# SPECTRAL PROJECTION PROFILES FOR OUT-OF-DISTRIBUTION DETECTION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Out-of-distribution (OOD) detection is essential for the safe deployment of deep models. However, the accuracy of OOD detection methods varies strongly across backbone models. We analyze this dependence through *spectral projection profiles*, measuring how in-distribution (ID) and OOD samples spread across the right singular vectors of the classifier's weight matrix. We find that in "well-behaved" models, which are accurate and robust across multiple dimensions, ID samples spread across the spectrum, whereas OOD inputs concentrate on the leading directions with large singular values. Less well-behaved models lack this distinction. This helps explain why logit-based and activation-shaping methods degrade on well-behaved models: the concentration of OOD inputs on the leading directions with larger singular values inflates the resulting logits, breaking the assumption that logits yield a reliable confidence measure. We further find that leading directions mainly encode local texture cues, whereas trailing directions encode global structure. Building on these insights, we propose *OMASHU*, a post-hoc OOD detection score with two complementary components that target the limitations of existing methods in well-behaved models. OMA normalizes the logits with a spectrally weighted norm that attenuates the effect of the leading directions. SHU exploits the fact that under patch shuffling, which destroys global structure but preserves local cues, the trailing projections of ID samples change more than those of OOD samples. On the OpenOOD benchmark, OMASHU and its variant OMASHU++, which also uses training statistics, achieve state-of-the-art OOD detection accuracy among methods without and with training data, respectively, for well-behaved transformer and convolutional backbones.

## 1 INTRODUCTION

As deep neural networks are deployed in an increasing number of safety-critical, real-world applications, ensuring their proper functioning is crucial (Huang et al., 2020). In this context, out-of-distribution (OOD) samples pose a particular challenge, as they violate the closed-world assumption of most models today and often result in degenerate outputs (Hendrycks & Gimpel, 2017). To address this problem, OOD detection methods aim to classify inputs as in-distribution (ID) or OOD, flagging predictions from input samples that do not stem from the training distribution.

Various approaches to OOD detection have been proposed, including logit-based, activation-shaping, and distance-based methods (see Sayyed et al., 2026, for an overview). Interestingly, OOD detection accuracy strongly depends on the employed backbone model. While a specific OOD approach may reliably identify OOD inputs on one model, the same approach can perform worse on another one (Zhang et al., 2024b), or when training with a different schedule (Krumpl et al., 2026). We seek to understand why and use these insights to design a new OOD detection approach.

We first show that models vary widely in how they represent ID inputs. To analyze this, we turn to *spectral projection profiles*, measuring how each sample utilizes the right singular vectors of the classifier's weight matrix. We find that the ID–OOD profile difference grows with a model's "well-behavedness," *i.e.*, with how accurate and robust it is across multiple dimensions (*cf.* Hesse et al., 2026). In a study across 11 backbones, we find that less well-behaved networks yield similar ID and OOD profiles, whereas well-behaved ones exhibit a pronounced projection gap (Figure 1 *(e)*).



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

These differences in spectral projection profiles also reveal *why* specific OOD detection methods favor certain models (Figure 1 *(a)–(d)*). In well-behaved models, OOD samples project more strongly than ID samples onto the leading directions, which have the largest singular values and dominate the resulting logits. OOD logits are therefore inflated, undermining the premise of logit-based and activation-shaping methods that ID inputs yield larger logits. In less well-behaved models, in contrast, ID and OOD profiles are similar, both concentrated on the leading directions, leaving distance-based methods little to separate. Hence, logit-based methods work well on less well-behaved models but degrade on well-behaved ones, while distance-based methods show the opposite trend (Figure 3).

Supported by our insights into well-behaved models, we propose OMASHU, a post-hoc OOD score with two *complementary* components. OMA targets the leading spectral directions: it normalizes the logits by a spectrally weighted norm that penalizes projections onto them, restoring logits as a reliable confidence measure. SHU targets the trailing directions, which OMA largely ignores. We find that leading directions mainly encode local texture cues, whereas trailing directions encode global structure: patch shuffling, which destroys global structure but preserves local cues, changes the trailing projections more for ID than for OOD samples (Figure 2). SHU measures this change. OMASHU combines both and needs no training data; OMASHU++ also uses training statistics.

To summarize, our contributions are: *(i)* We establish a new connection between model well-behavedness, a model's spectral projection profile, and the model dependence of OOD detection methods; moreover, we show that the leading and trailing directions encode local texture and global structure, respectively. *(ii)* We propose OMASHU(++), a post-hoc score that combines OMA, a spectrally weighted logit normalization targeting the leading directions, with SHU, which measures how trailing projections change under patch shuffling. *(iii)* We empirically demonstrate the effectiveness of OMASHU and OMASHU++, achieving state-of-the-art OOD detection accuracy on OpenOOD for well-behaved transformer and convolutional models, with and without training data.

## 2 Related Work

OOD detection aims to identify samples that differ significantly from the training data distribution. For classification, OOD samples are commonly defined as inputs that do not belong to any semantic category the model was trained on. In practice, most methods derive a scalar score whose ID and OOD distributions are separable. Existing methods fall into two main lines: training-based and post-hoc methods. *Training-based methods* include exposing the model to auxiliary data (Hendrycks et al., 2019; Jiang et al., 2024) or synthesized outliers (Du et al., 2022; 2023; Tao et al., 2023; Li & Zhang, 2025) during training; shaping representations and decision boundaries through contrastive learning (Sehwag et al., 2021), prototype-based learning (Ming et al., 2023; Lu et al., 2024), or semantic grouping (Huang & Li, 2021); and normalizing logits (Wei et al., 2022; Ding et al., 2026) or features (Ming et al., 2023; Zhang & Xiang, 2023) to improve ID–OOD separation.

In contrast, *post-hoc methods*, including ours, detect OOD inputs without retraining. They typically rely on prediction confidence or features of test samples. *Logit-based methods* derive scores from the distribution and magnitudes of logits (Liu et al., 2020; Hendrycks et al., 2022; Zhang et al., 2023; Hofmann et al., 2024; Djurisic et al., 2025), assuming logits yield a reliable ID confidence measure. *Activation-shaping methods* prune or scale intermediate features to amplify the gap between ID and OOD logits (Sun et al., 2021; Sun & Li, 2022; Zhu et al., 2022; Ahn et al., 2023; Djurisic et al., 2023; Xu et al., 2023; 2024; Yuan et al., 2024; Regmi, 2026). We show that this assumption fails for well-behaved models, as OOD samples project onto the leading directions of the spectral projection profile with the highest singular values; our OMA score compensates for this. *Softmax-based methods* use the output of the softmax layer in the form of prediction confidence or information entropy (Hendrycks & Gimpel, 2017; Chan et al., 2021; Liu et al., 2023). *Perturbation-consistency methods* apply image-level perturbations, assuming ID scores are more stable under them than OOD scores (Zhang et al., 2024a; Chen et al., 2025; Regmi, 2026). Whereas these methods perturb local cues and expect stable ID scores, our SHU score uses patch shuffling, which disrupts global structure but leaves OOD projections more stable. Finally, *gradient-based methods* derive scores from weight gradients induced by test samples (Huang et al., 2021; Behpour et al., 2023; Chen et al., 2023).

*Distance-based methods* additionally rely on training data statistics, scoring test samples by their distance to training samples (Sastry & Oore, 2020; Sun et al., 2022; Park et al., 2023; Zöngür et al., 2025), class prototypes (Lee et al., 2018; Ren et al., 2021; Müller & Hein, 2025; Pasarkar & Dieng,

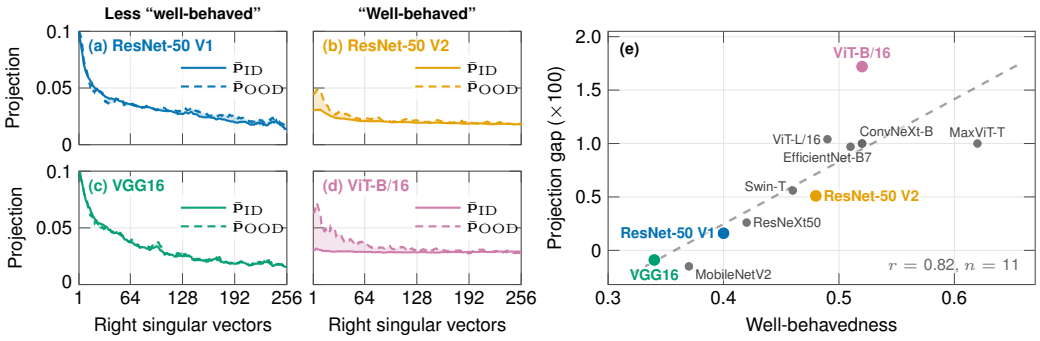



Figure 1: **Spectral projection profiles & well-behavedness.** *(a)–(d):* Mean normalized spectral projection profiles of ID (solid) and OOD (dashed) data, shown for the first 256 right singular vectors; shaded areas show the ID–OOD difference. For ID and OOD data, we use ImageNet-1k (Russakovsky et al., 2015) and OpenImage-O (Wang et al., 2022), respectively. *(e):* Projection gap $G$ from Equation (4) *vs.* well-behavedness for 11 TorchVision models (TorchVision maintainers and contributors, 2016); highlighted points correspond to (a)–(d). Similar to Hesse et al. (2026), we define well-behavedness as the unweighted average of classification accuracy, adversarial robustness, corruption robustness, OOD robustness, and shape bias. The more well-behaved a network, the more strongly OOD samples concentrate on the leading directions relative to ID inputs ($r = 0.82$).

2026), or feature subspaces (Song et al., 2025; Wang et al., 2022; Guan et al., 2023; Fang et al., 2024; Zöngür et al., 2025). MDS (Lee et al., 2018) uses the Mahalanobis distance to class centers. RMDS (Ren et al., 2021) measures class fit relative to the overall ID distribution in whitened space. fDBD (Liu & Qin, 2024) uses the feature distance to decision boundaries. NCI (Liu & Qin, 2025) exploits neural collapse to detect OOD inputs through feature alignment. MDS++ and RMDS++ (Müller & Hein, 2025) show that feature normalization restores the Gaussianity assumed by the Mahalanobis distance. KNN (Sun et al., 2022) relies on the $L_2$ distance to the nearest training features. SAE (Oh et al., 2026) quantifies the divergence from activation profiles of a sparse autoencoder. VNS (Pasarkar & Dieng, 2026) scores how much a test sample increases the Vendi Score of an ID reference set. *Hybrid methods*, including our OMASHU++, combine such scores with logits (Cook et al., 2020; Wang et al., 2022; Guan et al., 2023; Park et al., 2023; Fang et al., 2024; Ling et al., 2025; Zöngür et al., 2025). ViM (Wang et al., 2022) combines the residual-subspace feature norm with the logits. ActSub (Zöngür et al., 2025) combines feature similarity and logits from complementary activation subspaces. CADRef (Ling et al., 2025) splits relative deviations from class-average features considering positive and negative logit contributions.

Our analysis relates OOD detection to model *well-behavedness*, which Hesse et al. (2026) introduced to characterize classifiers along multiple quality dimensions beyond accuracy. We adopt a more focused, robustness-oriented definition based on accuracy, adversarial robustness, corruption robustness, OOD robustness, and shape bias. We omit dimensions such as parameter count or class balance, which relate less to robustness or vary little across models. Well-behavedness is not tied to architecture: modern CNNs (Tan & Le, 2019; Liu et al., 2022), vision transformers (Dosovitskiy et al., 2021; Liu et al., 2021; Tu et al., 2022), and ResNets trained with modern recipes (Vryniotis, 2021; Wightman et al., 2021) are well-behaved in this sense, whereas older ResNets (He et al., 2016) and VGG networks (Simonyan & Zisserman, 2015) are less so. See Appendix B for further discussion of related OOD detection methods and well-behavedness.

## 3 Spectral Projection Profile, Well-Behavedness, and OOD Detection

We now analyze how a model's spectral projection profile relates to its well-behavedness and to the accuracy of OOD detection methods. We formalize the *spectral projection profile*, central to our analysis, as follows. We decompose a classification model into an embedding function $f : \mathcal{X} \to \mathbb{R}^d$ and a linear classifier $g(\mathbf{z}) = W\mathbf{z} + \mathbf{b}$, with classifier weights $W \in \mathbb{R}^{C\times d}$. Let

$$W = U\Sigma V^\top \tag{1}$$

denote the SVD of $W$, with singular values ordered as $\sigma_1 \geq \ldots \geq \sigma_r$ and corresponding right singular vectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$, where $r = \min(C, d)$. We define the spectral projection profile of an input $x$ as

$$\mathbf{p}(x) = \left(|\mathbf{v}_1^\top f(x)|, |\mathbf{v}_2^\top f(x)|, \ldots, |\mathbf{v}_r^\top f(x)|\right) \in \mathbb{R}^r. \tag{2}$$

For $d > C$, $\mathbf{p}(x)$ ignores the part of $f(x)$ in the null space of $W$, which cannot affect the logits. To compare profiles across samples and models independently of the feature magnitude, we use the normalized profile $\hat{\mathbf{p}}(x) = \mathbf{p}(x)/\|f(x)\|$, where $\|\cdot\|$ denotes the $L_2$ norm. Since the right singular vectors are orthonormal, $\hat{p}_i(x)^2$ is the fraction of feature energy along $\mathbf{v}_i$, and $\|\hat{\mathbf{p}}(x)\| \leq 1$. For a set of samples $\mathcal{D}$, we define the mean normalized profile

$$\bar{\mathbf{p}}_{\mathcal{D}} = \frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} \hat{\mathbf{p}}(x), \tag{3}$$

and write $\bar{\mathbf{p}}_{\text{ID}}$ and $\bar{\mathbf{p}}_{\text{OOD}}$ for ID and OOD data, respectively.

Figure 1 *(a)–(d)* shows mean normalized profiles for four models. For the less well-behaved ones, VGG16 and ResNet-50 V1, the ID profile is dominated by the leading directions, with large singular values, and decays quickly, whereas the more well-behaved networks spread their representations over many more directions. Since the logit norm satisfies $\|Wf(x)\| = \|\Sigma\,\mathbf{p}(x)\|$, see Appendix A, the logits of less well-behaved models are then dominated by a few high-singular-value directions. One explanation is that less well-behaved networks rely more on texture cues (Geirhos et al., 2019), which, as we show below (Observation 3), are predominantly captured by the leading directions.

**Observation 1:** **Well-behavedness is reflected in the spectral projection profile.** In less well-behaved models, ID representations concentrate on the leading right singular directions, whereas well-behaved networks distribute them over many more spectral directions.

Beyond model differences, the profiles also reveal a characteristic difference between ID and OOD samples (Figure 1 *(a)–(d)*, dashed *vs.* solid). OOD inputs consistently concentrate on the leading directions. For less well-behaved models such as VGG16 or ResNet-50 V1, ID representations also concentrate on the leading directions, so that ID and OOD profiles are similar. For the more well-behaved networks, the profiles separate: ID representations spread across many directions, whereas OOD representations remain concentrated on the leading ones, resembling the profiles of less well-behaved models. This is consistent with OOD samples not matching the global, semantic features learned for ID data and thus relying more on local (texture) cues.

To quantify this difference, we measure how much more OOD than ID projection mass lies in the leading directions. Let $k$ be the smallest index with $\sum_{i=1}^{k} \sigma_i^2 \geq 0.25 \sum_{i=1}^{r} \sigma_i^2$, *i.e.*, the leading directions accounting for 25 % of the classifier's spectral energy. We define the projection gap as

$$G = \frac{1}{k} \sum_{i=1}^{k} \left(\bar{p}_{\text{OOD},i} - \bar{p}_{\text{ID},i}\right). \tag{4}$$

Figure 1 *(e)* shows Pearson's correlation $r = 0.82$ between $G$ and well-behavedness; leave-one-model-out correlations range from $r = 0.77$ (without VGG16) to $0.91$ (without MaxViT-T). Other cut-off thresholds show similar trends, see Appendix A.

**Observation 2:** **OOD samples exhibit distinct spectral projection profiles for well-behaved models.** In well-behaved networks, OOD representations concentrate on the leading directions, whereas ID representations spread across the spectrum; this gap grows with well-behavedness.

To connect these observations to image content, we examine what leading directions and trailing directions, *i.e.*, those with small singular values, encode. We use ViT-B/16 and two experiments (Figure 2). First, we reweight the classifier's spectrum by replacing $W$ with $W_\alpha = c_\alpha U \Sigma^\alpha V^\top$, where $c_\alpha$ preserves the mean singular value; $\alpha = 1$ recovers the original classifier, and decreasing $\alpha$ shifts weight from the leading to the trailing directions. As $\alpha$ decreases, shape bias (Geirhos et al., 2019), *i.e.*, the fraction of shape-based decisions on cue-conflict images, increases monotonically (*left*). For $\alpha \in [-1.5, 2]$, it rises by 10 percentage points, while classification accuracy changes by at most 2 points; only beyond this range does accuracy drop sharply. Second, we divide each image into equal square patches and shuffle them with a random permutation shared across images, destroying

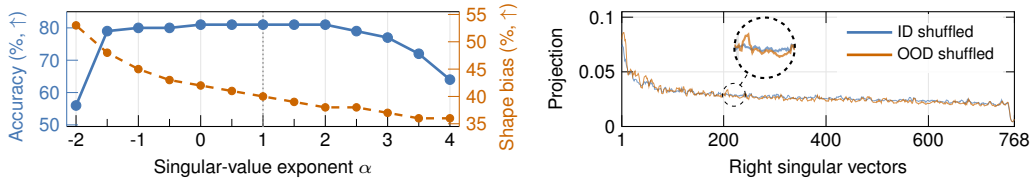



Figure 2: **What leading and trailing directions encode (ViT-B/16).** *Left:* Classification accuracy (in %) and shape bias (in %) when the classifier weights are replaced by $W_\alpha = c_\alpha U \Sigma^\alpha V^\top$. Shape bias decreases as more weight is put on the leading directions, while accuracy remains stable for $\alpha \in [-1.5, 2]$. *Right:* Mean normalized profiles of patch-shuffled ID and OOD images (dashed circle: magnified inset). ID profiles lose trailing-direction mass and resemble OOD profiles.

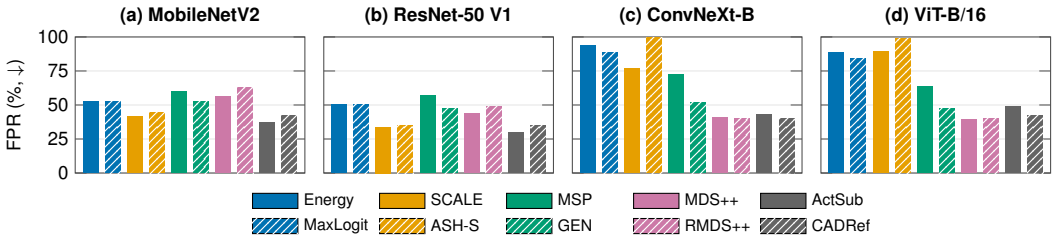



Figure 3: **OOD detection accuracy of five method families on four backbones.** False positive rate (FPR, in %, ↓) over OpenOOD (Zhang et al., 2024b) OOD datasets, with ImageNet-1k (Russakovsky et al., 2015) as ID. Panels *(a)–(d)* are ordered by increasing well-behavedness. Within each panel, bars are grouped by method family, from *left* to *right*: logit-based, with Energy (Liu et al., 2020) & MaxLogit (Hendrycks et al., 2022); activation-shaping, with SCALE (Xu et al., 2024) & ASH-S (Djurisic et al., 2023); softmax-based, with MSP (Hendrycks & Gimpel, 2017) & GEN (Liu et al., 2023); distance-based, with MDS++ & RMDS++ (Müller & Hein, 2025); and hybrid, with ActSub (Zöngür et al., 2025) & CADRef (Ling et al., 2025).

global structure while preserving local cues. Under shuffling, OOD profiles remain stable, whereas ID profiles lose projection mass in the trailing directions and come to resemble OOD profiles (*right*).

> **Observation 3:** **Leading directions encode local texture, trailing directions encode global structure.** Emphasizing the trailing directions increases shape bias, and destroying global structure removes the trailing-direction mass of ID representations but barely affects OOD ones.

**Implications for OOD detection methods.** Observations 1 and 2 help explain why OOD detection methods favor some networks over others. Since $\|Wf(x)\| = \|\Sigma\,\mathbf{p}(x)\|$, a representation concentrated on the leading directions yields larger logits than one of equal norm that is spread across the spectrum. OOD samples in well-behaved models therefore obtain large logits relative to their feature norm (Observation 2), undermining the premise of logit-based methods that ID inputs yield larger logits, *i.e.*, higher confidence. Activation-shaping methods inherit this premise, as they reshape features but evaluate the result with logit-based scores such as energy. Softmax-based methods are less affected but not immune, as scaling the logits acts as an inverse temperature and leads to overconfident predictions. Distance-based methods instead assume that ID samples lie closer to the training data or class prototypes than OOD inputs. This premise suits well-behaved models, where ID representations extend into directions that OOD samples barely use, but weakens for less well-behaved networks, where ID and OOD profiles are both concentrated on the leading directions.

Figure 3 confirms these predictions for two less and two more well-behaved backbones. Logit-based and activation-shaping methods perform well on MobileNetV2 and ResNet-50 V1 (FPR@95 of 34–53 %), but degrade strongly on ConvNeXt-B and ViT-B/16 (77–99 %). Distance-based methods show the opposite trend: they achieve the lowest FPR@95 on the well-behaved models (≈40 %) and outperform softmax-based methods there (48–73 %). Hybrid methods, which combine distance and logit information, perform best on the less well-behaved networks and remain competitive on the well-behaved ones, but, except for CADRef on ConvNeXt-B, do not reach distance-based methods.

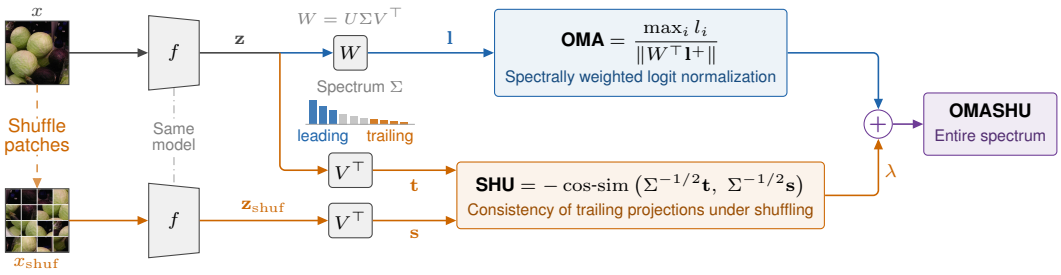



Figure 4: **Overview of OMASHU.** An input $x$ and its patch-shuffled version $x_{\text{shuf}}$ are fed into the backbone $f$. OMA normalizes the maximum logit by $\|W^\top \mathbf{l}^+\|$, which attenuates the influence of leading directions. SHU compares the projections $\mathbf{t}$ and $\mathbf{s}$ of the original and shuffled features onto right singular vectors, emphasizing trailing directions. OMASHU combines both, weighted by $\lambda$.

Observation 3 adds a further implication. Since logits weight each direction by its singular value, logit- and softmax-based scores largely ignore the trailing directions, which carry the global structure that distinguishes ID samples. Perturbation-consistency methods perturb local cues and are thus expected to mainly affect the leading directions, onto which both ID and OOD inputs project strongly; perturbing global structure instead changes ID representations more than OOD ones in well-behaved models, a signal that our SHU score exploits (Section 4).

## 4 OMASHU Method

Building on Section 3, we propose OMASHU, an OOD score with two complementary components (Figure 4) that targets modern, well-behaved models. OMA corrects the logits for the strong projections of OOD samples onto the leading directions (Observation 2), while SHU measures how consistently the trailing directions respond to patch shuffling (Observation 3). We describe both components and their combination below, followed by a variant that uses training statistics.

**OMA: Spectrally weighted logit normalization.** Let $\mathbf{z} = f(x)$ and $\mathbf{l} = W\mathbf{z}$ denote the features and logits of an input $x$, where we omit the bias. Since $\|\mathbf{l}\| = \|\Sigma\,\mathbf{p}\|$, OOD samples in well-behaved models obtain large logits relative to their feature norm, which undermines logit-based scores (Section 3). We therefore normalize the logits by a norm of the spectral projection profile in which the leading directions dominate. Weighting $\mathbf{p}$ with powers $\Sigma^\alpha$ of the singular values yields a family of normalizers $\|\Sigma^\alpha \mathbf{p}\|$ that includes three familiar terms:

$$\|\Sigma^0\mathbf{p}\| = \|\mathbf{z}\| \quad \text{if } d \le C, \qquad \|\Sigma^1\mathbf{p}\| = \|\mathbf{l}\|, \qquad \|\Sigma^2\mathbf{p}\| = \|W^\top\mathbf{l}\|, \tag{5}$$

where the last identity follows from $W^\top W = V\Sigma^2 V^\top$. The feature norm weights all directions uniformly, the logit norm weights them by their singular values, and the Gram norm $\|W^\top\mathbf{l}\|$ sharpens this weighting further, penalizing projections onto the leading directions most strongly. We use $\alpha = 2$: it yields the best ID–OOD separation of the three on average (Table 2) and can be computed directly from the logits, without an SVD at test time.

Since the Gram norm aggregates all logits, it is sensitive to strongly negative logits, whose magnitude can grow arbitrarily with little effect on the softmax output. We thus restrict the normalizer to the positive part $\mathbf{l}^+ = \max(\mathbf{l}, \mathbf{0})$ of the logits (element-wise), which further improves separation (Table 2). The resulting OMA score (**po**sitive nor**ma**lized logits) is the maximum normalized logit,

$$\text{OMA}(x) = \frac{\max_i l_i}{\|W^\top \mathbf{l}^+\|}. \tag{6}$$

By construction, OMA attenuates the influence of the leading directions on the logits; SHU focuses on complementary information in the trailing directions.

**SHU: Consistency of trailing projections under patch shuffling.** By Observation 3, patch shuffling removes the trailing-direction mass of ID representations while leaving OOD representations largely unchanged. SHU turns spectral profile consistency under patch **shu**ffling into a score. We

shuffle non-overlapping square patches of $x$ with a fixed random permutation, shared across images, to obtain $x_{\text{shuf}}$; the patch size is a hyperparameter (Figure 5). Let $\mathbf{t} = V^\top f(x)$ and $\mathbf{s} = V^\top f(x_{\text{shuf}})$ denote the (signed) projections of the original and shuffled images onto the right singular vectors. SHU is their negative cosine similarity after weighting each direction by $\Sigma^{-1/2}$,

$$\text{SHU}(x) = -\cos\text{-sim}\Big(\Sigma^{-1/2}\mathbf{t},\, \Sigma^{-1/2}\mathbf{s}\Big) = -\frac{\mathbf{t}^\top\Sigma^{-1}\mathbf{s}}{\sqrt{(\mathbf{t}^\top\Sigma^{-1}\mathbf{t})\,(\mathbf{s}^\top\Sigma^{-1}\mathbf{s})}}, \tag{7}$$

so that ID samples, whose projections change more under shuffling, obtain higher scores. The weighting scales the contribution of direction $i$ by $1/\sigma_i$, emphasizing the trailing directions; it can be seen as a standardization if the singular values reflect the spread of the projections along each direction. Unlike $\mathbf{p}$, SHU uses signed projections, which also capture sign changes under shuffling.

**OMASHU: Combining both components.** OMA and SHU focus on different parts of the spectrum, the leading and trailing directions. We combine them into

$$\text{OMASHU}(x) = \text{OMA}(x) + \lambda\ \text{SHU}(x), \tag{8}$$

where $\lambda \in \mathbb{R}^+$ balances the score scales and is chosen per backbone on a held-out validation set. OMASHU is post-hoc and needs no training data; its main cost is a second forward pass for $x_{\text{shuf}}$.

**OMASHU++: Using training statistics.** When training data is available, we additionally account for class-dependent variations of both scores. For OMA, we normalize the score by its class-wise average on the training data. With the predicted class $c = \arg\max_i l_i$ and the set $\mathcal{N}_c$ of training samples of class $c$, we define

$$\text{OMA}_{++}(x) = \frac{\text{OMA}(x)}{\mu_c}, \qquad \mu_c = \frac{1}{|\mathcal{N}_c|}\sum_{j\in\mathcal{N}_c}\frac{\max_i l_{j,i}}{\|\mathbf{l}_j^+\|}. \tag{9}$$

The class averages use the logit norm $\|\mathbf{l}^+\|$ rather than the Gram norm, so that the class normalization does not cancel the additional spectral weighting of OMA. For SHU, we replace the singular values by class-wise statistics of the signed training projections: we center $\mathbf{t}$ and $\mathbf{s}$ by the class mean $\boldsymbol{\mu}_c^t$ and weight each direction by the class-wise standard deviations $D_c = \text{diag}(\boldsymbol{\sigma}_c^t)$,

$$\text{SHU}_{++}(x) = -\cos\text{-sim}\Big(D_c^{-1/2}(\mathbf{t}-\boldsymbol{\mu}_c^t),\, D_c^{-1/2}(\mathbf{s}-\boldsymbol{\mu}_c^t)\Big). \tag{10}$$

OMASHU++ combines both as in Equation (8), *i.e.*, $\text{OMASHU}_{++} = \text{OMA}_{++} + \lambda\ \text{SHU}_{++}$, with the weighting factor $\lambda \in \mathbb{R}^+$. Appendix C discusses the limitations of OMASHU(++).

## 5 Experiments

**Setup.** We evaluate the OOD-detection accuracy of OMASHU(++) on four widely used well-behaved models: ViT-B/16 (Dosovitskiy et al., 2021), Swin-T (Liu et al., 2021), ConvNeXt-B (Liu et al., 2022), and ResNet-50 V2 (Vryniotis, 2021), using TorchVision checkpoints. See Appendix A for results on less well-behaved models. We use the OpenOOD benchmark (Zhang et al., 2024b) setup with ImageNet-1k val. as ID data. The near-OOD datasets SSB-Hard (Vaze et al., 2022) and NINCO (Bitterwolf et al., 2023) represent small shifts; the far-OOD datasets iNaturalist (Van Horn et al., 2018), Textures (Cimpoi et al., 2014), and OpenImage-O (Wang et al., 2022) represent large shifts relative to ImageNet-1k. Following common practice, we optimize OMASHU's hyperparameters ($\lambda$ & patch size) on OpenOOD val. per backbone. Patch size is optimized with SHU(++) only. We report the area under the ROC curve (AUC) and the false positive rate of OOD samples at 95 % true positive rate on ID samples (FPR). Higher AUC & lower FPR indicate better OOD detection.

**Baselines.** We split the evaluation into whether methods access the training data. We compare OMASHU with existing state-of-the-art approaches that do not use training data and OMASHU++ with those that do. We further evaluate the individual components of each variant: OMA and SHU for OMASHU, and OMA++ and SHU++ for OMASHU++ as baselines. Without training data, we compare against MSP (Hendrycks & Gimpel, 2017), Energy (Liu et al., 2020), MaxLogit (Hendrycks et al., 2022), ReAct (Sun et al., 2021), GEN (Liu et al., 2023), ASH (Djurisic et al., 2023), SCALE (Xu et al., 2024), LTS (Djurisic et al., 2025), PRO (Chen et al., 2025), and AdaSCALE (Regmi, 2026). Baselines with training data and their abbreviations are introduced in Section 2. For approaches with multiple variants, we pick the best on OpenOOD val. Unless noted otherwise, we reproduced all approaches with available code, tuned on the same validation set.

Table 1: **OOD detection accuracies for transformer & convolutional models.** Near- and far-OOD detection accuracy using AUC (in %, ↑) and FPR (in %, ↓, @95 %) for ImageNet-1k transformer (ViT-B/16 & Swin-T) and convolutional models (ResNet-50 V2 & ConvNeXt-B). We report results for existing approaches and ours, w/ & w/o access to training data at inference time. Avg. denotes the average results over all datasets. † results taken from the original papers; others are reproduced. Best results **bold** and highlighted; second best underlined and highlighted (gray rows omitted).

| | | Transformer Models | | | | | | Convolutional Models | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Near | | Far | | Avg. | | Near | | Far | | Avg. | |
| | **Method** | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ |
| **Without training data** | MSP | 75.14 | 76.47 | 86.17 | 50.41 | 81.76 | 60.83 | 75.40 | 78.16 | 81.90 | 71.89 | 79.30 | 74.40 |
| | Energy | 67.82 | 88.25 | 80.15 | 80.47 | 75.22 | 83.58 | 59.61 | 90.17 | 52.00 | 93.18 | 55.04 | 91.98 |
| | MaxLogit | 71.98 | 86.51 | 84.17 | 73.52 | 79.29 | 78.72 | 71.94 | 86.01 | 75.02 | 86.65 | 73.79 | 86.39 |
| | ReAct | 72.45 | 78.57 | 86.96 | 48.47 | 81.15 | 60.51 | 56.48 | 82.33 | 58.95 | 78.12 | 57.96 | 79.80 |
| | GEN | 77.63 | 67.20 | 91.16 | 32.41 | 85.75 | 46.32 | 79.63 | 64.18 | 89.55 | 43.69 | 85.58 | 51.88 |
| | ASH | 33.73 | 96.93 | 20.21 | 97.95 | 25.61 | 97.54 | 49.66 | 92.95 | 49.47 | 92.08 | 49.55 | 92.43 |
| | SCALE | 63.72 | 90.04 | 73.90 | 85.81 | 69.83 | 87.51 | 66.23 | 82.77 | 70.07 | 80.06 | 68.53 | 81.15 |
| | LTS | 72.11 | 84.17 | 84.81 | 65.87 | 79.73 | 73.19 | 58.14 | 90.91 | 53.70 | 92.20 | 55.48 | 91.69 |
| | PRO | 77.53 | 66.95 | 90.88 | 32.85 | 85.54 | 46.49 | 79.81 | **62.38** | 89.61 | 41.78 | 85.69 | 50.02 |
| | AdaSCALE | 74.00 | 74.65 | 88.86 | 41.23 | 82.92 | 54.60 | 68.25 | 78.77 | 85.17 | 46.01 | 78.40 | 59.11 |
| | OMA *(Ours)* | 79.86 | 62.93 | 91.52 | 31.74 | 86.85 | 44.22 | 80.38 | 61.92 | 90.74 | 34.79 | 86.59 | 45.64 |
| | SHU *(Ours)* | 61.45 | 87.98 | 85.26 | 54.83 | 75.74 | 68.09 | 63.64 | 89.31 | 84.17 | 59.67 | 75.96 | 71.52 |
| | OMA *(Ours)* + CoVer | 79.24 | 64.96 | 91.92 | 30.39 | 86.85 | 44.21 | 80.38 | 61.92 | 90.74 | 34.79 | 86.59 | 45.64 |
| | **OMASHU** *(Ours)* | **79.93** | **63.22** | **95.04** | **22.52** | **89.00** | **38.80** | **80.51** | 63.14 | **94.40** | **26.07** | **88.84** | **40.89** |
| **With training data** | MDS | 77.11 | 67.38 | 92.04 | 29.92 | 86.07 | 44.90 | 78.16 | 66.76 | 93.34 | 26.46 | 87.27 | 42.58 |
| | RMDS | 79.22 | 65.91 | 92.47 | 28.80 | 87.17 | 43.64 | 81.01 | 63.50 | 90.31 | 33.57 | 86.59 | 45.54 |
| | KNN | 72.86 | 71.12 | 90.09 | 33.03 | 83.20 | 48.26 | 74.42 | 70.21 | 90.76 | 31.68 | 84.23 | 47.09 |
| | ViM | 76.14 | 74.32 | 93.02 | 26.99 | 86.27 | 45.92 | 71.61 | 79.68 | 93.16 | 26.83 | 84.54 | 47.97 |
| | fDBD | 73.15 | 73.24 | 89.47 | 36.94 | 82.94 | 51.46 | 72.03 | 73.85 | 89.26 | 38.77 | 82.37 | 52.80 |
| | NCI | 74.82 | 71.92 | 89.49 | 38.11 | 83.62 | 51.64 | 72.98 | 74.46 | 91.11 | 36.17 | 83.86 | 51.48 |
| | CADRef | 78.17 | 66.51 | 92.38 | 26.81 | 86.70 | 42.69 | 60.40 | 76.61 | 72.52 | 55.25 | 67.67 | 63.79 |
| | ActSub | 76.16 | 65.66 | 88.88 | 32.90 | 83.79 | 46.00 | 77.68 | 63.55 | 89.50 | 35.91 | 84.77 | 46.97 |
| | MDS++ | 80.89 | 65.18 | 94.12 | 22.27 | 88.83 | 39.44 | 81.46 | 63.77 | 92.47 | 26.02 | 88.07 | 41.12 |
| | RMDS++ | 81.61 | 64.21 | 93.14 | 24.85 | 88.53 | 40.59 | 82.35 | 61.96 | 88.69 | 33.81 | 86.15 | 45.07 |
| | SAE† | 77.87 | 66.87 | 92.50 | 25.87 | 86.65 | 42.28 | – | – | – | – | – | – |
| | VNS† | 81.47 | 60.26 | 92.58 | 26.86 | 88.14 | 40.22 | – | – | – | – | – | – |
| | OMA++ *(Ours)* | 82.71 | 58.92 | 91.71 | 31.21 | 88.11 | 42.30 | 83.63 | 56.62 | 91.71 | 32.19 | 88.48 | 41.97 |
| | SHU++ *(Ours)* | 79.60 | 64.58 | 94.82 | 22.98 | 88.73 | 39.62 | 80.53 | 62.23 | 94.69 | 23.10 | 89.03 | 38.75 |
| | **OMASHU++** *(Ours)* | **82.58** | **59.50** | **95.46** | **20.38** | **90.31** | **36.03** | **83.98** | **56.41** | **95.20** | **21.45** | **90.71** | **35.44** |

### 5.1 Comparison to the State of the Art

Table 1 compares OMASHU(++) with existing methods and additionally reports OMA(++) and SHU(++) alone, with near-OOD, far-OOD, and average results across the five OpenOOD datasets for well-behaved transformer and convolutional models. *Without access to training data* during inference, OMA *alone* already achieves state-of-the-art accuracy for both model families. SHU alone falls short of the baselines and OMA. However, combining OMA with SHU (*i.e.*, OMASHU) substantially improves over OMA and outperforms all existing methods, notably even all baselines using training data. Against the second-best methods, GEN for transformer and PRO for convolutional models, OMASHU improves average AUC by 3.25 pp and 3.15 pp and FPR by 7.52 pp and 9.13 pp, respectively. The gain of OMASHU over OMA confirms that the two components are complementary. This synergy is specific to SHU: pairing OMA with CoVer (Zhang et al., 2024a), an alternative perturbation-consistency method, yields only marginal improvements (*cf.* Appendix B).

*With training data* (Table 1, *lower part*), OMA++ alone is competitive with the baselines. SHU++ *alone* also attains competitive results (SoTA in convolutional models), unlike SHU, since training statistics let SHU++ capture inter-class variations. Combining both (OMASHU++) achieves state-of-the-art accuracy *consistently* across near- and far-OOD and model families on average. OMASHU++ outperforms the closest baseline, MDS++, on average, improving AUC by 1.48 pp and 2.64 pp and FPR by 3.41 pp and 5.68 pp for transformer and convolutional models, respectively.

### 5.2 Analyzing OMASHU(++)

**OMA with different normalizations.** Table 2 compares OMA's spectrally amplified normalization with alternative normalizations. Specifically, we keep OMA's MaxLogit numerator fixed and vary the normalization. Starting from a constant (1) and feature norm ($\|\mathbf{z}\|$, Zhang & Xiang, 2023),

Table 2: **OMA normalization analysis.** We analyze different normalization choices for OMA and report near- and far-OOD detection accuracy using AUC (in %, ↑) and FPR (in %, ↓, @95 %) for ImageNet-1k transformers (TFNs; ViT-B/16 & Swin-T) and convolutional models (CNNs; ResNet-50 V2 & ConvNeXt-B). Avg. is taken over all datasets. Positive pooling over logits (OMA) scores consistently better. Best results **bold** and highlighted; second best underlined and highlighted.

| Normalization → | | $1$ | | | $\|\mathbf{z}\|$ | | | $\|\mathbf{1}\|$ | | | $\|W^\top \mathbf{1}\|$ | | | $\|W^\top \mathbf{1}^+\|$ **(OMA)** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | TFNs | CNNs | Avg. | TFNs | CNNs | Avg. | TFNs | CNNs | Avg. | TFNs | CNNs | Avg. | TFNs | CNNs | Avg. |
| **Near** | AUC↑ | 71.98 | 71.94 | 71.96 | 73.49 | 72.11 | 72.80 | 75.42 | 77.05 | 76.23 | 76.56 | 77.88 | 77.22 | **79.86** | **80.38** | **80.12** |
| | FPR↓ | 86.51 | 86.01 | 86.26 | 73.04 | 74.53 | 73.78 | 69.35 | 67.35 | 68.35 | 67.55 | 66.04 | 66.79 | **62.93** | **61.92** | **62.43** |
| **Far** | AUC↑ | 84.17 | 75.02 | 79.59 | 89.20 | 89.07 | 89.13 | 90.64 | 90.72 | 90.68 | 90.99 | 90.46 | 90.72 | **91.52** | **90.74** | **91.13** |
| | FPR↓ | 73.52 | 86.65 | 80.09 | 38.51 | 39.37 | 38.94 | 33.97 | 34.98 | 34.47 | 32.68 | 36.34 | 34.51 | **31.74** | **34.79** | **33.26** |
| **Avg.** | AUC↑ | 79.29 | 73.79 | 76.54 | 82.92 | 82.29 | 82.60 | 84.55 | 85.25 | 84.90 | 85.22 | 85.43 | 85.32 | **86.85** | **86.59** | **86.72** |
| | FPR↓ | 78.72 | 86.39 | 82.56 | 52.32 | 53.44 | 52.88 | 48.12 | 47.93 | 48.02 | 46.63 | 48.22 | 47.42 | **44.22** | **45.64** | **44.93** |

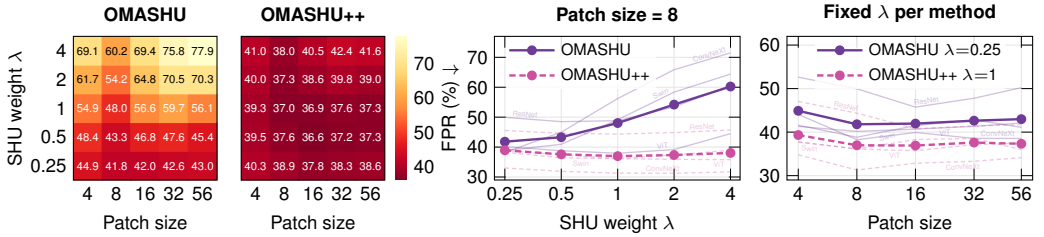



Figure 5: **Hyperparameter sensitivity of OMASHU(++).** *(left)* Heatmaps for different combinations of patch size (shuffle) and $\lambda$, averaged across all models and datasets. *(right)* Sensitivity of each hyperparameter when the other is fixed. We report FPR (in %, ↓). Model averages in bold.

we progressively sharpen the focus on leading directions via logit norm ($\|\mathbf{1}\|$, Wei et al., 2022) and Gram norm ($\|W^\top \mathbf{1}\|$). Although logit and Gram norms improve accuracy, OMA's normalization ($\|W^\top \mathbf{1}^+\|$) consistently outperforms all alternatives. Compared to the Gram norm, it reduces FPR by 2.49 pp on average across both model families and near- and far-OOD datasets.

**Hyperparameter sensitivity analysis.** Figure 5 analyzes the sensitivity of OMASHU and OMASHU++ to their two hyperparameters: the SHU weight $\lambda$, balancing the contributions of OMA and SHU, and the patch size, controlling how many local cues are preserved under SHU's patch shuffling. When varying both (*left*), OMASHU++ is highly robust, reaching state-of-the-art accuracy in 20 of 25 hyperparameter configurations. OMASHU remains stable for $\lambda \in \{0.25, 0.5\}$ and patch sizes of 8 or 16. For a fixed patch size of 8 (*right*), OMASHU++ is highly robust to $\lambda$, whereas OMASHU requires a low $\lambda$, down-weighting SHU. These trends largely hold across all models, while convolutional networks are more sensitive. We tune both on the OpenOOD validation set, selecting the patch size that yields the strongest SHU(++). For OMASHU(++), $\lambda = 0.50, 0.25, 0.25, 0.25\,/\,(2.00, 1.00, 0.50, 0.50)$ for ViT-B/16, Swin-T, ConvNeXt-B, and ResNet-50 V2, respectively. The patch size is 16 for ResNet-50 V2 and 8 for all other models.

## 6 Conclusion

The accuracy of OOD detection methods is bound by their premises about how ID & OOD samples behave. Our analysis of spectral projection profiles reveals how this behavior varies with model well-behavedness. In well-behaved models, OOD samples mainly occupy the leading directions of the profile, which relate to local texture cues, whereas ID samples also project onto the trailing spectral directions, which capture global structure. Building on this, the first component of our score, OMA, recovers the discriminative power of logits in well-behaved models by suppressing the contribution of the leading directions. The second component, SHU, measures how the trailing projections change under patch shuffling, which destroys global structure. Combining both, OMASHU(++) achieves state-of-the-art OOD detection accuracy among methods without and with training data, respectively, for modern, well-behaved transformer and convolutional models.

### AI USE STATEMENT

In this work, we used generative AI tools to obtain feedback on methodology, experiments, and interpretation of results, in the form of a critical review of a paper draft. We have *not* used generative AI tools to generate synthetic datasets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in writing proofs, propose or refine hypotheses, implement methods, assist with translation, clean and reformat datasets, or support qualitative and thematic data analysis; the remaining tasks with required disclosure are not applicable to this work. Additionally, we used generative AI tools in preparing the manuscript. Based on our drafts and instructions, they edited the text for readability, conciseness, and consistent terminology; drafted revised versions of parts of the paper; proofread the manuscript; suggested references; created and modified scientific figures, including the layout and labels of existing figures; and edited software code. The research ideas, methods, and experimental results are the authors' own. We have reviewed all AI-assisted work: we as authors revised all AI-assisted text and checked its statements against our results and the cited works, and we critically assessed the feedback and decided ourselves which points to address. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

### ETHICS STATEMENT

We believe that OOD detection or our proposed analysis and method generally *do not* pose major ethical concerns. For completeness, we would like to point out a few potential risks and highlight positive opportunities. While OOD detection can help increase the reliability of AI systems by flagging OOD samples, overreliance on such detection methods could also cause harm. For example, if a method incorrectly flags an OOD input as ID, blindly relying on the detection result could create false confidence. Similarly, when ID samples are incorrectly classified as OOD, an AI system could unnecessarily refrain from processing the input. Further, the notion of being "out-of-distribution" is somewhat ill-defined. While our experiments rely on established benchmarks, in real-world applications, whether a sample is considered in- or out-of-distribution depends on the task at hand. For instance, in some applications, inputs associated with underrepresented demographic groups may be classified as OOD despite being legitimate members of the intended population, *i.e.*, they are ID. Finally, while our analysis advances the understanding of OOD detection methods and how they rely on models being less or more "well-behaved," such understanding can also be exploited for adversarial applications. *E.g.*, AI systems could potentially be deliberately manipulated to classify malicious samples as ID, such that they are processed by the system. In conclusion, our proposed method and other OOD detection methods should be applied with a clear understanding of their limitations and without relying on them blindly. At the same time, when applied with the necessary caution, these approaches can help to significantly improve the reliability of real-world AI systems.

### REPRODUCIBILITY STATEMENT

We have made various efforts to ensure the reproducibility of our work. First, we follow established practices for evaluating OOD detection algorithms by using the OpenOOD benchmark (Zhang et al., 2024b) and associated evaluation protocols. Second, we use open models from TorchVision (TorchVision maintainers and contributors, 2016), as described in Section 5. Third, details for implementing our method are provided in Section 4 and Appendix A. Fourth, we will publish the code to reproduce the main experiments upon acceptance of the paper.

## REFERENCES

Yong Hyun Ahn, Gyeong-Moon Park, and Seong Tae Kim. LINe: Out-of-distribution detection by leveraging important neurons. In *CVPR*, pp. 19852–19862, 2023.

Sima Behpour, Thang Long Doan, Xin Li, Wenbin He, Liang Gou, and Liu Ren. GradOrth: A simple yet efficient out-of-distribution detection with orthogonal projection of gradients. In *NeurIPS*, volume 36, pp. 38206–38230, 2023.

Julian Bitterwolf, Maximilian Müller, and Matthias Hein. In or out? Fixing ImageNet out-of-distribution detection evaluation. In *ICML*, volume 202, pp. 2471–2506, 2023.

Robin Chan, Matthias Rottmann, and Hanno Gottschalk. Entropy maximization and meta classification for out-of-distribution detection in semantic segmentation. In *ICCV*, pp. 5108–5117, 2021.

Jinggang Chen, Junjie Li, Xiaoyang Qu, Jianzong Wang, Jiguang Wan, and Jing Xiao. GAIA: Delving into gradient-based attribution abnormality for out-of-distribution detection. In *NeurIPS*, volume 36, pp. 79946–79958, 2023.

Wenxi Chen, Raymond A. Yeh, Shaoshuai Mou, and Yan Gu. Leveraging perturbation robustness to enhance out-of-distribution detection. In *CVPR*, pp. 4724–4733, 2025.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, pp. 3606–3613, 2014.

Matthew Cook, Alina Zare, and Paul D. Gader. Outlier detection through null space analysis of neural networks. In *ICML Workshop (UDL)*, 2020.

Yifan Ding, Xixi Liu, Jonas Unger, and Gabriel Eilertsen. Enhancing out-of-distribution detection with extended logit normalization. In *CVPR*, pp. 24823–24832, 2026.

Andrija Djurisic, Nebojsa Bozanic, Arjun Ashok, and Rosanne Liu. Extremely simple activation shaping for out-of-distribution detection. In *ICLR*, 2023.

Andrija Djurisic, Rosanne Liu, and Mladen Nikolić. Logit scaling for out-of-distribution detection. *Mach. Vis. Appl.*, 36(5):108, 2025.

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 $16 \times 16$ words: Transformers for image recognition at scale. In *ICLR*, 2021.

Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. VOS: Learning what you don't know by virtual outlier synthesis. In *ICLR*, 2022.

Xuefeng Du, Yiyou Sun, Xiaojin Zhu, and Yixuan Li. Dream the impossible: Outlier imagination with diffusion models. In *NeurIPS*, volume 36, pp. 60878–60901, 2023.

Kun Fang, Qinghua Tao, Kexin Lv, Mingzhen He, Xiaolin Huang, and Jie Yang. Kernel PCA for out-of-distribution detection. In *NeurIPS*, volume 37, pp. 134317–134344, 2024.

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 *ICLR*, 2019.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Xiaoyuan Guan, Zhouwu Liu, Wei-Shi Zheng, Yuren Zhou, and Ruixuan Wang. Revisit PCA-based technique for out-of-distribution detection. In *ICCV*, pp. 19374–19382, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pp. 770–778, 2016.

Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

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

Dan Hendrycks, Mantas Mazeika, and Thomas G. Dietterich. Deep anomaly detection with outlier exposure. In *ICLR*, 2019.

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. In *ICCV*, pp. 8320–8329, 2021.

Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joseph Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. In *ICML*, volume 162, pp. 8759–8773, 2022.

Robin Hesse, Doğukan Bağcı, Bernt Schiele, Simone Schaub-Meyer, and Stefan Roth. Beyond accuracy: What matters in designing well-behaved image classification models? *Trans. Mach. Learn. Res.*, 2026.

Claus Hofmann, Simon Schmid, Bernhard Lehner, Daniel Klotz, and Sepp Hochreiter. Energy-based Hopfield boosting for out-of-distribution detection. In *NeurIPS*, volume 37, pp. 131859–131919, 2024.

Rui Huang and Yixuan Li. MOS: Towards scaling out-of-distribution detection for large semantic space. In *CVPR*, pp. 8706–8715, 2021.

Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. In *NeurIPS*, volume 34, pp. 677–689, 2021.

Xiaowei Huang, Daniel Kroening, Wenjie Ruan, James Sharp, Youcheng Sun, Emese Thamo, Min Wu, and Xinping Yi. A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Comput. Sci. Rev.*, 37:100270, 2020.

Wenyu Jiang, Hao Cheng, Mingcai Chen, Chongjun Wang, and Hongxin Wei. DOS: Diverse outlier sampling for out-of-distribution detection. In *ICLR*, 2024.

Gerhard Krumpl, Henning Avenhaus, and Horst Possegger. One model, many behaviors: Training-induced effects on out-of-distribution detection. In *WACV*, pp. 4128–4138, 2026.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*, volume 31, pp. 7167–7177, 2018.

Hengzhuang Li and Teng Zhang. Outlier synthesis via Hamiltonian Monte Carlo for out-of-distribution detection. In *ICLR*, 2025.

Zhiwei Ling, Yachen Chang, Hailiang Zhao, Xinkui Zhao, Kingsum Chow, and Shuiguang Deng. CADRef: Robust out-of-distribution detection via class-aware decoupled relative feature leveraging. In *CVPR*, pp. 4968–4977, 2025.

Litian Liu and Yao Qin. Fast decision boundary based out-of-distribution detector. In *ICML*, volume 235, pp. 31728–31746, 2024.

Litian Liu and Yao Qin. Detecting out-of-distribution through the lens of neural collapse. In *CVPR*, pp. 15424–15433, 2025.

Weitang Liu, Xiaoyun Wang, John D. Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *NeurIPS*, volume 33, pp. 21464–21475, 2020.

Xixi Liu, Yaroslava Lochman, and Christopher Zach. GEN: Pushing the limits of softmax-based out-of-distribution detection. In *CVPR*, pp. 23946–23955, 2023.

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 *ICCV*, pp. 9992–10002, 2021.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *CVPR*, pp. 11966–11976, 2022.

Haodong Lu, Dong Gong, Shuo Wang, Jason Xue, Lina Yao, and Kristen Moore. Learning with mixture of prototypes for out-of-distribution detection. In *ICLR*, 2024.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Yifei Ming, Yiyou Sun, Ousmane Dia, and Yixuan Li. How to exploit hyperspherical embeddings for out-of-distribution detection? In *ICLR*, 2023.

Maximilian Müller and Matthias Hein. Mahalanobis++: Improving OOD detection via feature normalization. In *ICML*, volume 267, pp. 45151–45184, 2025.

Ahyoung Oh, Wonseok Shin, and Songkuk Kim. Sparsity as a key: Unlocking new insights from latent structures for out-of-distribution detection. In *CVPR*, pp. 19298–19307, 2026.

Jaewoo Park, Yoon Gyo Jung, and Andrew Beng Jin Teoh. Nearest neighbor guidance for out-of-distribution detection. In *ICCV*, pp. 1686–1695, 2023.

Amey P. Pasarkar and Adji Bousso Dieng. Vendi novelty scores for out-of-distribution detection. *arXiv:2602.10062 [cs.LG]*, 2026.

Sudarshan Regmi. AdaSCALE: Adaptive scaling for OOD detection. In *ICML*, 2026.

Jie Ren, Stanislav Fort, Jeremiah Z. Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. A simple fix to Mahalanobis distance for improving near-OOD detection. In *ICML Workshop (UDL)*, 2021.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vision*, 115(3):211–252, 2015.

Chandramouli Shama Sastry and Sageev Oore. Detecting out-of-distribution examples with Gram matrices. In *ICML*, volume 119, pp. 8491–8501, 2020.

AQM Sazzad Sayyed, Rifat Shahriar, Milin Zhang, Ananthram Swami, Michael De Lucia, Nathaniel Bastian, and Francesco Restuccia. Out-of-distribution detection in computer vision: A comprehensive survey and research challenges. *ACM Comput. Surv.*, 58(14):364:1–364:37, 2026.

Vikash Sehwag, Mung Chiang, and Prateek Mittal. SSD: A unified framework for self-supervised outlier detection. In *ICLR*, 2021.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

Yue Song, Wei Wang, and Nicu Sebe. RankFeat&RankWeight: Rank-1 feature/weight removal for out-of-distribution detection. *IEEE T. Pattern Anal. Mach. Intell.*, 47(4):2505–2519, 2025.

Yiyou Sun and Yixuan Li. DICE: Leveraging sparsification for out-of-distribution detection. In *ECCV*, volume 13684, pp. 691–708, 2022.

Yiyou Sun, Chuan Guo, and Yixuan Li. ReAct: Out-of-distribution detection with rectified activations. In *NeurIPS*, volume 34, pp. 144–157, 2021.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *ICML*, volume 162, pp. 20827–20840, 2022.

Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *ICML*, volume 97, pp. 6105–6114, 2019.

Leitian Tao, Xuefeng Du, Xiaojin Zhu, and Yixuan Li. Non-parametric outlier synthesis. In *ICLR*, 2023.

TorchVision maintainers and contributors. TorchVision: PyTorch's computer vision library, 2016. URL `https://github.com/pytorch/vision`.

Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. MaxViT: Multi-axis vision transformer. In *ECCV*, volume 13684, pp. 459–479, 2022.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alexander Shepard, Hartwig Adam, Pietro Perona, and Serge J. Belongie. The iNaturalist species classification and detection dataset. In *CVPR*, pp. 8769–8778, 2018.

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

Vasilis Vryniotis. How to train state-of-the-art models using TorchVision's latest primitives. PyTorch Blog, 2021. URL https://pytorch.org/blog/how-to-train-state-of-the-art-models-using-torchvision-latest-primitives/.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *NeurIPS*, pp. 10506–10518, 2019.

Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. ViM: Out-of-distribution with virtual-logit matching. In *CVPR*, pp. 4911–4920, 2022.

Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating neural network overconfidence with logit normalization. In *ICML*, volume 162, pp. 23631–23644, 2022.

Ross Wightman, Hugo Touvron, and Hervé Jégou. ResNet strikes back: An improved training procedure in timm. *arXiv:2110.00476 [cs.CV]*, 2021.

Kai Xu, Rongyu Chen, Gianni Franchi, and Angela Yao. Scaling for training time and post-hoc out-of-distribution detection enhancement. In *ICLR*, 2024.

Mingyu Xu, Zheng Lian, Bin Liu, and Jianhua Tao. VRA: Variational rectified activation for out-of-distribution detection. In *NeurIPS*, volume 36, pp. 28941–28959, 2023.

Yue Yuan, Rundong He, Yicong Dong, Zhongyi Han, and Yilong Yin. Discriminability-driven channel selection for out-of-distribution detection. In *CVPR*, pp. 26171–26180, 2024.

Boxuan Zhang, Jianing Zhu, Zengmao Wang, Tongliang Liu, Bo Du, and Bo Han. What if the input is expanded in OOD detection? In *NeurIPS*, volume 37, pp. 21289–21329, 2024a.

Jingyang Zhang, Jingkang Yang, Pengyun Wang, Haoqi Wang, Yueqian Lin, Haoran Zhang, Yiyou Sun, Xuefeng Du, Yixuan Li, Ziwei Liu, Yiran Chen, and Hai Li. OpenOOD v1.5: Enhanced benchmark for out-of-distribution detection. *J. Data-centric Mach. Learn. Res.*, 2:1–32, 2024b.

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 *ICLR*, 2023.

Zihan Zhang and Xiang Xiang. Decoupling MaxLogit for out-of-distribution detection. In *CVPR*, pp. 3388–3397, 2023.

Yao Zhu, Yuefeng Chen, Chuanlong Xie, Xiaodan Li, Rong Zhang, Hui Xue, Xiang Tian, Bolun Zheng, and Yaowu Chen. Boosting out-of-distribution detection with typical features. In *NeurIPS*, volume 35, pp. 20758–20769, 2022.

Baris Zöngür, Robin Hesse, and Stefan Roth. Activation subspaces for out-of-distribution detection. In *ICCV*, pp. 3509–3519, 2025.

In this appendix, we provide further details that complement the main paper. Appendix A provides additional experiments that extend our analysis from Section 3, additional comparisons of OMASHU(++) complementing Section 5, as well as expanded derivations of our approach. Appendix B gives a more in-depth discussion of the relation to existing OOD detection approaches and discusses the notion of well-behavedness in context. Finally, we discuss the limitations of OMASHU(++) in Appendix C.

# A Additional Experiments and Derivations

## A.1 Additional analyses

**ID–OOD difference with varying energy cutoffs.** Figure 6 shows the correlation between the ID–OOD spectral projection gap and well-behavedness for different energy cutoffs (1 %, 5 %, 10 %, 50 %) beyond the 25 % cutoff in Figure 1. We observe high Pearson's correlation for 1 % and 50 %. For 5 % and 10 %, the gap measured in MaxViT-T is the main driver of the low correlation. When we omit MaxViT-T, we observe strong correlation across all energy cutoffs. We attribute MaxViT-T's behavior to a combination of unusually high variance in the first few directions of the spectral profile and the cutoff occurring after very few directions as a consequence of the high variance (cutoff after 3 and 5 dimensions for 5 % and 10 %, respectively). In other words, the first few directions carry unusually high spectral energy. Combined with high variation, this results in an outlier projection gap. Our selection of 25 % in Figure 1 avoids this regime; a sufficiently high cutoff normalizes the high variance in the first few dimensions for outliers.

Additionally, we observe similar trends in Pearson's correlation across all energy cutoffs when we omit shape bias in the well-behavedness calculation and consider only classification accuracy and robustness measures. Omitting shape bias yields $r = 0.79, -0.12, 0.10, 0.74, 0.80$ for energy cutoffs 1 %, 5 %, 10 %, 25 %, 50 %, respectively. When MaxViT-T is also omitted, the corresponding correlations without shape bias are $r = 0.85, 0.80, 0.85, 0.81, 0.73$.

**Additional visualizations.** Figure 7 shows spectral projection profiles of ID and OOD inputs for all 11 networks in our study. Consistent with Section 3, the ID–OOD difference grows with well-behavedness: the four least well-behaved networks (*(a)–(d)*) show little difference, whereas in the remaining networks, the ID profiles extend markedly further into the trailing directions than the OOD profiles. Figure 8 shows the profiles of patch-shuffled ID and OOD images for ViT-B/16, Swin-T, ConvNeXt-B, and ResNet-50 V2. Compared with the unshuffled profiles in Figure 7, shuffling changes the ID profiles more than the OOD profiles, making shuffled ID and OOD profiles more similar.

## A.2 Additional results

**Results on less well-behaved models.** Table 3 compares the OOD detection accuracy of OMASHU and OMASHU++ for less (ResNet-50 V1 & MobileNetV2) and more (ViT-B/16 & Swin-T & ResNet-50 V2 & ConvNeXt-B) well-behaved models. We follow the experimental setup in Table 1 and use OpenOOD datasets where the ID data is from ImageNet-1k. Unlike Table 1, we use the SCALE variant of ActSub, which yields higher overall OOD detection accuracy across less and more well-behaved networks. Recall that the ID–OOD projection gap for less well-behaved models is negligible. As for most OOD detection methods, the accuracy of OMASHU and OMASHU++ is inherently tied to their underlying assumptions holding. As discussed in Section 3, OMASHU and OMASHU++ are designed to address the distinctive behavior of well-behaved models. Like OMASHU(++), Mahalanobis variants (MDS, RMDS, MDS++, RMDS++) show subpar OOD detection accuracy, even though they otherwise perform well on well-behaved models. In contrast, methods that excel on less well-behaved models, such as ActSub and AdaSCALE, show subpar OOD detection accuracy on well-behaved models that violate their premise. While these results directly parallel the claims of Section 3, they also show that the distinctive behavior of different models calls for OOD detection methods that account for how specific models shape their representation space. While our analysis in Section 3 applies broadly, our proposed OOD detection approach is designed to address the requirements of modern, well-behaved models, which are in most widespread use today.

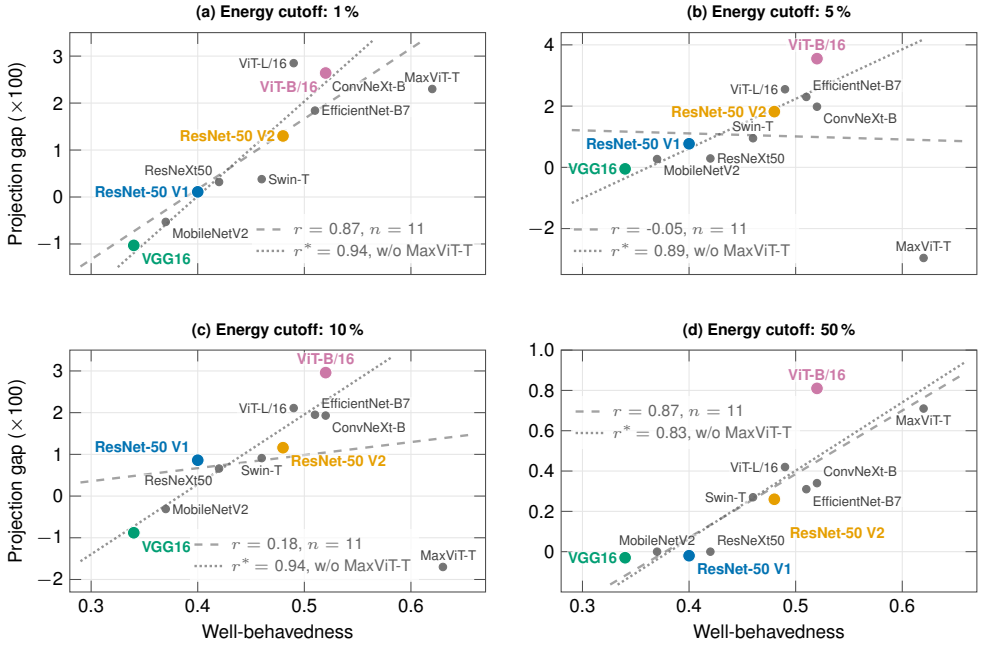



Figure 6: **Projection gap *vs.* well-behavedness correlation for different energy cutoffs.** Extending Figure 1, we visualize the correlation between the projection gap and well-behavedness for different energy cutoffs (*i.e.*, the cutoff of $\sum_{i=1}^{k} \sigma_i^2$). In particular, we use 1 %, 5 %, 10 %, and 50 % cutoffs and report $r$ and $r^*$ with and without MaxViT-T, respectively.

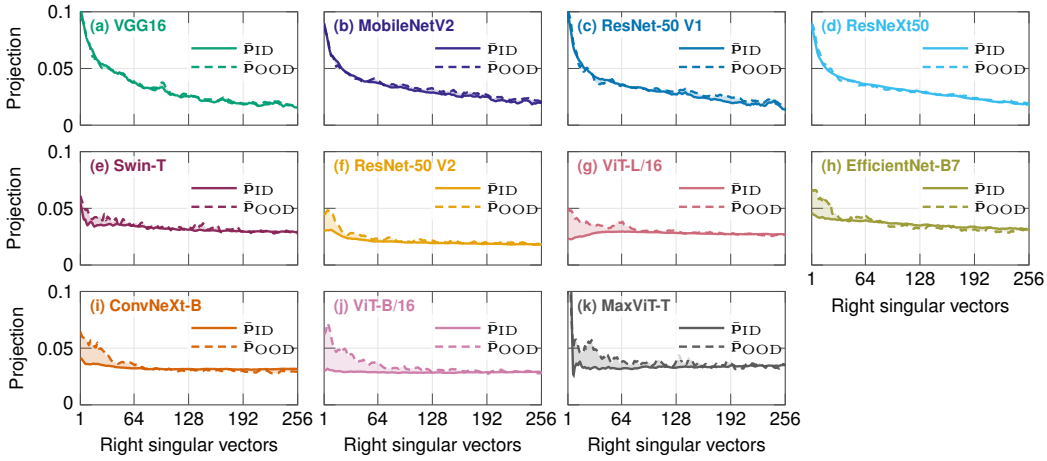



Figure 7: **Spectral projection profiles.** *(a)–(k):* Mean normalized spectral projection profiles of ID (solid) and OOD (dashed) data, shown for the first 256 right singular vectors; shaded areas show the ID–OOD difference. For ID and OOD data, we use ImageNet-1k (Russakovsky et al., 2015) and OpenImage-O (Wang et al., 2022), respectively. These profiles extend the results in Figure 1 of the main paper, visualizing *all* eleven models.

**Per-dataset results for each model.** We provide an extended version of the results reported in Table 1, following the same experimental setup as in Section 5. We report OOD detection accuracy for ViT-B/16 (Dosovitskiy et al., 2021) in Table 5, Swin-T (Liu et al., 2021) in Table 6, ConvNeXt-B (Liu et al., 2022) in Table 7, and ResNet-50 V2 (Vryniotis, 2021) in Table 8. Each table reports AUC (in %, ↑) and FPR (in %, ↓, @95 %) separately for the five OpenOOD (Zhang et al., 2024b) datasets: SSB-Hard (Vaze et al., 2022) and NINCO (Bitterwolf et al., 2023) for near-OOD, and iNaturalist

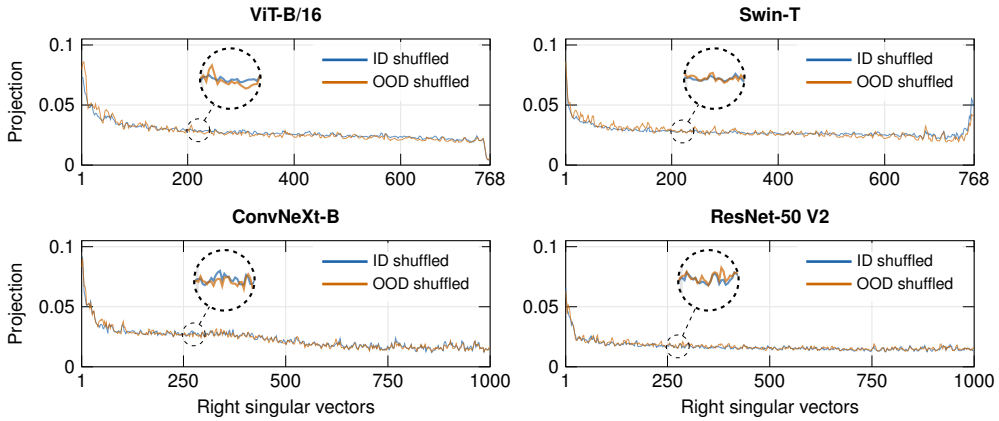



Figure 8: **Mean normalized spectral projection profiles.** Extending Figure 2, we provide additional projection profiles of shuffled ID and OOD samples. We report profiles for ViT-B/16, Swin-T, ConvNeXt-B, and ResNet-50 V2; dashed circles magnify the boxed regions. For all models, projection profiles of shuffled ID samples closely resemble those of shuffled OOD samples, supporting our analysis.

Table 3: **OOD detection accuracies for less & more well-behaved models.** Near- and far-OOD detection accuracy using AUC (in %, ↑) and FPR (in %, ↓, @95 %) on the OpenOOD datasets with ImageNet-1k as ID on less well-behaved (ResNet-50 V1 & MobileNetV2) and more well-behaved models (ViT-B/16 & Swin-T & ResNet-50 V2 & ConvNeXt-B). We report results for existing approaches and ours, w/ & w/o access to training data at inference time. Avg. denotes the average results over all datasets. † indicates results taken from the original papers; other results are reproduced. Best results **bold** and highlighted; second best underlined and highlighted.

| | | Less well-behaved | | | | | | More well-behaved | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Near | | Far | | Avg. | | Near | | Far | | Avg. | |
| | **Method** | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ |
| **Without training data** | MSP | 74.78 | 67.07 | 84.26 | 52.93 | 80.47 | 58.58 | 75.27 | 77.32 | 84.04 | 61.15 | 80.53 | 67.62 |
| | Energy | 73.56 | 71.24 | 89.09 | 38.65 | 82.88 | 51.68 | 63.72 | 89.21 | 66.08 | 86.83 | 65.13 | 87.78 |
| | MaxLogit | 74.23 | 70.43 | 89.12 | 38.37 | 83.17 | 51.19 | 71.96 | 86.26 | 79.60 | 80.09 | 76.54 | 82.56 |
| | ReAct | 74.53 | 70.39 | 92.68 | 28.98 | 85.42 | 45.54 | 64.47 | 80.45 | 72.96 | 63.30 | 69.56 | 70.16 |
| | GEN | 74.05 | 69.59 | 89.24 | 37.26 | 83.16 | 50.19 | 78.63 | 65.69 | 90.36 | 38.05 | 85.67 | 49.10 |
| | ASH | 76.49 | 66.46 | 95.12 | 21.77 | 87.67 | 39.64 | 41.70 | 94.94 | 34.84 | 95.02 | 37.58 | 94.99 |
| | SCALE | 78.12 | 64.57 | 95.64 | 19.97 | 88.63 | 37.81 | 64.98 | 86.41 | 71.99 | 82.94 | 69.18 | 84.33 |
| | LTS | 78.05 | 64.39 | 95.58 | 20.20 | 88.57 | 37.87 | 65.13 | 87.54 | 69.26 | 79.04 | 67.61 | 82.44 |
| | PRO | 74.59 | 67.52 | 88.34 | 37.69 | 82.84 | 49.62 | 78.67 | 64.67 | 90.25 | 37.32 | 85.62 | 48.26 |
| | AdaSCALE | **80.27** | **60.02** | **95.96** | **17.88** | **89.68** | **34.74** | 71.13 | 76.71 | 87.02 | 43.62 | 80.66 | 56.86 |
| | OMASHU | 65.70 | 81.06 | 75.09 | 71.51 | 71.33 | 75.33 | **80.22** | **63.18** | **94.72** | **24.30** | **88.92** | **39.85** |
| **With training data** | MDS | 52.75 | 90.13 | 63.45 | 78.40 | 59.17 | 83.09 | 77.64 | 67.07 | 92.69 | 28.19 | 86.67 | 43.74 |
| | RMDS | 75.22 | 68.49 | 83.92 | 48.09 | 80.44 | 56.25 | 80.12 | 64.71 | 91.39 | 31.19 | 86.88 | 44.59 |
| | KNN | 67.68 | 74.00 | 87.30 | 39.27 | 79.45 | 53.16 | 73.64 | 70.67 | 90.43 | 32.36 | 83.72 | 47.68 |
| | ViM | 70.73 | 73.09 | 90.16 | 29.22 | 82.39 | 46.77 | 73.88 | 77.00 | 93.09 | 26.91 | 85.41 | 46.95 |
| | fDBD | 75.37 | 66.65 | 91.95 | 28.68 | 85.31 | 43.87 | 72.59 | 73.55 | 89.37 | 37.86 | 82.66 | 52.13 |
| | NCI | 74.89 | 68.08 | 93.94 | 25.17 | 86.32 | 42.33 | 73.90 | 73.19 | 90.30 | 37.14 | 83.74 | 51.56 |
| | CADRef | 77.55 | 64.67 | 95.32 | 20.85 | 88.21 | 38.38 | 69.29 | 71.56 | 82.45 | 41.03 | 77.19 | 53.24 |
| | ActSub | **80.70** | **58.70** | **96.35** | **16.61** | **90.09** | **33.45** | 75.07 | 66.62 | 85.76 | 40.61 | 81.48 | 51.02 |
| | MDS++ | 69.90 | 74.74 | 90.28 | 33.51 | 82.13 | 50.00 | 81.18 | 64.48 | 93.30 | 24.15 | 88.45 | 40.28 |
| | RMDS++ | 76.40 | 69.54 | 88.31 | 47.09 | 83.54 | 56.07 | 81.98 | 63.09 | 90.92 | 29.33 | 87.34 | 42.83 |
| | **OMASHU++** *(Ours)* | 75.02 | 69.19 | 89.85 | 40.42 | 83.92 | 51.93 | **83.28** | **57.96** | **95.33** | **20.92** | **90.52** | **35.74** |

(Van Horn et al., 2018), Textures (Cimpoi et al., 2014), and OpenImage-O (Wang et al., 2022) for far-OOD. We additionally report the averages across near- and far-OOD datasets.

Across all networks, both OMASHU and OMASHU++ outperform their respective baselines in far-OOD detection and, for ViT-B/16 and ConvNeXt-B, in near-OOD detection. On Swin-T, OMASHU outperforms the methods without training data also for near-OOD detection. Finally, for near-OOD detection on ResNet-50 V2, OMASHU achieves the best AUC and OMASHU++ the best FPR within their respective groups. These results parallel the observations in Section 5.

### A.3 Expanded derivations

For completeness, we derive the identities for the logit norm and the Gram norm of Equation (5) in Section 4. Recall that we applied a singular value decomposition $W = U\Sigma V^\top$ to the classifier weights, where $\Sigma$ is diagonal and $U \in \mathbb{R}^{C\times r}$ and $V \in \mathbb{R}^{d\times r}$ have orthonormal columns. This implies that $U^\top U = V^\top V = I_r$, where $I_r$ is an identity matrix of $r$ dimensions. Also recall that multiplying a vector by a matrix $A$ with orthonormal columns preserves its norm, since $\|A\mathbf{y}\|^2 = \mathbf{y}^\top A^\top A\,\mathbf{y} = \|\mathbf{y}\|^2$. Note that this requires only $A^\top A = I_r$ and thus also holds for $V$ when $d > C$, even if $VV^\top \neq I_d$ in this case.

For the logit norm, we have:

$$\|W\mathbf{z}\| = \|U\Sigma V^\top \mathbf{z}\| \tag{11a}$$
$$= \|\Sigma V^\top \mathbf{z}\| \tag{11b}$$
$$= \|\Sigma |V^\top \mathbf{z}|\| \tag{11c}$$
$$= \|\Sigma \mathbf{p}\|. \tag{11d}$$

After substituting the SVD of $W$, we can drop $U$, since its orthonormal columns leave the norm unchanged. Because $\Sigma$ is diagonal, the squared norm $\|\Sigma\mathbf{y}\|^2 = \sum_i \sigma_i^2 y_i^2$ depends only on the magnitudes $|y_i|$, so taking element-wise absolute values does not change it either. What remains is the spectral projection profile $\mathbf{p} = |V^\top \mathbf{z}|$ from Equation (2), weighted by the singular values.

Next, we consider the Gram norm:

$$\left\|\Sigma^2 \mathbf{p}\right\| = \left\|\Sigma^2 V^\top \mathbf{z}\right\| \tag{12a}$$
$$= \left\|\Sigma U^\top U \Sigma V^\top \mathbf{z}\right\| \tag{12b}$$
$$= \left\|V\Sigma U^\top U \Sigma V^\top \mathbf{z}\right\| \tag{12c}$$
$$= \left\|W^\top W \mathbf{z}\right\| \tag{12d}$$
$$= \left\|W^\top \mathbf{l}\right\|. \tag{12e}$$

Here, we proceed in the opposite direction. As before, the absolute values in $\mathbf{p}$ can be dropped, since $\Sigma^2$ is diagonal. We then insert $U^\top U = I_r$ between the two factors of $\Sigma$ and multiply from the left by $V$, which again leaves the norm unchanged. The resulting product $V\Sigma U^\top U \Sigma V^\top$ is just $W^\top W$, and with $\mathbf{l} = W\mathbf{z}$, we obtain the Gram norm.

## B Further discussion on related work

### B.1 Relation to existing approaches

**ActSub and OMASHU(++).** ActSub (Zöngür et al., 2025) and OMASHU(++) both use the classifier spectrum but rely on different ID–OOD behavior. ActSub expects higher ID confidence after shaping leading features and greater similarity to training features in trailing directions. Similar to methods that use logit information, ActSub also assumes logit as a reliable confidence measure, which we show is violated in well-behaved networks. OMASHU(++) instead targets the behavior observed in well-behaved networks: OOD samples concentrate on leading directions, while ID trailing projections depend more on global structure. OMA compensates for this concentration through logit normalization; SHU measures the sensitivity to global structure through patch shuffling. This distinction appears in Table 3. ActSub excels in less well-behaved networks but struggles in well-behaved ones. OMASHU++ instead excels in well-behaved, modern models.

**MDS++ and OMASHU(++).** MDS++ (Müller & Hein, 2025) assumes samples are Gaussian-distributed around class centroids. Further, whitening implies that most of the feature spectrum carries an informative signal. Whitening increases variance along directions that initially have low variance. Thus, MDS++ has better OOD detection accuracy in well-behaved models, where ID projection is distributed more uniformly on the right singular spectrum. In less well-behaved models, the spectrum that the ID inputs occupy collapses to a tight subspace. This overclustering mitigates the discriminative signal from the trailing directions that MDS++ expects. Especially in less well-behaved models, ActSub shows that the trailing directions encode information unrelated to class

Table 4: **OMA normalization extension.** We complement the analysis in Table 2 with RankWeight (Song et al., 2025) using the Energy score, as originally proposed. We extend RankWeight to RankWeight++ by normalizing the features and using MaxLogit as the base score, ensuring a fair comparison with input-dependent spectral normalization terms such as $\|\mathbf{1}\|$ and $\|W^\top \mathbf{1}\|$. We report near- and far-OOD detection accuracy using AUC (in %, ↑) and FPR (in %, ↓, @95 %) for ImageNet-1k transformer (ViT-B/16 & Swin-T) and convolutional models (ResNet-50 V2 & ConvNeXt-B). We report on OpenOOD datasets for when the ID is ImageNet-1k. Avg. denotes the average results over all datasets. Positive pooling over logits (OMA) scores consistently better. Best results **bold** and highlighted; second best underlined and highlighted.

| | **Transformer Models** | | | | | | **Convolutional Models** | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Near | | Far | | Avg. | | Near | | Far | | Avg. | |
| **Normalization** | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ |
| RankWeight | 69.42 | 87.32 | 81.41 | 78.19 | 76.61 | 81.84 | 62.03 | 89.16 | 56.75 | 91.55 | 58.86 | 90.59 |
| RankWeight++ | 73.66 | 76.66 | 84.70 | 48.31 | 80.28 | 59.65 | 73.08 | 73.32 | 89.84 | 37.28 | 83.13 | 51.70 |
| $\|\mathbf{z}\|$ | 73.49 | 73.04 | 89.20 | 38.51 | 82.92 | 52.32 | 72.11 | 74.53 | 89.07 | 39.37 | 82.29 | 53.44 |
| $\|\mathbf{1}\|$ | 75.42 | 69.35 | 90.64 | 33.97 | 84.55 | 48.12 | 77.05 | 67.35 | 90.72 | 34.98 | 85.25 | 47.93 |
| $\|W^\top \mathbf{1}\|$ | 76.56 | 67.55 | 90.99 | 32.68 | 85.22 | 46.63 | 77.88 | 66.04 | 90.46 | 36.34 | 85.43 | 48.22 |
| $\|W^\top \mathbf{1}^+\|$ **(OMA)** | **79.86** | **62.93** | **91.52** | **31.74** | **86.85** | **44.22** | **80.38** | **61.92** | **90.74** | **34.79** | **86.59** | **45.64** |

discrimination, violating MDS++'s assumption. Although they are inherently different approaches, MDS++ and OMASHU(++) share a similar assumption of expected ID–OOD input behavior. This is shown in Table 3: both achieve high OOD detection accuracy in more well-behaved models and lower accuracy in the other. Our OMASHU++ outperforms MDS++ across both model families.

**LogitNorm, RankWeight, and OMA.** Let us contrast two other related approaches: Applying LogitNorm (Wei et al., 2022) post-hoc already introduces spectral weighting. OMA sharpens this weighting toward the leading directions and restricts the normalizer to the positive logits. As Table 2 shows, both steps improve OOD detection. The normalizer depends on each sample's spectral projections. RankWeight (Song et al., 2025) instead removes only the first singular component from a backbone weight matrix, using the same modification for every sample.

We extend our previous analysis in Table 2 by comparing RankWeight to dynamic spectral normalization scores such as logit normalization or OMA in Table 4. We evaluate the originally proposed RankWeight with Energy and, alternatively, with MaxLogit scores on normalized features. The latter controls feature magnitude and uses the same base score as OMA. We denote this as RankWeight++. RankWeight shows subpar OOD detection accuracy in both transformer and convolutional models. With feature normalization and MaxLogit (RankWeight++), accuracy does not improve beyond $\|\mathbf{z}\|$ for transformer networks, unlike with logit normalization or OMA. For convolutional networks, the gain remains limited compared with other normalization methods. These results demonstrate the effectiveness of our input-based spectral weighting over simply removing one singular direction as in RankWeight.

**Perturbation consistency and SHU.** SHU differs from existing perturbation-based approaches in two ways: (*i*) SHU applies a perturbation that distorts global cues, whereas existing approaches (Zhang et al., 2024a; Chen et al., 2025) distort local cues. (*ii*) Existing approaches assume ID consistency, and SHU utilizes the consistency of OOD samples in trailing directions under patch shuffling. To demonstrate the effectiveness of the SHU component, especially global patch perturbation, we compare OMASHU to OMA combined with CoVer (Zhang et al., 2024a) in Table 1. CoVer is a post-hoc perturbation consistency approach that combines multiple image perturbations of local texture cues. Combined with an existing OOD score, CoVer selects the perturbations that maximize ID–OOD separability using a validation set. We adapted CoVer to our evaluation setup by integrating a greedy search over 18 different types of local image perturbations for the validation set, each with 5 different intensity levels. We included each perturbation that improved OOD detection performance and excluded others during the greedy search. These local perturbations had minimal impact on OMA's performance, and even no perturbation increased the OOD detection accuracy for convolutional models. This observation parallels our claims regarding the complementary nature of OMA and SHU. Local-level perturbation mainly affects the leading directions, and OMA al-

ready captures this signal. In contrast, SHU provides information from the trailing directions using a global perturbation.

### B.2 Well-behavedness and OOD detection

Well-behavedness is central to our analysis of ID–OOD behavior across models. OOD detection in image classification can be viewed as identifying inputs whose visual cues do not match those learned for the model's ID classes. This motivates examining how reliably models use visual cues for classification and how their predictions respond when those cues change. Inspired by Hesse et al. (2026), we therefore adopt a definition of well-behavedness that focuses on robustness. Additionally, following Hesse et al., we include shape bias, since our spectral analysis (Section 3) relates feature directions to local texture and global structure. We omit broader notions of model quality, such as fairness (class balance), bias (object focus), and efficiency (number of parameters), as they are less relevant to OOD detection.

Let $A$, $R_{\text{adv}}$, $R_{\text{corr}}$, $R_{\text{OOD}}$, and $S$ denote a model's classification accuracy, adversarial robustness, corruption robustness, OOD robustness, and shape bias, respectively. We then formally define the well-behavedness of an image classification model as

$$\text{WB} = \frac{1}{5}\left(A + R_{\text{adv}} + R_{\text{corr}} + R_{\text{OOD}} + S\right). \tag{13}$$

Since all metrics are in a similar range, we omit a weighting for simplicity. Implementation details of each metric follow Hesse et al. (2026), and are briefly summarized below.

**Accuracy.** Accuracy ($A$) denotes the standard top-1 ImageNet-1k classification accuracy.

**Adversarial robustness.** Adversarial robustness ($R_{\text{adv}}$) measures a model's resistance to adversarial perturbations deliberately designed to cause incorrect predictions. We use the two popular adversarial attacks FGSM (Goodfellow et al., 2015) and PGD (Madry et al., 2018). They compute input gradients which are added to the original sample to maximally change the prediction. We measure the mean accuracy after applying the attacks relative to the clean classification accuracy.

**Corruption robustness.** Corruption robustness ($R_{\text{corr}}$) measures a model's ability to classify images affected by common corruptions, such as noise, blur, or contrast changes, as proposed in Hendrycks & Dietterich (2019). We again measure the mean accuracy after applying the corruptions relative to the clean classification accuracy.

**OOD robustness.** OOD robustness ($R_{\text{OOD}}$) measures a model's ability to generalize to images from visual domains that differ from its training data. We use commonly employed datasets – ImageNet-R (Hendrycks et al., 2021), ImageNet-Sketch (Wang et al., 2019), as well as Stylized-ImageNet, Edge, and Silhouette from Geirhos et al. (2019) – and again measure the mean accuracy on each dataset relative to the classification accuracy on the ImageNet-1k evaluation split. Please note that these datasets are only used for our analysis and not as part of our method OMASHU(++). Further, the notion of OOD in robustness differs from that of OOD detection used throughout the paper: in robustness, OOD refers to the same classes with different appearances, whereas in OOD detection, OOD refers to unseen classes during training.

**Shape bias.** Shape bias ($S$) characterizes whether a model relies more on object shape or texture for classification. It measures the fraction of predictions that follow the shape cue when shape and texture indicate different classes, considering predictions that match either cue (Geirhos et al., 2019).

## C Limitations

OMASHU and OMASHU++ leverage the distinctive spectral behavior of ID and OOD inputs in well-behaved networks, but this advantage diminishes in less well-behaved networks, where the separation is largely absent. We analyze the hyperparameter sensitivity and show that OMASHU and OMASHU++ are relatively robust to changes in $\lambda$ and patch size. Even so, both hyperparameters may vary depending on the backbone used. SHU requires an additional forward pass to calculate

representations for the shuffled inputs, which increases the inference cost. Finally, our evaluation is limited to image classification. Extending this evaluation to dense tasks such as semantic segmentation or more complex modalities could strengthen the scope of our methods.

Table 5: **OOD detection accuracies for ViT-B/16.** Near- and far-OOD detection accuracy using AUC (in %, ↑) and FPR (in %, ↓, @95 %) for ViT-B/16 (Dosovitskiy et al., 2021) (ImageNet-1k). We report results for existing approaches and OMASHU(++), with and without access to training data at inference time. We report results for all *five* OpenOOD datasets. † indicates results taken from the original papers; other results are reproduced. Best results **bold** and highlighted; second best underlined and highlighted. These results extend Table 1 of the main paper.

| | | Near-OOD | | | | | | Far-OOD | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SSB-HARD | | NINCO | | Near-Avg. | | iNaturalist | | Textures | | OpenImage-O | | Far-Avg. | |
| | **Method** | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ |
| **Without training data** | MSP | 68.94 | 86.41 | 78.11 | 77.35 | 73.53 | 81.88 | 88.19 | 42.42 | 85.06 | 56.44 | 84.87 | 56.11 | 86.04 | 51.66 |
| | Energy | 58.80 | 92.25 | 66.02 | 94.16 | 62.41 | 93.20 | 79.30 | 83.58 | 81.17 | 83.65 | 76.48 | 88.79 | 78.98 | 85.34 |
| | MaxLogit | 64.20 | 91.52 | 72.40 | 92.98 | 68.30 | 92.25 | 85.29 | 72.98 | 83.74 | 78.93 | 81.60 | 85.78 | 83.54 | 79.23 |
| | ReAct | 63.10 | 90.46 | 75.43 | 78.50 | 69.26 | 84.48 | 86.11 | 48.22 | 86.66 | 55.87 | 84.29 | 57.68 | 85.69 | 53.92 |
| | GEN | 70.09 | 82.24 | 82.51 | 59.31 | 76.30 | 70.78 | 93.54 | 22.94 | 90.23 | 38.31 | 90.27 | 35.43 | 91.35 | 32.23 |
| | ASH | 34.17 | 98.38 | 22.84 | 99.16 | 28.50 | 98.77 | 9.75 | 99.70 | 28.08 | 97.42 | 18.17 | 99.12 | 18.67 | 98.75 |
| | SCALE | 56.46 | 92.36 | 61.33 | 94.67 | 58.90 | 93.51 | 73.69 | 86.74 | 78.88 | 84.68 | 72.54 | 89.50 | 75.04 | 86.97 |
| | LTS | 64.35 | 91.40 | 75.48 | 82.62 | 69.92 | 87.01 | 88.92 | 50.43 | 86.53 | 61.75 | 83.45 | 69.26 | 86.30 | 60.48 |
| | PRO | 69.93 | 82.16 | 82.77 | 58.58 | 76.35 | 70.37 | 93.41 | 23.32 | 89.99 | 39.01 | 90.25 | 35.19 | 91.22 | 32.51 |
| | AdaSCALE | 67.29 | 86.26 | 80.37 | 63.64 | 73.83 | 74.95 | 89.90 | 35.90 | 86.96 | 53.28 | 87.68 | 44.19 | 88.18 | 44.46 |
| | **OMASHU** | **72.93** | **79.11** | **86.07** | **48.00** | **79.50** | **63.55** | **97.04** | **13.66** | **95.36** | **24.79** | **92.87** | **28.31** | **95.09** | **22.25** |
| **With training data** | MDS | 71.57 | 83.47 | 86.52 | 48.76 | 79.04 | 66.11 | 96.01 | 20.66 | 89.41 | 38.90 | 92.38 | 30.35 | 92.60 | 29.97 |
| | RMDS | 72.87 | 84.53 | 87.31 | 46.22 | 80.09 | 65.37 | 96.10 | 19.46 | 89.38 | 37.23 | 92.32 | 29.57 | 92.60 | 28.75 |
| | KNN | 65.97 | 86.22 | 82.25 | 54.73 | 74.11 | 70.47 | 91.46 | 27.74 | 91.12 | 33.23 | 89.86 | 34.82 | 90.81 | 31.93 |
| | ViM | 69.42 | 90.05 | 84.64 | 57.45 | 77.03 | 73.75 | 95.72 | 17.59 | 90.61 | 40.43 | 92.18 | 29.60 | 92.84 | 29.20 |
| | fDBD | 65.79 | 86.88 | 80.76 | 60.48 | 73.27 | 73.68 | 90.50 | 32.74 | 88.52 | 44.92 | 89.06 | 39.36 | 89.36 | 39.01 |
| | NCI | 66.61 | 85.42 | 80.83 | 61.27 | 73.72 | 73.35 | 90.19 | 34.64 | 87.90 | 48.71 | 88.54 | 41.56 | 88.88 | 41.64 |
| | CADRef | 71.10 | 83.57 | 85.17 | 49.05 | 78.14 | 66.31 | 93.84 | 21.10 | 91.35 | 30.95 | 91.56 | 28.33 | 92.25 | 26.79 |
| | ActSub | 70.79 | 80.81 | 84.32 | 49.20 | 77.56 | 65.01 | 91.69 | 23.73 | 88.86 | 38.01 | 89.44 | 31.83 | 89.99 | 31.19 |
| | MDS++ | 74.15 | 83.85 | 88.63 | 44.10 | 81.39 | 63.97 | 97.20 | **11.77** | 90.64 | 31.17 | 93.14 | **25.28** | 93.66 | 22.74 |
| | RMDS++ | 74.84 | 83.18 | **88.67** | 43.80 | 81.75 | 63.49 | 96.96 | 13.34 | 89.34 | 34.62 | 92.70 | 26.42 | 93.00 | 24.79 |
| | SAE† | 72.21 | 82.41 | 85.74 | 48.06 | 78.98 | 65.23 | 95.17 | 17.84 | 91.06 | 30.44 | 92.12 | 26.03 | 92.78 | 24.77 |
| | VNS† | 72.99 | 83.79 | 86.23 | 45.31 | 79.61 | 64.55 | 95.58 | 16.53 | 90.38 | 34.37 | 92.23 | 26.42 | 92.73 | 25.77 |
| | **OMASHU++** | **76.90** | **74.75** | 88.39 | **42.59** | **82.65** | **58.67** | **97.50** | 12.23 | **95.23** | **24.07** | **93.36** | 25.86 | **95.37** | **20.72** |

Table 6: **OOD detection accuracies for Swin-T.** Near- and far-OOD detection accuracy using AUC (in %, ↑) and FPR (in %, ↓, @95 %) for Swin-T (Liu et al., 2021) (ImageNet-1k). We report results for existing approaches and OMASHU(++), with and without access to training data at inference time. We report results for all *five* OpenOOD datasets. † indicates results taken from the original papers; other results are reproduced. Best results **bold** and highlighted; second best underlined and highlighted. These results extend Table 1 of the main paper.

| | | Near-OOD | | | | | | Far-OOD | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SSB-HARD | | NINCO | | Near-Avg. | | iNaturalist | | Textures | | OpenImage-O | | Far-Avg. | |
| | **Method** | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ |
| **Without training data** | MSP | 71.78 | 80.90 | 81.72 | 61.21 | 76.75 | 71.06 | 89.86 | 37.31 | 83.27 | 60.87 | 85.77 | 49.30 | 86.30 | 49.16 |
| | Energy | 68.21 | 87.45 | 78.25 | 79.16 | 73.23 | 83.31 | 85.13 | 61.52 | 78.96 | 84.45 | 79.86 | 80.81 | 81.32 | 75.59 |
| | MaxLogit | 70.44 | 86.52 | 80.89 | 75.00 | 75.66 | 80.76 | 88.99 | 49.19 | 81.68 | 79.80 | 83.72 | 74.45 | 84.80 | 67.81 |
| | ReAct | 69.30 | 85.02 | 81.97 | 60.29 | 75.64 | 72.66 | 90.03 | 31.55 | 86.99 | 54.66 | 87.66 | 42.81 | 88.23 | 43.01 |
| | GEN | 72.75 | 79.19 | 85.18 | 48.04 | 78.97 | 63.62 | 94.23 | 20.51 | 88.15 | 44.57 | 90.56 | 32.69 | 90.98 | 32.59 |
| | ASH | 44.58 | 94.37 | 33.33 | 95.82 | 38.95 | 95.10 | 23.03 | 96.05 | 20.26 | 98.10 | 21.95 | 97.32 | 21.74 | 97.16 |
| | SCALE | 64.64 | 88.94 | 72.47 | 84.20 | 68.55 | 86.57 | 78.60 | 74.76 | 68.52 | 91.30 | 71.18 | 87.91 | 72.77 | 84.66 |
| | LTS | 69.11 | 87.08 | 79.49 | 75.58 | 74.30 | 81.33 | 88.25 | 54.46 | 79.71 | 82.51 | 81.99 | 76.80 | 83.32 | 71.26 |
| | PRO | 72.22 | 79.29 | 85.19 | 47.75 | 78.70 | 63.52 | 93.81 | 20.84 | 87.44 | 45.60 | 90.33 | 33.12 | 90.53 | 33.19 |
| | AdaSCALE | 67.23 | 84.14 | 81.11 | 64.55 | 74.17 | 74.34 | 91.55 | 28.90 | 87.18 | 49.96 | 89.88 | 35.15 | 89.54 | 38.00 |
| | **OMASHU** | **73.75** | **78.68** | **86.98** | **47.10** | **80.36** | **62.89** | **98.31** | **8.11** | **93.60** | **31.32** | **93.06** | **28.91** | **94.99** | **22.78** |
| **With training data** | MDS | 68.61 | 83.60 | 81.75 | 53.71 | 75.18 | 68.65 | 93.59 | 21.56 | 89.81 | 37.49 | 91.05 | 30.56 | 91.49 | 29.87 |
| | RMDS | 71.77 | 82.77 | 84.92 | 50.13 | 78.34 | 66.45 | 95.58 | 18.44 | 89.26 | 39.18 | 92.17 | 28.91 | 92.34 | 28.84 |
| | KNN | 64.08 | 85.00 | 79.15 | 58.51 | 71.62 | 71.76 | 88.93 | 30.94 | 90.54 | 35.71 | 88.63 | 35.71 | 89.37 | 34.12 |
| | ViM | 68.79 | 88.58 | 81.69 | 61.19 | 75.24 | 74.88 | 94.61 | 17.98 | 92.67 | 29.77 | 92.32 | 26.55 | 93.20 | 24.77 |
| | fDBD | 65.93 | 86.23 | 80.12 | 59.39 | 73.02 | 72.81 | 90.91 | 27.35 | 88.35 | 42.29 | 89.48 | 35.00 | 89.58 | 34.88 |
| | NCI | 68.94 | 84.40 | 82.91 | 56.61 | 75.93 | 70.50 | 92.21 | 26.71 | 88.34 | 41.84 | 89.78 | 35.20 | 90.11 | 34.58 |
| | CADRef | 71.51 | 81.73 | 84.90 | 51.70 | 78.20 | 66.72 | 94.31 | 19.92 | 91.14 | 33.33 | 92.08 | 27.23 | 92.51 | 26.83 |
| | ActSub | 67.81 | 80.34 | 81.72 | 52.27 | 74.76 | 66.30 | 88.06 | 28.86 | 87.70 | 41.08 | 87.56 | 33.87 | 87.77 | 34.60 |
| | MDS++ | 73.65 | 83.31 | 87.15 | 49.48 | 80.40 | 66.39 | 97.27 | 10.83 | 92.67 | 30.34 | **93.82** | **24.22** | 94.59 | 21.80 |
| | RMDS++ | 75.14 | 81.26 | 87.79 | 48.58 | 81.47 | 64.92 | 97.32 | 11.06 | 89.32 | 38.00 | 93.18 | 25.65 | 93.27 | 24.91 |
| | SAE† | 70.50 | 81.62 | 83.01 | 55.40 | 76.76 | 68.51 | 94.07 | 20.37 | 90.86 | 32.64 | 91.71 | 27.90 | 92.21 | 26.97 |
| | VNS† | **78.71** | **70.01** | 87.93 | **41.93** | **83.32** | **55.97** | 96.20 | 15.22 | 88.34 | 42.94 | 92.76 | 25.69 | 92.43 | 27.95 |
| | **OMASHU++** | 76.99 | 75.06 | **88.01** | 45.58 | 82.50 | 60.32 | **98.68** | **5.97** | **94.54** | **25.86** | 93.41 | 28.30 | **95.54** | **20.04** |

Table 7: **OOD detection accuracies for ConvNeXt-B.** Near- and far-OOD detection accuracy using AUC (in %, ↑) and FPR (in %, ↓, @95 %) for ConvNeXt-B (Liu et al., 2022) (ImageNet-1k). We report results for existing approaches and OMASHU(++), with and without access to training data at inference time. We report results for all *five* OpenOOD datasets. † indicates results taken from the original papers; other results are reproduced. Best results **bold** and highlighted; second best underlined and highlighted. These results extend Table 1 of the main paper.

| | | Near-OOD | | | | | | Far-OOD | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SSB-HARD | | NINCO | | Near-Avg. | | iNaturalist | | Textures | | OpenImage-O | | Far-Avg. | |
| | **Method** | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ |
| **Without training data** | MSP | 75.47 | 76.75 | 81.08 | 71.52 | 78.28 | 74.14 | 89.54 | 43.45 | 74.36 | 96.46 | 83.10 | 74.67 | 82.33 | 71.53 |
| | Energy | 65.59 | 90.04 | 64.05 | 92.19 | 64.82 | 91.11 | 55.75 | 92.24 | 49.48 | 98.27 | 52.68 | 96.42 | 52.64 | 95.65 |
| | MaxLogit | 72.93 | 85.60 | 76.43 | 86.43 | 74.68 | 86.02 | 82.63 | 77.81 | 64.66 | 98.11 | 73.40 | 93.54 | 73.56 | 89.82 |
| | ReAct | 73.81 | 76.74 | 81.29 | 62.86 | 77.55 | 69.80 | 87.87 | 36.76 | 76.30 | 91.24 | 82.94 | 58.73 | 82.37 | 62.24 |
| | GEN | **79.40** | **67.32** | 86.51 | 50.72 | 82.96 | 59.02 | 95.36 | 19.22 | 83.63 | 86.14 | 90.92 | 36.83 | 89.97 | 47.40 |
| | ASH | 33.41 | 99.38 | 29.30 | 99.76 | 31.35 | 99.57 | 4.51 | 99.97 | 41.89 | 99.50 | 23.06 | 99.92 | 23.15 | 99.80 |
| | SCALE | 61.13 | 86.16 | 67.75 | 72.36 | 64.44 | 79.26 | 57.73 | 74.92 | 70.08 | 78.90 | 65.19 | 73.48 | 64.33 | 75.76 |
| | LTS | 65.14 | 90.55 | 64.45 | 92.44 | 64.80 | 91.50 | 66.31 | 88.51 | 46.03 | 99.09 | 57.31 | 95.87 | 56.55 | 94.49 |
| | PRO | 78.33 | 67.96 | 86.85 | 47.20 | 82.59 | 57.58 | 95.04 | 19.62 | 84.25 | 80.04 | 91.26 | 33.62 | 90.18 | 44.43 |
| | AdaSCALE | 71.89 | 81.34 | 84.81 | 56.74 | 78.35 | 69.04 | 94.28 | 19.90 | 87.27 | 47.38 | 91.82 | **28.90** | 91.12 | 32.06 |
| | **OMASHU** | 79.36 | 69.61 | **88.58** | **45.44** | **83.97** | **57.53** | **98.30** | **8.25** | **93.03** | **36.26** | **92.59** | 31.59 | **94.64** | **25.37** |
| **With training data** | MDS | 74.19 | 77.09 | 85.21 | 51.61 | 79.70 | 64.35 | 94.31 | 19.60 | 91.49 | 32.81 | 93.00 | 25.68 | 92.93 | 26.03 |
| | RMDS | 75.99 | 73.84 | 86.29 | 47.79 | 81.14 | 60.82 | 94.50 | 21.26 | 90.67 | 36.79 | 93.08 | 27.05 | 92.75 | 28.37 |
| | KNN | 72.46 | 77.80 | 83.71 | 56.09 | 78.08 | 66.95 | 92.94 | 21.83 | 92.14 | 31.16 | 92.29 | 26.94 | 92.46 | 26.64 |
| | ViM | 71.69 | 87.01 | 84.50 | 57.26 | 78.10 | 72.13 | 94.97 | 15.50 | 93.37 | **26.80** | 93.39 | 22.46 | 93.91 | 21.59 |
| | fDBD | 73.27 | 76.52 | 84.35 | 54.55 | 78.81 | 65.53 | 94.35 | 19.91 | 90.33 | 38.92 | 92.53 | 27.84 | 92.40 | 28.89 |
| | NCI | 72.40 | 78.87 | 83.49 | 56.25 | 77.95 | 67.56 | 93.49 | 22.78 | 89.84 | 41.05 | 91.78 | 30.54 | 91.71 | 31.46 |
| | CADRef | 76.72 | 72.68 | 86.81 | 47.56 | 81.76 | 60.12 | 94.64 | 18.62 | 90.95 | 35.76 | 92.89 | 24.52 | 92.83 | 26.30 |
| | ActSub | 74.50 | 72.36 | 85.20 | 49.54 | 79.85 | 60.95 | 93.68 | 19.61 | 91.08 | 34.99 | 92.11 | 26.02 | 92.29 | 26.88 |
| | MDS++ | 76.85 | 77.10 | 87.13 | 48.51 | 81.99 | 62.80 | 94.78 | 18.49 | 89.89 | 34.95 | 93.04 | 24.88 | 92.57 | 26.11 |
| | RMDS++ | 78.15 | 73.10 | 88.02 | 45.80 | 83.09 | 59.45 | 95.24 | 19.19 | 89.49 | 37.86 | 93.18 | 25.24 | 92.63 | 27.43 |
| | **OMASHU++** | **82.20** | **64.46** | **90.58** | **37.84** | **86.39** | **51.15** | **98.45** | **7.51** | **94.46** | 28.33 | **94.96** | **21.33** | **95.95** | **19.06** |

Table 8: **OOD detection accuracies for ResNet-50 V2.** Near- and far-OOD detection accuracy using AUC (in %, ↑) and FPR (in %, ↓, @95 %) for ResNet-50 V2 (Vryniotis, 2021) (ImageNet-1k). We report results for existing approaches and OMASHU(++), with and without access to training data at inference time. We report results for all *five* OpenOOD datasets. † indicates results taken from the original papers; other results are reproduced. Best results **bold** and highlighted; second best underlined and highlighted. These results extend Table 1 of the main paper.

| | | Near-OOD | | | | | | Far-OOD | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SSB-HARD | | NINCO | | Near-Avg. | | iNaturalist | | Textures | | OpenImage-O | | Far-Avg. | |
| | **Method** | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ | AUC ↑ | FPR ↓ |
| **Without training data** | MSP | 68.32 | 84.46 | 76.73 | 79.92 | 72.53 | 82.19 | 84.80 | 62.99 | 77.34 | 84.24 | 82.29 | 69.50 | 81.48 | 72.25 |
| | Energy | 54.10 | 89.38 | 54.68 | 89.07 | 54.39 | 89.22 | 51.16 | 88.03 | 49.14 | 94.29 | 53.78 | 89.84 | 51.36 | 90.72 |
| | MaxLogit | 65.55 | 87.01 | 72.85 | 84.98 | 69.20 | 85.99 | 80.37 | 77.52 | 71.35 | 90.78 | 77.70 | 82.15 | 76.47 | 83.48 |
| | ReAct | 39.90 | 92.68 | 30.91 | 97.06 | 35.40 | 94.87 | 33.63 | 90.34 | 43.51 | 94.24 | 29.44 | 97.40 | 35.53 | 94.00 |
| | GEN | 70.37 | 79.80 | 82.23 | 58.89 | 76.30 | 69.35 | 92.65 | 27.28 | 85.36 | 57.12 | 89.36 | 35.53 | 89.12 | 39.98 |
| | ASH | 64.52 | 87.46 | 71.39 | 85.22 | 67.96 | 86.34 | 80.19 | 78.10 | 71.08 | 90.28 | 76.12 | 84.69 | 75.79 | 84.36 |
| | SCALE | 64.58 | 87.40 | 71.46 | 85.17 | 68.02 | 86.28 | 80.17 | 78.18 | 71.05 | 90.31 | 76.19 | 84.59 | 75.80 | 84.36 |
| | LTS | 51.91 | 90.24 | 51.06 | 90.43 | 51.48 | 90.33 | 51.60 | 86.20 | 50.47 | 92.37 | 50.47 | 91.16 | 50.85 | 89.91 |
| | PRO | **71.04** | **78.60** | 82.99 | **55.75** | 77.02 | **67.17** | 92.39 | 27.19 | 85.27 | 55.94 | 89.44 | 34.22 | 89.03 | 39.12 |
| | AdaSCALE | 53.82 | 92.00 | 62.47 | 85.00 | 58.15 | 88.50 | 78.78 | 58.80 | 83.62 | 51.87 | 75.23 | 69.24 | 79.21 | 59.97 |
| | **OMASHU** | 70.59 | 81.72 | **83.49** | 55.77 | **77.04** | 68.75 | **97.18** | **14.69** | **93.13** | **34.00** | **92.16** | **31.58** | **94.16** | **26.76** |
| **With training data** | MDS | 70.49 | 82.74 | 82.76 | 55.59 | 76.62 | 69.16 | 95.42 | 17.58 | 93.52 | 34.19 | 92.28 | 28.89 | 93.74 | 26.89 |
| | RMDS | 75.59 | 79.20 | 86.17 | 53.17 | 80.88 | 66.19 | 91.18 | 28.07 | 83.38 | 53.91 | 89.08 | 34.35 | 87.88 | 38.78 |
| | KNN | 62.84 | 86.08 | 78.68 | 60.88 | 70.76 | 73.48 | 86.44 | 34.73 | 93.82 | 35.76 | 86.95 | 39.63 | 89.07 | 36.71 |
| | ViM | 57.88 | 92.07 | 72.38 | 82.39 | 65.13 | 87.23 | 92.31 | 30.77 | **95.04** | 26.44 | 89.90 | 39.00 | 92.42 | 32.07 |
| | fDBD | 58.80 | 88.45 | 71.71 | 75.90 | 65.26 | 82.17 | 86.43 | 46.02 | 86.52 | 51.44 | 85.37 | 48.51 | 86.11 | 48.65 |
| | NCI | 61.54 | 87.82 | 74.48 | 74.91 | 68.01 | 81.36 | 91.90 | 35.80 | 90.90 | 43.75 | 88.73 | 43.08 | 90.51 | 40.88 |
| | CADRef | 38.87 | 93.92 | 39.18 | 92.27 | 39.03 | 93.10 | 42.65 | 87.27 | 69.19 | 75.29 | 44.81 | 90.04 | 52.22 | 84.20 |
| | ActSub | 69.10 | 77.99 | 81.90 | 54.32 | 75.50 | 66.15 | 83.79 | 42.49 | 90.72 | 46.24 | 85.62 | 46.12 | 86.71 | 44.95 |
| | MDS++ | 75.33 | 80.40 | 86.53 | 49.07 | 80.93 | 64.74 | 91.95 | 26.81 | 94.58 | **20.15** | 90.58 | 30.81 | 92.37 | 25.92 |
| | RMDS++ | **76.65** | 78.82 | 86.58 | 50.10 | **81.61** | 64.46 | 87.78 | 31.13 | 79.84 | 55.03 | 86.61 | 34.42 | 84.74 | 40.19 |
| | **OMASHU++** | 76.29 | **75.37** | **86.88** | **47.98** | 81.58 | **61.67** | **97.86** | **11.06** | 92.59 | 32.35 | **92.88** | **28.11** | **94.44** | **23.84** |