Exploiting Discrepancy in Feature Statistic for Out-of-Distribution Detection

Xiaoyuan Guan[1,2], Jiankang Chen[1,2], Shenshen Bu[1], Yuren Zhou[1*], Wei-Shi Zheng[1,2], Ruixuan Wang[1,2,3*]

[1]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2]Key Laboratory of Machine Intelligence and Advanced Computing, MOE, Guangzhou, China
[3]Peng Cheng Laboratory, Shenzhen, China
{guanxy36,chenjk36,bushsh}@mail2.sysu.edu.cn, zhouyuren@mail.sysu.edu.cn,wszheng@ieee.org, wangruix5@mail.sysu.edu.cn

Abstract

Recent studies on out-of-distribution (OOD) detection focus on designing models or scoring functions that can effectively distinguish between unseen OOD data and in-distribution (ID) data. In this paper, we propose a simple yet novel approach to OOD detection by leveraging the phenomenon that the average of feature vector elements from convolutional neural network (CNN) is typically larger for ID data than for OOD data. Specifically, the average of feature vector elements is used as part of the scoring function to further separate OOD data from ID data. We also provide mathematical analysis to explain this phenomenon. Experimental evaluations demonstrate that, when combined with a strong baseline, our method can achieve state-of-the-art performance on several OOD detection benchmarks. Furthermore, our method can be easily integrated into various CNN architectures and requires less computation. Source code address: https://github.com/SYSU-MIA-GROUP/statistical_discrepancy_ood.

Introduction

Out-of-distribution (OOD) detection is a crucial task in the field of machine learning, particularly in scenarios where models are deployed in the real world. In such cases, it is necessary to ensure that the model is able to recognize inputs that are outside of the distribution on which it was trained, as failure to do so could potentially lead to significant consequences. For instance, in autonomous driving (Geiger, Lenz, and Urtasun 2012), a model that is unable to detect OOD inputs may fail to recognize a new type of obstacle on the road, leading to fatal accidents.

Various techniques have been developed to address OOD detection, including uncertainty estimation methods (Lakshminarayanan, Pritzel, and Blundell 2017; Malinin and Gales 2018, 2019; Nandy, Hsu, and Lee 2020), confidence-based methods (DeVries and Taylor 2018; Hein, Andriushchenko, and Bitterwolf 2019), and density estimation methods (Jiang, Sun, and Yu 2021; Ren et al. 2019; Serrà et al. 2020; Zisselman and Tamar 2020). For example, one of the confidence-based methods called the maximum softmax probability (MSP) (Hendrycks and Gimpel 2017) can be used as a score for detecting OOD inputs. However, OOD data often result in high prediction confidence. Density estimation methods based on generative models can also suffer from the same vulnerability to OOD inputs. Nalisnick (Nalisnick et al. 2019) found that using generative models to estimate the data distribution can result in OOD inputs obtaining higher density estimates than in-distribution data.

The vulnerabilities of existing out-of-distribution detection methods have highlighted the importance for researchers to capture more potential differences between ID and OOD data from various aspects. To this end, ReAct (Sun, Guo, and Li 2021) proposed performing activation pruning on feature vectors that are output from the penultimate layer of CNN networks, which can reduce more activations for the OOD feature and increase the gap between ID and OOD scoring value. LINe (Ahn, Park, and Kim 2023) was proposed to clip feature activation output from the penultimate layer and consider the difference in the number of activated units. Similar methods that perform rectification on feature vectors include BATS (Zhu et al. 2022) and RankFeat (Song, Sebe, and Wang 2022). ViM (Wang et al. 2022) was proposed to combine multiple sources of information to improve OOD detection performance. In particular, by discarding some information from the feature vector and using the discarded information to generate an extra logit, ViM can capture differences between ID and OOD data from both the feature space and the logit space. The aforementioned studies either enlarge the differences between ID and OOD data by post-hoc operations or enhance the OOD detection performance by fusing differences from multiple sources.

This paper further exploits potential difference in features between ID and OOD data and introduces a novel approach to OOD detection based on the discrepancy between the statistical information of feature vectors. Specifically, we observe that the average of feature vector elements from the penultimate layer of a CNN classifier tends to be larger for ID data than for OOD data. By combining the feature average information with certain existing strong OOD methods, we propose a method that outperforms state-of-the-art approaches on multiple OOD detection benchmarks. The main contributions are summarized below.

- We propose a simple and effective OOD detection method by exploiting the statistical discrepancy between in-distribution and out-of-distribution data. Our method

*Corresponding author

needs no extra training or auxiliary OOD data, and the implementation is rather simple.

- Theoretical analysis was provided to help understand the observed discrepancy between ID and OOD data.
- Extensive evaluations on the widely used benchmarks (ImageNet and CIFAR) demonstrate the superior performance of the proposed method over strong post-hoc OOD detection baselines.

Preliminaries

We consider the setting of supervised classification task, denote by $\mathcal{X} = \mathbb{R}^M$ the input space with dimension M and by $\mathcal{Y} = \{1, 2, \cdots, C\}$ the label space. The training set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ is drawn i.i.d. from the joint distribution $\mathcal{P}_{\text{in}} \times \mathcal{P}_{\mathcal{Y}}$, where $\mathcal{P}_{\text{in}}$ and $\mathcal{P}_{\mathcal{Y}}$ represent the marginal distribution of the ID training data and the label, respectively. Let $f \circ h : \mathcal{X} \to \mathcal{Y}$ denote a neural network trained on the training set $\mathcal{D}$ for the classification task, where h denotes the feature extractor which encodes input $\mathbf{x} \in \mathcal{X}$ as the feature vector $h(\mathbf{x}) \in \mathbb{R}^d$, where d represents the dimension of the feature space, and f denotes the classifier head.

Out-of-Distribution Detection

Out-of-distribution (OOD) detection is the task of identifying samples that do not come from the distribution of ID data, which can be viewed as a binary classification problem. At test time, the goal of OOD detection is to decide whether a test sample $\mathbf{x} \in \mathcal{X}$ is from the distribution $\mathcal{P}_{\text{in}}$ or from out-of-distribution $\mathcal{P}_{\text{out}}$. The decision can be made via a decision function

$$G(\mathbf{x}) = \begin{cases} 1 & \text{if} \quad \mathbf{x} \sim \mathcal{P}_{\text{in}} \\ 0 & \text{if} \quad \mathbf{x} \sim \mathcal{P}_{\text{out}} \end{cases} . \tag{1}$$

The difficulty of OOD detection depends on the separation between $\mathcal{P}_{\text{in}}$ and $\mathcal{P}_{\text{out}}$.

Energy-Based OOD Detection

For a sample point $(\mathbf{x}, y)$, let $f_k \circ h(\mathbf{x})$ denote the k^{th} output of the last layer, and let $E(\mathbf{x})$ denote the energy of $\mathbf{x}$ which can be defined as (Grathwohl et al. 2020; Liu et al. 2020)

$$E(\mathbf{x}) = -\log \sum_{k=1}^{C} \exp(f_k \circ h(\mathbf{x})) . \tag{2}$$

Liu et al. (Liu et al. 2020) proposed to use the opposite of the energy $E(\mathbf{x})$ as a scoring function to detect OOD samples, with a higher energy score indicating higher ID-ness of the test example.

Method

In this section, we start by revealing an observation that the element-mean of feature vector obtained from the penultimate layer of CNN model is larger for ID than for OOD. On the basis of this observation, we propose to combine element-mean of feature vector with existing OOD score (energy score) to further improve the separability between ID and OOD data. Furthermore, we provide mathematical analysis to explain why the discrepancy exists between ID and OOD data.

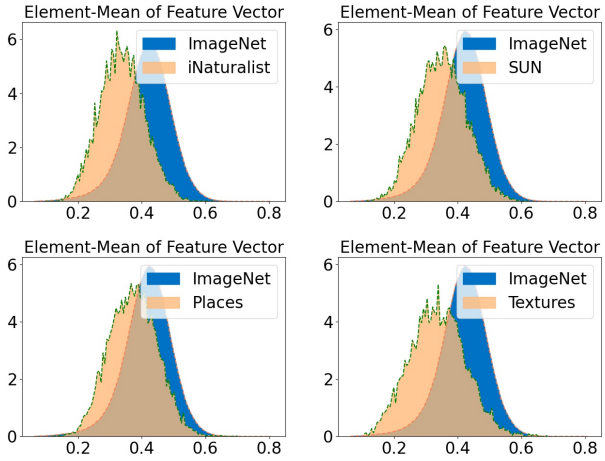

Figure 1: Distributions of the element-mean of feature vector output from the penultimate layer for ID dataset (blue) and each of the four OOD datasets (green). The model is ResNet50 (He et al. 2016) pre-trained on the ID dataset ImageNet1K (Deng et al. 2009).

Element-Mean Discrepancy

For classifiers trained on the ID dataset, there exist many potential differences between ID and OOD data around the output layer of the classifier, which has been partly explored in recent studies (Song, Sebe, and Wang 2022; Sun and Li 2022; Wang et al. 2022). Here, we observed another statistical difference in the output of the penultimate layer of CNN classifier between ID and OOD data, i.e., the average of feature output elements ('element-mean' in short) is often different between ID and OOD data (Figure 1). Denote by $h(\mathbf{x}) = (z_1, z_2, \cdots, z_d)$ the feature vector from the penultimate layer. The element-mean of feature vector is defined as

$$Z(\mathbf{x}) = \frac{1}{d} \sum_{i=1}^{d} z_i . \tag{3}$$

As demonstrated in Figure 1, the element-mean of ID data's feature vector is statistically larger than that of each OOD dataset (iNaturalist, SUN, Places, and Textures).

We leverage this element-mean discrepancy to further improve OOD detection performance by combining element-mean of feature vector with the existing energy score, resulting in the new scoring function

$$D(\mathbf{x}) = (1 + Z(\mathbf{x})) \cdot \log \sum_{k=1}^{C} \exp(f_k \circ h(\mathbf{x})) . \tag{4}$$

ID data are expected to result in a higher score $D(\mathbf{x})$, while OOD data would result in a lower score. The inclusion of the element-mean discrepancy would enlarge the original difference in the energy score between ID and OOD data. The multiplication rather than the addition operator is adopted to fuse the element-mean discrepancy and energy score such that additional coefficient can be avoided to adjust the difference in scale between the two terms.

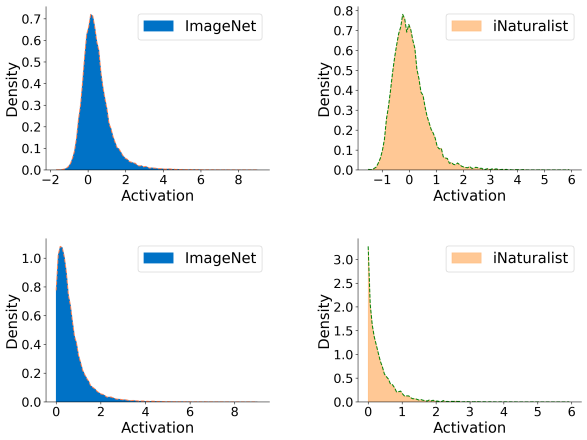

Figure 2: Empirical distributions (at randomly selected dimension) of v_i's (first row) and z_i's (second row) on the ID dataset ImageNet1K (first column) and the OOD dataset iNaturalist (second column). Model is ResNet50 pre-trained on ImageNet1K.

Theoretical Analysis

Here mathematical insight is provided to better understand the discrepancy in element-mean of feature vector between ID and OOD data. Without loss of generality, suppose the ReLU activation function $g(\cdot)$ is part of the penultimate layer of the classifier, and z_i in Equation 3 is from the output of the ReLU activation, i.e., $z_i = g(v_i)$, where v_i is the corresponding input to ReLU for each z_i. Figure 2 plots the empirical distribution of v_i's on the ID dataset ImageNet1K and the OOD dataset iNaturalist (first row) at a randomly selected dimension. As we can see, on both ID and OOD datasets, distributions of v_i's exhibit certain degrees of right skewness, i.e., the right tail has a higher density than the left tail. Such right-skewed distribution can be modeled as epsilon-skew-norm (ESN) distribution (Mudholkar and Hutson 2000) which has the following probability density function

$$q(v_i) = \begin{cases} \frac{1}{\sigma_i}\phi(\frac{v_i-\mu_i}{\sigma_i(1+\epsilon_i)}) & \text{if } v_i < \mu_i \\ \frac{1}{\sigma_i}\phi(\frac{v_i-\mu_i}{\sigma_i(1-\epsilon_i)}) & \text{if } v_i \geq \mu_i \end{cases}, \tag{5}$$

where $\phi(\cdot)$ represents the probability distribution function of standard normal distribution, μ_i and σ_i denote the mode parameter and standard deviation parameter respectively, and the hyper-parameter $\epsilon_i \in (-1, 1)$ controls the skewness. $q(v_i)$ will become the well-known half-normal distribution when $\epsilon_i \to \pm 1$, and will be reduced to the normal distribution when $\epsilon_i = 0$. In particular, positively skewed ESN distribution has the property $\epsilon_i < 0$.

Since z_i is the ReLU activation output with the corresponding v_i as input, the distribution of each z_i can be modeled as a *rectified* ESN distribution (Mudholkar and Hutson 2000). The second row of Figure 2 plots the corresponding empirical distribution of each z_i, with each distribution equivalent to the ReLU-rectified version of the corresponding distribution of v_i (first row).

For a rectified ESN distribution of z_i which is derived from ESN distribution of v_i with mode μ_i, standard deviation σ_i, and skewness ϵ_i, the expectation of z_i is (Mudholkar and Hutson 2000; Sun, Guo, and Li 2021)

$$\begin{aligned} \mathbb{E}[z_i] = \mu_i - (1+\epsilon_i)\Phi(\frac{-\mu_i}{(1+\epsilon_i)\sigma_i}) \cdot \mu_i \\ + (1+\epsilon_i)^2\phi(\frac{-\mu_i}{(1+\epsilon_i)\sigma_i}) \cdot \sigma_i - \frac{4\epsilon_i}{\sqrt{2\pi}} \cdot \sigma_i \,, \end{aligned} \tag{6}$$

where $\Phi(\cdot)$ denotes the cumulative distribution function of the standard normal distribution. Considering that the ID training set contains enough samples (e.g., around 1.28 million images in ImageNet1K), the average of activation z_i over all training samples is an unbiased estimation of $\mathbb{E}[z_i]$ for ID dataset.

However, for each OOD dataset, the number of samples is not large enough, and therefore the hyperparameters in Equation 6 need to be estimated appropriately. For ESN distribution, ϵ_i with value closer to -1 indicates stronger right-skewness of the distribution. Currently, there is no analytical method to calculate the value of ϵ_i. Here ϵ_i was set to -0.4 by referring to the reference graph in the literature (Mudholkar and Hutson 2000). The ESN distribution with $\epsilon_i = -0.4$ exhibits stronger right-skewness than the distributions in Figure 2 and therefore would give us an upper bound estimate of $\mathbb{E}[z_i]$. Given the estimated skewness parameter ϵ_i, each standard deviation parameter σ_i can be estimated by (Mudholkar and Hutson 2000)

$$\sigma_i = \sqrt{\frac{\pi}{(3\pi-8)\epsilon_i^2 + \pi} \cdot \mathrm{Var}[v_i]}\,, \tag{7}$$

where the empirical variance $\mathrm{Var}[v_i]$ of the ReLU input v_i can be directly computed on all the real OOD data of a specific OOD set. Additionally, considering that z_i is the output of ReLU activation layer with v_i as input, the sample mode of v_i can be directly used as the mode μ_i of z_i. Consequently, the expectation of each z_i on any specific OOD set can be roughly estimated with Equation 6 based on the approximate estimate of the hyperparameters μ_i, ϵ_i and σ_i. Table 2 summarizes the summation of expected z_i over all feature elements, which shows that the summation is clearly larger for ID than for OOD data, supporting the obersation of element-mean discrepancy.

In summary, our analysis suggests that the discrepancy in element-mean of feature vector between ID and OOD datasets can be attributed to the larger sum of activation expectations in ID data. It is supported in Figure 2 that a considerable portion of ID activations exceed the maximum value of OOD activations.

Experiments

Experimental Setting

Dataset. Our method was extensively evaluated on three OOD detection benchmarks, including the large-scale image benchmark *ImageNet1K* and the small-scale image benchmarks *CIFAR*. For the ImageNet1K benchmark, we use the ImageNet1K (Deng et al. 2009) as ID set and use four

ID Dataset Model	Method	OOD Datasets									
		iNaturalist		SUN		Places		Textures		Average	
		F↓	A↑	F↓	A↑	F↓	A↑	F↓	A	F↓	A↑
ImageNet1K ResNet50	MSP	54.99	87.74	70.83	80.86	73.99	79.76	68.00	79.61	66.95	81.99
	ODIN	47.66	89.66	60.15	84.59	67.89	81.78	50.23	85.62	56.48	85.41
	Mahalanobis	97.00	52.65	98.50	42.41	98.40	41.79	55.80	85.01	87.43	55.47
	Energy	55.72	89.95	59.26	85.89	64.92	82.86	53.72	85.99	58.41	86.17
	ViM	68.86	87.13	79.62	81.67	83.81	77.80	**14.95**	**96.74**	61.81	85.83
	BATS	42.26	92.75	44.70	90.22	55.85	86.48	33.24	93.33	44.01	90.69
	DICE	26.66	94.49	36.08	90.98	47.63	87.73	32.46	90.46	35.71	90.92
	ReAct	20.38	96.22	24.20	94.20	33.85	91.58	47.30	89.80	31.43	92.95
	FeatureNorm	22.01	95.76	42.93	90.21	56.80	84.99	20.07	95.39	35.45	91.59
	DICE+ReAct	20.08	96.11	26.50	93.83	38.34	90.61	29.36	92.65	28.57	93.30
	LINe	**12.26**	97.56	19.48	95.26	28.52	92.85	22.54	94.44	20.70	95.03
	Ours	12.40	**97.74**	**17.68**	**95.86**	**26.88**	**93.47**	19.72	95.87	**19.17**	**95.73**
ImageNet1K MobileNet	MSP	64.29	85.32	77.02	77.10	79.23	76.27	73.51	77.30	73.51	79.00
	ODIN	58.54	87.51	57.00	85.83	59.87	84.77	52.07	85.04	56.87	85.79
	Mahalanobis	62.11	81.00	47.82	83.66	52.09	83.63	92.38	33.06	63.60	71.01
	Energy	59.50	88.91	62.65	84.50	69.37	81.19	58.05	85.03	62.39	84.91
	ViM	91.83	77.47	94.34	70.24	93.97	68.26	37.62	92.65	79.44	77.15
	BATS	49.57	91.50	57.81	85.96	64.48	82.83	39.77	91.17	52.91	87.87
	DICE	43.28	90.79	38.86	90.41	53.48	85.67	33.14	91.26	42.19	89.53
	ReAct	43.07	92.72	52.47	87.26	59.91	84.07	40.20	90.96	48.91	88.75
	FeatureNorm	33.10	92.71	42.41	88.60	58.46	81.79	**8.60**	**98.26**	35.64	90.34
	DICE+ReAct	41.75	89.84	39.07	90.39	54.41	84.03	19.98	95.86	38.80	90.03
	LINe	24.95	95.53	33.19	92.94	47.95	88.98	12.30	97.05	29.60	93.62
	Ours	**22.39**	**95.88**	**26.08**	**94.56**	**40.88**	**90.66**	12.55	97.08	**25.48**	**94.55**

Table 1: Comparison between different methods in OOD detection on the ImageNet1K Benchmark with two different model backbones. ↑ indicates that larger values are better and ↓ indicates that smaller values are better. All values are percentages.

Dataset	ImageNet1K	iNaturalist	SUN	Places	Textures
ϵ	-	-0.4	-0.4	-0.4	-0.4
$\mathbb{E}[\sum z_i]$	862.31	643.58	722.22	758.63	750.93

Table 2: The sum of expectation activation over all feature elements respectively on the ID training set ImageNet1K and the four OOD datasets.

datasets iNaturalist (Van Horn et al. 2018), SUN (Xiao et al. 2010), Places (Zhou et al. 2017), and Textures (Cimpoi et al. 2014) as the OOD sets. The CIFAR benchmarks respectively use CIFAR10 and CIFAR100 (Krizhevsky and Hinton 2009) as ID sets, and both use six datasets SVHN (Netzer et al. 2011), LSUN-Crop (Yu et al. 2015), LSUN-Resize (Yu et al. 2015), iSUN (Xu et al. 2015), Textures (Cimpoi et al. 2014), and Places365 (Zhou et al. 2017) as the OOD sets. There is no semantic overlap between ID sets and OOD sets. Please refer to Supplementary Section A for more details of the datasets.

Baselines. Multiple types of competitive OOD detection methods were adoped as baselines for comprehensive evaluation, including the Maximum Softmax Probability (MSP) (Hendrycks and Gimpel 2017), ODIN (Liang, Li, and Srikant 2017), Energy (Liu et al. 2020), Mahalanobis (Lee et al. 2018), ViM (Wang et al. 2022), DICE (Sun and Li 2022), ReAct (Sun, Guo, and Li 2021), BATS (Zhu et al. 2022), FeatureNorm (Yu et al. 2023) and LINe (Ahn, Park, and Kim 2023). All the above baselines are post-hoc and can obtain the OOD score based on a pre-trained CNN classifier. In addition, LogitNorm (Wei et al. 2022) and CIDER (Ming et al. 2023), which need model retraining, were used on CIFAR benchmarks considering that they achieve the state-of-the-art performance on the CIFAR10 benchmark.

Metrics. For all experiments, *FPR95* (abbr. **F**, i.e., the false positive rate when the true positive rate of ID samples is 95%) and *AUROC* (abbr. **A**, the area under the receiver operating characteristic curve) in OOD detection were used as metrics, with lower FPR95 values and higher AUROC values indicating better OOD detection performance.

Consistent with previous studies (Sun, Guo, and Li 2021; Zhu et al. 2022), we evaluated our method and each baseline with ResNet50 and the lightweight MobileNet-v2 (Sandler et al. 2018) as the classifier backbones on the ImageNet1K benchmark. For the CIFAR benchmark, we trained ResNet18 (He et al. 2016) and WideResNet28-10 (Zhu et al. 2022) as the classifier backbones from scratch on the associated training ID dataset. To train each classifier, we used the stochastic gradient descent optimizer with momentum (0.9) and weight decay (0.0005) for up to 200 epochs on the CIFAR datasets, with a batch size of 128. The initial learning rate was set to 0.1, and it was decayed by a factor of 10 at the

ID Dataset Model	Methods	OOD Datasets													
		SVHN		LSUN-R		LSUN-C		iSUN		Textures		Places365		Average	
		F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑
CIFAR10 RN18	MSP	61.22	86.99	41.62	93.84	34.30	95.40	43.14	93.21	53.40	90.19	54.51	88.74	48.03	91.40
	Mahalanobis	67.25	89.51	48.37	92.38	91.65	74.55	44.24	92.68	45.92	91.96	66.11	85.79	60.59	87.81
	ODIN	53.56	77.48	17.31	94.63	13.64	96.09	**19.87**	93.55	46.65	80.85	49.72	79.92	33.46	87.09
	Energy	41.25	87.69	24.19	95.01	11.37	97.63	26.40	94.16	42.52	89.10	40.04	88.71	30.96	92.05
	ViM	53.75	88.67	34.17	94.34	82.31	87.18	31.41	94.25	**36.15**	**92.83**	49.64	88.86	47.90	91.02
	DICE	**36.42**	**91.46**	31.57	93.77	**7.10**	**98.67**	36.94	92.05	47.02	88.41	46.74	86.05	34.30	91.73
	BATS	41.42	87.84	24.17	95.02	11.35	97.63	26.36	94.16	42.13	89.29	40.04	88.71	**30.91**	92.11
	ReAct	43.19	87.56	24.82	95.12	12.23	97.53	26.90	94.31	41.95	90.02	40.78	89.00	31.65	92.26
	DICE+ReAct	36.90	91.31	31.59	93.71	7.29	98.64	37.15	92.10	46.76	88.61	46.76	86.12	34.41	91.75
	LINe	45.38	87.96	39.25	92.61	9.75	98.19	41.52	91.74	58.37	84.14	53.02	85.70	41.22	90.06
	Ours	50.22	87.80	**24.83**	**95.96**	16.11	97.21	26.60	**95.30**	39.56	92.49	**41.91**	**90.54**	33.20	**93.22**
CIFAR10 WRN	MSP	46.92	87.25	33.04	94.62	17.32	97.08	35.15	94.29	44.54	90.61	43.50	90.28	36.74	92.35
	Mahalanobis	31.11	92.68	87.47	62.37	91.28	59.06	86.72	62.30	47.15	81.96	83.56	64.65	71.21	70.50
	ODIN	61.09	73.64	**15.22**	**96.10**	7.07	98.54	**16.75**	95.33	43.63	82.73	46.49	82.67	31.71	88.17
	Energy	41.25	87.69	24.19	95.01	11.37	97.63	26.40	94.16	42.52	89.10	40.04	88.71	30.96	92.05
	ViM	**10.80**	**97.95**	20.11	95.98	13.85	97.38	19.89	**95.97**	**17.87**	**95.85**	42.78	88.33	**20.88**	**95.24**
	DICE	46.51	84.57	25.66	93.15	**0.25**	**99.89**	29.95	92.10	51.70	81.83	45.70	84.49	33.29	89.34
	BATS	38.15	90.97	22.57	96.13	10.60	98.07	24.52	95.83	34.36	92.93	32.71	93.23	27.15	94.53
	ReAct	46.05	89.06	26.87	95.40	30.89	93.91	28.99	94.88	41.01	92.01	**28.98**	**93.80**	33.80	93.23
	DICE+ReAct	45.84	85.39	26.47	93.60	0.39	99.87	30.41	92.62	48.65	85.41	47.49	84.79	33.21	90.28
	LINe	40.48	85.92	18.80	96.27	3.37	99.26	19.94	95.93	43.88	87.32	37.87	89.30	27.39	92.33
	Ours	39.42	87.82	21.87	96.13	8.37	98.49	23.76	95.72	34.72	92.13	32.90	92.58	26.84	93.81

Table 3: Comparison between different methods in OOD detection on the CIFAR10 Benchmark with two different model backbones. ↑ indicates that larger values are better and ↓ indicates that smaller values are better. All values are percentages

Dataset	Metric	iNaturalist	SUN	Places	Textures	Avg
ImageNet	F↓	37.08	45.06	56.82	39.75	44.68
	A↑	90.84	87.15	82.12	88.22	87.08

Table 4: OOD detection with only element-mean discrepancy. Model backbone is ResNet50.

100th and 150th epoch on CIFAR. To prepare the images for training, we padded each training image from 32×32 pixels to 36×36 pixels and randomly cropped it to 32×32 pixels on the CIFAR10 and CIFAR100 datasets. We also applied random horizontal flipping together with random cropping on each training image. During testing, we used only center cropping with resizing on the test dataset. All experiments were run on NVIDIA GeForce RTX 2080ti GPUs.

Quantitative Evaluations

Table 1 summarizes the out-of-distribution detection performance of each method on each OOD set (together with the ID test set) and the average performance over the four OOD sets for the ImageNet1K benchmark. Our method, which builds on the strong baseline ReAct that uses the energy score for OOD detection, achieves state-of-the-art performance on three of the four OOD sets and outperforms the best baseline LINe (A 95.73% vs. 95.03%, and FPR95 19.17% vs. 20.70%) on average with the ResNet50 backbone. Similar results are obtained with the MobileNet-v2 backbone (Table 1, lower half), with our method achieving the best AUROC performance on three OOD sets and state-of-the-art average performance over the four OOD sets. All the results support the efficacy of the element-mean in improving the OOD detection performance. Table 4 shows that the AUROC of OOD detection on each OOD set is significantly greater than 50% when using only element-mean, further demonstrating the feasibility of improving OOD detection performance by our method. We also perform statistical tests to validate the significance of our method's superiority, please refer to Supplementary Section B for details.

The discrepancy in element of feature vector can also help achieve competitive or state-of-the-art performance on average across the six OOD sets for both CNN backbones on the CIFAR benchmarks, as shown in Table 3 and Table 5. One exception is on the CIFAR10 benchmark with the WideResNet28 backbone (Table 3, lower half) where the element-mean discrepancy did not help achieve a new state-of-the-art performance, although it did outperform the corresponding strong baseline ReAct. One possible reason is that for the CIFAR10 classification task, due to the smaller number of ID classes and higher feature dimensionality from WideResNet28 than from ResNet18, sufficient separability in feature representations of different ID classes can be relatively easily achieved by simply increasing angular distance between different classes of feature representations during classifier training. In this case, it is not necessary to increase the norm of feature representations for ID classes during classifier training (Vaze et al. 2022), which in turn leads to smaller or even indiscernible gap between the feature norms

ID Dataset Model	Methods	OOD Datasets													
		SVHN		LSUN-R		LSUN-C		iSUN		Textures		Places365		Average	
		F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑
CIFAR100 RN18	MSP	69.74	84.73	66.89	85.65	77.08	81.83	69.40	84.77	80.08	77.65	78.38	78.81	73.60	82.24
	Mahalanobis	92.62	66.80	89.00	68.46	98.83	49.58	88.45	68.44	72.68	74.57	92.87	63.26	89.07	65.18
	ODIN	79.74	81.40	**37.63**	**93.21**	72.66	85.93	**39.59**	**92.58**	73.07	80.42	80.39	77.22	63.85	85.13
	Energy	68.90	87.66	59.71	88.58	73.21	84.46	64.03	87.50	79.61	78.22	77.74	79.64	70.53	84.34
	ViM	73.70	84.45	61.30	88.05	92.76	69.87	61.92	87.34	**57.93**	86.31	81.01	76.54	71.43	82.09
	DICE	53.60	90.22	79.84	81.17	40.03	92.52	79.79	80.96	78.65	77.46	82.31	76.76	69.04	83.18
	BATS	62.05	89.31	50.38	91.21	73.70	84.55	55.97	90.30	72.93	84.50	72.61	82.03	64.61	86.98
	ReAct	58.24	90.02	50.82	90.98	70.70	85.75	55.91	90.18	70.85	85.39	**71.85**	**82.25**	63.06	87.43
	DICE+ReAct	48.20	91.19	84.18	78.79	**32.05**	**93.71**	82.23	79.65	66.74	83.96	80.28	77.96	65.61	84.21
	LINe	52.02	91.01	65.66	86.87	47.76	91.23	69.27	85.90	71.22	83.37	80.90	77.21	64.47	85.93
	Ours	**48.92**	**91.33**	51.27	90.44	63.17	88.43	54.58	89.96	58.78	**87.98**	73.58	81.93	**58.38**	**88.34**
CIFAR100 WRN	MSP	70.24	84.42	74.49	83.05	65.60	87.25	75.70	82.57	81.91	77.05	76.87	80.23	74.14	82.43
	Mahalanobis	55.26	89.79	46.46	91.17	97.91	64.37	43.56	91.45	**25.39**	**95.08**	82.65	76.20	58.54	84.68
	ODIN	81.23	80.24	46.57	91.16	55.07	90.76	48.34	90.64	77.62	77.97	79.10	78.41	64.66	84.86
	Energy	66.49	87.81	68.04	86.38	45.53	92.65	70.83	85.69	82.89	77.33	80.47	79.88	69.04	84.96
	ViM	52.74	89.95	**38.90**	**92.20**	69.26	86.06	33.54	93.11	31.84	93.23	78.21	78.31	50.75	88.81
	DICE	73.03	80.77	80.46	82.01	10.64	97.96	84.04	80.47	77.50	77.23	80.62	76.99	67.72	82.57
	BATS	60.19	90.17	57.22	89.85	58.42	87.66	60.36	89.25	70.43	85.03	76.68	81.74	63.88	87.28
	ReAct	64.93	88.75	66.51	87.19	45.54	92.58	69.45	86.60	80.69	81.22	79.59	80.54	67.78	86.15
	DICE+ReAct	67.24	86.81	80.23	82.78	**8.59**	**98.22**	82.15	82.52	68.19	84.11	78.64	79.14	64.17	85.60
	LINe	66.67	87.32	64.59	87.98	68.01	86.73	**26.25**	**95.53**	83.67	75.70	81.44	77.20	65.11	85.08
	Ours	**38.03**	**92.85**	53.05	90.08	26.98	94.12	52.62	90.33	48.78	89.07	**76.07**	**81.76**	**49.25**	**89.70**

Table 5: Comparison between different methods in OOD detection on the CIFAR100 Benchmark with two different model backbones. ↑ indicates that larger values are better and ↓ indicates that smaller values are better. All values are percentages.

of ID and OOD classes. Consequently, the observed discrepancy in element-mean of feature vector between ID and OOD data would become much smaller. This is supported by the results on the CIFAR100 benchmark, where the number of ID classes in CIFAR100 is much larger than that in CIFAR10, and therefore the separability of ID categories cannot be easily achieved by increasing the angular distance alone. In this case, enhancing feature norms also becomes necessary. As a result, on benchmarks with more ID classes, the effect of element-mean discrepancy is relatively stable. For instance, on CIFAR100 (Table 5), by integrating the discrepancy, our method can help achieve a new state-of-the-art performance on both model backbones. Notably, it reduces FPR95 from 63.06% (ReAct) to 58.38% with ResNet18, and from 67.78% to 49.25% with WideResNet.

Here we primarily perform comparison with post-hoc methods. Notably, the state-of-the-art performance on the CIFAR benchmark is mostly achieved by model re-training methods. We leave the comparison with the model re-training method to a later section where we enhance our approach using the training trick proposed by Vaze (Vaze et al. 2022).

Discussion and Further Analysis

Element-Mean Discrepancy From Shallow Blocks

Every CNN backbone consists of four convolutional blocks. Here we further investigate the effects of element-mean discrepancy from the first three blocks for OOD detection. Specifically, we integrate these discrepancies from the first three blocks into the employed OOD score, i.e.,

$$D_B(\mathbf{x}) = \prod_{b=0}^{B} (1 + Z_{4-b}(\mathbf{x})) \cdot \log \sum_{k=1}^{C} \exp\left(f_k \circ h(\mathbf{x})\right), \quad (8)$$

where the subscript B represents the number of blocks, with Z_{4-b} representing the mean of feature map activations from the $(4-b)$-th block ($b \in \{0, 1, 2, 3\}$).

From Table 6, we can see that integrating the element-mean of feature map activations from blocks 1-3 into OOD score does not always improve the average OOD detection performance. On individual OOD data set, for example on iNaturalist, incorporating blocks 1-3 leads to better AUROC score than using only block 4 alone, indicating a positive contribution of early layer features. On the other hand, for Textures, adding blocks 1-3 results in a lower AUROC score than using block 4 alone, suggesting that blocks 1-3 are not helpful for distinguishing Textures from ID samples (see Figure 3).

One possible reason for this inconsistency is that features from earlier layers may capture different types of visual patterns than later layer, and their effect on OOD detection depends on the nature of the ID and OOD data sets. In general, blocks 1-3 encode more basic or generic features that are shared across object categories and ID/OOD domains, while block 4 often represents more specialized or discriminative features that are specific to ID categories. Therefore, when OOD samples are dissimilar to ID samples in terms of basic or generic features, early layer information may help

ID Dataset Model	Method	OOD Datasets									
		iNaturalist		SUN		Places		Textures		Average	
		F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑
ImageNet1K ResNet50	Block4	12.40	97.74	17.68	95.86	26.88	93.47	19.72	95.87	19.17	95.73
	Block4+3	11.79	97.84	17.37	95.93	26.53	93.61	21.68	95.52	19.34	95.72
	Block4+3+2	12.83	97.66	16.95	96.05	25.89	93.79	22.55	95.28	19.56	95.69
	Block4+3+2+1	12.01	97.81	16.32	96.26	25.03	94.04	23.28	95.05	19.16	95.79

Table 6: Discrepancy from additional shallow blocks for OOD detection. It can be seen that including activation mean of shallow blocks can slightly further improve the final average OOD detection performance on three out of the four OOD datasets.

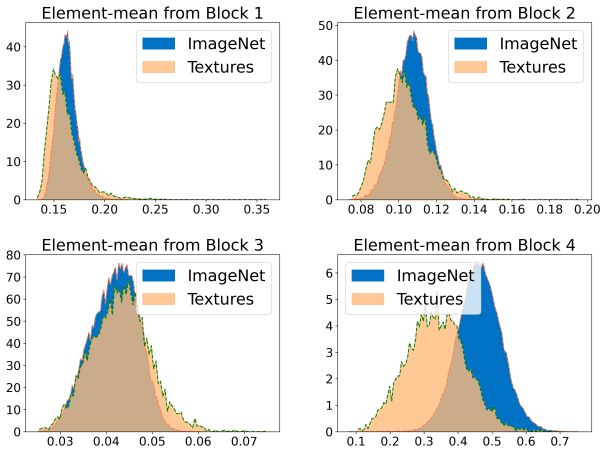

Figure 3: The discrepancy of element-mean from blocks 1-3 is less obvious than from block 4. The low-level features of the OOD set Textures encode abundant information that is shared with ID dataset.

improve the detection, whereas when OOD samples are similar to ID samples in terms of specialized or discriminative features, early layer information may hurt the detection.

Better Feature Representation Leads to Larger Difference in Feature Element

Our observation that the element-mean of feature vector is larger for ID than for OOD data is consistent with the finding by Vaze et al. (Vaze et al. 2022) that feature norms are often larger for ID than for OOD data. Since there are no negative elements in feature vector due to the ReLU operator, the size of feature norm is positively correlated to each unit activation. According to Vaze's finding, over the training progresses, the norm of ID data features is continuously increased, which pulls ID features away from the range of OOD features in the feature space. Therefore, if further optimization is performed to obtain better feature representations, e.g., larger norm of ID features, the difference in feature element between ID and OOD features will be also likely increased, leading to better OOD detection performance.

To verify the inference above, relevant experiments were performed by following the training strategy as in the related study (Vaze et al. 2022). Specifically, each model was trained up to 600 epochs and RandAugment strategy was adopted for data augmentation. As shown in Table 7, model training with more epochs and stronger data augmentation techniques (resulting in the stronger version of our method 'Ours+') can improve the classification performance that is comparable with those re-training methods. For example, with ResNet18 model on CIFAR10 benchmark, our method plus stronger traning trick can achieve average AUROC up to 96.76% which is already similar to the model re-training method LogitNorm. On CIFAR100, stronger training tricks can further improve the AUROC performance of our method from 88.34% to 89.72%. Notably, with WideResNet28-10 architecture on CIFAR100 benchmark, stronger training tricks can significantly improve the average AUROC from 89.70% to 91.30%, outperforming all existing post-hoc and re-training methods.

Related Work

OOD detection has been an active research area in machine learning and deep learning, and the objective of OOD detection study is to identify OOD data with distinct characteristics from ID data. Numerous research efforts have been dedicated to developing effective methods for distinguishing OOD samples from ID samples. Confidence-based approaches accomplish this by quantifying OOD scores using various scoring functions. For example, Hendrycks et al. (Hendrycks and Gimpel 2017) proposed the maximum softmax probability (MSP) of the model as a basic confidence-based OOD scoring function. ODIN (Liang, Li, and Srikant 2017) utilizes input perturbations and temperature scaling on the softmax layer to increase the difference between ID and OOD samples. Recently, Liu et al. (Liu et al. 2020) introduces an energy-based score with a theoretical interpretation from a likelihood perspective to enhance the effectiveness of confidence-based scores.

Different from confidence-based approaches, distance-based approaches measure the distance between input samples and typical ID samples or their centroids (Lee et al. 2018; Ming et al. 2023; Sun et al. 2022). These methods rely on the simple observation that OOD samples often result in larger distance compared to ID samples. Similarly, density-based methods (Kobyzev, Prince, and Brubaker 2020; Malinin and Gales 2019; Nandy, Hsu, and Lee 2020) identify OOD samples based on the distribution of the training samples and use density (or likelihood) as a metric.

Another line of studies focuses on OOD detection using post-hoc approaches. Unlike confidence-based and distance-

ID Dataset Model	Method	OOD Datasets													
		SVHN		LSUN-R		LSUN-C		iSUN		Textures		Places365		Average	
		F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓	A↑
CIFAR10 RN18	LogitNorm	12.68	97.75	15.29	97.45	0.53	99.82	15.36	97.43	31.56	94.09	32.31	93.92	17.96	96.75
	FeatureNorm	7.13	98.65	27.08	95.25	**0.07**	**99.96**	26.02	95.38	31.18	92.31	62.54	84.62	25.67	94.36
	CIDER	**1.61**	**99.68**	21.06	96.57	3.09	99.36	19.63	96.80	**10.35**	**98.29**	**25.13**	**95.16**	**13.48**	**97.64**
	Ours +	6.81	98.37	**8.30**	**98.29**	6.61	98.65	**9.59**	**98.16**	29.79	93.29	30.05	93.77	15.19	96.76
CIFAR10 WRN	LogitNorm	10.48	97.81	12.76	97.64	0.50	99.78	12.90	97.71	34.98	92.06	25.91	94.06	16.25	96.51
	FeatureNorm	3.83	99.18	8.13	98.32	0.32	99.81	5.98	98.71	14.23	97.06	48.69	90.91	13.53	97.33
	CIDER	**2.42**	**99.54**	**9.82**	**98.30**	**0.71**	**99.79**	**9.09**	**98.40**	**6.28**	**98.95**	**18.03**	**96.42**	**7.73**	**98.57**
	Ours +	3.36	99.20	10.32	97.76	3.41	99.15	10.02	97.85	22.22	95.10	25.06	93.64	12.40	97.12
CIFAR100 RN18	LogitNorm	51.34	91.79	88.80	78.67	**6.82**	**98.70**	90.16	75.55	77.02	77.52	77.79	79.56	65.32	83.63
	CIDER	**31.36**	**93.47**	80.39	81.54	43.68	89.45	78.23	81.33	**35.51**	**91.70**	82.80	72.71	58.66	85.03
	Ours +	41.45	92.87	**57.37**	**89.26**	23.24	95.38	**54.80**	**89.94**	49.15	89.82	**73.19**	**81.04**	**49.87**	**89.72**
CIFAR100 WRN	LogitNorm	47.31	92.79	78.92	81.05	**6.08**	**98.93**	78.58	80.85	64.57	81.92	**75.04**	**81.84**	58.42	86.23
	CIDER	18.66	96.24	69.22	85.78	35.95	89.68	64.86	85.91	**27.22**	**93.79**	81.68	73.71	49.60	87.52
	Ours +	**17.83**	**96.73**	**56.81**	**90.75**	23.50	95.47	**53.34**	**91.25**	37.75	92.71	77.09	80.89	**44.39**	**91.30**

Table 7: With stronger training strategy (indicated by symbol $+$), which leads to a better feature representation, our method can be comparable with state-of-the-art re-training methods.

based approaches that solely rely on the model's output to define OOD scores, post-hoc methods introduce modifications on model's output in order to amplify the differences between in-distribution and out-of-distribution data. Post-hoc methods offer notable advantages in real-world applications as they eliminate the need for model re-training which can potentially degrade the model's performance and increase training costs. Significant advancements have been made with post-hoc methods. For instance, ReAct (Sun, Guo, and Li 2021) has shown that OOD data can display remarkable excessive values in the penultimate layer activations. By truncating such overhigh activations, ReAct effectively enhances OOD detection performance. For another example, DICE (Sun and Li 2022) applies a weight selection method for selecting important weights of the overparameterized network, which further separates the energy score between ID and OOD. Our method also belongs to the category of post-hoc methods, and therefore our method was compared mainly with other post-hoc approaches in this study.

Conclusion

In this study, we propose a new method for OOD detection by leveraging the observation that the element-mean of feature vector obtained from CNN models is typically larger for in-distribution data than for out-of-distribution data. The proposed method is simple, efficient, and does not require additional training or computational resources. We provided mathematical analysis to help understand this discrepancy and evaluated our method on several benchmark datasets. The results demonstrate that our method can often improve OOD detection performance. Our method is also model-agnostic and can be easily integrated into various CNN architectures. Future work includes its generalization to various image processing and natural language processing tasks, and its combinations with more existing OOD techniques.

Acknowledgements

This work is supported in part by the Major Key Project of PCL (grant No. PCL2023AS7-1), the National Natural Science Foundation of China (grant No. 62071502), and Guangdong Excellent Youth Team Program (grant No. 2023B1515040025).

References

Ahn, Y. H.; Park, G.-M.; and Kim, S. T. 2023. LINe: Out-of-Distribution Detection by Leveraging Important Neurons. In *CVPR*.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; and Vedaldi, A. 2014. Describing textures in the wild. In *CVPR*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: a large-scale hierarchical image database. In *CVPR*.

DeVries, T.; and Taylor, G. W. 2018. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*.

Geiger, A.; Lenz, P.; and Urtasun, R. 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*.

Grathwohl, W.; Wang, K.-C.; Jacobsen, J.-H.; Duvenaud, D.; Norouzi, M.; and Swersky, K. 2020. Your classifier is secretly an energy based model and you should treat it like one. In *ICLR*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.

Hein, M.; Andriushchenko, M.; and Bitterwolf, J. 2019. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *CVPR*.

Hendrycks, D.; and Gimpel, K. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*.

Jiang, D.; Sun, S.; and Yu, Y. 2021. Revisiting flow generative models for out-of-distribution detection. In *ICLR*.

Kobyzev, I.; Prince, S. J.; and Brubaker, M. A. 2020. Normalizing flows: An introduction and review of current methods. *PAMI*, 43(11): 3964–3979.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. *Technical report, University of Toronto*.

Lakshminarayanan, B.; Pritzel, A.; and Blundell, C. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *NeurIPS*.

Lee, K.; Lee, K.; Lee, H.; and Shin, J. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*.

Liang, S.; Li, Y.; and Srikant, R. 2017. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*.

Liu, W.; Wang, X.; Owens, J.; and Li, Y. 2020. Energy-based Out-of-distribution Detection. In *NeurIPS*.

Malinin, A.; and Gales, M. 2018. Predictive uncertainty estimation via prior networks. *NeurIPS*.

Malinin, A.; and Gales, M. 2019. Reverse kl-divergence training of prior networks: Improved uncertainty and adversarial robustness. *NeurIPS*.

Ming, Y.; Sun, Y.; Dia, O.; and Li, Y. 2023. How to Exploit Hyperspherical Embeddings for Out-of-Distribution Detection? In *ICLR*.

Mudholkar, G. S.; and Hutson, A. D. 2000. The epsilon–skew–normal distribution for analyzing near-normal data. *Journal of statistical planning and inference*, 83(2).

Nalisnick, E.; Matsukawa, A.; Teh, Y. W.; Gorur, D.; and Lakshminarayanan, B. 2019. Do Deep Generative Models Know What They Don't Know? In *ICLR*.

Nandy, J.; Hsu, W.; and Lee, M. L. 2020. Towards maximizing the representation gap between in-domain & out-of-distribution examples. *NeurIPS*.

Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; and Ng, A. Y. 2011. Reading Digits in Natural Images with Unsupervised Feature Learning. In *NeurIPS*.

Ren, J.; Liu, P. J.; Fertig, E.; Snoek, J.; Poplin, R.; Depristo, M.; Dillon, J.; and Lakshminarayanan, B. 2019. Likelihood ratios for out-of-distribution detection. *NeurIPS*.

Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; and Chen, L.-C. 2018. Mobilenetv2: inverted residuals and linear bottlenecks. In *CVPR*.

Serrà, J.; Álvarez, D.; Gómez, V.; Slizovskaia, O.; Núñez, J. F.; and Luque, J. 2020. Input Complexity and Out-of-distribution Detection with Likelihood-based Generative Models. In *ICLR*.

Song, Y.; Sebe, N.; and Wang, W. 2022. RankFeat: Rank-1 Feature Removal for Out-of-distribution Detection. In *NeurIPS*.

Sun, Y.; Guo, C.; and Li, Y. 2021. ReAct: Out-of-distribution Detection With Rectified Activations. In *NeurIPS*.

Sun, Y.; and Li, Y. 2022. DICE: leveraging Sparsification for Out-of-Distribution Detection. In *ECCV*.

Sun, Y.; Ming, Y.; Zhu, X.; and Li, Y. 2022. Out-of-Distribution Detection with Deep Nearest Neighbors. In *ICML*.

Van Horn, G.; Mac Aodha, O.; Song, Y.; Cui, Y.; Sun, C.; Shepard, A.; Adam, H.; Perona, P.; and Belongie, S. 2018. The inaturalist species classification and detection dataset. In *CVPR*.

Vaze, S.; Han, K.; Vedaldi, A.; and Zisserman, A. 2022. Open-Set Recognition: A Good Closed-Set Classifier is All You Need. In *ICLR*.

Wang, H.; Li, Z.; Feng, L.; and Zhang, W. 2022. Vim: Out-of-distribution with virtual-logit matching. In *CVPR*.

Wei, H.; Xie, R.; Cheng, H.; Feng, L.; An, B.; and Li, Y. 2022. Mitigating Neural Network Overconfidence with Logit Normalization. In *ICML*.

Xiao, J.; Hays, J.; Ehinger, K. A.; Oliva, A.; and Torralba, A. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*.

Xu, P.; Ehinger, K. A.; Zhang, Y.; Finkelstein, A.; Kulkarni, S. R.; and Xiao, J. 2015. Turkergaze: crowdsourcing saliency with webcam based eye tracking. *arXiv preprint arXiv:1504.06755*.

Yu, F.; Seff, A.; Zhang, Y.; Song, S.; Funkhouser, T.; and Xiao, J. 2015. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. *arXiv e-prints*.

Yu, Y.; Shin, S.; Lee, S.; Jun, C.; and Lee, K. 2023. Block Selection Method for Using Feature Norm in Out-of-distribution Detection. In *CVPR*.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million image database for scene recognition. *PAMI*, 40(6): 1452–1464.

Zhu, Y.; Chen, Y.; Xie, C.; Li, X.; Zhang, R.; Tian, X.; Chen, Y.; et al. 2022. Boosting Out-of-distribution Detection with Typical Features. In *NeurIPS*.

Zisselman, E.; and Tamar, A. 2020. Deep residual flow for out of distribution detection. In *CVPR*.