ViM: Out-Of-Distribution with Virtual-logit Matching

Haoqi Wang[1*] Zhizhong Li[1*] Litong Feng[1] Wayne Zhang[12†]
[1]SenseTime Research [2]Qing Yuan Research Institute, Shanghai Jiao Tong University
{wanghaoqi,lizz,fenglitong,wayne.zhang}@sensetime.com

Abstract

Most of the existing Out-Of-Distribution (OOD) detection algorithms depend on single input source: the feature, the logit, or the softmax probability. However, the immense diversity of the OOD examples makes such methods fragile. There are OOD samples that are easy to identify in the feature space while hard to distinguish in the logit space and vice versa. Motivated by this observation, we propose a novel OOD scoring method named Virtual-logit Matching (ViM), which combines the class-agnostic score from feature space and the In-Distribution (ID) class-dependent logits. Specifically, an additional logit representing the virtual OOD class is generated from the residual of the feature against the principal space, and then matched with the original logits by a constant scaling. The probability of this virtual logit after softmax is the indicator of OOD-ness. To facilitate the evaluation of large-scale OOD detection in academia, we create a new OOD dataset for ImageNet-1K, which is human-annotated and is $8.8\times$ the size of existing datasets. We conducted extensive experiments, including CNNs and vision transformers, to demonstrate the effectiveness of the proposed ViM score. In particular, using the BiT-S model, our method gets an average AUROC 90.91% on four difficult OOD benchmarks, which is 4% ahead of the best baseline. Code and dataset are available at https://github.com/haoqiwang/vim.

1. Introduction

Considering most deep image classification models are trained in the closed-world setting, the *out-of-distribution* (OOD) issue arises and deteriorates customer experience when the models are deployed in production, facing inputs coming from the open world [9]. For instance, a model may wrongly but confidently classify an image of *crab* into the *clapping* class, even though no crab-related concepts appear in the training set. OOD detection is to decide whether an input belongs to the training distribution. OOD detection complements classification and finds its application in fields such as autonomous driving [19], medical analysis [30] and industrial inspection [1]. A comprehensive review of OOD and related topics including open set recognition, novelty detection and anomaly detection can be found in [38].

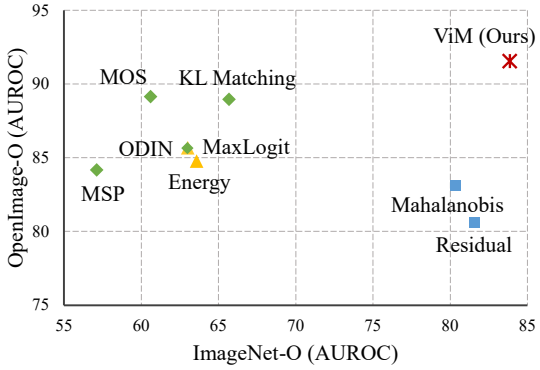

Figure 1. The AUROC (in percentage) of nine OOD detection algorithms applied to a BiT model trained on ImageNet-1K. The OOD datasets are ImageNet-O (x-axis) and OpenImage-O (y-axis). Methods marked with box □ use the feature space; methods with triangle △ use the logit; and methods with diamond ◊ use the softmax probability. The proposed method ViM (marked with *) uses information from both features and logits.

The core of an OOD detector is a scoring function ϕ that maps an input feature $\boldsymbol{x}$ to a scalar in $\mathbb{R}$, indicating to what extent the sample is likely to be OOD. In testing, a threshold τ is decided, ensuring that the validation set retains at least a given true-positive rate (TPR), *e.g.* the typical value of 0.95. The input example is regarded as OOD if $\phi(\boldsymbol{x}) > \tau$ and as ID (*i.e.*, in-distribution) otherwise. In cases where a score indicating the ID-ness is convenient, we can mentally use the negative of OOD score as the ID score.

Researchers have designed quite a few scoring functions by seeking properties that are naturally held by ID examples and easily violated by OOD examples, or *vice versa*. Scores are mainly derived from three sources: (1) *the probability*, such as the maximum softmax probabilities [13], the minimum KL-divergence between the softmax and the mean class-conditional distributions [12]; (2) *the logit*, such as the maximum logits [12], the $\mathrm{logsumexp}$ function over log-

[*] These two authors contribute equally to the work.
[†] Corresponding author: Wayne Zhang.

its [25]; and (3) *the feature*, such as the norm of the residual between feature and the pre-image of its low-dimensional embedding [27], the minimum Mahalanobis distance between the feature and the class centroids [23], *etc*. In these methods, OOD scores can be directly computed from existing models without re-training, making the deployment effortless. However, as illustrated in Fig. 1, their performances are limited by the singleness of their information source: using features exclusively disregards the classification weights with class-dependent information; using the logit or the softmax solely misses feature variations in the null space [3], which carries class-agnostic information; and the softmax further discards the norm of logits. To cope with the immense diversity that manifests in OOD samples, we ask the question, *is it helpful to design an OOD score that utilizes multiple sources?*

Built upon the success of prior arts, we design a novel scoring function termed the *Virtual-logit Matching* (ViM) score, which is the softmax score of a constructed virtual OOD class whose logit is jointly determined by the feature and the existing logits. To be specific, the scoring function first extracts the residual of the feature against a principal subspace, and then converts it to a valid logit by matching its mean over training samples to the average maximum logits. Finally, the softmax probability of the devised OOD class is the OOD score. From the construction of ViM, we can see intuitively that the smaller the original logits and the greater the residual, the more likely it is to be OOD.

Different from the aforementioned methods, another line of works tailors the features learned by the network to better identify ID and OOD by imposing dedicated regularization losses [5, 16, 18, 40] or by exposing generated or real collected OOD samples [22, 37]. As they all require the re-training of the network, we briefly mention them here and will not delve into the details.

Recently, OOD detection in large-scale semantic space has attracted increasing attention [12, 15, 18, 29], advancing OOD detection methods toward real-world applications. However, the current shortage of clean and realistic OOD datasets for large-scale ID datasets becomes an impediment to the field. Previous OOD datasets were curated from public datasets which were collected with a predefined tag list, such as iNaturalist, Texture, and ImageNet-21k (Tab. 1). This may lead to a biased performance comparison, specifically, the hackability of small coverage as described in Sec. 5. To avoid this risk, we build a new OOD benchmark for ImageNet-1K [4] models, *OpenImage-O*, from OpenImage dataset [21] with natural class distribution. It contains 17,632 manually filtered images, and is 7.8× larger than the recent ImageNet-O [15] dataset.

We extensively evaluate our method on various models using ImageNet-1K as the ID dataset. The model architectures range from the classical ResNet-50 [11], to the recent BiT [20], and to the latest ViT-B16 [8], RepVGG [7], DeiT [33] and Swin Transformer [26]. From the results on four OOD datasets, including OpenImage-O, ImageNet-O, Texture, and iNaturalist, we found that model selection affected the performance of many baseline methods, while our method performs stably well. Specially, our method achieved an average AUROC of 90.91% using the BiT model, which greatly surpasses the best baseline whose average AUROC is 86.62%.

Dataset	Image Distribution	#Image	Labeling Method
OpenImage-O	natural class statistics	17,632	image-level manual
Texture [2]	predefined tag list	5,160	tag-level manual
iNaturalist [18, 34]	predefined tag list	10,000	tag-level manual
ImageNet-O [18]	hard adversarial OOD	2,000	image-level manual

Table 1. OpenImage-O follows natural class statistics, while ImageNet-O is adversarially built to be hard. Both datasets have image-level OOD annotation. Texture and iNaturalist are selected by tags, and their OOD labels are annotated in tag-level.

Our contributions are threefold. (1) We proposed a novel OOD detection method ViM, that works well for a large range of models and datasets, owing to the effective fusion of information from both features and logits. The method is lightweight and fast, requiring neither extra OOD data nor re-training. (2) We conducted comprehensive experiments and ablation studies on the ImageNet-1K dataset, including CNNs and vision transformers. (3) We curated a new OOD dataset for ImageNet-1K called OpenImage-O, which is very diverse and contains complex scenes. We believe it will facilitate research on large-scale OOD detection.

2. Related Work

OOD/ID Score Design Hendrycks *et al*. [13] presented a baseline method using the maximum predicted softmax probability (MSP) as the ID score. ODIN [24] enhances MSP by perturbing the inputs and rescaling the logits. Hendrycks *et al*. [12] also experimented with the MaxLogit and the KL matching method on the ImageNet dataset. The energy score [25] computes the $\mathrm{logsumexp}$ on logits, and ReAct [32] strengthens the energy score by feature clipping. In [27] the norm of the difference between the feature and the pre-image of its low-dimensional manifold embedding is used. Lee *et al*. [23] computes the minimum Mahalanobis distance between the feature and the class-wise centroids. NuSA [3] uses the ratio of the norm of feature projected onto the column space of the classification weight matrix to the original norm as the ID score. The gradients are also used as evidence for ID and OOD distinction in [17]. For methods using logits/probabilities, feature variations on the null space of the weight matrix are completely ignored; while for methods that operate on the features space, the class-dependent information on weight matrix is dropped.

Our method combines the strengths of feature-based scores and logit-based scores by the novel mechanism of virtual logit, and gets substantial improvements.

Network/Loss Design Many works redesign the training loss to be OOD-aware [5] or add regularization terms [18, 40] to push part ID/OOD features. DeVries *et al*. [5] augment the network with a confidence estimation branch that uses misclassified in-distribution examples as a proxy for out-of-distribution examples. MOS [18] modifies the loss to use the pre-defined group structure so that the minimum group-wise "else" class probability can indicate the OOD-ness. Zaeemzadeh *et al*. [40] forces the ID samples to embed into a union of 1-dimensional subspaces during training and computes the minimum angular distance from the feature to the class-wise subspaces. Generalized ODIN [16] uses a dividend/divisor structure to encode the prior knowledge of decomposing the confidence of class probability. Different from these methods, our method does not require model retraining, thus not only is it easier to apply, but the ID classification accuracy is also preserved.

OOD Data Exposure Outlier Exposure [14] utilizes an auxiliary OOD dataset to improve OOD detection. Dhamija *et al*. [6] regularize samples from extra background classes to have uniform logits and to have small feature norms. Lee *et al*. [22] use GAN to generate OOD samples that lie near the ID samples and push the prediction of OOD samples to the uniform distribution. Several methods, including MCD [39], NGC [36] and UDG [37], can utilize external unlabeled noisy data to enhance the OOD detection performances. Different from these methods, our method does not require additional OOD data and thus avoids biases towards the introduced OOD samples [31].

3. Motivation: The Missing Info in Logits

For a series of OOD detection methods that are based on logits or softmax probabilities, we find that their performances are limited. In Fig. 1, feature-based OOD scores such as Mahalanobis and Residual are good at detecting OOD in ImageNet-O, while all methods that are based on logit/probability lag behind. This is not an accident, as is again shown in Fig. 2. The AUROC of the state-of-the-art probability-based method KL Matching is still lower than straightforwardly designed OOD scores in feature space on Texture dataset. This motivates us to study the influence of the lost information going from features to logits.

Consider a C-class classification model whose logit $\boldsymbol{l} \in \mathbb{R}^C$ is transformed from the feature $\boldsymbol{x} \in \mathbb{R}^N$ by a fully connected layer with weight $\boldsymbol{W} \in \mathbb{R}^{N\times C}$ and bias $\boldsymbol{b} \in \mathbb{R}^C$, *i.e.* $\boldsymbol{l} = \boldsymbol{W}^T\boldsymbol{x} + \boldsymbol{b}$. The predicted probability is $p(\boldsymbol{x}) = \mathrm{softmax}(\boldsymbol{l})$. For convenience, we set the point

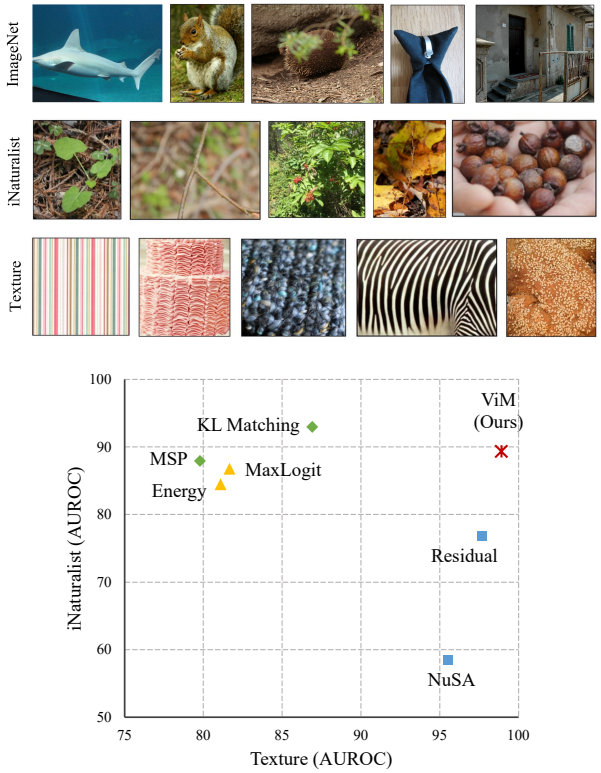

Figure 2. Comparison of AUROC for OOD detection algorithms that are based on probability (marked with diamond ◊), logit (△), and feature (□) of 9 OOD detection algorithms applied to a BiT model trained on ImageNet-1K. The OOD datasets are Texture (x-axis) and iNaturalist (y-axis). Example images for the ID dataset ImageNet-1K and the two OOD datasets are illustrated at the top.

$\boldsymbol{o} := -(\boldsymbol{W}^T)^+\boldsymbol{b}$, where $(\cdot)^+$ is the Moore-Penrose inverse, as the origin of a new coordinate system of feature space,

$$\boldsymbol{l} = \boldsymbol{W}^T\boldsymbol{x}' = \boldsymbol{W}^T(\boldsymbol{x} - \boldsymbol{o}), \quad \forall \boldsymbol{x}. \tag{1}$$

Geometrically, each logit l_i is the inner product between the feature $\boldsymbol{x}'$ and the class vector $\boldsymbol{w}_i$ (the i-th column of $\boldsymbol{W}$). Later when generalizing logits to virtual logits, we will replace $\boldsymbol{w}_i$ with a subspace, and replace the inner product with a projection. The bias term is safely omitted in the new coordinate system. In the remaining part of the paper, we assume the feature space uses the new coordinate system.

Logits contain class-dependent information, yet there is class-agnostic information in feature space that is not recoverable from logits. We study two cases (null space and principal space) and discuss the two OOD scores (NuSA and Residual) that rely on them, respectively.

OOD Score Based on Null Space A feature $\boldsymbol{x}$ can be decomposed into $\boldsymbol{x} = \boldsymbol{x}^{W^\perp} + \boldsymbol{x}^W$, where W is the column space of $\boldsymbol{W}$, $\boldsymbol{x}^{W^\perp}$ and $\boldsymbol{x}^W$ are projections of $\boldsymbol{x}$ to $W^\perp$ and W, respectively. $W^\perp$ is the null space of $\boldsymbol{W}^T$, and we have

$\boldsymbol{W}^T \boldsymbol{x}^{W^\perp} = \boldsymbol{0}$. The component $\boldsymbol{x}^{W^\perp}$ does not affect classification, but it influences OOD detection. It is demonstrated in [3] that one can perturb an image intensely yet constrain the difference between the features in $W^\perp$. The resulting outlier images are not like any of the ID images but retains high confidence in classification. Taking advantage of this, they define an ID score NuSA (null space analysis) as

$$\text{NuSA}(\boldsymbol{x}) = \frac{\sqrt{\|\boldsymbol{x}\|^2 - \|\boldsymbol{x}^{W^\perp}\|^2}}{\|\boldsymbol{x}\|}. \tag{2}$$

Intuitively, NuSA uses the angle ($= \arccos(\text{NuSA}(\boldsymbol{x}))$) between $\boldsymbol{x}$ and W to indicate the OOD-ness. From Fig. 2 we can see that the simple angle information clearly distinguishes OOD examples in Texture with an AUROC 95.50%, surpassing methods based on logits and the competitive method KL Matching based on softmax probability.

OOD Score Based on Principal Space It is generally assumed that features lie in low-dimensional manifolds [27, 40]. For simplicity, we use linear subspace (in the new coordinate system) passing through the origin $\boldsymbol{o}$ as the model. We define the *principal space* as the D-dimensional subspace P spanned by eigenvectors of the largest D eigenvalues of the matrix $\boldsymbol{X}^T\boldsymbol{X}$, where $\boldsymbol{X}$ is the ID data matrix. Features that deviate from the principal space are likely to be OOD examples. We can define

$$\text{Residual}(\boldsymbol{x}) = \|\boldsymbol{x}^{P^\perp}\|, \tag{3}$$

to capture the deviation of features from the principal space. Here $\boldsymbol{x} = \boldsymbol{x}^P + \boldsymbol{x}^{P^\perp}$ and $\boldsymbol{x}^{P^\perp}$ is the projection of $\boldsymbol{x}$ to $P^\perp$. The residual score is similar to the reconstruction error in [27] except that they employ nonlinear manifold learning for dimension reduction. Note that after the projection onto logits, this deviation is corrupted since the matrix $\boldsymbol{W}^T$ projects to a lower dimensional space than the feature space. Fig. 2 shows that Residual score improves over the NuSA score on both datasets, making the performance contrast between feature-based methods with logit/probability-based methods more striking.

Fusing Class-dependent and Class-agnostic Information In contrast to methods on logit/probability, both the NuSA and the Residual do not consider information that is specific to individual ID classes, namely they are class-agnostic. As a consequence, these scores ignore the feature similarity to each ID class, and are ignorant about which class the input resembles most. This gives an explanation of their worse performance on the iNaturalist OOD benchmark, as iNaturalist samples need to distinguish subtle differences between fine-grained classes. We hypothesize that unifying the information from feature space and the logits could improve the detection performance on a broader type of OOD samples. Such a solution is presented in Sec. 4 using the concept of virtual logit.

4. Virtual-logit Matching

To unify the class-agnostic and class-dependent information for OOD detection, we propose an OOD score by Virtual-logit Matching, abbreviated as ViM. The pipeline is illustrated in Fig. 3, where there are three steps, operating at the feature, the logit, and the probability, respectively. To be specific, for feature $\boldsymbol{x}$, (1) extract the residual $\boldsymbol{x}^{P^\perp}$ of $\boldsymbol{x}$ against the principal subspace P; (2) convert the norm $\|\boldsymbol{x}^{P^\perp}\|$ to a virtual logit by rescaling; and (3) output the softmax probability of the virtual logit as the ViM score. Below we give more details. Recall the notations: C is the number of classes, N is the feature dimension, and $\boldsymbol{W}$ and $\boldsymbol{b}$ are the classification weight and bias, respectively.

Principal Subspace and Residual Firstly we offset the feature space by a vector $\boldsymbol{o} = -(\boldsymbol{W}^T)^+\boldsymbol{b}$ so that it is bias-free in the computation of logits as Eq. (1). The principal subspace P is defined by the training set $\boldsymbol{X}$, where rows are features in the new coordinate system with origin $\boldsymbol{o}$. Suppose the eigendecomposition on the matrix $\boldsymbol{X}^T\boldsymbol{X}$ is

$$\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{-1}, \tag{4}$$

where eigenvalues in $\boldsymbol{\Lambda}$ are sorted decreasingly, then the span of the first D columns is the D-dimensional principal subspace P. The residual $\boldsymbol{x}^{P^\perp}$ is the projection of $\boldsymbol{x}$ onto $P^\perp$, Let the $(D+1)$-th column to the last column of $\boldsymbol{Q}$ in Eq. (4) be a new matrix $\boldsymbol{R} \in \mathbb{R}^{N\times(N-D)}$, then $\boldsymbol{x}^{P^\perp} = \boldsymbol{R}\boldsymbol{R}^T\boldsymbol{x}$. The residual $\boldsymbol{x}^{P^\perp}$ is sent to the next step.

Virtual-logit Matching The virtual logit

$$l_0 := \alpha\|\boldsymbol{x}^{P^\perp}\| = \alpha\sqrt{\boldsymbol{x}^T\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{x}} \tag{5}$$

is the norm of the residual rescaled by a per-model constant α. The norm $\|\boldsymbol{x}^{P^\perp}\|$ cannot be used as a new logit directly since the latter softmax will normalize over the exponential of logits and thus is very sensitive to the scale of logits. If the residual is very small compared to the largest logit, then after the softmax the residual will be buried in the noise of logits. To match the scales of the virtual logit, we compute the average norm of the virtual logit on the training set and also the mean of the maximum logit on the training set, then

$$\alpha := \frac{\sum_{i=1}^K \max_{j=1,\dots,C}\{l_j^i\}}{\sum_{i=1}^K \|\boldsymbol{x}_i^{P^\perp}\|}, \tag{6}$$

where $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_K$ are uniformly sampled K training examples, and l_j^i is the j-th logit of $\boldsymbol{x}_i$. In this way, on average, the scale of the virtual logit is the same as the maximum of the original logits.

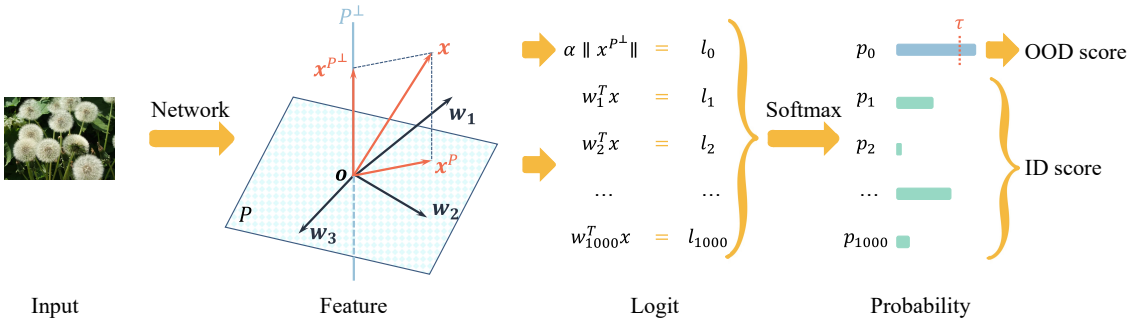

Figure 3. The pipeline of ViM. The principal space P and the matching constant α are determined by the training set beforehand using Eq. (4) and Eq. (6). In inference, feature $\boldsymbol{x}$ is computed by the network, and the virtual logit $\alpha\|\boldsymbol{x}^{P^\perp}\|$ is computed by projection and scaling. After softmax, the probability corresponding to the virtual logit is the OOD score. It is OOD if the score is larger than threshold τ.

The ViM Score We append the virtual logit to the original logits and compute the softmax. The probability corresponding to the virtual logit is defined as ViM. Mathematically, let the i-th logit of $\boldsymbol{x}$ be l_i, and then the score is

$$\text{ViM}(\boldsymbol{x}) = \frac{e^{\alpha\sqrt{\boldsymbol{x}^T\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{x}}}}{\sum_{i=1}^{C} e^{l_i} + e^{\alpha\sqrt{\boldsymbol{x}^T\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{x}}}}. \tag{7}$$

This equation reveals that two factors affect the ViM score: if its original logits are larger, then it is less of an OOD example; while if the norm of residual is larger, it is more likely to be OOD. The computational overhead is comparable to the last fully-connected layer (mapping from feature to logit) in the classification network, which is small.

Connection to Existing Methods Note that applying a strictly increasing function to the scores does not affect the OOD evaluation. Apply the function $t(x) = -\ln\left(\frac{1}{x} - 1\right)$ to the ViM score, then we have an equivalent expression

$$\alpha\|\boldsymbol{x}^{P^\perp}\| - \ln\sum_{i=1}^{C} e^{l_i}. \tag{8}$$

The first term is the virtual logit in Eq. (5) while the second term is the energy score [25]. ViM completes the energy method by feeding extra residual information from features. The performance is much superior to energy and residual.

5. OpenImage-O Dataset

We build a new OOD dataset called OpenImage-O for the ID dataset ImageNet-1K. It is manually annotated, comes with a naturally diverse distribution, and has a large scale with 17,632 images. It is built to overcome several shortcomings of existing OOD benchmarks. OpenImage-O is selected image-by-image from the test set of OpenImage-V3, including 125,436 images collected from Flickr without a predefined list of class names or tags, leading to natural class statistics and avoiding an initial design bias.

Necessity for Image-Level Annotation Some previous works on large-scale OOD detection select a portion of other datasets solely based on class labels. While class-level annotation costs less, the resulting dataset might be much noisier than expected. For example, the Places and the SUN dataset selected by [18] have a large portion of images that are indistinguishable from ID samples. Another example is the Texture [2, 18], in which the *bubbly* texture overlaps with the *bubble* class in ImageNet. Thus creating OOD datasets by querying tags is not reliable and per-image human inspection is needed for the confirmation of validity.

Hackability of Small Coverage If the OOD dataset has a central topic such as the Texture, featuring a less diverse distribution, then it might be easy to be "hacked". In Tab. 2, the gap between the highest and the average AUROC over nine methods for BiT are: OpenImage-O 5.61, iNaturalist 6.06, Texture 10.52, and ImageNet-O 14.39. Having larger gaps implies that the dataset is easier to improve.

Construction Process of OpenImage-O We construct the OpenImage-O based on the OpenImage-v3 dataset [21]. For every image in its testing set, we let human labelers to determine whether it is an OOD sample. To assist labeling, we simplified the task as distinguishing the image from the top-10 categories predicted by an ImageNet-1K classification model, i.e., the image is OOD if it does not belong to any of the 10 categories. Category labels as well as the most similar image to the test image in each category, measured by cosine similarity in the feature space, were presented for visualization. To further improve the annotation quality, we design several schemes: (1) Labelers can choose "Difficult", if they cannot decide whether the image belongs to any of the 10 categories; (2) Each image was labeled by at least two labelers independently, and we took the set of OOD images having consensus from the two; (3) Random inspection was performed to guarantee the quality.

6. Experiment

In this section, we compare our algorithm with state-of-the-art OOD detection algorithms. Following the prior work on large-scale OOD detection, we choose ImageNet-1K as the ID dataset. We benchmark the algorithms using both the CNN-based and the transformer-based models. Detailed experimental settings are as follows.

OOD Datasets Four OOD datasets (Tab. 1) are used to comprehensively benchmark the algorithms. OpenImage-O is our newly collected large-scale OOD dataset. Texture [2] consists of natural textural images and we removed four categories (*bubbly, honeycombed, cobwebbed, spiralled*) that overlapped with ImageNet. iNaturalist [34] is a fine-grained species classification dataset. We use the subset from [18]. Images in ImageNet-O [15] are adversarially filtered so that they can fool OOD detectors.

Evaluation Metrics Two commonly used metrics are reported. The AUROC is a threshold-free metric that computes the area under the receiver operating characteristic curve. Higher value indicates better detection performance. FPR95 is short for FPR@TPR95, which is the false positive rate when the true positive rate is 95%. The smaller FPR95 the better. We report both their numbers in percentage.

Experiment Settings BiT (Big Transfer) [20] is a variant of ResNet-v2, which employs group normalization and weight standardization. The BiT-S model series is pre-trained on ImageNet-1K, and we take the officially released checkpoint of BiT-S-R101$\times$1 for experiments. ViT (Vision Transformer) [8] is a transformer-based image classification model which treats images as sequences of patches. We use the officially released ViT-B/16 model, which is pre-trained on ImageNet-21K and fine-tuned on ImageNet-1K. Since the compared algorithms do not require re-training, the ID accuracies are not affected. Results on more model architectures, including CNN-based RepVGG [7], ResNet-50d [11], and transformer based Swin [26] and DeiT [33], are listed in Sec. 6.3. Their pre-trained weights are obtained from the timm repo [35]. When estimating the principal space, $K = 200,000$ images are randomly sampled from the training set. For features spaces with dimension $N > 1500$, we set the dimension of principal space to $D = 1000$, and set $D = 512$ otherwise.

Baseline Methods We compare ViM with eight baselines that do not require fine-tuning. They are MSP [13], Energy [25], ODIN [24], MaxLogit [12], KL Matching [12], Residual, ReAct [32] and Mahalanobis [23]. For Mahalanobis, we followed the setting in [10], which uses only the final feature instead of an ensemble of multiple layers [18, 23]. For ReAct, we use the Energy+ReAct setting with rectification percentile $p = 99$. The Residual is defined in Eq. (3).

6.1. Results on BiT

We present the results of the BiT model at the first half of Tab. 2. The best AUROC is shown in bold and the second and third place ones are shown with underlines.

ViM *vs.* Baselines On three datasets, including OpenImage-O, Texture, and ImageNet-O, ViM achieves the largest AUROC and the smallest FPR95. On average ViM has 90.91% AUROC, which surpasses the second place by 4.29%. The average FPR95 is also the lowest among them. In particular, regarding Eq. (8), an interpretation of ViM in terms of the Residual score and the Energy score, the results show that ViM is significantly better than the two methods on all datasets. This indicates that ViM non-trivially combined the OOD information in Residual and in Energy. However, on iNaturalist, ViM is only on the third place. We hypothesize that its moderate performance on iNaturalist relates to how much information is contained in the residual, because iNaturalist has the smallest average residual norm among four OOD datasets (iNaturalist 4.65, OpenImage-O 5.04, ImageNet-O 5.16, and Texture 8.16).

Effect of Information Source For OOD detection performances on BiT model, Tab. 2 shows an interesting pattern regarding the information source. If feature variations in the null space are absent, such as in methods that rely on logits and softmax, performances on Texture and ImageNet-O are restricted. For example, on the Texture dataset, the best performing method that relies on logit and softmax is KL Matching, which has 86.92% AUROC and is far behind ViM, Mahalanobis, and Residual, which operate on the feature space. In contrast, if the class-dependent information is dropped, such as in the Residual method, performances in iNaturalist and OpenImage-O are also limited. The proposed ViM score, however, is competent regardless of dataset types.

6.2. Results on ViT

[10] has discussed the benefit of large-scale pre-trained transformers on OOD tasks. However, their experiments are conducted on CIFAR100/10 and only two baseline methods are compared. We provide a comprehensive OOD evaluation on ImageNet-1K over a wide range of methods in the second half of Tab. 2.

ViM *vs.* Baselines The two best-performing methods for the ViT model are ViM and Mahalanobis. Their AU-

Model	Method	Source	OpenImage-O		Texture		iNaturalist		ImageNet-O		Average	
			AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓
BiT	MSP [13]	prob	84.16	73.72	79.80	76.65	87.92	64.09	57.12	96.85	77.25	77.83
	Energy [25]	logit	84.77	73.42	81.09	73.91	84.47	74.98	63.59	96.40	78.48	79.68
	ODIN [24]	prob+grad	85.64	72.83	81.60	74.07	86.73	70.75	63.00	96.85	79.24	78.63
	MaxLogit [12]	logit	85.67	72.68	81.66	73.72	86.76	70.59	63.01	96.85	79.27	78.46
	KL Matching [12]	prob	88.96	51.51	86.92	51.05	**92.95**	**33.28**	65.68	86.65	83.63	55.62
	Residual†	feat	80.58	67.85	97.66	11.16	76.76	80.41	81.57	65.50	84.14	56.23
	ReAct [32]	feat+logit	88.94	54.97	90.64	50.25	91.45	48.60	67.07	91.70	84.53	61.38
	Mahalanobis [23]	feat+label	83.10	64.32	97.33	14.05	85.70	64.95	80.37	70.05	86.62	53.34
	ViM (Ours)	feat+logit	**91.54**	**43.96**	**98.92**	**4.69**	89.30	55.71	**83.87**	**61.50**	**90.91**	**41.46**
ViT	MSP [13]	prob	92.53	34.18	87.10	48.55	96.11	19.04	81.86	64.85	89.40	41.65
	Energy [25]	logit	97.11	14.04	93.39	28.22	98.66	6.16	90.46	41.30	94.90	22.43
	ODIN [24]	prob+grad	96.86	15.68	93.01	30.60	98.57	6.58	89.85	44.15	94.57	24.25
	MaxLogit [12]	logit	96.87	15.68	93.01	30.60	98.57	6.58	89.85	44.15	94.57	24.25
	KL Matching [12]	prob	93.80	28.49	88.76	44.09	96.88	14.79	84.12	55.70	90.89	35.77
	Residual†	feat	92.72	32.63	92.21	33.80	98.57	6.63	88.23	47.85	92.93	30.23
	ReAct [32]	feat+logit	97.38	13.50	93.34	28.49	99.00	4.31	90.71	42.60	95.11	22.22
	Mahalanobis [23]	feat+label	97.48	13.54	94.24	25.17	**99.54**	**2.12**	**92.81**	36.95	96.02	19.45
	ViM (Ours)	feat+logit	**97.61**	**12.61**	**95.34**	**20.31**	99.41	2.60	92.55	**36.75**	**96.23**	**18.07**

Table 2. OOD detection for ViM and baseline methods. The ID dataset is ImageNet-1K, and OOD datasets are OpenImage-O, Texture, iNaturalist and ImageNet-O. Both metrics AUROC and FPR95 are in percentage. A pre-trained BiT-S-R101×1 model and a pre-trained ViT-B/16 model is tested. The best method is emphasized in bold, and the 2nd and 3rd ones are underlined. ODIN needs backpropagation for producing input perturbations, so it is *prob+grad*. ReAct clips feature and uses Energy subsequently, so it is *feat+logit*. Mahalanobis need gt labels to compute the class-wise mean feature, so it is *feat+label*. †: Residual is defined in Eq. (3).

Method	RepVGG [7]		Res50d [11]		Swin [26]		DeiT [33]	
	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓
MSP	78.10	70.55	77.99	67.96	87.57	43.44	79.48	66.43
Energy	76.38	78.99	71.08	78.39	87.77	35.08	72.80	70.14
ODIN	77.72	72.68	75.27	68.56	88.00	36.58	77.13	**63.92**
MaxLogit	77.56	73.50	75.39	69.34	88.40	35.28	76.79	64.49
KL Matching	81.35	61.65	82.72	64.41	88.87	46.99	83.49	64.80
Residual	84.19	59.00	87.01	58.55	92.88	37.38	84.15	74.13
ReAct	49.14	98.96	82.93	58.63	90.17	31.36	77.37	67.00
Mahalanobis	86.07	59.39	88.33	55.70	92.16	40.39	85.03	73.18
ViM (Ours)	**87.81**	**50.50**	**89.22**	**52.61**	**94.11**	**31.04**	**85.25**	69.95

Table 3. Results on RepVGG, ResNet50-d, Swin and DeiT. Due to space limitation, only their average AUROC (A↑) and average FPR95 (F↓) are reported. The numbers are in percentage. All models are using pre-trained weights taken from timm [35].

ROCs are close on all four datasets. However, Mahalanobis needs to compute the class-wise Mahalanobis distance, which makes its computation costly. In contrast, our method is lightweight and fast. Four methods, ReAct, Energy, MaxLogit, and ODIN, are the second best ones, and the remaining three methods have relatively low AUROCs.

Difference between ViT and BiT Since the ViT model is pre-trained on the ImageNet-21K dataset, the semantics it has seen is much larger than the BiT model. The OOD performance is relatively saturated. Although on most OOD datasets ViT is significantly better than BiT, we observe that ViT performs less competitively on the Texture dataset. We hypothesize that it is related to the observation in [28] that higher layers of ViT maintain spatial location information more faithfully than ResNets. ViT has high responses for local patches. However, textural images with similar local patches but not revealing the whole object are regarded as OOD of ImageNet (see example images in Fig. 2).

6.3. Results on More Model Architectures

We show more results on a variety of model architectures. In particular, we choose two CNN-based models RepVGG [7] and ResNet-50d [11] and two transformer-based models Swin Transformer [26] and DeiT [33]. Their average AUROCs and average FPR95s over the four OOD datasets are listed in Tab. 3. It is shown that ViM is robust to model architecture changes. The detailed experiment setting and results are in the supplementary materials.

6.4. The Effect of Hyperparameter

The Dimension D of Principal Space In [40] the feature of each class is represented by a 1-dimensional subspace, so a natural choice for the dimension D of principal space

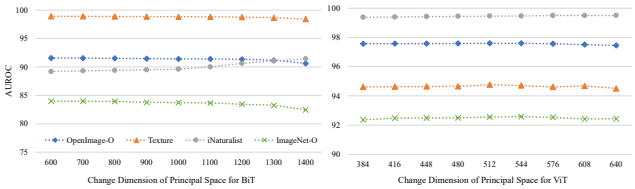

Figure 4. Robustness against principal space dimension. Left is BiT and right is ViT. The performance changes are small when D varies in a wide range of values.

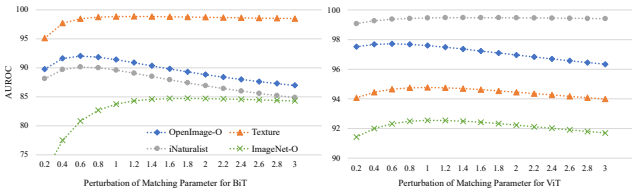

Figure 5. Perturbation of α by multiplying a factor. Left is BiT, and right is ViT. For both models, the proposed matching parameter fits well for the trends.

is the number of classes C. For models like ViT whose feature dimension N may be less than the number of classes C, we empirically suggest taking a number in the range $[N/3, 2N/3]$. We show in Fig. 4 that our method is robust to the selection of dimensions. However, if the application permits, one can adjust this parameter according to a hold-out OOD dataset. In our experiments, we set $D = 1000$ for BiT and $D = 512$ for ViT.

The Matching Parameter α The matching parameter controls the relative importance of the trade-off between different OOD features. Since OOD distribution is unknown, we suggest keeping them to be of equal importance. This is how α is defined in Eq. (6). It is easy to tune the parameter to fit some types of OOD datasets, but it is hard to improve all datasets at the same time. We show the result of perturbing the matching parameter by multiplying a factor in Fig. 5. If the multiple is larger, then information from the feature space is given more weight. Otherwise, information from logits is given more importance. Overall the best choice is no perturbation, suggesting that the defined α is a good choice.

6.5. The Effect of Grouping

In addition, we also compare with MOS [18], which exploits grouping structure in large-scale semantic spaces. Two methods are added to the comparison. (1) *MaxGroup* is the group version of MSP, which first obtains the group-wise probability by summing over the constituent classes, and then takes the maximum group probability as the ID score. (2) *ViM+Group* also takes the maximum group probability as the ID score, except that the probabilities are taken from the $(C+1)$ dimensional vector, with an extra ViM virtual class participating in the softmax normalization.

Method	OpenImage-O		Texture		iNaturalist		ImageNet-O	
	A↑	F↓	A↑	F↓	A↑	F↓	A↑	F↓
MOS* [18]	89.14	**41.97**	82.35	59.30	**98.15**	**9.28**	60.62	86.65
MaxGroup	84.75	71.22	80.42	77.87	89.50	57.18	63.93	92.45
ViM+Group	**91.92**	42.26	**98.91**	**4.69**	90.16	52.74	**83.43**	**62.00**

Table 4. AUROC of methods with grouping information. A↑ is AUROC and F↓ is FPR95. All numbers are in percentage. BiT is used and the grouping is defined in [18] based on taxonomy. * MOS needs fine-tuning while others do not.

MaxGroup and ViM+Group are evaluated on the pre-trained weights of BiT, while MOS needs to fine-tune the model using group-based learning. Results are shown in Tab. 4. We observe that (1) the average AUROC of MaxGroup improves over the vanilla MSP from 77.25% to 79.23%, showing the usefulness of group information; and (2) both our original ViM and the group version of ViM are better than MOS on three of four datasets by large margins.

6.6. Limitation of ViM

As we have noticed in Sec. 6.1, ViM shows less performance gains on OOD datasets that have small residuals, such as iNaturalist. Besides, the property that ViM does not need training is a double-edged sword. It means that ViM is limited by the feature quality of the original network.

7. Conclusion

In this paper, we present a novel OOD detection method: the Virtual-logit Matching (ViM) score. It combines the information from both the feature space and the logits, which provides the class-agnostic information and the class-dependent information, respectively. Extensive experiments on the large-scale OOD benchmarks show the effectiveness and robustness of the method. Especially, we tested ViM on both CNN-based models and transformer-based models, showing its robustness across model architectures. To facilitate the evaluation of large-scale OOD detection, we create the OpenImage-O dataset for ImageNet-1K, which is of high-quality and large-scale.

Acknowledgement

This work was supported in part by Innovation and Technology Commission of the Hong Kong Special Administrative Region, China (Enterprise Support Scheme under the Innovation and Technology Fund B/E030/18). Haoqi Wang was also supported by the Technology Leaders of Tomorrow (TLT) Programme of HKSTP InnoAcademy.

References

[1] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. MVTec AD–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9592–9600, 2019. 1

[2] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, 2014. 2, 5, 6

[3] Matthew Cook, Alina Zare, and Paul Gader. Outlier detection through null space analysis of neural networks. *arXiv preprint arXiv:2007.01263*, 2020. 2, 4

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. Ieee, 2009. 2

[5] Terrance DeVries and Graham W Taylor. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*, 2018. 2, 3

[6] Akshay Raj Dhamija, Manuel Günther, and Terrance Boult. Reducing network agnostophobia. *Advances in Neural Information Processing Systems*, 31, 2018. 3

[7] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. RepVGG: Making VGG-style ConvNets great again. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13733–13742, 2021. 2, 6, 7

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 2, 6

[9] Nick Drummond and Rob Shearer. The open world assumption. In *eSI Workshop: The Closed World of Databases meets the Open World of the Semantic Web*, volume 15, 2006. 1

[10] Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. *Advances in Neural Information Processing Systems*, 34, 2021. 6

[11] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2019. 2, 6, 7

[12] Dan Hendrycks, Steven Basart, Mantas Mazeika, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. *arXiv preprint arXiv:1911.11132*, 2019. 1, 2, 6, 7

[13] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017. 1, 2, 6, 7

[14] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019. 3

[15] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021. 2, 6

[16] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized ODIN: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10951–10960, 2020. 2, 3

[17] Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[18] Rui Huang and Yixuan Li. MOS: Towards scaling out-of-distribution detection for large semantic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8710–8719, 2021. 2, 3, 5, 6, 8

[19] Bernd Kitt, Andreas Geiger, and Henning Lategahn. Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. In *2010 IEEE Intelligent Vehicles Symposium*, pages 486–492, 2010. 1

[20] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (BiT): General visual representation learning. In *European Conference on Computer Vision*, pages 491–507. Springer, 2020. 2, 6

[21] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, et al. OpenImages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2(3):18, 2017. 2, 5

[22] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018. 2, 3

[23] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in Neural Information Processing Systems*, 31, 2018. 2, 6, 7

[24] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018. 2, 6, 7

[25] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 33:21464–21475, 2020. 2, 5, 6, 7

[26] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 2, 6, 7

[27] Ibrahima Ndiour, Nilesh Ahuja, and Omesh Tickoo. Out-of-distribution detection with subspace techniques and probabilistic modeling of features. *arXiv preprint arXiv:2012.04250*, 2020. 2, 4

[28] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34, 2021. 7

[29] Ryne Roady, Tyler L Hayes, Ronald Kemker, Ayesha Gonzales, and Christopher Kanan. Are open set classification methods effective on large-scale datasets? *PLOS ONE*, 15(9):1–18, 09 2020. 2

[30] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*, pages 146–157. Springer, 2017. 1

[31] Alireza Shafaei, Mark Schmidt, and James J Little. A less biased evaluation of out-of-distribution sample detectors. In *30th British Machine Vision Conference 2019, BMVC 2019, Cardiff, UK, September 9-12, 2019*, page 3. BMVA Press, 2019. 3

[32] Yiyou Sun, Chuan Guo, and Yixuan Li. ReAct: Out-of-distribution detection with rectified activations. *Advances in Neural Information Processing Systems*, 34, 2021. 2, 6, 7

[33] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 2, 6, 7

[34] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The iNaturalist species classification and detection dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, 2018. 2, 6

[35] Ross Wightman. PyTorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019. 6, 7

[36] Zhi-Fan Wu, Tong Wei, Jianwen Jiang, Chaojie Mao, Mingqian Tang, and Yu-Feng Li. NGC: A unified framework for learning with open-world noisy data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 62–71, 2021. 3

[37] Jingkang Yang, Haoqi Wang, Litong Feng, Xiaopeng Yan, Huabin Zheng, Wayne Zhang, and Ziwei Liu. Semantically coherent out-of-distribution detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8301–8309, 2021. 2, 3

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

[39] Qing Yu and Kiyoharu Aizawa. Unsupervised out-of-distribution detection by maximum classifier discrepancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9518–9526, 2019. 3

[40] Alireza Zaeemzadeh, Niccolò Bisagno, Zeno Sambugaro, Nicola Conci, Nazanin Rahnavard, and Mubarak Shah. Out-of-distribution detection using union of 1-dimensional subspaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9452–9461, 2021. 2, 3, 4, 7

ViM: Out-Of-Distribution with Virtual-logit Matching Supplementary Material

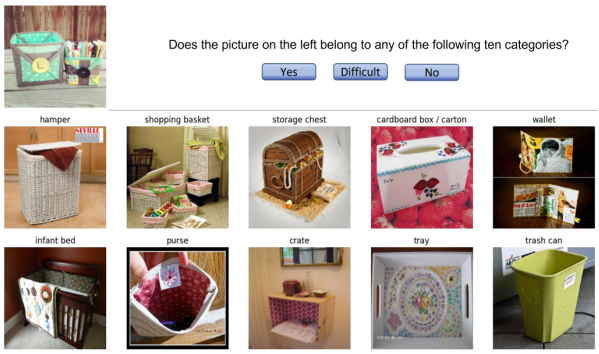

Figure 6. A demonstrative UI for the labelers. The image on the left-top corner is the candidate OOD image to be labeled. The two rows of images below are from the 10 most similar ID classes. Labelers choose from *yes/difficult/no* according to these information.

A. Detailed Information of Models (Sec. 6)

In the experiment, we benchmarked a collection of deep classification models. Their detailed information, including the specification, the architecture, the pre-train information, and the top-1 accuracy, is listed in Tab. 5. To summarize, half of them are CNN-based, and half are transformer-based. Vision Transformer and Swin Transformer are pre-trained on ImageNet-21K before training on the ImageNet-1K.

B. Detailed Results of Four Models (Sec. 6.3)

In Sec. 6.3 we gave the average AUROC and FPR95 for RepVGG, ResNet-50d, Swin Transformer and DeiT. We provide the detailed AUROC and FPR95 on OpenImage-O, Texture, iNaturalist, and ImageNet-O in Tab. 6.

C. Details on OpenImage-O (Sec. 5)

An illustrative software interface for labelers is shown in Fig. 6. For each candidate OOD image to be labeled, we find the top 10 classes in ImageNet-1K predicted by a classification model. Then we gather the most similar images in those top 10 classes by cosine similarity in the feature space. Next, we patch them as well as their labels with the corresponding OpenImage samples, and let the labelers distinguish whether the OpenImage sample belongs to any of the top 10 categories. We also set a choice called difficult, so that labelers can put the undistinguishable hard samples into the difficult category. To reduce annotation noises, each image is labeled twice from different group of labelers. Then we take the set of OOD images having consensus from the two groups, resulting in an OOD dataset with 17,632 unique images. In the end, a random inspection process is performed to guarantee the quality of the OOD dataset.

The OpenImage-O follows a natural image distribution as both the source dataset and the labeling process do not involve any filtration based on pre-defined list of labels. To get a sense of its distribution, we use the BiT model to find the most similar ID class in ImageNet for each OOD image. Then the histogram is illustrated in Fig. 7. It shows that the coverage of OpenImage-O is broader compared to the other three OOD datasets.

D. Details on Grouping (Sec. 6.5)

MOS [7] is trained using the officially released code and its default parameter setting. For all experiments in Sec. 6.5, the grouping strategy follows the taxonomy grouping defined in [7].

Grouping Results on ViT The grouping strategy is less effective for the ViT model, as seen from results in Tab. 7. Comparing MSP with its group version, MaxGroup, we can see that the improvement on AUROC is very small, while FPRs become even worse. Examining ViM with its group variant ViM+Group, we can see that their difference is very small, and the original version of ViM is slightly better than ViM+Group.

E. Details on Baselines (Sec. 6)

Mahalanobis On the BiT model, when including lower level features, the performance of Mahalanobis degrades a lot. The average AUROC on the four OOD datasets is 56%, which is much worse than the baseline MSP. Similar results is also found in [7, Table 1]. In this paper, we implement the

Model	Specification	Architecture	Pre-Trained Dataset	Top1 (%)
BiT [8]	BiT-S-R101x1	CNN	—	81.30
ViT [2]	ViT-B/16	Transformer	ImageNet-21K	85.43
RepVGG [1]	RepVGG-b3	CNN	—	80.52
Res50d [4]	ResNet-50d	CNN	—	80.52
Swin [12]	Swin-base-patch4-window7-224	Transformer	ImageNet-21K	85.27
DeiT [14]	DeiT-base-patch16-224	Transformer	—	81.98

Table 5. Detailed information on the used models. The detailed specification and the top-1 accuracy of the model are provided. Three of them are CNN-based, and the other three are transformer-based. Both ViT and Swin Transformer are pre-trained on ImageNet-21K before training on ImageNet-1K, so their general OOD performances are much better than alternatives.

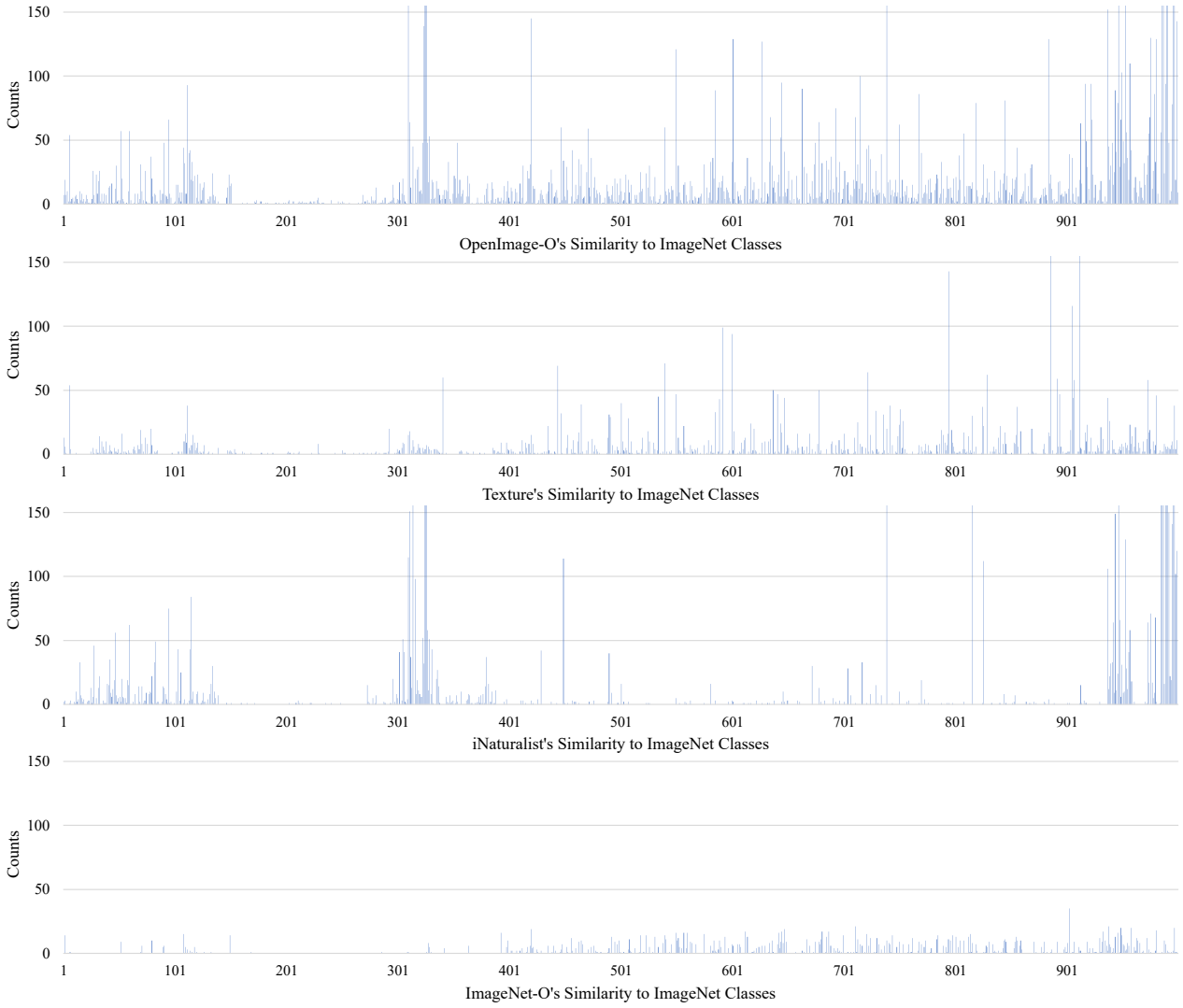

Figure 7. The diversity of four OOD datasets shown by how they look similar to the ImageNet-1K classes. We use the BiT model to predict which ID class the image most resembles, and count the number of such OOD images for each class. Results are shown above. Due to space limitation, the y-axis is clipped at 155. Our newly created OpenImage-O has a wider coverage on ImageNet ID classes.

Model	Method	Source	OpenImage-O		Texture		iNaturalist		ImageNet-O		Average	
			AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓
RepVGG [1]	MSP [6]	prob	85.06	63.36	78.58	72.62	87.11	54.93	61.65	91.30	78.10	70.55
	Energy [11]	logit	83.64	69.92	74.53	82.97	83.92	75.31	63.36	87.75	76.36	78.99
	ODIN [10]	prob+grad	85.22	63.48	76.77	76.14	86.37	61.40	62.50	89.70	77.72	72.68
	MaxLogit [5]	logit	84.81	65.04	76.33	76.86	86.22	62.20	62.87	89.90	77.56	73.50
	KL Matching [5]	prob	86.80	57.48	83.18	62.09	89.06	**42.07**	66.36	84.95	81.35	61.65
	Residual†	feat	82.51	65.13	93.05	28.66	86.09	62.40	75.11	79.80	84.19	59.00
	ReAct [13]	feat	46.08	99.65	54.56	97.66	47.18	99.88	48.76	98.65	49.14	98.96
	Mahalanobis [9]	feat+label	85.71	64.93	92.71	32.03	89.17	58.79	76.68	81.80	86.07	59.39
	ViM (Ours)	feat+logit	**89.27**	**52.40**	**93.69**	**23.76**	**91.35**	46.79	**76.93**	**79.05**	**87.81**	**50.50**
Res50d [4]	MSP [6]	prob	84.50	63.53	82.75	64.40	88.58	50.05	56.13	93.85	77.99	67.96
	Energy [11]	logit	75.95	76.83	73.93	75.31	80.50	71.32	53.95	90.10	71.08	78.39
	ODIN [10]	prob+grad	81.53	64.49	80.21	63.93	86.48	52.58	52.87	93.25	75.27	68.56
	MaxLogit [5]	logit	81.50	65.50	79.25	66.20	86.42	53.00	54.39	92.65	75.39	69.34
	KL Matching [5]	prob	87.31	60.58	86.07	61.36	**90.48**	**47.22**	67.00	88.50	82.72	64.41
	Residual†	feat	87.64	59.65	94.62	25.89	84.63	75.81	**81.15**	**72.85**	87.01	58.55
	ReAct [13]	feat	85.30	60.79	91.12	39.26	87.27	56.03	68.02	78.45	82.93	58.63
	Mahalanobis [9]	feat+label	89.52	55.91	94.15	28.22	89.48	62.69	80.15	76.00	88.33	55.70
	ViM (Ours)	feat+logit	**90.76**	**50.45**	**95.84**	**20.58**	89.26	64.59	81.02	74.80	**89.22**	**52.61**
Swin [12]	MSP [6]	prob	91.35	34.96	85.21	51.90	94.76	23.19	78.97	63.70	87.57	43.44
	Energy [11]	logit	90.93	27.58	82.62	51.57	95.22	15.47	82.29	45.70	87.77	35.08
	ODIN [10]	prob+grad	91.38	28.42	85.74	44.59	94.24	19.65	80.62	53.65	88.00	36.58
	MaxLogit [5]	logit	91.91	26.79	84.67	47.42	95.72	15.41	81.28	51.50	88.40	35.28
	KL Matching [5]	prob	91.92	40.05	86.89	52.93	94.77	27.62	81.91	67.35	88.87	46.99
	Residual†	feat	94.64	32.19	91.31	43.97	98.89	4.81	86.68	68.55	92.88	37.38
	ReAct [13]	feat	93.58	**23.07**	85.51	47.91	97.51	9.98	84.09	**44.50**	90.17	31.36
	Mahalanobis [9]	feat+label	94.57	33.41	89.92	49.17	98.69	5.43	85.46	73.55	92.16	40.39
	ViM (Ours)	feat+logit	**96.04**	23.88	**92.34**	**38.49**	**99.28**	**2.60**	**88.78**	59.20	**94.11**	**31.04**
DeiT [14]	MSP [6]	prob	84.04	62.03	81.99	64.48	88.25	52.00	63.65	87.20	79.48	66.43
	Energy [11]	logit	74.50	67.21	77.47	64.77	78.63	65.82	60.60	82.75	72.80	70.14
	ODIN [10]	prob+grad	80.19	**59.53**	81.26	**59.38**	85.36	51.81	61.70	84.95	77.13	**63.92**
	MaxLogit [5]	logit	80.11	60.83	80.45	60.89	85.22	52.54	61.38	83.70	76.79	64.49
	KL Matching [5]	prob	87.49	60.66	**84.89**	63.47	90.54	**50.47**	71.05	84.60	83.49	64.80
	Residual†	feat	88.07	69.21	82.68	77.75	91.32	58.30	74.54	91.25	84.15	74.13
	ReAct [13]	feat	80.29	63.11	80.45	63.99	84.43	59.07	64.32	**81.85**	77.37	67.00
	Mahalanobis [9]	feat+label	89.03	66.51	83.58	77.31	91.56	58.67	75.95	90.25	85.03	73.18
	ViM (Ours)	feat+logit	**89.13**	64.58	84.42	73.02	**92.15**	52.79	**95.30**	89.40	**85.25**	69.95

Table 6. OOD detection for ViM and baseline methods on RepVGG, ResNet-50d, Swin Transformer, and DeiT. Their pre-trained weights are used. The ID dataset is ImageNet-1K, and OOD datasets are OpenImage-O, Texture, iNaturalist, and ImageNet-O. Both metrics AUROC and FPR95 are in percentage. The best performing item is bolded, and the second and the third places are underlined. The proposed ViM has the largest AUROC and the lowest FPR in most cases. †: Residual is defined in Equ. (4).

Method	OpenImage-O		Texture		iNaturalist		ImageNet-O	
	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓
MSP	92.53	34.18	87.10	48.55	96.11	19.04	81.86	64.85
MaxGroup	92.60	48.08	87.84	60.08	95.39	31.40	84.45	71.90
ViM	97.61	12.61	**95.34**	**20.31**	**99.41**	**2.60**	**92.55**	**36.75**
ViM+Group	**97.64**	**12.51**	95.29	20.41	99.40	2.70	92.50	37.05

Table 7. Comparison of effect of grouping on ViT. All numbers are in percentage. The grouping is defined in [7] based on taxonomy. MaxGroup is the group version of MSP and ViM+Group is the group version of ViM.

Method	OpenImage-O	Texture	iNaturalist	ImageNet-O
Residual	1.70s	0.56s	1.00s	0.19s
KL Matching	249.97s	78.65s	141.63s	33.51s
Mahalanobis	2135.13s	626.80s	1210.82s	243.69s
ViM	1.49s	0.51s	0.86s	0.18s

Table 8. Score computation time for four methods on four OOD datasets. We assume that the features have been extracted, so the network forward time is not included. The implementation uses numpy and runs on Intel Xeon (Skylake) 23.20GHz CPU.

Mahalanobis score using the feature vector before the final classification fc layer, as in [3]. The precision matrix and the class-wise average vector are estimated using 200,000 random training samples. The ground-truth class label is used during computation.

KL Matching We estimate the class-wise average probability using 200,000 random training samples. Following the practice of [5], the predicted class is used instead of ground-truth labels. We would like to note that the hyperparameter selection for OOD methods should not base on the ID set that is used for computing FPR95 and AUROC (in our case, its the validation set of ImageNet), because once the OOD method overfits the validation set, the evaluation result can be higher than the actual performance.

ReAct For ReAct, we use the Energy+ReAct setting, which is the most effective settings in [13]. In the original paper, they recommended the 90-th percentile of activations estimated on the ID data for the clipping threshold. However, for BiT and ViT, we found that the rectification percentile $p = 99$ works much better than 90. So we report results using $p = 99$.

F. OOD Examples Detected by KL Matching and Residual (Sec. 3)

In Sec. 3, we showed that feature-based OOD scores (*e.g.* Residual) and logit/softmax-based OOD scores (*e.g.* KL Matching) have different performances on the Texture OOD dataset. Here we visualize the OOD examples found by the two methods in Fig. 8.

G. Running Time of Four Methods (Sec. 6.2)

From Tab. 2 and Tab. 6, it is clear that the four most competitive methods are ViM, Mahalanobis, KL Matching, and Residual. Our ViM is the fastest among all four methods. We show their inference time on the four datasets in Tab. 8.

References

[1] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. RepVGG: Making VGG-style ConvNets great again. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13733–13742, 2021. 12, 13

[2] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 12

[3] Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. *Advances in Neural Information Processing Systems*, 34, 2021. 14

[4] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2019. 12, 13

[5] Dan Hendrycks, Steven Basart, Mantas Mazeika, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. *arXiv preprint arXiv:1911.11132*, 2019. 13, 14

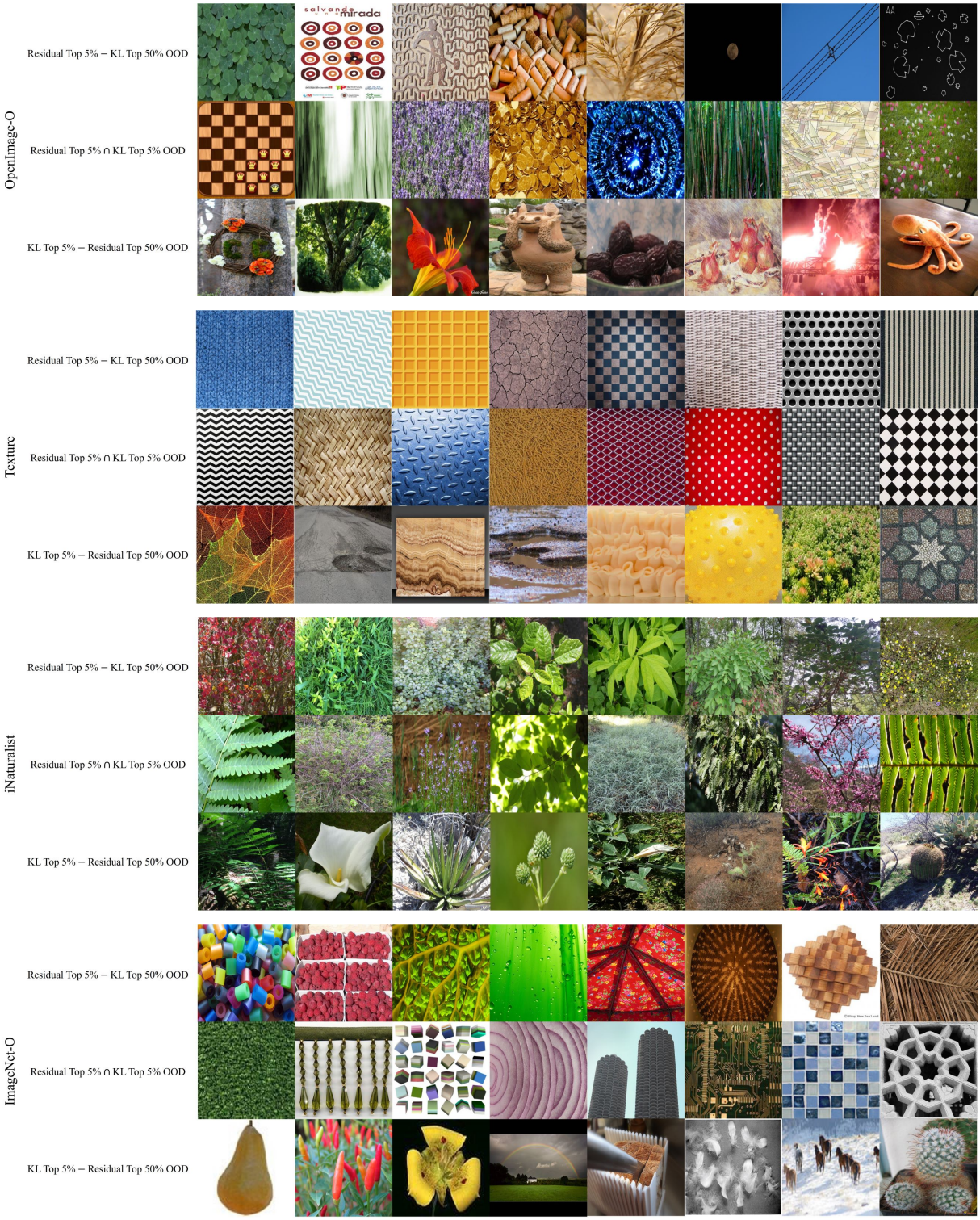

Figure 8. OOD examples detected by Residual and KL Matching. There are three rows for each OOD dataset. The first row shows images from the top 5% OODs detected by Residual, with overlapping images in the top 50% list of KL Matching removed. The second row displays images from the intersection of the top 5% OODs detected by Residual and the top 5% OODs detected by KL Matching. The third row shows images from the top 5% OODs detected by KL Matching, with overlapping ones in the top 50% list of Residual removed.

[6] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017. 13

[7] Rui Huang and Yixuan Li. MOS: Towards scaling out-of-distribution detection for large semantic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8710–8719, 2021. 11, 14

[8] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (BiT): General visual representation learning. In *European Conference on Computer Vision*, pages 491–507. Springer, 2020. 12

[9] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in Neural Information Processing Systems*, 31, 2018. 13

[10] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018. 13

[11] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 33:21464–21475, 2020. 13

[12] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 12, 13

[13] Yiyou Sun, Chuan Guo, and Yixuan Li. ReAct: Out-of-distribution detection with rectified activations. *Advances in Neural Information Processing Systems*, 34, 2021. 13, 14

[14] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 12, 13