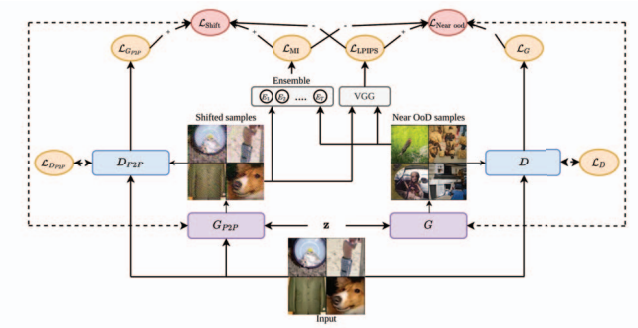


Figure 3: **Schematic of the proposed method to generate Shifted and Near OoD samples.** G_{P2P} and D_{P2P} represent the Pix-2-Pix to generate shifted samples and G and D represent the GAN to generate Near OoD images. The dotted lines show paths for gradient propagation.

T networks, on an input $\mathbf{x}$. The average softmax distribution for the ensemble is $p(y|\mathbf{x}, \theta) = \frac{1}{T} \sum_{t=1}^{T} p(y|\mathbf{x}, \theta_t)$. Then, the MI for the ensemble on input $\mathbf{x}$ can be approximated as:

$$\mathcal{L}_{\text{MI}}(\mathbf{x}) = \hat{\mathbb{I}}[y, \theta | \mathcal{D}, \mathbf{x}] \approx \mathbb{H}[p(y|\mathbf{x}, \theta)] - \frac{1}{T} \sum_{t=1}^{T} \mathbb{H}[p(y|\mathbf{x}, \theta_t)] \tag{2}$$

where $\mathbb{H}[.]$ represents the entropy of a distribution. We use $\mathcal{L}_{\text{MI}}$ of a pre-trained ensemble as the semantic loss to quantify semantic similarity of an image to the training distribution. Semantically similar images (eg., images belonging to training classes) should have low $\mathcal{L}_{\text{MI}}$ and vice versa.

Generative Model Having defined perceptual and semantic similarity as quantifiable loss functions, we discuss two different GAN architectures for the two different types of OoD data. For distribution shift, we intend to learn a transformation on iD data and hence, use a conditional GAN architecture, Pix-2-Pix [30] in particular. For Near OoD, we intend to learn a distribution instead and hence, use a standard GAN. We do not change the loss for the discriminator in any of the GANs and only regularise the loss for the generator to produce the desired OoD type.

Distribution Shift: For distribution shift, we want to maximize the perceptual loss $\mathcal{L}_{\text{LPIPS}}$ (to generate perceptually different images) and minimize the semantic loss $\mathcal{L}_{\text{MI}}$ (to preserve the semantic meaning of generated images). Thus, the objective for the Pix-2-Pix generator is:

$$\begin{aligned} \mathcal{L}_{\text{Shift}} = & \underbrace{\mathbb{E}_{\mathbf{x},\mathbf{z}}[\log(1 - D_{P2P}(\mathbf{x}, G_{P2P}(\mathbf{x}, \mathbf{z})))]}_{\text{Pix-2-Pix Generator Loss}} \\ & - \underbrace{\lambda_p \mathbb{E}_{\mathbf{x},\mathbf{z}}[\mathcal{L}_{\text{LPIPS}}(\mathbf{x}, G_{P2P}(\mathbf{x}, \mathbf{z}))]}_{\text{maximize perceptual loss}} \\ & + \underbrace{\lambda_s \mathbb{E}_{\mathbf{x},\mathbf{z}}[\mathcal{L}_{\text{MI}}(G_{P2P}(\mathbf{x}, \mathbf{z}))]}_{\text{minimize semantic loss}}, \end{aligned} \tag{3}$$

where G_{P2P} and D_{P2P} represent the generator and dis-

criminator of the Pix-2-Pix GAN and λ_p and λ_s are the regularisation coefficients for the perceptual and semantic losses.

Near OoD: Similarly, for Near OoD, we want to minimize the perceptual loss $\mathcal{L}_{\text{LPIPS}}$ (to encourage perceptually similar images) and maximize the semantic loss $\mathcal{L}_{\text{MI}}$ (to generate semantically different images which don't belong to training set classes). We use a GAN for Near OoD distributions with the generator objective as shown below:

$$\begin{aligned}\mathcal{L}_{\text{Near ood}} = & \underbrace{\mathbb{E}_{\mathbf{z}}[\log(1 - D(G(\mathbf{z})))]}_{\text{GAN Generator Loss}} \\ & + \underbrace{\lambda_p \mathbb{E}_{\mathbf{x},\mathbf{z}}[\mathcal{L}_{\text{LPIPS}}(\mathbf{x}, G(\mathbf{z}))]}_{\text{minimize perceptual loss}} \\ & - \underbrace{\lambda_s \mathbb{E}_{\mathbf{z}}[\mathcal{L}_{\text{MI}}(G(\mathbf{z}))]}_{\text{maximize semantic loss}},\end{aligned} \tag{4}$$

where G and D denote the generator and discriminator of the GAN respectively. Figure 3 shows a schematic of our model along with the loss functions.

4. Experiments & Discussion

4.1. Implementation Details

Setting λ_p and λ_s and MI thresholding: In eq. (3) and eq. (4), we introduce the regularisation coefficients λ_p and λ_s. To set these, we generate samples using different combinations of λ_s and λ_p. We then measure the MI on the training ensemble for all generated samples and filter them such that MI is neither too high (avoid samples which are too dissimilar), nor too low (samples which coincide with iD). The lower bound is selected to be the lowest value which minimizes MI overlap between val and near OoD sets and the upper bound is chosen to be lower bound + 0.4 (see fig. 10 in appendix). In particular, we use $[0.1, 0.5]$ for MNIST, $[0.2, 0.6]$ for CIFAR-10 (C10) and $[0.4, 0.8]$ for CIFAR-100 (C100) and ImageNet. We find these settings to empirically produce good OoD samples across datasets.

Ensemble for $\mathcal{L}_{\text{MI}}$: We implement the semantic loss $\mathcal{L}_{\text{MI}}$ on MNIST using an ensemble of 4 different networks: LeNet [37], AlexNet [35], VGG-11 [58] and ResNet-18 [18]. On C10/100, we use 6 networks: DenseNet-121 [28], ResNet-50/110, VGG-16, Wide-ResNet-28-10 and Inception-v3 [60] and on ImageNet, we use 3 networks: ResNet-18, MobileNet-v3-Large [26] and EfficientNet-B0 [61]. All the ensemble models are trained on their respective training sets.

Generating Near OoD Datasets We use a GAN to generate Near OoD samples from the training set using the $\mathcal{L}_{\text{Near ood}}$ loss. In particular, for MNIST, we use DCGAN for its simplicity and on C10/100 and ImageNet, we use BigGAN due to its superior performance in terms of FID scores. For all the GANs, we set λ_p to 1 and perform a grid search for λ_s over the interval $[0.5, 3]$ at steps of 0.25 . As mentioned above, we use all the resulting trained GANs and filter samples out by thresholding on MI for the ensemble.

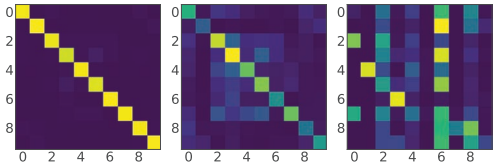
Generating Shifted Datasets We generate shifted samples using a Pix-2-Pix model trained on $\mathcal{L}_{\text{shift}}$ loss. After a grid search over different values, we found $\lambda_s = 1$ and $\lambda_p = 2$ to produce the best results on C10 and ImageNet. Further training details can be found in appendix A. We show qualitative examples of both Near OoD and Shifted samples in fig. 1 and additional samples in appendix D .

4.2. Sanity Check on Benchmarks

We perform a sanity check using CIFAR-10 to verify our claims on the generated samples: i.e., Near OoD samples are semantically dissimilar and perceptually similar while Shifted samples are semantically similar and perceptually dissimilar to the training set.

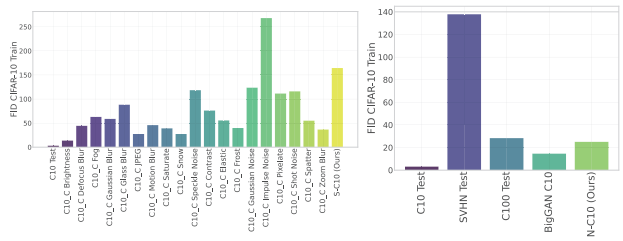
Semantic similarity to the training distribution To verify that Near OoD samples are semantically dissimilar and Shifted samples are semantically similar to iD, we compare the predictions of an ensemble of 6 models: DenseNet-121, ResNet-50/110, VGG-16, Wide-ResNet-28-10 and Inception-v3 on the CIFAR-10 test set with both near OoD and shifted versions of CIFAR-10. Note that the ensembles used for testing are different from the ones used in training. We present the corresponding confusion matrices in fig. 4. Clearly, in case of shifted samples, the label information is preserved as the ensemble's predicted classes broadly match the correct labels. Such is not the case, however, for near OoD samples where the predictions for each class are mostly incorrect, indicating that the dataset has not preserved the label information from the training distribution.

Perceptual similarity to the training distribution To measure perceptual similarity between images, we use the well-known FID score [25]. Again, note that the FID score is an independent metric which has not been used during our training. It is normally used to evaluate generative models on the perceptual quality of their outputs. We show the FID between the CIFAR-10 training set and Shifted (S) CIFAR-10 comparing with all the corruption types at intensity 5 of CIFAR-10-C [21]. Simlarly, we compare the FID of Near OoD (N) CIFAR-10 with SVHN, CIFAR-100 and samples generated by a BigGAN trained on CIFAR-10. We present the results in fig. 5. It is evident that S CIFAR-10 has a very high FID score indicating perceptual dissimilarity from the training set. Note however, that it is not the highest among all the corruption types in CIFAR-10-C. On the other hand, N CIFAR-10 has a significantly lower FID than SVHN and slightly lower than CIFAR-100, providing evidence in favour of the fact that N CIFAR-10 is perceptually more similar to the training set as compared to CIFAR-100 or SVHN.



(a) iD (b) Shifted (c) Near OoD

Figure 4: **Confusion matrices for ensemble predictions on versions of CIFAR-10.** The ensemble broadly predicts Shifted samples correctly unlike Near OoD samples.



(a) Shifted (b) Near OoD

Figure 5: **FID scores between CIFAR-10 and generated OoD sets.** FID of Shifted CIFAR-10 is higher than most corruption types in CIFAR-10-C, indicating perceptual dissimilarity from CIFAR-10, whereas FID of N CIFAR-10 is significantly lower than SVHN and even lower than CIFAR-100 indicating perceptual similarity to CIFAR-10 samples.

4.3. Evaluating Near OoD Datasets

Having sanity checked our claims on the perceptual and semantic similarity of generated samples, we evaluate Near OoD samples using two experiments. Firstly, we use generated Near OoD datasets obtained from MNIST, CIFAR-10/100 and ImageNet and compare them with conventional OoD benchmarks using multiple well-known OoD detection baselines. Secondly, in Appendix B we also compare performance on models trained using outlier exposure [23] on our datasets with conventional ones.

OoD Detection: Architectures & Baselines For models trained on ImageNet, we use 6 different architectures: 2 convolutional models, ResNet-50 and Wide-ResNet-50-2, and 4 vision transformers: ViT-B/16, ViT-B/32, ViT-L/16 and ViT-L/32. For each of the Vision transformers, we evaluate 5 well-known OoD detection baselines: softmax confidence, entropy [22], predictive entropy of a 5-ensemble [36], MaxLogit [19] and Energy [43] based OoD detection. Additionally, we also evaluate Mahalanobis distance [39] as a baseline for the 2 convolutional models. Note that the predictive entropy of a deep ensemble is the entropy of the average softmax distribution (first term in eq. (2)). The ensemble used to compute $\mathcal{L}_{\text{MI}}$ during training contains models with different architectures to encourage variability in predictions. During evaluation however, we follow standard procedure and use the same architecture for each model in the ensemble.

Architecture	Baseline	AUROC % ImageNet-O	Species	OpenImage-O	Texture	N-ImageNet (Ours)
ViT-B-16	SC	82.13	59.8	90.8	90.2	75.72
	SE	89.15	63.2	94.5	94.6	80.63
	Ensemble	95.1	75.6	98.2	98.3	88.12
	MaxLogit	90.3	68.1	95.4	95.6	82.1
	Energy	92.0	70.02	96.0	97.9	84.15
ViT-B-32	SC	75.96	53.4	85.4	87.8	74.2
	SE	78.94	56.5	87.3	89	76.98
	Ensemble	88.1	67.3	95.6	97	85.2
	MaxLogit	82.1	61.6	90.2	93.2	80.1
	Energy	85.6	63.5	93.0	94.0	83.21
ViT-L-16	SC	90.69	64.1	93.3	95.8	82.6
	SE	92.36	66.23	94.0	96.71	84.7
	Ensemble	95.8	77.3	98.62	99.2	88.6
	MaxLogit	94.2	72.1	96.8	97.8	86.8
	Energy	93.5	70.4	95.1	96.8	85.3
ViT-L-32	SC	86.91	56.6	88.6	90.3	80.06
	SE	89.51	60.2	91.0	94.2	81.65
	Ensemble	95.6	70.1	97.2	97.0	88.1
	MaxLogit	94.1	63.4	94.0	95.2	84.2
	Energy	92.8	62.1	93.2	94.7	83.0
RN50	SC	52.8	51.34	58.2	61.4	53.4
	SE	53.1	52.27	60.8	62.7	54.2
	Maha	50.1	50.3	55.4	54.3	51.1
	Ensemble	62.5	59.8	67.4	68.3	60.1
	MaxLogit	58.4	56.1	63.5	67.4	57.45
	Energy	57.6	55.3	63.2	66.2	56.8
WRN-50-2	SC	54.6	52.1	59.5	63.4	55.3
	SE	57.0	54.3	61.8	66.1	57.6
	Maha	53.1	50.6	57.2	57.6	52.8
	Ensemble	64.2	61.3	69.23	71.4	62.4
	MaxLogit	60.1	57.87	66.4	69.2	58.2
	Energy	58.5	57.1	65.3	67.8	57.8

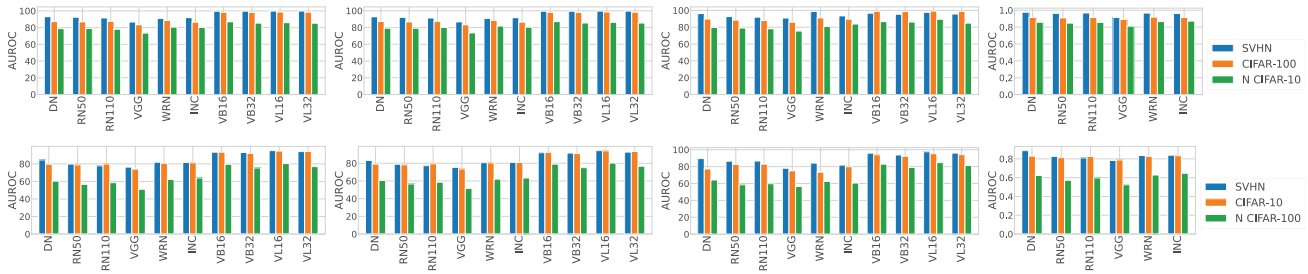
Table 1: AUROC% of OoD detection on 4 transformer and 2 convolutional architectures trained on ImageNet using ImageNet-O, Species, OpenImage-O, Texture and Near OoD ImageNet (N-ImageNet) as OoD.

For CIFAR-10/100, we use 10 different model architectures: 6 convolutional, DenseNet-121, ResNet-50/110, VGG-16, Wide-ResNet-28-10, Inception-v3 and the 4 ViT based models mentioned above. On each of these architectures, we evaluate 3 OoD detection methods: softmax confidence, entropy and predictive entropy of a 5-ensemble.

Competitive Benchmarks For comparison, we use conventionally used OoD detection benchmarks. In particular, we compare with (MNIST vs Fashion-MNIST), (CIFAR-10 vs SVHN/CIFAR-100), (CIFAR-100 vs SVHN/CIFAR-10) and (ImageNet vs ImageNet-O [24]/ Texture [5]/ Species [19]/ OpenImage-O [65]). We present the test set accuracies and AUROC scores for MNIST in table 4 of the appendix, CIFAR-10/100 and ImageNet test accuracies in table 5 of the appendix, corresponding AUROC scores for CIFAR-10/100 shown as bar plots in fig. 6 and finally, AUROC scores for ImageNet models in table 1. Related AUPRC scores are shown appendix C.

Observations Our observations are as follows:

1. *Except for the ImageNet vs Species benchmark, AUROC & AUPRC for all model architectures across all training datasets and all baselines are lower for our Near OoD samples as compared to conventionally used OoD datasets*. This brings into question, the performance of baselines which are considered to be robust to OoD inputs, as evidenced from their performance on standard benchmarks. It also provides evidence in favour of more challenging OoD detection benchmarks for evaluation which are far from saturation.

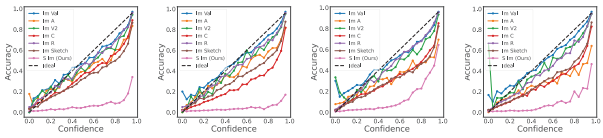


(a) Entropy (b) Confidence (c) Mahalanobis (d) Ensemble

Figure 6: AUROC % for different models, DenseNet-121 (DN), ResNet-50 (RN50), ResNet-110 (RN110), VGG-16, Wide-ResNet-28-10 (WRN) and Inception-v3 (INC), ViT-B-16/32 (VB16/32) and ViT-L-16/32 (VL16/32) trained on CIFAR-10 (first row) and CIFAR-100 (second row) using SVHN, CIFAR-10/100 and Near OoD (N) CIFAR-10/100 as OoD datasets.

Model	Im-val	Im-A	Im-v2	Im-C		Im-R	Im-Sketch	S Im (Ours)
		ECE %		*Avg ECE %*	*Max ECE %*		*ECE %*	
ViT-B-16	3.62	14.16	7.43	11.18	18.61	5.38	15.44	**26.15**
ViT-B-32	3.70	23.13	8.15	11.32	19.21	7.69	17.64	**29.15**
ViT-L-16	2.35	12.67	7.30	9.42	13.44	4.79	14.97	**22.76**
ViT-L-32	2.51	13.20	7.62	11.03	15.76	4.85	15.12	**23.54**

Table 2: ECE % on standard ImageNet (Im) shifts compared to our Shifted ImageNet (S Im).



(a) VB16 (b) VB32 (c) VL16 (d) VL32

Figure 7: Reliability plots for ImageNet shifts

2. *The order of performance is preserved.* We consistently observe that if a model M_1 outperforms a model M_2 on our Near OoD benchmark, it broadly outperforms M_2 on all conventionally used real-world benchmarks and vice-versa. To see this, we present the normalized AUROC scores for ImageNet trained models (in table 1) as a radar plot in fig. 8.

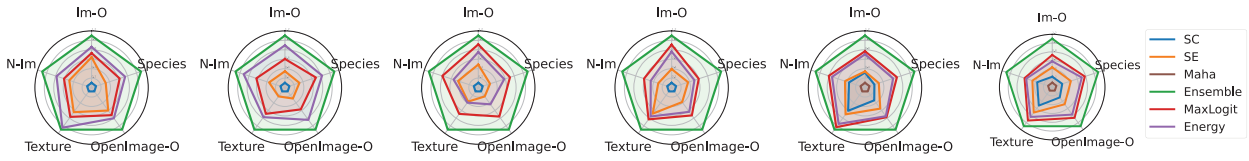
Discussion & Visualisation Firstly, as mentioned before, quite a few conventional OoD benchmarks (like CIFAR-10 vs SVHN) are nearing saturation and might become redundant in future for evaluating OoD detection methods. Our Near OoD datasets are much more effective at measuring performance as all the baselines broadly underperform in our setup. Secondly, even though the Near OoD samples might not look like real-world objects, the fact that the ordering of performance is consistently preserved between our benchmarks and conventional ones implies their potential to estimate real-world OoD detection performance. We visualise Near OoD samples from 10 ImageNet classes in fig. 9b and observe that such images contain patches from original classes in an odd order which makes the images unrecognizable, while still preserving close perceptual proximity to original classes. We find this to be true in general for other classes too. Thus, we posit that *it is not necessary for an OoD image to represent a real-world object as long as it captures certain desirable properties. For near OoD, the desirable property is to be semantically dissimilar while lying in the close perceptual vicinity of the training distribution*. Thus, our method provides a relatively easy alternative to gathering real-world OoD samples even for datasets where conventional benchmarks cannot be used.

4.4. Evaluating Shifted Datasets

In the second part of our experiments, we evaluate our method of learning distribution shifts by comparing the shifted datasets with other well-known synthetic and real-world shifts. For CIFAR-10, we compare with CIFAR-10-C [21] and for ImageNet, we compare with synthetic shifts: ImageNet-C, ImageNet-R (renditions) [20], ImageNet-Sketch [64] and real-world shifts: ImageNet-A [24] and ImageNet-V2 [55]. For CIFAR-10-C and ImageNet-C, we use corrupted images at the highest intensity 5.

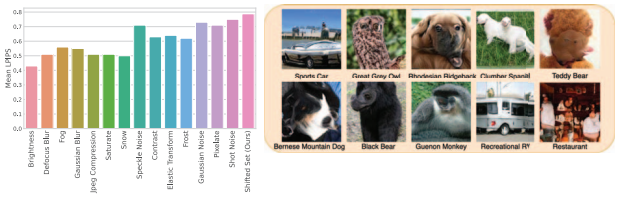
We report the Expected Calibration Error (ECE) in table 10 of the appendix for CIFAR-10 models and table 2 for ImageNet. Detailed results for each corruption type can be found in appendix C. For each model, the ECE for every competitive dataset is starkly lower than the ECE obtained on our Shifted datasets.

To better understand the effect of learned shifts, we compute the VGG LPIPS between each image in the ImageNet val set and the corresponding corrupted image obtained from using transformations in ImageNet-C (eg: brightness, Gaussian noise etc.). For the same validation image, we also compute the LPIPS with the transformed image using our Shift method. This is different from the FID analysis in fig. 5 as unlike FID, here, we are computing difference between individual pairs of images. Note that we ensure that the L2 norm of the difference between normal and corrupted images in the image space is same (set to 50) for all corruption types. We present the average LPIPS for each corruption type in fig. 9a. Although the input space L2 norm between image pairs remains the same for



(a) VB16 (b) VB32 (c) VL16 (d) VL32 (e) RN50 (f) WRN

Figure 8: **Ordering of AUROC scores of OoD detection baselines on 4 ViT and 2 convolutional architectures** trained on ImageNet-1K using OoD sets: ImageNet-O, Texture, Species, OpenImage-O and our Near OoD ImageNet (N-Im).



(a) (b)

Figure 9: (a) **average LPIPS between ImageNet val and corrupted/shifted samples**, (b) **Near OoD samples from 10 ImageNet classes.** We observe the Near OoD samples to contain patches from their respective original classes but in an odd order, making the images unrecognizable compared to the original class.

all corruption types, the high LPIPS for our shifted samples indicates that the corruptions learnt in our shifted dataset are again, not random transformations, rather they seem to be optimized to decrease feature space similarity (or increase feature space distance) between generated images and their original counterparts, in a neural network. At the same time, minimizing an ensemble's MI in $\mathcal{L}_{\text{Shift}}$ encourages the Pix-2-Pix model to learn transformations which make the neural network confident on its predictions. Such a transformation should therefore lead to inaccurate but confident, and thereby miscalibrated predictions from pre-trained models. To empirically verify this, we present reliability plots for Shifted-ImageNet vs all other shifts in fig. 7. Clearly, for Shifted ImageNet, as expected, models consistently have lower accuracy as compared to confidence, i.e., they are overconfident and miscalibrated, thereby providing evidence in favour of our claim above.

4.5. Evaluation of OoD detection in real life

Having thoroughly evaluated our method, in order to use them in real-life applications, we need to verify if we can compare baselines sensibly using our benchmarks. Indeed that is the very purpose of a benchmark. We make the observation here that in all our experiments across datasets, the ordering of performance between baselines is broadly consistent between our benchmarks and conventional ones. *Models which perform well on our benchmarks also perform well on conventional benchmarks and vice versa.* We show this consistency for our ImageNet baselines in a set of radar plots in fig. 8 (using observations in table 1), where we have normalized the AUROC scores of all baselines to lie between $[0.1, 1.1]$. Assuming that performance on conventional OoD detection benchmarks generalises to and is indicative of real-world OoD detection performance, we can then use the above observation as validation for our proposed benchmarks. Interestingly, our observations also indicate that we might not need an OoD dataset in the first place to evaluate a model's OoD detection performance. We could reliably estimate the OoD detection performance of any model just from the training set by following our proposed method. This has implications when working with new in-distribution training sets where OoD detection performance cannot be judged using conventional benchmarks. Even in such cases, our method can be used to provide a reasonable estimate of OoD detection performance.

5. Conclusion

Reliably evaluating the behaviour of models in unknown scenarios is an open problem and is extremely difficult because of infinitely many situations that could potentially fulfill our notion of "unknown" with respect to in-distribution. To our knowledge, our work is the first step in the direction where we do not advocate the naive approach of testing on arbitrarily chosen "other" datasets. Rather we propose to learn the distribution of samples satisfying constraints mimicking our notion of what is out-of-distribution. Through numerous experiments, we show that our generated samples provide for a more challenging and reliable benchmark for popularly used, state-of-the-art OoD detection baselines. In future, we want to explore how our approach can be extended to text-to-image generative models and if we can encode desirable out-of-distribution properties in natural language text. We thus hope that our work leads to an improvement in the standard procedure of evaluating OoD detection methods.

Acknowledgements The majority of this work was done in Meta AI. The Oxford authors are partially supported by the UKRI grant: Turing AI Fellowship EP/W002981/1 and EPSRC/MURI grant: EP/N019474/1. The Oxford authors would also like to thank the Royal Academy of Engineering and FiveAI. Meta AI authors are not supported by the UKRI grants nor have any relationships whatsoever to the grant.

References

[1] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pages 1613–1622. PMLR, 2015. 3

[2] Sebastian Bosse, Dominique Maniry, Klaus-Robert Müller, Thomas Wiegand, and Wojciech Samek. Deep neural networks for no-reference and full-reference image quality assessment. *IEEE Transactions on Image Processing*, 27(1):206–219, 2018. 4

[3] Clemens-Alexander Brust and Joachim Denzler. Not just a matter of semantics: the relationship between visual similarity and semantic similarity. *arXiv preprint arXiv:1811.07120*, 2018. 2

[4] Guangyao Chen, Peixi Peng, Xiangqian Wang, and Yonghong Tian. Adversarial reciprocal points learning for open set recognition. *arXiv preprint arXiv:2103.00953*, 2021. 3

[5] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014. 6, 13

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2, 3

[7] Thomas Deselaers and Vittorio Ferrari. Visual and semantic similarity in imagenet. In *CVPR 2011*, pages 1777–1784. IEEE, 2011. 2

[8] Nikolaos Dionelis. Omasgan: Out-of-distribution minimum anomaly score gan for sample generation on the boundary. *arXiv preprint arXiv:2110.15273*, 2021. 3

[9] Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. Vos: Learning what you don't know by virtual outlier synthesis. *arXiv preprint arXiv:2202.01197*, 2022. 3

[10] Nikita Durasov, Timur Bagautdinov, Pierre Baque, and Pascal Fua. Masksembles for uncertainty estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13539–13548, 2021. 3

[11] Angelos Filos, Panagiotis Tigkas, Rowan McAllister, Nicholas Rhinehart, Sergey Levine, and Yarin Gal. Can autonomous vehicles identify, recover from, and adapt to distribution shifts? In *International Conference on Machine Learning*, pages 3145–3153. PMLR, 2020. 1

[12] Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection, 2021. 1, 2, 3

[13] Yarin Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016. 3, 4

[14] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016. 3

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 2

[16] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020. 3

[17] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017. 3, 4

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[19] Dan Hendrycks, Steven Basart, Mantas Mazeika, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. *arXiv preprint arXiv:1911.11132*, 2019. 3, 6

[20] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020. 3, 7

[21] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 2, 3, 5, 7

[22] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016. 1, 3, 6

[23] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018. 6, 12, 13

[24] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021. 3, 6, 7

[25] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 5

[26] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324, 2019. 5

[27] 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*, pages 10951–10960, 2020. 3

[28] 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*, pages 4700–4708, 2017. 5

[29] Haiwen Huang, Zhihan Li, Lulu Wang, Sishuo Chen, Bin Dong, and Xinyu Zhou. Feature space singularity for out-

of-distribution detection. *arXiv preprint arXiv:2011.14654*, 2020. 3

[30] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 2, 4, 12

[31] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *arXiv preprint arXiv:1703.04977*, 2017. 3

[32] Pang Wei Koh, Shiori Sagawa, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021. 2, 3

[33] Shu Kong and Deva Ramanan. Opengan: Open-set recognition via open data generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 813–822, 2021. 3

[34] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2

[35] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 5

[36] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *arXiv preprint arXiv:1612.01474*, 2016. 1, 2, 3, 6

[37] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2, 3, 5

[38] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *arXiv preprint arXiv:1711.09325*, 2017. 3

[39] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018. 1, 6

[40] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017. 1, 3

[41] Jeremiah Zhe Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax-Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *arXiv preprint arXiv:2006.10108*, 2020. 1, 3

[42] Jeremiah Zhe Liu, Shreyas Padhy, Jie Ren, Zi Lin, Yeming Wen, Ghassen Jerfel, Zack Nado, Jasper Snoek, Dustin Tran, and Balaji Lakshminarayanan. A simple approach to improve single-model deep uncertainty via distance-awareness. *arXiv preprint arXiv:2205.00403*, 2022. 3

[43] Weitang Liu, Xiaoyun Wang, John D Owens, and Yixuan Li. Energy-based out-of-distribution detection. *arXiv preprint arXiv:2010.03759*, 2020. 3, 6

[44] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2537–2546, 2019. 1, 3

[45] Stephan Mandt, Matthew D Hoffman, and David M Blei. Stochastic gradient descent as approximate bayesian inference. *arXiv preprint arXiv:1704.04289*, 2017. 3

[46] Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip HS Torr, and Yarin Gal. Deterministic neural networks with appropriate inductive biases capture epistemic and aleatoric uncertainty. *arXiv preprint arXiv:2102.11582*, 2021. 1

[47] Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip HS Torr, and Yarin Gal. Deep deterministic uncertainty: A new simple baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24384–24394, 2023. 3

[48] Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip HS Torr, and Puneet K Dokania. Calibrating deep neural networks using focal loss. *arXiv preprint arXiv:2002.09437*, 2020. 3

[49] Martin Mundt, Sagnik Majumder, Iuliia Pliushch, Yong Won Hong, and Visvanathan Ramesh. Unified probabilistic deep continual learning through generative replay and open set recognition. *arXiv preprint arXiv:1905.12019*, 2019. 1

[50] Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012. 3, 4

[51] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011. 2

[52] Phuc Cuong Ngo, Amadeus Aristo Winarto, Connie Khor Li Kou, Sojeong Park, Farhan Akram, and Hwee Kuan Lee. Fence gan: Towards better anomaly detection. In *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 141–148. IEEE, 2019. 3

[53] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua V Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *arXiv preprint arXiv:1906.02530*, 2019. 3, 4

[54] Francesco Pinto, H Yang, Ser-Nam Lim, Philip HS Torr, and Puneet K Dokania. Regmixup: Mixup as a regularizer can surprisingly improve accuracy and out distribution robustness. In *arXiv: 2206.14502*, 2022. 1

[55] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pages 5389–5400. PMLR, 2019. 7

[56] Abhijit Guha Roy, Jie Ren, Shekoofeh Azizi, Aaron Loh, Vivek Natarajan, Basil Mustafa, Nick Pawlowski, Jan Freyberg, Yuan Liu, Zach Beaver, et al. Does your dermatology classifier know what it doesn't know? detecting the long-tail of unseen conditions. *arXiv preprint arXiv:2104.03829*, 2021. 1

[57] Chandramouli Shama Sastry and Sageev Oore. Detecting out-of-distribution examples with gram matrices. In *International Conference on Machine Learning*, pages 8491–8501. PMLR, 2020. 3

[58] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[59] Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. *Advances in Neural Information Processing Systems*, 34:144–157, 2021. 3

[60] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 5

[61] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019. 5

[62] Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In *International Conference on Machine Learning*, pages 9690–9700. PMLR, 2020. 1, 3

[63] Chu Wang, Yan-Ming Zhang, and Cheng-Lin Liu. Anomaly detection via minimum likelihood generative adversarial networks. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 1121–1126. IEEE, 2018. 3

[64] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32, 2019. 3, 7

[65] Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4921–4930, 2022. 3, 6

[66] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 4

[67] Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating neural network overconfidence with logit normalization. *arXiv preprint arXiv:2205.09310*, 2022. 3

[68] Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. *arXiv preprint arXiv:2002.06715*, 2020. 3

[69] Andrew G Wilson and Pavel Izmailov. Bayesian deep learning and a probabilistic perspective of generalization. *Advances in neural information processing systems*, 33:4697–4708, 2020. 3, 4

[70] Jim Winkens, Rudy Bunel, Abhijit Guha Roy, Robert Stanforth, Vivek Natarajan, Joseph R Ledsam, Patricia MacWilliams, Pushmeet Kohli, Alan Karthikesalingam, Simon Kohl, et al. Contrastive training for improved out-of-distribution detection. *arXiv preprint arXiv:2007.05566*, 2020. 1, 2

[71] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 2, 3

[72] Kai Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. *arXiv preprint arXiv:2006.09994*, 2020. 2, 3

[73] Jingkang Yang, Pengyun Wang, Dejian Zou, Zitang Zhou, Kunyuan Ding, Wenxuan Peng, Haoqi Wang, Guangyao Chen, Bo Li, Yiyou Sun, et al. Openood: Benchmarking generalized out-of-distribution detection. *Advances in Neural Information Processing Systems*, 35:32598–32611, 2022. 3

[74] Jianhao Yuan, Francesco Pinto, Adam Davies, Aarushi Gupta, and Philip Torr. Not just pretty pictures: Text-to-image generators enable interpretable interventions for robust representations. *arXiv preprint arXiv:2212.11237*, 2022. 3

[75] Muhammad Zaigham Zaheer, Jin-ha Lee, Marcella Astrid, and Seung-Ik Lee. Old is gold: Redefining the adversarially learned one-class classifier training paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14183–14193, 2020. 3

[76] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011. 4

[77] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 4

[78] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 13