OpenOOD v1.5: Enhanced Benchmark for Out-of-Distribution Detection

Jingyang Zhang[1], **Jingkang Yang**[2], **Pengyun Wang**[3], **Haoqi Wang**[4], **Yueqian Lin**[5], **Haoran Zhang**[1], **Yiyou Sun**[6], **Xuefeng Du**[6], **Kaiyang Zhou**[7], **Wayne Zhang**[4], **Yixuan Li**[6], **Ziwei Liu**[2], **Yiran Chen**[1], **Hai Li**[1]
[1]Duke University, Durham, USA [2]S-Lab, Nanyang Technological University, Singapore
[3]Beijing University of Posts and Telecommunications, Beijing, China
[4]SenseTime Research, Shenzhen, China [5]Duke Kunshan University, Kunshan, China
[6]University of Wisconsin-Madison, Madison, USA
[7]Hong Kong Baptist University, Hong Kong SAR, China
Code: https://github.com/Jingkang50/OpenOOD/
Leaderboard: https://zjysteven.github.io/OpenOOD/

Abstract

Out-of-Distribution (OOD) detection is critical for the reliable operation of open-world intelligent systems. Despite the emergence of an increasing number of OOD detection methods, the evaluation inconsistencies present challenges for tracking the progress in this field. OpenOOD v1 initiated the unification of the OOD detection evaluation but faced limitations in scalability and scope. In response, this paper presents OpenOOD v1.5, a significant improvement from its predecessor that ensures accurate and standardized evaluation of OOD detection methodologies at large scale. Notably, OpenOOD v1.5 extends its evaluation capabilities to large-scale datasets (ImageNet) and foundation models (*e.g.*, CLIP and DINOv2), and expands its scope to investigate full-spectrum OOD detection which considers *semantic* and *covariate* distribution shifts at the same time. This work also contributes in-depth analysis and insights derived from comprehensive experimental results, thereby enriching the knowledge pool of OOD detection methodologies. With these enhancements, OpenOOD v1.5 aims to drive advancements and offer a more robust and comprehensive evaluation benchmark for OOD detection research. A longer version of this paper is available at https://arxiv.org/pdf/2306.09301.pdf.

1 Introduction

For intelligent recognition systems to reliably operate in the open world, it is crucial for them to have the capability of detecting and handling unknown inputs. This problem is commonly formulated as Out-of-Distribution (OOD) detection [1] or Open-Set Recognition (OSR) [2]. In the context of image classification, OOD detection seeks to enable the identification of images that do not belong to any of the known, in-distribution (ID) categories of the classifier.

Recent years have witnessed a surge of over a hundred papers on OOD detection [3]. Despite the increasing attention and the importance of this research problem, tracking the progress in the field has been hindered by three *evaluation pitfalls* that are often overlooked by researchers: **1)** confusing terminologies, **2)** inconsistent datasets, and **3)** erroneous practices (we refer readers to Appendix A for a detailed discussion). As a result, the OOD detection community has a pressing need for a unified test platform and benchmark for accurate evaluation of current and future methodologies. One work that comes close to this goal is the first version of OpenOOD [4], which yet is limited in *scale* and *scope*. For example, OpenOOD v1's evaluation was mostly performed on small-scale datasets like

Workshop on Distribution Shifts, 37th Conference on Neural Information Processing Systems (NeurIPS 2023).

MNIST [5] and CIFAR [6, 7], while larger datasets such as ImageNet [8] obviously carry greater importance [9, 10, 11, 12, 13, 14].

Building upon OpenOOD v1, in this work we present OpenOOD v1.5 which features fair and accurate evaluation of OOD detection on a larger scale and with a broader scope. Concretely, we make the following extensions and contributions.[1]

Large-scale experiments and results. In addition to the small datasets included in v1, OpenOOD v1.5 provide extensive experiment results for nearly 40 methods (and their combinations) on ImageNet-1K, which serve as a comprehensive reference for later works. To facilitate future research on large-scale settings with affordable computational cost, we also introduce a new benchmark constructed with ImageNet-200, a subset of ImageNet-1K. Furthermore, we evaluate two large-scale foundation models (CLIP [15] and DINOv2 [16]) to provide initial inspection of their performance on OOD detection tasks.

Investigation on full-spectrum detection. Besides the *standard* setting considered in v1, OpenOOD v1.5 (for the first time) closely studies *full-spectrum* OOD detection [17], an important setting that considers OOD *generalization* [18, 19] and OOD *detection* simultaneously. Compared with the standard setting which is studied by most existing works, we show that full-spectrum detection poses significant challenge for all current approaches.

New insights. With comprehensive results from OpenOOD v1.5, we are able to provide several valuable observations. For example, we identify that there is *no single winner* that always outperforms others across multiple datasets. Meanwhile, we observe that *data augmentations* [20, 21, 22, 19, 23, 24] *help* with OOD detection in both standard and full-spectrum setting. Our insights help assess the current state of OOD detection and provide future directions for the community.

2 Overview of OpenOOD v1.5

In this section, we briefly introduce the 1) problem statement, 2) evaluation metrics, and 3) supported benchmarks and methods of OpenOOD v1.5. Due to space limit, we leave extensive details in the supplemental, including evaluation protocol (Appendix C), breakdown of datasets and methods (Appendices D and E), and experiment setup (Appendix F). Note that we focus on OOD detection in the context of multi-class image classification in this work.

Standard OOD detection. Given an image classification problem, there will always exist a pre-defined set of semantic categories/labels $\mathcal{Y}_{\text{ID}}$, which is considered in-distribution (ID). In an open world, an OOD label space $\mathcal{Y}_{\text{OOD}} = \{y | y \notin \mathcal{Y}_{\text{ID}}\}$ also exists. In the case of a CIFAR-10 classifier, for example, $\mathcal{Y}_{\text{ID}} = \{\texttt{airplane}, \texttt{bird}, ..., \texttt{truck}\}$, and $\mathcal{Y}_{\text{OOD}} = \{\texttt{apple}, \texttt{mountain}, ...\}$. At inference time, for any image x with ground-truth label y, an ideal classifier with OOD detection capability should behave in a way that: 1) it can identify whether $y \in \mathcal{Y}_{\text{ID}}$ or $y \in \mathcal{Y}_{\text{OOD}}$, and 2) if $y \in \mathcal{Y}_{\text{ID}}$, it classifies x to one of the ID categories accurately. While the second goal is fundamental for any image classifier, the first goal is the unique focus of OOD detection.

Full-spectrum OOD detection. Standard OOD detection essentially studies the *semantic* distribution shifts (between $\mathcal{Y}_{\text{ID}}$ and $\mathcal{Y}_{\text{OOD}}$); it yet ignores another type of distribution shifts that are prevalent in real-world, *i.e.*, *covariate* shifts [18, 25, 19]. Full-spectrum detection [17] fills this gap by further taking covariate-shifted ID images (csID) into the picture. Note that in the context of OOD detection, csID images are still *in-distribution* since their semantic labels are still within $\mathcal{Y}_{\text{ID}}$ (*e.g.*, a blurry dog image should be recognized as `dog` nonetheless). A concrete illustration of full-spectrum detection is shown in Figure 5 in the supplemental.

Evaluation metrics. OOD detection is a binary classification problem. Following convention [1], we treat the "anomalous" OOD samples as *positive* and the "normal" ID samples as *negative*. We use three well established metrics: 1) area under the receiver operating characteristic (AUROC), 2) area under the Precision-Recall curve (AUPR), and 3) false positive rate at 95% true positive rate (FPR@95). AUROC and AUPR are threshold-independent measurements, while FPR@95 reflects the performance at a specific threshold. In the manuscript we report AUROC as the main metric, while full results are available in this online document.

[1] As a benchmarking tool that continuously evolves, OpenOOD v1.5 also introduces several new features and updates that make itself more accurate, and easier to use. We describe them in Appendix B.

Table 1: Main results from OpenOOD v1.5 on standard OOD detection. In this table we use AUROC as the metric for OOD detection. Whenever applicable, we report the average number and the corresponding standard deviation obtained from 3 training runs. The best result within each group is **bolded**. Notes on missing results are in Appendix F. Full result table including other metrics and per-dataset statistics can be found in an online document, which we hide for now for double-blind review.

	CIFAR-10 (ResNet-18)			**CIFAR-100** (ResNet-18)			**ImageNet-200** (ResNet-18)			**ImageNet-1K** (ResNet-50)		
	Near-OOD	Far-OOD	ID Acc.	Near-OOD	Far-OOD	ID Acc.	Near-OOD	Far-OOD	ID Acc.	Near-OOD	Far-OOD	ID Acc.
- Post-hoc Inference Methods												
OpenMax [2] (CVPR'16)	87.62$_{(\pm 0.29)}$	89.62$_{(\pm 0.19)}$	95.06$_{(\pm 0.30)}$	76.41$_{(\pm 0.25)}$	79.48$_{(\pm 0.41)}$	77.25$_{(\pm 0.10)}$	80.27$_{(\pm 0.10)}$	90.20$_{(\pm 0.17)}$	86.37$_{(\pm 0.08)}$	74.77	89.26	76.18
MSP [1] (ICLR'17)	88.03$_{(\pm 0.25)}$	90.73$_{(\pm 0.43)}$	95.06$_{(\pm 0.30)}$	80.27$_{(\pm 0.11)}$	77.76$_{(\pm 0.44)}$	77.25$_{(\pm 0.10)}$	83.34$_{(\pm 0.06)}$	90.13$_{(\pm 0.09)}$	86.37$_{(\pm 0.08)}$	76.02	85.23	76.18
TempScale [26] (ICML'17)	88.09$_{(\pm 0.31)}$	90.97$_{(\pm 0.52)}$	95.06$_{(\pm 0.30)}$	80.90$_{(\pm 0.07)}$	78.74$_{(\pm 0.51)}$	77.25$_{(\pm 0.10)}$	**83.69**$_{(\pm 0.04)}$	90.82$_{(\pm 0.09)}$	86.37$_{(\pm 0.08)}$	77.14	87.56	76.18
ODIN [27] (ICLR'18)	82.87$_{(\pm 1.85)}$	87.96$_{(\pm 0.61)}$	95.06$_{(\pm 0.30)}$	79.90$_{(\pm 0.11)}$	79.28$_{(\pm 0.21)}$	77.25$_{(\pm 0.10)}$	80.27$_{(\pm 0.08)}$	91.71$_{(\pm 0.19)}$	86.37$_{(\pm 0.08)}$	74.75	89.47	76.18
MDS [28] (NeurIPS'18)	84.20$_{(\pm 2.40)}$	89.72$_{(\pm 1.36)}$	95.06$_{(\pm 0.30)}$	58.69$_{(\pm 0.09)}$	69.39$_{(\pm 1.39)}$	77.25$_{(\pm 0.10)}$	61.93$_{(\pm 0.51)}$	74.72$_{(\pm 0.26)}$	86.37$_{(\pm 0.08)}$	55.44	74.25	76.18
MDSEns [28] (NeurIPS'18)	60.43$_{(\pm 0.26)}$	73.90$_{(\pm 0.27)}$	95.06$_{(\pm 0.30)}$	46.31$_{(\pm 0.24)}$	66.00$_{(\pm 0.69)}$	77.25$_{(\pm 0.10)}$	54.32$_{(\pm 0.24)}$	69.27$_{(\pm 0.57)}$	86.37$_{(\pm 0.08)}$	49.67	67.52	76.18
RMDS [29] (arXiv'21)	89.80$_{(\pm 0.28)}$	92.20$_{(\pm 0.21)}$	95.06$_{(\pm 0.30)}$	80.15$_{(\pm 0.11)}$	**82.92**$_{(\pm 0.42)}$	77.25$_{(\pm 0.10)}$	82.57$_{(\pm 0.25)}$	88.06$_{(\pm 0.34)}$	86.37$_{(\pm 0.08)}$	76.99	86.38	76.18
Gram [30] (ICML'20)	58.66$_{(\pm 4.83)}$	71.73$_{(\pm 3.20)}$	95.06$_{(\pm 0.30)}$	51.66$_{(\pm 0.77)}$	73.36$_{(\pm 1.08)}$	77.25$_{(\pm 0.10)}$	67.67$_{(\pm 1.07)}$	71.19$_{(\pm 0.24)}$	86.37$_{(\pm 0.08)}$	61.70	79.71	76.18
EBO [31] (NeurIPS'20)	87.58$_{(\pm 0.46)}$	91.21$_{(\pm 0.92)}$	95.06$_{(\pm 0.30)}$	80.91$_{(\pm 0.08)}$	79.77$_{(\pm 0.61)}$	77.25$_{(\pm 0.10)}$	82.50$_{(\pm 0.05)}$	90.86$_{(\pm 0.21)}$	86.37$_{(\pm 0.08)}$	75.89	89.47	76.18
OpenGAN [32] (ICCV'21)	53.71$_{(\pm 7.68)}$	54.61$_{(\pm 15.51)}$	95.06$_{(\pm 0.30)}$	65.98$_{(\pm 1.26)}$	67.88$_{(\pm 7.16)}$	77.25$_{(\pm 0.10)}$	59.79$_{(\pm 3.39)}$	73.15$_{(\pm 4.07)}$	86.37$_{(\pm 0.08)}$	N/A	N/A	N/A
GradNorm [33] (NeurIPS'21)	54.90$_{(\pm 0.98)}$	57.55$_{(\pm 3.22)}$	95.06$_{(\pm 0.30)}$	70.13$_{(\pm 0.47)}$	69.14$_{(\pm 1.05)}$	77.25$_{(\pm 0.10)}$	72.75$_{(\pm 0.48)}$	84.26$_{(\pm 0.87)}$	86.37$_{(\pm 0.08)}$	72.96	90.25	76.18
ReAct [9] (NeurIPS'21)	87.11$_{(\pm 0.61)}$	90.42$_{(\pm 1.41)}$	95.06$_{(\pm 0.30)}$	80.77$_{(\pm 0.05)}$	80.39$_{(\pm 0.49)}$	77.25$_{(\pm 0.10)}$	81.87$_{(\pm 0.98)}$	92.31$_{(\pm 0.56)}$	86.37$_{(\pm 0.08)}$	77.38	93.67	76.18
MLS [10] (ICML'22)	87.52$_{(\pm 0.47)}$	91.10$_{(\pm 0.89)}$	95.06$_{(\pm 0.30)}$	**81.05**$_{(\pm 0.07)}$	79.67$_{(\pm 0.57)}$	77.25$_{(\pm 0.10)}$	82.90$_{(\pm 0.04)}$	91.11$_{(\pm 0.19)}$	86.37$_{(\pm 0.08)}$	76.46	89.57	76.18
KLM [10] (ICML'22)	79.19$_{(\pm 0.80)}$	82.68$_{(\pm 0.21)}$	95.06$_{(\pm 0.30)}$	76.56$_{(\pm 0.25)}$	76.24$_{(\pm 0.52)}$	77.25$_{(\pm 0.10)}$	80.76$_{(\pm 0.08)}$	88.53$_{(\pm 0.11)}$	86.37$_{(\pm 0.08)}$	76.64	87.60	76.18
VIM [11] (CVPR'22)	88.68$_{(\pm 0.28)}$	**93.48**$_{(\pm 0.24)}$	95.06$_{(\pm 0.30)}$	74.98$_{(\pm 0.13)}$	81.70$_{(\pm 0.62)}$	77.25$_{(\pm 0.10)}$	78.68$_{(\pm 0.24)}$	91.26$_{(\pm 0.19)}$	86.37$_{(\pm 0.08)}$	72.08	92.68	76.18
KNN [12] (ICML'22)	**90.64**$_{(\pm 0.20)}$	92.96$_{(\pm 0.14)}$	95.06$_{(\pm 0.30)}$	80.18$_{(\pm 0.15)}$	82.40$_{(\pm 0.17)}$	77.25$_{(\pm 0.10)}$	81.57$_{(\pm 0.17)}$	93.16$_{(\pm 0.22)}$	86.37$_{(\pm 0.08)}$	71.10	90.18	76.18
DICE [34] (ECCV'22)	78.34$_{(\pm 0.79)}$	84.23$_{(\pm 1.89)}$	95.06$_{(\pm 0.30)}$	79.38$_{(\pm 0.23)}$	80.01$_{(\pm 0.18)}$	77.25$_{(\pm 0.10)}$	81.78$_{(\pm 0.14)}$	90.80$_{(\pm 0.31)}$	86.37$_{(\pm 0.08)}$	73.07	90.95	76.18
RankFeat [35] (NeurIPS'22)	79.46$_{(\pm 2.52)}$	75.87$_{(\pm 5.06)}$	95.06$_{(\pm 0.30)}$	61.88$_{(\pm 1.28)}$	67.10$_{(\pm 1.42)}$	77.25$_{(\pm 0.10)}$	56.92$_{(\pm 1.59)}$	38.22$_{(\pm 3.85)}$	86.37$_{(\pm 0.08)}$	50.99	53.93	76.18
ASH [13] (ICLR'23)	75.27$_{(\pm 1.04)}$	78.49$_{(\pm 2.58)}$	95.06$_{(\pm 0.30)}$	78.20$_{(\pm 0.15)}$	80.58$_{(\pm 0.66)}$	77.25$_{(\pm 0.10)}$	82.38$_{(\pm 0.19)}$	**93.90**$_{(\pm 0.27)}$	86.37$_{(\pm 0.08)}$	**78.17**	**95.74**	76.18
SHE [14] (ICLR'23)	81.54$_{(\pm 0.51)}$	85.32$_{(\pm 1.43)}$	95.06$_{(\pm 0.30)}$	78.95$_{(\pm 0.18)}$	76.92$_{(\pm 1.16)}$	77.25$_{(\pm 0.10)}$	80.18$_{(\pm 0.25)}$	89.81$_{(\pm 0.61)}$	86.37$_{(\pm 0.08)}$	73.78	90.92	76.18
- Training Methods (w/o Outlier Data)												
ConfBranch [36] (arXiv'18)	89.84$_{(\pm 0.24)}$	92.85$_{(\pm 0.29)}$	94.88$_{(\pm 0.05)}$	71.60$_{(\pm 0.62)}$	68.90$_{(\pm 1.83)}$	76.59$_{(\pm 0.27)}$	79.10$_{(\pm 0.24)}$	90.43$_{(\pm 0.18)}$	85.92$_{(\pm 0.07)}$	70.66	83.94	75.63
RotPred [37] (NeurIPS'19)	**92.68**$_{(\pm 0.27)}$	96.62$_{(\pm 0.18)}$	**95.35**$_{(\pm 0.52)}$	76.43$_{(\pm 0.16)}$	**88.40**$_{(\pm 0.13)}$	76.03$_{(\pm 0.38)}$	81.59$_{(\pm 0.20)}$	92.56$_{(\pm 0.09)}$	**86.37**$_{(\pm 0.16)}$	**76.52**	90.00	**76.55**
G-ODIN [38] (CVPR'20)	89.12$_{(\pm 0.57)}$	95.51$_{(\pm 0.31)}$	94.70$_{(\pm 0.25)}$	77.15$_{(\pm 0.28)}$	85.67$_{(\pm 1.58)}$	74.46$_{(\pm 0.04)}$	77.28$_{(\pm 0.10)}$	92.33$_{(\pm 0.11)}$	84.56$_{(\pm 0.28)}$	70.77	85.51	74.85
CSI [39] (NeurIPS'20)	89.51$_{(\pm 0.19)}$	92.00$_{(\pm 0.30)}$	91.16$_{(\pm 0.14)}$	71.45$_{(\pm 0.27)}$	66.31$_{(\pm 1.21)}$	61.60$_{(\pm 0.46)}$	N/A	N/A	N/A	N/A	N/A	N/A
ARPL [40] (TPAMI'21)	87.44$_{(\pm 0.15)}$	89.31$_{(\pm 0.32)}$	93.66$_{(\pm 0.11)}$	74.94$_{(\pm 0.93)}$	73.69$_{(\pm 1.80)}$	70.70$_{(\pm 1.08)}$	82.02$_{(\pm 0.10)}$	89.23$_{(\pm 0.11)}$	83.95$_{(\pm 0.32)}$	76.30	85.50	75.87
MOS [41] (CVPR'21)	71.45$_{(\pm 3.09)}$	76.41$_{(\pm 5.93)}$	94.83$_{(\pm 0.37)}$	80.40$_{(\pm 0.18)}$	80.17$_{(\pm 1.21)}$	76.98$_{(\pm 0.20)}$	69.84$_{(\pm 0.46)}$	80.46$_{(\pm 0.92)}$	85.60$_{(\pm 0.20)}$	72.85	82.75	72.81
VOS [42] (ICLR'22)	87.70$_{(\pm 0.48)}$	90.83$_{(\pm 0.92)}$	94.31$_{(\pm 0.64)}$	**80.93**$_{(\pm 0.29)}$	81.32$_{(\pm 0.09)}$	77.20$_{(\pm 0.10)}$	82.51$_{(\pm 0.11)}$	91.00$_{(\pm 0.28)}$	86.23$_{(\pm 0.19)}$	N/A	N/A	N/A
LogitNorm [43] (ICML'22)	92.33$_{(\pm 0.08)}$	**96.74**$_{(\pm 0.06)}$	94.30$_{(\pm 0.25)}$	78.47$_{(\pm 0.31)}$	81.53$_{(\pm 1.26)}$	76.34$_{(\pm 0.17)}$	**82.66**$_{(\pm 0.15)}$	93.04$_{(\pm 0.21)}$	86.04$_{(\pm 0.15)}$	74.62	91.54	76.45
CIDER [44] (ICLR'23)	90.71$_{(\pm 0.16)}$	94.71$_{(\pm 0.36)}$	N/A	73.10$_{(\pm 0.39)}$	80.49$_{(\pm 0.68)}$	N/A	80.58$_{(\pm 1.75)}$	90.66$_{(\pm 1.68)}$	N/A	68.97	**92.18**	N/A
NPOS [45] (ICLR'23)	89.78$_{(\pm 0.33)}$	94.07$_{(\pm 0.49)}$	N/A	78.35$_{(\pm 0.37)}$	82.29$_{(\pm 1.55)}$	N/A	79.40$_{(\pm 0.39)}$	**94.49**$_{(\pm 0.07)}$	N/A	N/A	N/A	N/A
- Training Methods (w/ Outlier Data)												
OE [46] (NeurIPS'18)	**94.82**$_{(\pm 0.21)}$	**96.00**$_{(\pm 0.13)}$	94.63$_{(\pm 0.26)}$	**88.30**$_{(\pm 0.10)}$	**81.41**$_{(\pm 1.49)}$	**76.84**$_{(\pm 0.42)}$	**84.84**$_{(\pm 0.16)}$	**89.02**$_{(\pm 0.18)}$	85.82$_{(\pm 0.21)}$	N/A	N/A	N/A
MCD [47] (ICCV'19)	91.03$_{(\pm 0.12)}$	91.00$_{(\pm 1.10)}$	94.95$_{(\pm 0.04)}$	77.07$_{(\pm 0.32)}$	74.72$_{(\pm 0.78)}$	75.83$_{(\pm 0.04)}$	83.62$_{(\pm 0.09)}$	88.94$_{(\pm 0.10)}$	**86.12**$_{(\pm 0.17)}$	N/A	N/A	N/A
UDG [48] (ICCV'21)	89.91$_{(\pm 0.25)}$	94.06$_{(\pm 0.90)}$	92.36$_{(\pm 0.84)}$	78.02$_{(\pm 0.10)}$	79.59$_{(\pm 1.77)}$	71.54$_{(\pm 0.64)}$	74.30$_{(\pm 1.63)}$	82.09$_{(\pm 2.78)}$	68.11$_{(\pm 1.24)}$	N/A	N/A	N/A
MixOE [49] (WACV'23)	88.73$_{(\pm 0.82)}$	91.93$_{(\pm 0.69)}$	94.55$_{(\pm 0.32)}$	80.95$_{(\pm 0.20)}$	76.40$_{(\pm 1.44)}$	75.13$_{(\pm 0.06)}$	82.62$_{(\pm 0.03)}$	88.27$_{(\pm 0.41)}$	85.71$_{(\pm 0.07)}$	N/A	N/A	N/A

Benchmarks. OpenOOD v1.5 supports 6 benchmarks, including 4 for standard detection and 2 for full-spectrum detection. The standard benchmarks are constructed by taking CIFAR-10 [6], CIFAR-100 [7], ImageNet-200, and ImageNet-1K [8] as the ID data, respectively. The full-spectrum benchmarks are built around ImageNet-200 and ImageNet-1K, respectively. In Appendix D, we describe the used OOD datasets and discuss how our benchmarks exhibits higher fairness and usefulness than previous ones.

Supported methods. OpenOOD v1.5 now implements in total 40 advanced and most recent methods. We go over each of them in Appendix E.

3 Analysis

In this section, we discuss multiple observations and insights that arise from the benchmarking efforts of OpenOOD v1.5. We start by analyzing Table 1, which presents main results of standard OOD detection on 4 benchmarks. Then we specifically look at full-spectrum detection, whose results are summarized in Figure 1. Lastly, we provide initial inspection of large foundation models (CLIP [15] and DINOv2 [16]) on the task of OOD detection.

Standard detection. From the comprehensive results over 4 benchmarks in Table 1 we can identify several general observations for standard OOD detection. One example is that *there is no single winner* that consistently outperform others across benchmarks, and the ranking between methods can be quite different from one dataset to another. For example, ReAct [9] and ASH [13] are extremely powerful on ImageNet but less competitive on CIFAR. In contrast, methods that yield remarkable performance on small datasets (*e.g.*, KNN [12] and RotPred [37]) do not show clear advantage on large datasets. We believe the absence of a clear winner can be attributed, in part, to the evaluation inconsistencies of current methods, underscoring the importance of our benchmark. Due to space limit, we refer readers to Appendix G for more results and observations (*e.g.*, that data augmentation is consistently beneficial for OOD detection).

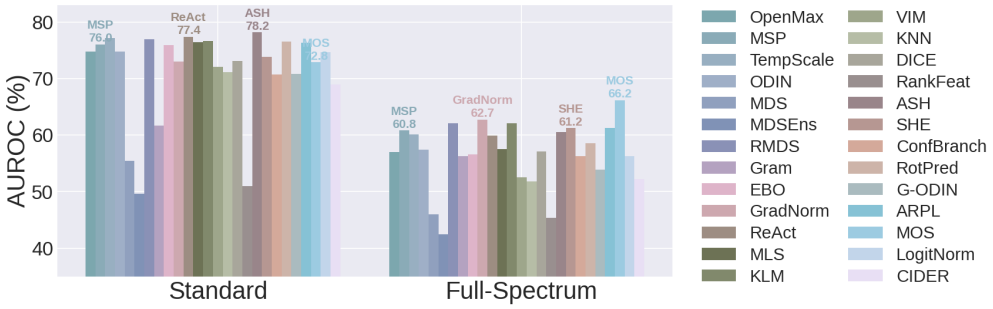

Figure 1: Comparison between standard and full-spectrum detection on ImageNet-1K (near-OOD). Many detectors suffer significant performance degradation in the full-spectrum setting.

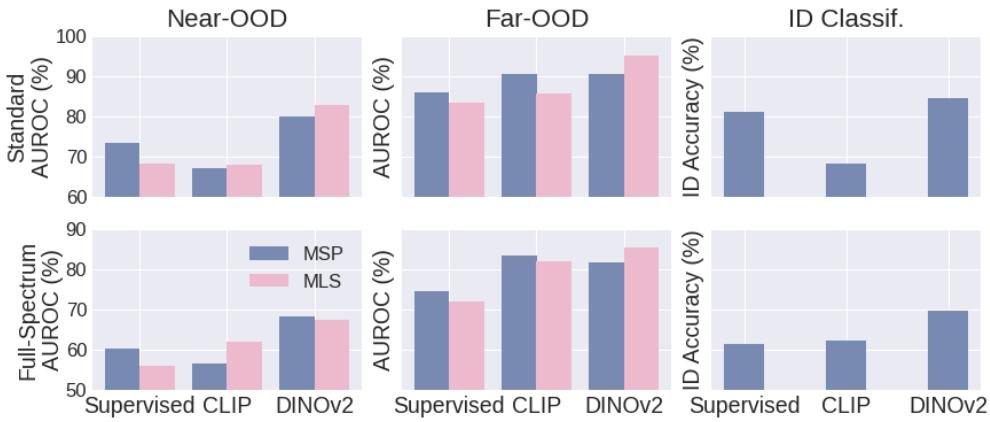

Figure 2: Performance of foundation models on ImageNet-1K. Here we consider OOD detectors that are compatible with the zero-shot CLIP (MSP and MLS).

Full-Spectrum detection. In Figure 1, it is obvious that the near-OOD AUROC of most methods decreases by >10% when changing from standard scenario to full-spectrum scenario (similar trend holds for far-OOD). The performance drop suggests that *existing OOD detectors can be sensitive to the non-semantic covariate shift and are likely to flag covariate-shifted ID samples as OOD*. Such behaviour is not ideal because: 1) It does not align with human perception/decision (*e.g.*, a human annotator classifying `dog` and `car` wouldn't mark a covariate-shifted dog image as something unknown or novel). 2) It can harm the classifier's generalization capability on covariate-shifted ID data, as often times it is assumed that the classifier would refrain from making predictions when the sample is identified as OOD. As a result, we firmly believe that full-spectrum detection is an important open problem. For example, it would be interesting to see if OOD generalization methods specifically designed to handle covariate-shifted data can help with full-spectrum OOD detection.

Foundation models. Foundation models—large pretrained models that can be adapted for a wide range of tasks—have demonstrated superior recognition capability over the task-specific strictly-supervised counterparts [15]. In particular, zero-shot foundation models generalize significantly better when facing *covariate*-shifted samples when it comes to OOD generalization [50]. This observation motivates us to see whether the same advantage exists when foundation models handle *semantic*-shifted samples, *i.e.*, in the context of OOD detection.

To this end, we evaluate two foundation models (CLIP [15] and DINOv2 [16]) on our ImageNet-1K benchmark and compare them with a ImageNet-1K trained classifier. We consider zero-shot classification for CLIP (using the prompt-ensemble to obtain the classifier weight as in [15]) for the reason mentioned above, and use linear probe for DINOv2 which does not have zero-shot capability. We do not consider fine-tuning, whose effect on OOD detection has been studied in [51]. All three classifiers share the same ViT-Base backbone.

The results are summarized in Figure 2. Compared with the task-specific supervised model, zero-shot CLIP shows substantial improvements on far-OOD detection in both standard and full-spectrum setting, yet the comparison on near-OOD detection is nuanced and gives no conclusive remark. In fact, giving any conclusion at this moment will be too early: Most existing detectors are *not* compatible with zero-shot CLIP, and the only detector that specifically targets CLIP [52] adopts a simple design that is equivalent to MSP [1]. We hypothesize that CLIP's power may not be fully unleashed until more advanced or appropriate detector is developed. Meanwhile, we notice that the linear probe of self-supervised DINOv2 consistently and remarkably outperforms the fully-supervised counterpart, in *all* OOD detection cases and in ID classification. This demonstrates that large-scale self-supervised foundation models are also powerful for handling semantic distribution shifts.

4 Conclusion

In this work we present OpenOOD v1.5, an enhanced OOD detection benchmark that features comprehensive evaluation with large-scale datasets on both standard and full-spectrum detection. We provide several observations from our results to identify open problems and provide future directions. In addition, we evaluate two foundation models and find that while self-supervised DINOv2 already shows improvements, how to leverage zero-shot CLIP for better OOD detection requires more investigation. We hope that OpenOOD can accelerate the progress and foster collective efforts towards advancing the state-of-the-art in OOD detection. For more discussions including related work and limitation, please see Appendix H.

Acknowledgments and Disclosure of Funding

This study is supported by the Ministry of Education, Singapore, under its MOE AcRF Tier 2 (MOE-T2EP20221- 0012), NTU NAP, and under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s).

References

[1] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*, 2017. 1, 2, 3, 5, 10, 13, 15, 16, 17

[2] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *CVPR*, 2016. 1, 3, 10, 13

[3] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*, 2021. 1

[4] Jingkang Yang, Pengyun Wang, Dejian Zou, Zitang Zhou, Kunyuan Ding, WenXuan Peng, Haoqi Wang, Guangyao Chen, Bo Li, Yiyou Sun, Xuefeng Du, Kaiyang Zhou, Wayne Zhang, Dan Hendrycks, Yixuan Li, and Ziwei Liu. OpenOOD: Benchmarking generalized out-of-distribution detection. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL https://openreview.net/forum?id=gT6j4_tskUt. 1, 10, 12

[5] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 2012. 2, 12

[6] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2, 3, 12

[7] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 and cifar-100 datasets. *URl: https://www. cs. toronto. edu/kriz/cifar. html*, 6(1):1, 2009. 2, 3, 12

[8] 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, 12

[9] Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. In *NeurIPS*, 2021. 2, 3, 14, 15, 16, 17

[10] Dan Hendrycks, Steven Basart, Mantas Mazeika, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. In *ICML*, 2022. 2, 3, 14, 16

[11] Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *CVPR*, 2022. 2, 3, 12, 14

[12] Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. *ICML*, 2022. 2, 3, 14, 17

[13] Andrija Djurisic, Nebojsa Bozanic, Arjun Ashok, and Rosanne Liu. Extremely simple activation shaping for out-of-distribution detection. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ndYXTEL6cZz. 2, 3, 14, 15, 16, 17

[14] Jinsong Zhang, Qiang Fu, Xu Chen, Lun Du, Zelin Li, Gang Wang, xiaoguang Liu, Shi Han, and Dongmei Zhang. Out-of-distribution detection based on in-distribution data patterns memorization with modern hopfield energy. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=KkazG4lgKL. 2, 3, 14, 17

[15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 3, 4

[16] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 3, 4

[17] Jingkang Yang, Kaiyang Zhou, and Ziwei Liu. Full-spectrum out-of-distribution detection. *arXiv preprint arXiv:2204.05306*, 2022. 2

[18] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019. 2, 12

[19] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021. 2, 12, 14, 15, 16, 17

[20] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bygh9j09KX. 2, 14, 15, 16, 17

[21] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. 2, 14, 15, 16, 17

[22] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. AugMix: A simple data processing method to improve robustness and uncertainty. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. 2, 14, 15, 16, 17

[23] Dan Hendrycks, Andy Zou, Mantas Mazeika, Leonard Tang, Dawn Song, and Jacob Steinhardt. Pixmix: Dreamlike pictures comprehensively improve safety measures. *arXiv preprint arXiv:2112.05135*, 2021. 2, 14, 15, 16, 17

[24] Francesco Pinto, Harry Yang, Ser-Nam Lim, Philip Torr, and Puneet K. Dokania. Using mixup as a regularizer can surprisingly improve accuracy & out-of-distribution robustness. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=5j6fWcPccO. 2, 14, 15, 16, 17

[25] 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. 2, 12

[26] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *ICML*, 2017. 3, 13

[27] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *ICLR*, 2018. 3, 10, 11, 13, 18

[28] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*, 2018. 3, 11, 13, 16

[29] Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. A simple fix to mahalanobis distance for improving near-ood detection. *arXiv preprint arXiv:2106.09022*, 2021. 3, 13, 16

[30] Chandramouli Shama Sastry and Sageev Oore. Detecting out-of-distribution examples with gram matrices. In *ICML*, 2020. 3, 13

[31] Weitang Liu, Xiaoyun Wang, John D Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *NeurIPS*, 2020. 3, 13

[32] Shu Kong and Deva Ramanan. Opengan: Open-set recognition via open data generation. In *ICCV*, 2021. 3, 10, 11, 13, 15

[33] Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. In *NeurIPS*, 2021. 3

[34] Yiyou Sun and Sharon Li. Dice: Leveraging sparsification for out-of-distribution detection. In *ECCV*, 2022. 3, 14

[35] Yue Song, Nicu Sebe, and Wei Wang. Rankfeat: Rank-1 feature removal for out-of-distribution detection. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=-deKNiSOXLG. 3, 14

[36] Terrance DeVries and Graham W Taylor. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*, 2018. 3, 14

[37] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/a2b15837edac15df90721968986f7f8e-Paper.pdf. 3, 14, 16

[38] 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 *CVPR*, 2020. 3, 11, 14

[39] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. In *NeurIPS*, 2020. 3, 10, 13, 14, 15, 18

[40] Guangyao Chen, Peixi Peng, Xiangqian Wang, and Yonghong Tian. Adversarial reciprocal points learning for open set recognition. *arXiv preprint arXiv:2103.00953*, 2021. 3, 14

[41] Rui Huang and Yixuan Li. Mos: Towards scaling out-of-distribution detection for large semantic space. In *CVPR*, 2021. 3, 12, 14

[42] Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. Vos: Learning what you don't know by virtual outlier synthesis. In *ICLR*, 2022. 3, 14, 15

[43] Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating neural network overconfidence with logit normalization. In *ICML*, 2022. 3, 14, 16

[44] Yifei Ming, Yiyou Sun, Ousmane Dia, and Yixuan Li. How to exploit hyperspherical embeddings for out-of-distribution detection? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=aEFaE0W5pAd. 3, 14, 15

[45] Leitian Tao, Xuefeng Du, Jerry Zhu, and Yixuan Li. Non-parametric outlier synthesis. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JHklpEZqduQ. 3, 14, 15

[46] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *ICLR*, 2019. 3, 11, 13, 14, 16

[47] Qing Yu and Kiyoharu Aizawa. Unsupervised out-of-distribution detection by maximum classifier discrepancy. In *ICCV*, 2019. 3, 11, 13, 14

[48] Jingkang Yang, Haoqi Wang, Litong Feng, Xiaopeng Yan, Huabin Zheng, Wayne Zhang, and Ziwei Liu. Semantically coherent out-of-distribution detection. In *ICCV*, 2021. 3, 11, 12, 13, 14

[49] Jingyang Zhang, Nathan Inkawhich, Randolph Linderman, Yiran Chen, and Hai Li. Mixture outlier exposure: Towards out-of-distribution detection in fine-grained environments. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5531–5540, January 2023. 3, 11, 13, 14, 18

[50] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7959–7971, 2022. 4

[51] Yifei Ming and Yixuan Li. How does fine-tuning impact out-of-distribution detection for vision-language models? *arXiv preprint arXiv:2306.06048*, 2023. 4

[52] Yifei Ming, Ziyang Cai, Jiuxiang Gu, Yiyou Sun, Wei Li, and Yixuan Li. Delving into out-of-distribution detection with vision-language representations. In *Advances in Neural Information Processing Systems*, 2022. 5

[53] Pramuditha Perera and Vishal M Patel. Deep transfer learning for multiple class novelty detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11544–11552, 2019. 10

[54] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015. 12

[55] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011. 12

[56] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014. 12

[57] 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. 12

[58] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need. In *ICLR*, 2022. 12

[59] Julian Bitterwolf, Maximilian Müller, and Matthias Hein. In or out? fixing imagenet out-of-distribution detection evaluation. *arXiv preprint arXiv:2306.00826*, 2023. 12, 18

[60] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018. 12

[61] Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses, 2021. 12

[62] 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: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. 12

[63] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 13

[64] Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng-Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In *ECCV*, 2018. 13

[65] Faruk Ahmed and Aaron Courville. Detecting semantic anomalies. In *AAAI*, 2020. 13

[66] Antonio Torralba, Rob Fergus, and William T Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *TPAMI*, 2008. 13

[67] Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. In *NeurIPS*, 2021. 13, 17

[68] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 14

[69] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 14

[70] TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016. 15

[71] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy. 15, 16

[72] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 15, 16

[73] Justin Gilmer and Dan Hendrycks. A discussion of 'adversarial examples are not bugs, they are features': Adversarial example researchers need to expand what is meant by 'robustness'. *Distill*, 2019. doi: 10.23915/distill.00019.1. https://distill.pub/2019/advex-bugs-discussion/response-1. 16

[74] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *NeurIPS*, 2019. 16

[75] Yifei Ming, Hang Yin, and Yixuan Li. On the impact of spurious correlation for out-of-distribution detection. In *The AAAI Conference on Artificial Intelligence (AAAI)*, 2022. 16

[76] Fahim Tajwar, Ananya Kumar, Sang Michael Xie, and Percy Liang. No true state-of-the-art? ood detection methods are inconsistent across datasets. *arXiv preprint arXiv:2109.05554*, 2021. 18

[77] Ido Galil, Mohammed Dabbah, and Ran El-Yaniv. A framework for benchmarking class-out-of-distribution detection and its application to imagenet. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Iuubb9W6Jtk. 18

[78] Konstantin Kirchheim, Marco Filax, and Frank Ortmeier. Pytorch-ood: A library for out-of-distribution detection based on pytorch. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4351–4360, June 2022. 18

[79] Nathan Inkawhich, Jingyang Zhang, Eric K. Davis, Ryan Luley, and Yiran Chen. Improving out-of-distribution detection by learning from the deployment environment. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:2070–2086, 2022. doi: 10.1109/JSTARS.2022.3146362. 18

[80] Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ml safety. *arXiv preprint arXiv:2109.13916*, 2021. 18

[81] Ev Zisselman and Aviv Tamar. Deep residual flow for out of distribution detection. In *CVPR*, 2020. 18, 19

[82] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Why normalizing flows fail to detect out-of-distribution data. In *NeurIPS*, 2020. 18, 19

[83] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *NeurIPS*, 2018. 18

[84] 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. 2020. 18, 19

[85] Jingyang Zhang, Yiran Chen, and Hai Li. Privacy leakage of adversarial training models in federated learning systems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 108–114, June 2022. 19

[86] Nathan Inkawhich, Gwendolyn McDonald, and Ryan Luley. Adversarial attacks on foundational vision models. *arXiv preprint arXiv:2308.14597*, 2023. 19

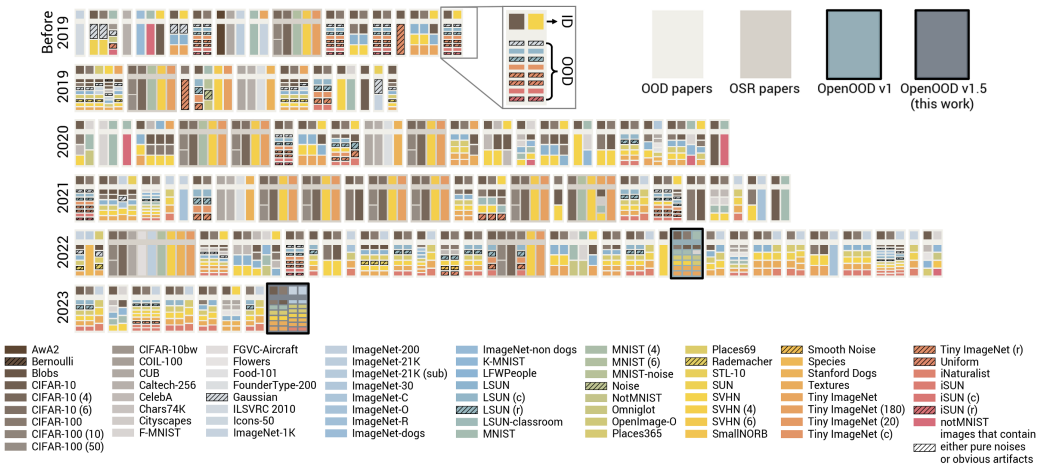

Figure 3: Summarizing evaluation settings of 100+ recent OOD detection and OSR works from NeurIPS, AAAI, ICLR, CVPR, ICML, and ICCV/ECCV (zoom in to view better). Each box stands for a paper. Within the box, each column shows the ID dataset and corresponding OOD datasets which are represented by the color blocks. The lack of a consistent color pattern between boxes signifies the inconsistency in the evaluation setup of current works. Multiple works also adopted unrealistic or problematic data [39] in evaluation (marked by the hatch pattern). The community still suffers from such chaos after OpenOOD v1 [4] came out. Full paper list used to produce this figure can be found here.

A Evaluation Pitfalls of OOD Detection

Here we explain the three evaluation pitfalls that we identify in the current OOD detection research.

Confusing terminologies. Despite subtle differences in the way of constructing their test environments, OOD detection and Open-Set Recognition (OSR) (or sometimes, "novelty detection") are essentially pursuing the same goal [2, 1]. With two different terminologies, however, the two topics often diverge from each other in a counterproductive way, where methods are developed and compared separately within each branch using different benchmarks.

Inconsistent datasets. Given an ID dataset, the simplest practice is to use other datasets with semantically different visual categories as OOD datasets. Unfortunately, we have seen great inconsistency in the selected datasets for OOD detection evaluation. Such phenomenon is highlighted in Figure 3, where we summarize the evaluation settings of 100+ recent works from top-tier machine learning conferences. The lack of a consistent pattern indicates how different the used datasets are from paper to paper, causing great difficulty for straight comparison between methods. The evaluation settings within the OSR branch are more consistent yet are significantly limited in scale (see more discussion in Appendix D).

Erroneous practices could compromise evaluation if no extra care is taken. One example that is pervasive in OOD detection works is leaking information about OOD data that is used for evaluation. More specifically, some methods train the model or tune hyperparameters with test OOD data [53, 27, 32]. Such practices go against basic machine learning principles and will lead to overoptimistic results.

B New Features and Updates of OpenOOD v1.5

OpenOOD v1.5 introduces new features including a *leaderboard* hosted online to track the-state-of-the-art based on various methods and a lightweight *evaluator* which enables easy evaluation with a few lines of code (see Figure 4). Other updates such as adding newer methods and fixing implementation bugs are documented in the online changelog.

```python
from openood.evaluation_api import Evaluator
from openood.networks import ResNet50
from torchvision.models import ResNet50_Weights
from torch.hub import load_state_dict_from_url

# Load an ImageNet-pretrained model from torchvision
net = ResNet50()
weights = ResNet50_Weights.IMAGENET1K_V1
net.load_state_dict(load_state_dict_from_url(weights.url))
preprocessor = weights.transforms()
net.eval(); net.cuda()

# Initialize an evaluator and evaluate with
# ASH postprocessor
evaluator = Evaluator(net, id_name='imagenet',
    preprocessor=preprocessor, postprocessor_name='ash')
metrics = evaluator.eval_ood()
```

Rank	Training	Postprocessor	Near-OOD AUROC	Far-OOD AUROC	ID Accuracy	Outlier Data	Model Arch.	Additional Description
1	AugMix	ASH	82.16	96.05	77.63	✗	ResNet50	official ckpt
2	CrossEntropy	RMDS	80.09	92.60	81.54	✗	ViT-B/16	torchvision ckpt
3	AugMix	ReAct	79.94	93.70	77.63	✗	ResNet50	official ckpt
4	RandAugment	ASH	79.83	95.01	76.90	✗	ResNet50	finetuned 30 epochs
5	CrossEntropy	MDS	79.04	92.60	81.54	✗	ViT-B/16	torchvision ckpt
6	RegMixup	ASH	78.45	95.35	76.68	✗	ResNet50	finetuned 30 epochs
7	CrossEntropy	RMDS	78.34	92.34	81.59	✗	Swin-T	torchvision ckpt
8	RandAugment	ReAct	78.30	93.50	76.90	✗	ResNet50	finetuned 30 epochs
9	CrossEntropy	ASH	78.17	95.74	76.18	✗	ResNet50	torchvision ckpt
10	RegMixup	ReAct	77.68	92.45	76.68	✗	ResNet50	finetuned 30 epochs
11	AugMix	MSP	77.49	86.67	77.63	✗	ResNet50	official ckpt
12	CrossEntropy	ReAct	77.38	93.67	76.18	✗	ResNet50	torchvision ckpt
13	CrossEntropy	TempScale	77.14	87.56	76.18	✗	ResNet50	torchvision ckpt
14	RegMixup	MSP	77.04	86.31	76.68	✗	ResNet50	finetuned 30 epochs
15	CrossEntropy	VIM	77.03	92.84	81.54	✗	ViT-B/16	torchvision ckpt

Figure 4: **Left:** An example of evaluting ImageNet-1K models in a few lines with our `Evaluator`. **Right:** Screenshot of top entries on our ImageNet-1K leaderboard. Zoom in to view better.

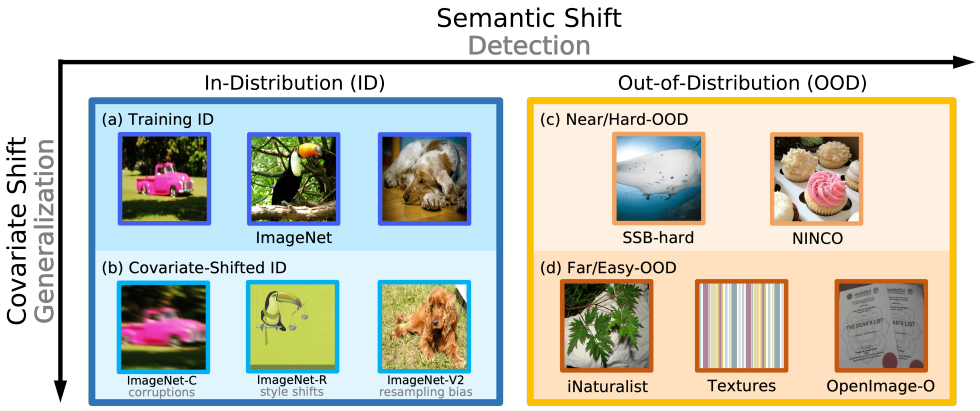

Figure 5: Illustration of full-spectrum OOD detection [48]. Standard detection only concerns *semantic* shift by detecting (c) + (d) from (a), while full-spectrum detection takes into account *covariate* shift and aims to separate (c) + (d) from (a) + (b). An ideal system should be robust to the non-semantic covariate shift (OOD *generalization*) while being able to identify semantic shift (OOD *detection*).

C Detailed Design of Evaluation Protocol

Here we expand upon Section 2 to provide details on the curated evaluation protocol of OpenOOD which is designed to enable the most rigorous assessment of current and future methodologies.

Overview. We consider the dataset that characterizes the given image classification task as in-distribution (ID) dataset $\mathcal{D}_{\text{ID}}$. Following common practices, we take publicly available datasets whose categories are OOD w.r.t. $\mathcal{Y}_{\text{ID}}$ as the source of OOD samples $\mathcal{D}_{\text{OOD}}$. In general, the image classifier will be trained with ID training images $\mathcal{D}_{\text{ID}}^{\texttt{train}}$ and evaluated with ID test images $\mathcal{D}_{\text{ID}}^{\texttt{test}}$ and OOD test images $\mathcal{D}_{\text{OOD}}^{\texttt{test}}$. For each $\mathcal{D}_{\text{ID}}$, we evaluate the classifier and detector with multiple $\mathcal{D}_{\text{OOD}}$ for comprehensiveness. Moreover, we divide the considered OOD datasets into two groups: near-OOD and far-OOD (or equivalently, hard-OOD and easy-OOD). The grouping is based on either image semantics or empirical difficulty, which can give a more fine-grained evaluation of OOD detectors in the face of different OOD samples (see Appendix D for more details).

Validation data for hyperparameter tuning. Multiple OOD detectors have tunable hyperparameters. In contrast to earlier works which determine hyperparameter values directly using test samples $\mathcal{D}_{\text{ID}}^{\texttt{test}}$ and $\mathcal{D}_{\text{OOD}}^{\texttt{test}}$ [27, 28, 38, 32], we instead introduce ID and OOD validation samples $\mathcal{D}_{\text{ID}}^{\texttt{val}}$ and $\mathcal{D}_{\text{OOD}}^{\texttt{val}}$ to ensure realistic evaluation and avoid reporting overoptimistic results. Specifically, $\mathcal{D}_{\text{ID}}^{\texttt{val}}$ is a small subset held out from $\mathcal{D}_{\text{ID}}^{\texttt{test}}$, and $\mathcal{D}_{\text{OOD}}^{\texttt{val}}$ is carefully constructed such that $\mathcal{Y}_{\text{OOD}}^{\texttt{val}} \cap \mathcal{Y}_{\text{OOD}}^{\texttt{test}} = \varnothing$.

OOD training data. A line of works choose to incorporate OOD images at training time to improve OOD detection capability [46, 47, 48, 49]. To avoid trivial evaluation in such cases [46], we make distinction between OOD training samples $\mathcal{D}_{\text{OOD}}^{\texttt{train}}$ and OOD test samples $\mathcal{D}_{\text{OOD}}^{\texttt{test}}$, where they should have non-overlapping categories between each other (*i.e.*, $\mathcal{Y}_{\text{OOD}}^{\texttt{train}} \cap \mathcal{Y}_{\text{OOD}}^{\texttt{test}} = \varnothing$).

Table 2: Summary of the 4 standard OOD detection benchmarks of OpenOOD v1.5. Datasets without a reference are ones that we curate from existing ones; see text for details. Full-spectrum benchmarks only adds additional covariate-shifted samples into ID test set, which we also describe in text.

ID samples	OOD test samples $\mathcal{D}_{\text{OOD}}^{\texttt{test}}$		OOD training samples	OOD validation samples
$\mathcal{D}_{\text{ID}}$	Near-OOD	Far-OOD	$\mathcal{D}_{\text{OOD}}^{\texttt{train}}$	$\mathcal{D}_{\text{OOD}}^{\texttt{val}}$
CIFAR-10 [6]	CIFAR-100 [7], TIN [54]	MNIST [5], SVHN [55], Textures [56], Places365 [57]	TIN-597	20-class hold-out set of TIN
CIFAR-100 [7]	CIFAR-10 [6], TIN [54]	MNIST [5], SVHN [55], Textures [56], Places365 [57]	TIN-597	20-class hold-out set of TIN
ImageNet-200	SSB-hard [58], NINCO [59]	iNaturalist [60], Textures [56], OpenImage-O [11]	ImageNet-800	Hold-out set of OpenImage-O
ImageNet-1K [8]	SSB-hard [58], NINCO [59]	iNaturalist [60], Textures [56], OpenImage-O [11]	N/A	Hold-out set of OpenImage-O

Evaluating full-spectrum detection. As mentioned earlier, full-spectrum detection additionally considers covariate-shifted ID samples (csID). In practice, this is done by simply incorporating csID samples $\mathcal{D}_{\text{csID}}^{\texttt{test}}$ together with $\mathcal{D}_{\text{ID}}^{\texttt{test}}$ to serve as the ID test data.

D Benchmark Breakdown

We describe each benchmark in detail in this section. A high-level summary is provided in Table 2.

CIFAR-10. The first benchmark considers CIFAR-10 [6] as ID. We use the official train set as $\mathcal{D}_{\text{ID}}^{\texttt{train}}$ and hold out 1,000 samples from the test set to form $\mathcal{D}_{\text{ID}}^{\texttt{val}}$, while the remaining 9,000 test samples are taken as $\mathcal{D}_{\text{ID}}^{\texttt{test}}$. The *near*-OOD group contains CIFAR-100 [7] and Tiny ImageNet (TIN) [54]. 1,203 images are removed from TIN due to the overlap with CIFAR [48]. Another 1,000 TIN images covering 20 categories are held out to serve as $\mathcal{D}_{\text{OOD}}^{\texttt{val}}$ which is disjoint with $\mathcal{D}_{\text{OOD}}^{\texttt{test}}$. The *far*-OOD group consists of MNIST [5], SVHN [55], Textures [56], and Places365 [57] with 1,305 images removed due to semantic overlap [4]. The OOD group is determined by image content and semantics: Near-OOD images are similar to CIFAR-10 as they all include specific objects, while far-OOD images are either numerical digits, textural patterns, or scene imagery.

CIFAR-100. The CIFAR-100 benchmark is similar to the CIFAR-10 one. We take 1,000 samples out of the ID test set as ID validation data. The *near*-OOD split is made of CIFAR-10 and TIN. The *far*-OOD group is the same as for CIFAR-10.

ImageNet-1K. We use 45K images from the ImageNet validation set [8] as $\mathcal{D}_{\text{ID}}^{\texttt{test}}$, while the remaining 5K images serve as $\mathcal{D}_{\text{ID}}^{\texttt{val}}$. We do not modify the original 1.2M ImageNet training set such that *any pre-trained models can be directly evaluated with OpenOOD*.

We include SSB-hard [58] and NINCO [59] in the *near*-OOD group for ImageNet-1K. SSB-hard consists of 49K images and covers 980 categories selected from ImageNet-21K [61]. NINCO is a new dataset of 5879 images manually curated by Bitterwolf et al. [59]. The *far*-OOD group considers iNaturalist [60], Textures [56], and OpenImage-O [11]. The first two datasets were first used as benchmarks in the MOS paper [41] and later have become popular for evaluating ImageNet models. OpenImage-O is curated from Open Images [62]. 1,763 images from OpenImage-O are picked out as $\mathcal{D}_{\text{OOD}}^{\texttt{val}}$. Unlike CIFAR, for ImageNet which has 1K diverse visual categories, it is hard to define near/far-OOD based on label semantics. Instead, here we make the categorization according to the empirical difficulty that we observe in experiments.

ImageNet-1K (full-spectrum). The only difference with the standard benchmark is that we further include covariate-shifted ID samples $\mathcal{D}_{\text{csID}}^{\texttt{test}}$ and consider $\mathcal{D}_{\text{csID}}^{\texttt{test}}$ and $\mathcal{D}_{\text{ID}}^{\texttt{test}}$ together as ID. We use three different $\mathcal{D}_{\text{csID}}^{\texttt{test}}$: ImageNet-C [18] with image corruptions, ImageNet-R [19] with style changes, and ImageNet-V2 [25] with resampling bias. They are all commonly used for evaluating classifiers' ability to generalize to covariate-shifted ID images. ImageNet-C has 15 corruption types each with 5 severities. We randomly sample 10K images uniformly across the 75 combinations to form the test set that is used in OpenOOD. OOD datasets remain the same so that a straight comparison can be made between the standard and full-spectrum setting.

ImageNet-200 and the full-spectrum version. We further consider a 200-class subset of ImageNet-1K which is still relatively large yet requires less compute to experiment with compared to ImageNet-1K. ImageNet-200 has the same 200 categories as ImageNet-R [19]. It shares the same OOD datasets as our ImageNet-1K benchmark. In the full-spectrum setting, we again incorporate ImageNet-C, ImageNet-R, and ImageNet-V2.

OOD training data. Our benchmark also specifies OOD training samples $\mathcal{D}_{\text{OOD}}^{\texttt{train}}$ which can be incorporated into training when applicable [46, 47, 48, 49]. To construct $\mathcal{D}_{\text{OOD}}^{\texttt{train}}$ for CIFAR-10/100, we start from the 800 categories in ImageNet-1K that are apart from the 200 classes of TIN and filter out 203 categories relevant to CIFAR-10/100 based on WordNet [63]. Named as TIN-597, the resulting dataset has 597 classes which do not overlap with any of the categories from $\mathcal{D}_{\text{OOD}}^{\texttt{test}}$ and serves as a good candidate for $\mathcal{D}_{\text{OOD}}^{\texttt{train}}$. For ImageNet-200, we directly take the rest 800 categories' images from ImageNet-1K as $\mathcal{D}_{\text{OOD}}^{\texttt{train}}$ (namely ImageNet-800). We do not consider $\mathcal{D}_{\text{OOD}}^{\texttt{train}}$ for ImageNet-1K since it is hard to find images that do not overlap with $\mathcal{D}_{\text{OOD}}^{\texttt{test}}$, and no relevant methods that train with OOD data have considered ImageNet-1K.

Comparison with prior OOD and OSR benchmarks. Most of the OOD datasets considered in OOD detection literature fall into our far-OOD category, meaning that the more difficult near-OOD detection was less emphasized than our benchmarks do. Meanwhile, we exclude problematic OOD datasets (*e.g.*, the resized version of LSUN and TIN [27] with obvious artifacts [39]) which make detection trivial and much less meaningful [39]. Following one of the seminal works [64], OSR papers often construct ID-OOD pairs by partitioning a single dataset into two splits (*e.g.*, using a 6-class subset of CIFAR-10 as ID and the other 4-class subset as OOD). While such practice is well-suited for studying near-OOD detection [65] (*i.e.*, minimum covariate shift exhibits between ID and OOD data), it inevitably reduces the scale and complexity of the problem (in terms of the number of ID and OOD categories), making the resulted benchmarks less representative for real-world scenarios. In contrast, our benchmarks hold the covariate shift between ID and near-OOD samples to a low level, while not sacrificing the scale. Specifically, all of our near-OOD groups include a certain dataset that comes from the same source as ID data (or essentially we are partitioning a much larger dataset to create ID-OOD pair). For example, in the CIFAR-10 benchmark, CIFAR-100 is one of the near-OOD datsets. They both are subset of Tiny Images [66] and thus have minimum covariate shift between each other.

Comparison with the benchmarks in v1. Among the 6 benchmarks in v1.5, CIFAR-10/100 and ImageNet-1K standard benchmark are adapted from v1 release with necessary changes for fairness and usefulness (*e.g.*, unlike v1, v1.5 ensures that $\mathcal{D}_{\text{OOD}}^{\texttt{train}}$, $\mathcal{D}_{\text{OOD}}^{\texttt{val}}$, and $\mathcal{D}_{\text{OOD}}^{\texttt{test}}$ are strictly disjoint with each other). ImageNet-200 and full-spectrum benchmarks are newly introduced in v1.5.

E Supported Methods

We overview the supported methods of OpenOOD v1.5. Like in v1 we prioritize methods that have public implementations and thus can be more easily and reliably adapted into our framework. They are categorized into four groups. **Post-hoc inference methods** designs post-processors that are applied to the base classifier to generate the "OOD score" (which is then thresholded to produce the binary ID/OOD prediction). They only take effect at inference phase and by default assume that the classifier is trained with the standard cross-entropy loss. In contrast, **training methods** involve training-time regularization. Most of them assume no access to the auxiliary OOD training data (**w/o outlier data**), while some methods do (**w/ outlier data**). We also consider several **data augmentation** methods.

Post-Hoc Inference Methods. Given an input image, OOD detectors often function by assigning an "OOD score" that is computed based on certain outputs from the base classifier, which will then be thresholded to give the binary prediction. The first line of works focus on designing such post-processors/scoring mechanisms that best separate ID and OOD samples. **MSP** [1] takes the maximum softmax probability over ID categories as the score. **OpenMax** [2] is a replacement for the softmax layer which directly estimates the probability of an input being from an unknown class. **TempScale** [26] calibrates softmax probabilities with temperature scaling. **ODIN** [27] further introduces input preprocessing on top of TempScale. **MDS** [28] fits class-conditional Gaussian distribution on the penultimate layer features of the classifier and derives OOD score with Mahalanobis distance. **MDSEns** [28] is another version of MDS which leverages multiple intermediate layers and forms a feature ensemble. **RMDS** [29] improves MDS by considering the "background score" computed from an unconditional Gaussian distribution. **Gram** [30] identifies abnormal patterns from the Gram Matrices of intermediate feature maps. **EBO** [31] applies energy function to the logits to compute OOD score. **OpenGAN** [32] trains a GAN in the classifier's feature space and uses the discriminator as the post-processor. **GradNorm** [67] computes the KL divergence between the softmax probability distribution and the uniform distribution and takes the gradients of penultimate layer weights w.r.t.

the KL divergence as OOD score. **ReAct** [9] rectifies feature vectors by thresholding their elements with a certain magnitude. **MLS** [10] uses the maximum logit. **KLM** [10] looks at the KL divergence between the softmax probability distribution and a "template" distribution. **VIM** [11] augments the logits with the norm of feature residual compared with ID training samples' features to compute the OOD score. **KNN** [12] applies KNN to the penultimate layer's features. **DICE** [34] sparsifies the last linear layer before computing the logits. **RankFeat** [35] transforms the feature matrices such that their rank is 1. **ASH** [13] shapes later layer activations by removing a large portion of the elements and simplifing the rest. **SHE** [14] maintains a template representation for each ID category and detects OOD samples by measuring the distance between the representation of an input to that template.

Training methods without outlier data. Unlike post-hoc methods that only interfere with the inference process, training methods involve training-time regularization to enhance OOD detection capability. **ConfBranch** [36] trains another branch in addition to the classification one to explicitly learn the estimate of model uncertainty. **RotPred** [37] includes an extra head to predict the rotation angle of rotated inputs in a self-supervised manner, and the rotation head together with the classification head is used for OOD detection. **G-ODIN** [38] utilizes a dividend/divisor structure and decomposes the softmax confidence for better ID-OOD separation. **CSI** [39] explores self-supervised contrastive learning objectives for OOD detectors. **ARPL** [40] introduces "reciprocal points" for each ID category and trains the model by pushing the reciprocal point away from the corresponding ID cluster and encouraging OOD samples to gather around the reciprocal point. **MOS** [41] incorporates a two-level hierarchical classifier and designs an accompanying OOD score to benefit OOD detection especially in large-scale settings. **VOS** [42] regularizes the feature space of the classifier under the assumption that the learned representations follow conditional Gaussian distributions. **LogitNorm** [43] mitigates the over-confidence issue by training and testing with normalized logit vectors. **CIDER** [44] regularizes the model's hyperspherical space by increasing inter-class separability and intra-class compactness. **NPOS** [45] is a non-parametric version of VOS which removes the Gaussian assumption and instead adopts KNN to model the feature distribution.

Training methods with outlier data. While most methods consider the standard ID-only training, some works assume the access to auxiliary OOD training samples. **OE** [46] is the seminal work in this thread, which lets the classifier learn OOD detection in a supervised fashion. **MCD** [47] considers an ensemble of multiple classification heads and promotes the disagreement between each head's prediction on OOD samples. **UDG** [48] proposes a clustering-based method to practically extract OOD samples from a mixed pool of auxiliary data and to improve the learned representation quality with unsupervised learning. **MixOE** [49] performs pixel-level mixing operations between ID and OOD samples and regularizes the model such that the prediction confidence smoothly decays as the input transitions from ID to OOD.

Data augmentations. We consider several data augmentation methods which have demonstrated success for improving the generalization ability of image classifiers. **StyleAugment** [20] applies style transfer to clean images to emphasize the shape bias over the texture bias. **RandAugment** [21] randomly sample the augmentation operation and magnitude to increase the diversity of augmented images. **AugMix** [22] linearly interpolate between the clean and the augmented image to preserve the natural looking/fidelity of training images for better generalization. **DeepAugment** [19] manipulates the low-level statistics of clean images by sending them through image-to-image network and distorting the network's weights. **PixMix** [23] mixes two images with conical combination to create various new inputs with similar semantics. **RegMixup** [24] trains the model with both clean images and mixed images obtained from convex combination.

F Experiment Setup

We perform extensive experiments to evaluate a wide range of methods on the supported benchmarks. This section describes the training and evaluation setup of our benchmarking experiments.

Training. For CIFAR-10/100 and ImageNet-200, we train a ResNet-18 [68] for 100 epochs. We consider the standard cross-entropy training for post-hoc methods. The optimizer is SGD with a momentum of 0.9. We use a learning rate of 0.1 with cosine annealing decay schedule [69]. A weight decay of 0.0005 is applied. The batch size is 128 for CIFAR-10/100 and 256 for ImageNet-200.

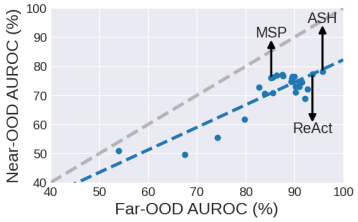

Figure 6: Near-OOD improvements are proportional to, yet slower than, far-OOD improvements on ImageNet-1K.

	ImageNet-200			ImageNet-1K			
	MSP [1]	ASH [13]	ID Acc.	MSP [1]	ReAct [9]	ASH [13]	ID Acc.
CrossEntropy	83.34 / 90.13	82.38 / 93.90	86.37	76.02 / 85.23	77.38 / 93.67	78.17 / 95.74	76.18
StyleAugment [20]	80.99 / 88.44	80.65 / 93.70	83.41	75.78 / 85.73	76.70 / 91.88	78.21 / 94.90	74.68
RandAugment [21]	83.17 / 90.34	81.56 / 94.53	86.58	76.60 / 85.27	78.30 / 93.50	79.81 / 95.01	76.90
AugMix [22]	83.49 / 90.68	**82.87** / 94.66	87.01	**77.49 / 86.67**	**79.94 / 93.70**	**82.16 / 96.05**	**77.63**
DeepAugment [19]	81.39 / 88.79	80.61 / 93.84	85.00	76.67 / 86.26	78.43 / 92.12	79.14 / 93.90	76.77
PixMix [23]	82.15 / 90.23	81.36 / **95.01**	85.79	76.86 / 85.63	79.12 / 91.59	78.92 / 92.17	77.44
RegMixup [24]	**84.13 / 90.81**	79.38 / 92.74	**87.25**	77.04 / 86.31	77.68 / 92.45	78.45 / 95.35	76.68

Table 3: Data augmentation methods (column headers) are beneficial for OOD detection and amplify the performance gain when combined with post-hoc methods (row headers). The cell numbers represent the near-OOD / far-OOD AUROC.

Some methods have specific setup, and we adopt their official implementations and hyperparameters whenever possible.

For ImageNet-1K, we evaluate post-hoc methods with pre-trained models from torchvision [70]. In addition to ResNet-50 that is considered in OpenOOD v1, v1.5 further includes ViT [71] and Swin Transformer [72] for comprehensive evaluation. For training methods, we focus on ResNet-50 and use official checkpoints when possible. Otherwise, we fine-tune the torchvision pre-trained checkpoint for 30 epochs with a learning rate of 0.001. Again, we use a batch size of 256 and weight decay of 0.0005. CIFAR and ImageNet models are trained using 1 and 2 Quadro RTX 6000 GPUs (24GB memory), respectively. Except ImageNet-1K experiments, we perform 3 independent training runs for each method. All training runs can be easily reproduced by running OpenOOD and providing corresponding configuration files (*e.g.*, training a ResNet-18 on CIFAR-10 is as simple as `python main.py --config configs/cifar10.yml configs/resnet18_32x32.yml`). We refer to our github repo for details, which thoroughly documents all bash training scripts.

Evaluation. As aforementioned, we use AUROC, AUPR, and FPR@95 as metrics. In the paper we focus on near- and far-OOD AUROC which are averaged over all OOD datasets in each group. AUROC can be interpreted as the probability that the detector correctly separates ID and OOD samples; the random-guessing baseline is 50%, and the higher the better. Results under other metrics and per-dataset statistics are available in our full result table. For post-hoc methods, OpenOOD supports automatic hyperparameter search using ID and OOD validation samples. The hyperparameter that yields the best AUROC is used for the final test. Similar to training, evaluation can be performed by running simple bash scripts, which can be found in our online code repo.

Notes on missing results. The main results for standard OOD detection are presented in Table 1. A few numbers are missing for the following reasons. OpenGAN [32] has not shown success on ImageNet-1K, and substantial changes are required to make it work with ImageNet models. CSI [39], VOS [42], and NPOS [45] are infeasible with our compute resources on ImageNet. CIDER [44] and NPOS trains the CNN backbone without the final linear classifier, and the exact code for evaluating ID accuracy is not provided in their official implementations. Lastly, we do not consider training with outlier data $\mathcal{D}_{\text{OOD}}^{\texttt{train}}$ on ImageNet-1K since it is difficult to find OOD training samples that do not overlap with test OOD data, and no relevant methods have considered ImageNet-1K.

Experiments on foundation models. For zero-shot CLIP, we follow the exact way of constructing prompt ensemble and zero-shot classifier weights recommended by the authors of CLIP.[2] We take the outputs of zero-shot CLIP as the logits of each ID category. For DINOv2 linear probe, we use the official weights[3] for the classification head.

G Additional Results and Observations

In this section we present remaining results and observations that we do not cover in Section 3 due to space limit.

[2] https://github.com/openai/CLIP/blob/main/notebooks/Prompt_Engineering_for_ImageNet.ipynb

[3] https://github.com/facebookresearch/dinov2

Data augmentations help. While data augmentations have been shown beneficial for standard classification [21] and OOD generalization [20, 22, 19, 23, 24], their effects for OOD detection remain unclear. In Table 3, we find that several data augmentation methods, despite not being designed to improve OOD detection, can actually boost detection rates in many cases. More interestingly, the performance gain is amplified when they are combined with powerful post-processors. For example, compared with the baseline of cross-entropy training on ImageNet-1K near-OOD, AugMix [22] achieves 76.02 + **1.47** = 77.49% AUROC and 78.17 + **3.99** = 82.16% AUROC when working with MSP [1] and ASH [13], respectively. 82.16% is the current best score among all methods. The results indicate that the effects from data augmentations and post-processors are complementary.

As to why data augmentations benefit OOD detection, one possible explanation is that the diverse training samples they introduce can help model better capture semantic-correlated features rather than spurious features [73, 74] (*i.e.*, features that correlate to the label but are not semantically meaningful; examples include high-frequency pattern [73] and adversarial noise [74]). Without data augmentations, models may be easily activated by OOD samples that contain spurious features [75]. In contrast, models with data augmentations will activate less in the face of OOD samples, making ID and OOD samples more separable.

Near-OOD remains more challenging than far-OOD. Figure 6 plots the trend of near-OOD AUROC v.s far-OOD AUROC on ImageNet-1K. Not surprisingly, near-OOD AUROC is (roughly) proportional to far-OOD AUROC, meaning that the improvement for one group is likely to help with the other as well. Meanwhile, we notice that the progress on near-OOD is slower than that on far-OOD. Besides the fact that near-OOD detection is more difficult, this may also be due to that previous works mainly focus on far-OOD datasets when designing and evaluating their methods.

Vision transformers do not outperform ResNets. We visualize in Figure 7 the performance of a few powerful post-hoc methods on ImageNet-1K with ResNet-50, ViT-B-16 [71], and Swin-T [72] as the classifier. We use the ImageNet-1K-pre-trained checkpoints provided by torchvision. As reference, they have 25.6M, 86.6M, and 28.3M parameters, and their ID accuracy is 76.18%, 81.14%, and 81.59%, respectively. Interestingly, despite their better ID classification performance, we find that vision transformers do *not* show noticeable improvements over ResNets for OOD detection, which aligns with the observation in [10]. Meanwhile, different post-processor may favor different architecture. For instance, the top-2 post-processor on ResNet-50, *i.e.*, ASH [13] and ReAct [9], both have significant performance degradation when operating on transformer. RMDS [29], in contrast, suits transformers much better than ResNets.

Training methods excel at small datasets. On CIFAR-10/100, we find that training-time regularizations (the second group in Table 1) can provide better OOD detection capability than post-hoc methods. In particular, RotPred [37] and LogitNorm [43] stand out as two powerful training methods (without using outlier data). On CIFAR-10, they both lead to $\sim$2% and $\sim$3% increase in near- and far-OOD AUROC, respectively, compared to the best-performing post-processors. Meanwhile, however, training methods in general do not outperform post-hoc ones on ImageNet-200/1K. This might be due to that more sophisticated training dynamics require larger models or longer training, and the exact reason needs further investigation in future works.

Post-hoc methods are more effective for large-scale settings. This is particularly true on ImageNet-1K, where applying post-processors to a model pre-trained with the standard cross-entropy loss are the top-performing solutions for both near- and far-OOD detection, according to Table 1.

Outlier data helps in certain cases. Compared with methods that do not have such consideration, incorporating OOD training data (the last group in Table 1) is helpful mainly when the test OOD samples are similar to the training ones. For instance, OE [46] yields the highest near-OOD AUROC on CIFAR-100 because the used OOD training set (TIN-597; see Appendix D for details) is similar to one of the near-OOD test sets (TIN). In contrast, it does not seem to be beneficial for detecting far-OOD samples (which are quite different from TIN images) and actually underperforms several other methods which do not use the outlier data.

Intermediate features are only useful for detecting differences in low-level statistics. Take MDSEns [28] as an example, which is a representative post-hoc method that leverages feature maps from the classifier's intermediate layers to compute OOD score. From our full result table (presented here), we observe that MDSEns achieves a near-perfect result of 99% AUROC when detecting MNIST (black and white handwritten digits) from CIFAR-10 (color images with objects), yet performs much

Figure 7: OOD detection rates of post-hoc methods with different architectures on ImageNet-1K. Some methods are sensitive to model architecture while some are not. Transformers do not seem to have clear advantage over ResNets.

Table 4: ImageNet-1K full-spectrum detection results of data augmentation methods. The numbers in each cell represent the near-OOD / far-OOD AUROC. Again, data augmentations are helpful especially when combined with approriate post-processors and when performing near-OOD detection.

	MSP [1]	GradNorm [67]	SHE [14]	ID Acc.
CrossEntropy	60.79 / 72.32	62.70 / **83.49**	61.21 / 83.04	54.35
StyleAugment [20]	62.09 / 74.37	65.27 / 81.62	66.64 / 82.64	55.44
RandAugment [21]	61.36 / 72.07	63.27 / 76.08	64.41 / 76.68	55.57
AugMix [22]	63.14 / 74.62	**67.10** / 81.29	**69.66** / **83.06**	57.46
DeepAugment [19]	63.51 / **75.40**	65.66 / 76.27	68.27 / 78.85	**57.82**
PixMix [23]	**62.51** / 73.47	61.07 / 70.00	65.02 / 77.03	57.27
RegMixup [24]	61.32 / 72.87	61.86 / 79.98	64.71 / 81.23	55.55

worse (∼60% AUROC) in all other cases where there is not much difference in the low-level statistics between ID and OOD. In contrast, the high-level semantic information from the penultimate layer or the last linear layer (used by most post-processors like MSP [1], ReAct [9], KNN [12], and ASH [13]) is more robust for detecting multiple types of OOD samples.

ID accuracy can be affected a little. Lastly, as shown in Table 1, most training methods incur slight drop (within 1%) in ID classification accuracy compared to the standard cross-entropy training. For some methods the drop could be large, and it is important for future works to monitor ID accuracy to maintain utility while improving OOD detection capability.

Data augmentations continue to help in full-spectrum settings. Similar to our observations in standard OOD detection, in Table 4 we see that data augmentations also boost full-spectrum detection rates especially when combined with powerful post-hoc methods. For example, compared to the cross-entropy baseline, while AugMix [22] increases near-OOD AUROC "only" by 2.35% with the MSP detector [1], it leads to a much significant improvement of 8.45% when working with SHE [14]. AugMix + SHE is the current best approach in terms of full-spectrum near-OOD AUROC on ImageNet-1K. In the meantime, we do notice that data augmentations do not clearly benefits full-spectrum far-OOD AUROC, and the reasons require future study. That said, our results demonstrate that in general "data augmentation + post-processor" is promising for both standard and full-spectrum OOD detection.

H Additional Discussions

H.1 Related Work

To the best of our knowledge, OpenOOD (especially the v1.5 release) is the only work that comprehensively benchmarks various OOD detection methods on multiple ID-OOD pairs. That said, there are still a few works that relate to OpenOOD in certain aspects.

Tajwar et al. [76] made the observation that "OOD detection methods are inconsistent across datasets" from experiments on 3 small datasets (CIFAR-10, CIFAR-100, and SVHN) with 3 specific post-hoc methods (MSP, ODIN, and MDS). While we draw a similar conclusion of "no single winner" in Section 3, our observation comes from the experimentation with 4 datasets and nearly 40 methods from different categories.

A recent work by Galil et al. [77] proposed a method for constructing OOD detection benchmark and evaluated the performance of 5 post-hoc methods with ImageNet-1K pre-trained models. Specifically, for a specific ImageNet-1K model with a specific post-processor, they consider ImageNet-21K images as OOD and categorizes OOD images into a sequence of difficulty groups based on the OOD score from the post-processor. Correspondingly, their evaluation looks at the OOD detection AUROC across all groups, intending to provide a spectrum of AUROC v.s. difficulty. We see two shortcomings of such practice for constructing a general benchmark. First, their benchmarking process is extremely time-consuming since it needs to iterate through nearly all of the samples in ImageNet-21K, which could be prohibitive even for the most lightweight method considering the compute required by common ImageNet models. Second, the resulting benchmark is diagnostic to both the classifier and the post-processor. For example, the first difficulty group of the benchmark for MSP and that for ASH would *not* contain the same OOD samples, making the comparison ambiguous and much less straightforward. In comparison, our carefully designed benchmarks are *standardized*, *i.e.*, agnostic to classifiers and post-processors. Plus, we consider a wide range of methods beyond a few specific post-hoc approaches.

One work that most closely relates to ours is PyTorch-OOD [78], which is a python library for evaluating OOD detection performance. There are several distinctions that separate OpenOOD from PyTorch-OOD. 1) **Number of supported methods.** PyTorch-OOD implements 19 methods as of May 2023 with the most recent one dating back to 2022, while OpenOOD supports 40 approaches including the most advanced ones published in 2023. 2) **Reliability of evaluation results.** PyTorch-OOD still includes as OOD images the LSUN-R and TIN-R [27] which contain obvious resizing artifacts [39]. It also considers ImageNet-O which is known to cause biased evaluation since it is constructed by adversarially targeting a ResNet-50 model with the MSP detector [77]. Their benchmarking results thus can be problematic and unreliable. 3) **Alignment between the goal reflected by the evaluation and human perception.** PyTorch-OOD's evaluation setup favors detectors that flag covariate-shifted ID samples (*e.g.*, those from ImageNet-R or ImageNet-C) as OOD. We argue that this does not align with human perception and is not an ideal behavior.

Another concurrent work by Bitterwolf et al. [59] put up a noise-free OOD dataset (NINCO) for ImageNet-1K in response to the observed noise that exists in popular OOD datasets. They then evaluate 8 post-hoc methods on NINCO and specifically study the effect of large-scale pre-training. OpenOOD is inherently complementary to the work of [59], as we intend to build a comprehensive benchmark for OOD detection by implementing and evaluating various types of methods (not restricting to post-hoc ones) on multiple datasets including ImageNet-1K. Meanwhile, our investigation on full-spectrum detection and findings regarding data augmentation techniques are unique.

H.2 Real-World Implications

OOD detection is important in many real-world cases such as remote sensing applications [79] and fine-grained novel category discovery [49]. It also has strong implications for safety-critical applications, as OOD detection can be used to detect unexpected anomalies, unknown unknowns, and Black Swans [80]. We believe that OpenOOD has laid a solid foundation and provided a good starting point for tackling OOD detection in those specific scenarios.

H.3 Limitation

In this work we focus on the context where there assumes to be a discriminative classifier for the ID classification in the first place. As a result, all the OOD detection methods considered in our work are discriminative. OOD detection approaches that are based on generative modeling (*e.g.*, [81, 82, 83, 84]) are not currently included in OpenOOD. To our knowledge, generative methods have not yet demonstrated scalability on ImageNet-level data and in general are less competitive than discriminative methods. Additional efforts will be required to integrate generative methods into

OpenOOD in the future, as they often rely on dedicated/specialized model architecture and training procedure [81, 82, 84].

H.4 Societal Impact

OOD detection is an important topic for machine learning safety as it studies how deep neural networks can handle unknown inputs desirably. As an open-sourced, unified, and comprehensive benchmark for OOD detection, OpenOOD is expected to benefit the whole community and facilitate relevant research, which we believe has positive societal impacts. On the other hand, while OpenOOD itself as a benchmark platform does not incur concerns, certain methods that are included in OpenOOD may pose privacy or security risks. Specifically, methods such as KNN and SHE rely on the extracted representation of training samples to compute the OOD score, making it vulnerable to privacy attacks [85]. Meanwhile, Inkawhich et al. [86] showed that OOD detectors enlarge the attack surface of deep learning systems, and existing methods can easily be compromised by adversarial attacks. We encourage future works to take this into consideration.

H.5 Future Work

First, we will keep maintaining OpenOOD's codebase and leaderboard. The codebase is hosted on Github, and the leaderboard is hosted using Github pages, which both are free services. We anticipate the benchmark to be community-driven: Reporting new results and submitting new entries to the leaderboard would be easy with our unified evaluator.

In addition to the maintenance, in the future v2 release we plan to extend the scope of OpenOOD beyond image classification and include more application scenarios such as object detection, semantic segmentation, and natural language processing tasks. Specifically, it would be interesting to see whether current OOD detectors, which are designed for image classifiers, can generalize to different problems and modalities.