Domain Feature Collapse: Implications for Out-of-Distribution Detection and Solutions

Anonymous submission

Abstract

The deployment of deep neural networks in safety-critical domains necessitates robust out-of-distribution (OOD) detection. While existing methods demonstrate strong performance on diverse multi-domain benchmarks, they are not specifically tailored to real-world applications where in-distribution data often originates from a single, narrow domain. This paper identifies and theoretically proves the existence of a critical issue that is unique to single-domain settings: *domain feature collapse*. We show, through information theory and bottleneck compression, that models trained on a single domain dataset with a class label objective learn representations that discard domain-specific features, relying predominantly on class features. This collapse results in significantly higher error rates for out-of-domain OOD detection, a failure mode not adequately captured by traditional multi-domain benchmarks. To combat this, we introduce a two-stage domain filtering process as a simple and effective solution. Domain filtering leverages an initial stage to determine if an input is in-domain before subsequently applying a standard OOD detector. Additionally, we propose the Domain Bench, a new benchmark comprised of various single-domain datasets, providing empirical evidence for domain feature collapse as well as a testbed for evaluating solutions. Our experimental results validate our theoretical findings, showing that existing methods struggle with out-of-domain detection in single-domain contexts and that incorporating domain filtering substantially improves performance in these scenarios, underscoring its importance for reliable OOD detection in single-domain applications.

1 Introduction

The deployment of deep neural networks (DNNs) in safety-critical domains – such as autonomous driving (Ramanagopal et al. 2018), biometric authentication (Wang and Deng 2021), and medical diagnostics (Bakator and Radosav 2018) – has spurred growing interest in ensuring their reliability. In these contexts, the traditional closed-world assumption (Krizhevsky, Sutskever, and Hinton 2012), where training and test data are drawn i.i.d. from the same in-distribution (ID), is no longer valid. Instead, models must operate under open-world conditions (Drummond and Shearer 2006), where inputs encountered at test time may stem from entirely different, out-of-distribution (OOD) sources. This reality necessitates robust OOD detection methods, which aim to flag inputs whose labels were not seen during training. The rationale is straightforward: we must prevent high-stakes systems from acting on predictions that are inherently invalid due to unseen or unfamiliar inputs.

Many state-of-the-art OOD detection methods demonstrate strong performance across established benchmarks (Zhang et al. 2023). However, these benchmarks almost exclusively use in-distribution sets that contain samples and classes from a wide variety of domains, such as (Zhang et al. 2023), which explicitly emphasize performance on CIFAR10/100 (Krizhevsky, Nair, and Hinton 2009) and ImageNet (Deng et al. 2009). While this approach provides a broad testbed for evaluating OOD robustness, it implicitly biases models and methods toward handling multi-domain in-distribution settings. As a result, there exists a gap in the literature: current OOD detection techniques are largely tailored to scenarios where the ID data is inherently diverse, rather than narrow or homogeneous. This raises concern as to how well these methods generalize to real-world applications where ID data may come from a single domain or task-specific distribution.

The single domain setting is understudied in bleeding edge OOD detection research, yet it has been heavily studied in the application of OOD methods for downstream tasks. Single-domain OOD detection is particularly important in application areas such as medical imaging (Zhang, Delbrouck, and Rubin 2021), satellite imagery (Ekim et al. 2024), and agriculture (Saadati et al. 2024), where models are often deployed in narrowly scoped environments with highly consistent data characteristics. This creates a domain mismatch between the theoretical efforts of the best OOD detection researchers and the practitioners who would benefit greatly from their research.

Surprisingly, there is a fatal flaw in OOD detection that only occurs in single domain settings. This paper introduces the concept of and theoretically proves the existence of **domain feature collapse**. Through information theory and bottleneck compression, we show that artificial neural networks will remove domain specific features from their learned representations, under the single domain setting. This leads to a situation where OOD detection relies solely on class-specific features, while ignoring domain-specific features (e.g., knowing that an image is an X-ray does not help in detecting the disease depicted by the X-ray, but would help in OOD detection). Unfortunately, this failure results in higher OOD detection error rates when the ID set is single domain versus multi-domain. Since this mode of OOD detection failure

rarely occurs for multi-domain ID sets, it is difficult to identify in the commonly used OOD benchmarks.

To address the issue of domain feature collapse, we introduce a new benchmark to evaluate the performance of OOD detection algorithms in the single domain setting. This benchmark covers data from a wide variety of singular domains, including medical imaging, agriculture, satellite imagery, and more. We also propose a simple but effective solution to domain feature collapse, a two stage domain filtering process. Our theory suggests that it is absolutely necessary to address the issue of domain feature collapse in order to ensure safe OOD detection in single domain settings.

Our key contributions are as follows:

- **Domain Feature Collapse**: Through bottleneck compression, we prove that training on a single domain results in a learned representation that contains no discriminative information regarding that particular domain, as this domain would be independent of the class within the context of the training dataset. We label this behavior as domain feature collapse, where a class label-supervised model will only learn class features and ignore domain features.
- **Domain Filtering**: We introduce a simple and consistent solution to the problem of domain feature collapse that allows any existing OOD detection algorithm to maintain high performance for both in-domain and out-of-domain OOD detection.
- **Domain Bench**: We introduce multiple single-domain datasets to provide empirical evidence of domain feature collapse across a wide variety of domains. We release source code to run these benchmarks following the OpenOOD framework (Zhang et al. 2023) in order to improve future research in single domain settings [1].

2 Preliminaries

2.1 Out-of-Distribution Detection

The task of out-of-distribution detection is to identify a semantic shift in the data (Yang et al. 2021). This is determining when no predicted label could match the true label $\mathbf{y} \notin \mathbb{Y}_{in}$, where $\mathbb{Y}_{in}$ represents the set of in-distribution training labels. In this case, we would consider the semantic space of the sample and the training distribution to be different; this represents a semantic shift. We can then express the probability that a sample is out-of-distribution as $P(\mathbf{y} \notin \mathbb{Y}_{in}|\mathbf{x})$. One baseline method to calculate $P(\mathbf{y} \notin \mathbb{Y}_{in}|\mathbf{x})$ is to take $1 - \texttt{MSP}(\mathbf{x})$, where $\texttt{MSP}$ is the maximum softmax probability extracted from a classifier for a particular datapoint.

Furthermore, we are only concerned with labels that can be generated using only $\mathbf{x}$, via function f which depends solely on $\mathbf{x}$ and no other information. Note that $f_{\mathbf{y}}$ may represent human labelers that generate $\mathbf{y}$. If we consider $\mathbb{Y}_{all}$ as the set of all possible labels that can be generated from $f_{\mathbf{y}}(\mathbf{x} \in \mathbb{X}_{all})$, a subset of $\mathbb{X}_{all}$ considered as $\mathbb{X}_{training}$ may not contain all labels in $\mathbb{Y}_{all}$. For real world datasets, it is possible that $\mathbb{Y}_{in} \subsetneq \mathbb{Y}_{all}$.

[1] see supplementary material for anonymized repository

2.2 Representation Learning and Bottleneck Compression

In Apendix A, we briefly review the information-theoretic properties used in this work.

Representation learning can be formulated as finding a distribution $p(\mathbf{z}|\mathbf{x})$ that maps the observations from $\boldsymbol{x} \in \mathbb{X}$ to $\boldsymbol{z} \in \mathbb{Z}$, while capturing relevant information for some primary task. When $\mathbf{y}$ represents some primary task, we consider only $\mathbf{z}$ that is sufficiently discriminative for accomplishing the task $\mathbf{y}$. For simplicity, we consider $\mathbf{y}$ as a classification label, but $\mathbf{y}$ can represent any objective or task. (Federici et al. 2020) show that this sufficiency is met when the information relevant for predicting $\mathbf{y}$ is unchanged when encoding $\mathbf{x} \to \mathbf{z}$.

Definition 2.1. *Sufficiency*: A representation $\mathbf{z}$ of $\mathbf{x}$ is sufficient for $\mathbf{y}$ if and only if $I(\mathbf{x}; \mathbf{y} \mid \mathbf{z}) = 0$.

Since there exists the sufficient statistic $\mathbf{x} = \mathbf{z}$, we must consider the minimal sufficient statistic which conveys only relevant information for predicting $\mathbf{y}$. A supervised learning algorithm will seek the minimal sufficient statistic via the information bottleneck framework (Tishby and Zaslavsky 2015), under idealized conditions.

Definition 2.2. *Minimal Sufficient Statistic*. A sufficient statistic $\mathbf{z}$ is minimal if, for any other sufficient statistic $\mathbf{s}$, there exists a function f such that $\mathbf{z} = f(\mathbf{s})$.

Information bottleneck optimization can be expressed as the minimization of the representation's complexity via $I(\mathbf{x}; \mathbf{z})$ while maximizing its utility $I(\mathbf{z}; \mathbf{y})$. This results in the information theoretic loss function below, where β is a trade-off between complexity and utility (Shwartz-Ziv and LeCun 2023). We can consider a supervised algorithm's loss function as a variation of the following function:

$$\mathcal{L} = I(\mathbf{x}; \mathbf{z}) - \beta I(\mathbf{z}; \mathbf{y}). \tag{1}$$

While real world conditions may not enforce the minimal sufficient statistic, e.g., overparameterization and lack of regularization, there still always exists some degree of compression such that $I(\mathbf{x}; \mathbf{z}) < H(\mathbf{x})$ (Shwartz-Ziv and Tishby 2017).

3 Methodology

3.1 Dataset Domain and Domain Features

We define the dataset's domain $\mathbf{d}$ as a value generated from some domain labeling function $f_{\mathbf{d}}(\mathbf{x})$. For the purposes of this paper, we are primarily concerned with cases where the data comes from a single domain $\mathbf{d}_1$, such that $\forall \mathbf{x} \in \{\mathbf{x} : f_{\mathbf{y}}(\mathbf{x}) \in \mathbb{Y}_{in}\}, f_{\mathbf{d}}(\mathbf{x}) = \mathbf{d}_1$. In these situations, we can conclude that $\forall \mathbf{x} \in \{\mathbf{x} : f_{\mathbf{d}}(\mathbf{x}) \neq \mathbf{d}_1\}, f_{\mathbf{y}}(\mathbf{x}) \notin \mathbb{Y}_{in}$, since any data outside of the domain cannot possibly have an in-distribution label. For this paper, we define domain features $\mathbf{x_d}$ such that they do not overlap with class features $\mathbf{x_y}$, implying $I(\mathbf{x_d}; \mathbf{x_y}) = 0$. The independence of domain and class features only applies to the training set, as domain features would provide useful information in the context of $\mathbb{X}_{all}$. Note that this also implies that $\neg(\forall \mathbf{x}, f_{\mathbf{y}}(\mathbf{x_y}) = f_{\mathbf{y}}(\mathbf{x}))$ and $\forall \mathbf{x}, f_{\mathbf{y}}(\mathbf{x_y}, \mathbf{x_d}) = f_{\mathbf{y}}(\mathbf{x})$. For both domain and class features, we refer to the minimal set of features, as per definition 2.2.

Definition 3.1. *Domain Features*. Given a dataset with domain $\mathbf{d}$ determined by the labeling function $f_{\mathbf{d}}(\mathbf{x})$, we define the domain features $\mathbf{x_d}$ as the minimal subset of features of $\mathbf{x}$ that is sufficient for $f_{\mathbf{d}}$, under the constraint that $\mathbf{x_d}$ is independent of the minimal sufficient class features $\mathbf{x_y}$, i.e., $I(\mathbf{x_d}; \mathbf{x_y}) = 0$.

Examples of single domain datasets could include a medical chest X-ray dataset (Yang et al. 2023), a geology dataset (Hossain et al. 2021), or a satellite imagery dataset (Helber et al. 2019). Further note that domains exist in a hierarchy; for instance, the domain of cats is a subdomain of mammals which is itself a subdomain of animals. This means that there is a domain that includes all things, but such a domain would have $\{\mathbf{x_d}\} = \emptyset$. For a wide domain with a wide variety of classes, we expect fewer domain features and more class features.

There exist datasets that could be labeled as a single domain $\mathbf{d}_1$ yet contain $|\{\mathbf{x_d}\}| \approx 0$. For example, if one were to treat ImageNet as a single domain, the set of domain features that do not overlap with class features is likely to be zero or nearly zero. We refer to such datasets as multi-domain datasets, as their diversity of classes require multiple domains.

3.2 Domain Feature Collapse

This section theoretically proves that any supervised model under a label-based training objective will learn a representation that contains no information on domain features, such that $I(\mathbf{x_d}, \mathbf{z}) = 0$, if full bottleneck compression occurs. This is considered to be strict domain feature collapse and relies on full information bottleneck compression:

Theorem 3.2. *Strict Domain Feature Collapse in the Minimal Sufficient Statistic.*
Let $\mathbf{x}$ come from a distribution. $\mathbf{x}$ is composed of two independent variables $\mathbf{x_d}$ and $\mathbf{x_y}$, where $\mathbf{x_d}$ is a set of domain features as per definition 3.1. Let $\mathbf{d}$ be a domain label generated from $f_{\mathbf{d}}(\mathbf{x_d}) = \mathbf{d}_1$, where $\mathbf{d}_1$ is a constant value for all $\mathbf{x}$. Let $\mathbf{y}$ be a class label generated from $f_{\mathbf{y}}(\mathbf{x_d}, \mathbf{x_y}) = \mathbf{y}$. Let $\mathbf{z}$ be any sufficient representation of $\mathbf{x}$ for $\mathbf{y}$ that satisfies the sufficiency definition 2.1 and minimizes the loss function $\mathcal{L} = I(\mathbf{x_d x_y}; \mathbf{z}) - \beta I(\mathbf{z}; \mathbf{y})$. The possible $\mathbf{z}$ that minimizes $\mathcal{L}$ and is sufficient must meet the condition $I(\mathbf{x_d}; \mathbf{z}) = 0$.

Detailed proof is given in Appendix B.4. Intuitively, the minimal sufficient representation cannot encode any information independent of the learning objective, otherwise it would not be minimal. Due to the definition 3.1 of $\mathbf{x_d}$ as domain features independent of class features, it is clear that compression results in the loss of domain features in the learned representation. This is contrary to the desired outcome, which is to learn $\hat{\mathbf{y}} = g(\mathbf{x_d}, \mathbf{x_y})$, as this would match the labeling function $\mathbf{y} = f_{\mathbf{y}}(\mathbf{x_d}, \mathbf{x_y})$. Instead, the model learns $\hat{\mathbf{y}} = g(\mathbf{x_y})$ because the domain features $\mathbf{x_d}$ are not predictive of the class in the context of the training data.

The lack of domain features is not problematic for safety purposes when $\forall \mathbf{x} \in \mathbb{X}_{all}, H(\mathbf{d}|\mathbf{x_y}) = 0$; it is safe when all out-of-domain data points contain no in-distribution class features. However, this is difficult to guarantee in an open world setting, as we do not possess information on the OOD distribution. For example, a model might learn that a Tyrannosaurus rex is a dinosaur that stands on two feet and proceed to classify "Barney" (a purple dinosaur character from a children's TV show) as a dinosaur, ignoring the fact that it is purple.

This issue can be further complicated by model overfitting, where a model may learn only a subset of $\mathbf{x_y}$ as opposed to the full set of features intended by the practitioner. Suppose we have a bird dataset made up of blue jays and cardinals. A model may only learn that blue jays are blue and assume that any blue object is a blue jay. Such a model would be safer if it could determine the domain of the blue object as a bird, before assuming it is a blue jay.

It should also be noted that full information bottleneck compression may not occur in real world scenarios, yet we can expect that some level of compression would still occur, as suggested by (Tishby and Zaslavsky 2015). In such cases, we can use Fano's Inequality to extend our theory of strict domain feature collapse onto partial compression cases. By Fano's Inequality, we would expect to observe unsafe and unreliable OOD detection conditions even with small $I(\mathbf{x_d}; \mathbf{z})$.

Theorem 3.3. *Fano's Inequality (See (Robert 1952)).*
Let $\mathbf{y}$ be a discrete random variable representing the true label with $\mathcal{Y}$ possible values and cardinality of $|\mathcal{Y}|$ and $\mathbf{x}$ be a random variable used to predict $\mathbf{y}$. Let e be the occurrence of an error such that $\mathbf{y} \neq \hat{\mathbf{y}}$ where $\hat{\mathbf{y}} = f(\mathbf{x})$. Let H_b represent the binary entropy function such that $H_b(e) = -P(e)\log P(e) - (1 - P(e))\log(1 - P(e))$. The lower bound for $P(e)$ increases with lower mutual information $I(\mathbf{x}; \mathbf{y})$.

$$H_b(e) + P(e)\log(|\mathcal{Y}| - 1) \geq H(\mathbf{y}) - I(\mathbf{x}; \mathbf{y}). \quad (2)$$

In summary, the critical safety risk in domain feature collapse is the existence of out-of-domain data that contains in-distribution features. This problem occurs when training with a class label-based learning objective combined with in ID data that consists of a single domain. This safety gap is under-studied in existing literature as current benchmarks use multi-domain datasets (e.g., CIFAR10/100 or ImageNet) as the in-distribution set, minimizing the amount of domain features $\mathbf{x_d}$ that are independent of the class features $\mathbf{x_y}$.

3.3 Limitations of Current Solutions

Fine Tuning. The use of fine tuning pretrained models is well-studied in OOD detection. Methods such as Energy (Liu et al. 2020) and MOS (Huang and Li 2021) utilize pretrained models to fine tune on ID data. However, fine tuning may not prevent domain feature collapse due to catastrophic forgetting (McCloskey and Cohen 1989), where the original pretrained weights are forgotten.

Pretrained Models. The use of pretrained models is also well-studied in OOD detection. Recent methods (Esmaeilpour et al. 2022) focus on zero-shot OOD using large artificial neural networks, such as CLIP (Radford et al. 2021).

However, work by (Yang, Yu, and Desell 2025) demonstrates that relying on zero-shot pretrained models for OOD detection is not effective for narrow domain data through the adjacent OOD detection benchmark. This benchmark considers the impact of out-of-distribution data that is of the same domain, e.g., a new type of disease, and is particularly difficult for pretrained models. (Yang, Yu, and Desell 2025) attributes this to a lack of relevant class features in the pretrained model due to the significant difference in the domain of the pretraining data and the ID data. In our experiments, we show that pretrained models struggle with the adjacent OOD benchmark when the in-distribution domain is narrow.

Self Supervised / Unsupervised / Unlabeled OOD Detection. Unlabeled OOD methods may address the issue of domain feature collapse by adding an information term to the loss function that explicitly encodes for $I(\mathbf{x_d}; \mathbf{z}) > 0$. These methods generally do not use labels and may use autoencoders (Zhou 2022), contrastive learning (Sehwag, Chiang, and Mittal 2021), or diffusion models (Liu et al. 2023). However, such methods may require adaptation to the target domain in order to properly capture relevant features. Furthermore, work by (Yang, Yu, and Desell 2025) has shown that these methods suffer similar issues as pretrained methods in the adjacent OOD detection benchmark, due to a lack of class features.

Auxiliary Loss Functions. Many of the unlabeled OOD methods explicitly optimize a learning objective that is not generated from the class labels. This can result in a situation where the learning objective may be closely aligned with the domain features needed to avoid domain feature collapse. However, the alignment of the auxiliary loss function and domain features is domain specific and would not generalize across different domains. This because the loss function learns domain features specific to domain A, which may be irrelevant for domain B.

3.4 Domain Filtering: A Solution

Two Stage Detector: Domain Filtering + OOD Detector To address the risk of domain feature collapse in supervised networks, we can utilize a two-stage process. In the first stage, a pretrained network is used to determine if a data sample is in-domain. In the second stage, an OOD detector is used to determine if in-domain samples are also in-distribution. This requires the assumption that there exists no in-distribution data sample that is out-of-domain, which is consistent with our earlier definitions.

This paper evaluates a K-nearest neighbors (KNN)-based domain filter, similar to a KNN-based OOD detector proposed by (Sun et al. 2022). To calibrate the domain filter, we calculate the domain threshold $\mathbf{t_d}$ such that $P(f_{knn}(\{\mathbf{x} \in \mathbb{X}_{train}\}) \leq \mathbf{t_d}) = p$, where f_{knn} is a KNN function considering the kth neighbor and p is a hyper parameter set to $p = 0.99$. Essentially, we select a distance such that 99% of the training data falls within that distance. The two stage process considers all samples with $f_{knn} > \mathbf{t}_d$ as OOD (due to it being out-of-domain) and uses the second stage detector to determine an OOD score for samples with $f_{knn} \leq \mathbf{t}_d$. This process ensures that only 1% of in-domain samples will be flagged as a false positive. See Algorithm 1 in Appendix D.

While there are alternative distance calculation methods and percentile thresholds available, this paper finds that a KNN filter at the 99th percentile with $K = 50$ works well as a first stage domain filter. In our experiments, we follow OpenOOD's hyperparameter tuning methods and also investigate two additional values $p = 0.98$ and $p = 0.999$. The two-stage process achieves significantly better results on out-of-domain OOD benchmarks while maintaining almost identical performance on in-domain OOD benchmarks.

Adjacent, Near, and Far OOD Benchmarks versus In and Out-of-Domain In most recent work, such as (Fort, Ren, and Lakshminarayanan 2021), and in the OpenOOD framework (Yang et al. 2022; Zhang et al. 2023), there is a distinction between near and far OOD. Near OOD refers to out-of-distribution samples that are semantically different from the training data but visually or structurally similar, e.g., similar textures or contexts. Far OOD refers to samples that are both semantically and visually dissimilar, often coming from completely unrelated domains.

However, by definition, both near and far OOD must be considered out-of-domain. If an in-distribution dataset is composed of a single domain, e.g., X-rays, where $\{\mathbf{x_d}\} \neq \emptyset$, existing near and far OOD benchmarks will be considered out-of-domain, as they would not be considered in the same domain by $f_{\mathbf{d}}$. We observe that the domain filter is very capable at detecting both near and far OOD benchmark datasets as out-of-domain.

This is in contrast to the adjacent OOD benchmark (Yang, Yu, and Desell 2025), which explicitly tests OOD detection performance on in domain samples that are out-of-distribution. The adjacent OOD benchmark constructs a new in-distribution set using a random subset of the training set classes. It then evaluates the OOD performance against the remaining training set classes as if they were OOD, allowing us to consider the impact of in-domain yet OOD samples. When used alone, the domain filter often performs poorly on the adjacent OOD benchmark, as it is unlikely to contain any class features. Work by (Yang, Yu, and Desell 2025) demonstrates that adjacent OOD is safety-critical because of the risk of unknown classes that come from the same domain.

Ensembling vs Filtering Ensemble methods have been used in uncertainty estimation and OOD detection before, such as (Lakshminarayanan, Pritzel, and Blundell 2017) and (Xu et al. 2024a). However, any ensemble would have to contend with a large performance gap between the two models. If we assume that the secondary model is good at out-of-domain OOD, its score would be dragged down by the primary model, which would be worse at out-of-domain OOD. Similarly, the primary model would be dragged down on in-domain OOD by the secondary model.

Domain filtering significantly reduces the negative impacts of ensembling by allowing the correct model to dominate the OOD score based on the domain of the sample. This allows us to maintain good in-domain OOD detection performance by limiting our negative impact on the primary model.

4 Experimental Results

4.1 Experimental Setup

For each narrow domain dataset, we generate a ID train, ID validation, ID test, and OOD test dataset using a unique seed. After training with ID data on one of the three methods below, we evaluate multiple OOD detection algorithms using the weights with the highest validation classification accuracy. For each OOD detection algorithm, we use the default postprocessor provided by OpenOOD (Zhang et al. 2023). We also implement two additional two-stage post processors, combining a pretrained DinoV2 ViTs14 (Oquab et al. 2023) with ReAct (Sun, Guo, and Li 2021) or KNN (Sun et al. 2022); more information can be found in Appendix E.

- **Cross Entropy Resnet50 (CE Resnet).** We fine tune a pretrained Resnet50 for 300 epochs using an SGD optimizer with an initial learning rate of 0.1.
- **Cross Entropy DinoV2 (CE DinoV2).** We fine tune a pretrained DinoV2 ViTs14 for 75 epochs using an Adam optimizer with an initial learning rate of 0.0001.
- **Supervised Contrastive Learning Resnet50 (SC Resnet).** We train a Resnet50 using supervised contrastive learning for 500 epochs using an SGD optimizer with an initial learning rate of 0.5 and a temperature of 0.5.

4.2 In and Out-of-Domain OOD Benchmarking

As noted in Section 3.4, all OOD datasets can be reduced into in-domain and out-of-domain. For the out-of-domain OOD benchmark, we use the following datasets as provided by OpenOOD (Zhang et al. 2023): MNIST (LeCun et al. 1998), SVHN (Netzer et al. 2011), Texture (Cimpoi et al. 2014), Places365 (Zhou et al. 2017), Cifar10/100 (Krizhevsky, Nair, and Hinton 2009), and Tiny Image Net (Deng et al. 2009). We also add samples from Chest X-rays (Yang et al. 2023) into the out-of-domain OOD benchmark, as it does not share a domain with the Tissue or Colon ID datasets.

To evaluate in domain OOD detection performance, we use the adjacent OOD benchmark proposed by (Yang, Yu, and Desell 2025). Because the adjacent OOD benchmark samples a subset of ID classes to be considered as OOD, we repeat our experiments 5 times with 5 different random seeds.

4.3 Variance Across Seeds

A consequence of repeating the adjacent OOD benchmark across multiple seeds is a significant variance in performance across seeds, due to selecting different classes as OOD. However, we observe that adding the domain filter consistently improves out-of-domain OOD performance while having very little impact on in-domain performance. We provide additional analysis on the statistical significance in Appendix F.4.

4.4 Narrow Domain Datasets

We use the following 11 narrow domain datasets as the in-distribution dataset to evaluate the impact of domain feature collapse. These datasets often contain class features that are quite subtle and difficult for pretrained models to pick up. They are also considered single domain as per our earlier domain definition. Additional descriptions and sample images are provided in Appendix G.

Butterfly – A butterfly species classification dataset (AI-Planet 2023).

Cards – A playing card classification dataset by rank and suit (Gerry 2023).

Colon – A colon pathology dataset with different diseases labeled (Yang et al. 2023).

Eurosat – A satellite images dataset for classifying different types of land use (Helber et al. 2019).

Fashion – The FashionMNIST dataset describing different articles of clothing (Xiao, Rasul, and Vollgraf 2017).

Food – The Food101 datasets (Bossard, Guillaumin, and Van Gool 2014) with 101 classes of different types of food.

Garbage – A dataset to classify the material of different waste objects (Single, Iranmanesh, and Raad 2023).

Plant – A plant leaves dataset detailing different types of disease (Hughes and Salathé 2015).

Rock – A dataset of different types of rocks and minerals (Hossain et al. 2021).

Tissue – A kidney cortex microscope dataset with various types of tissue labeled (Yang et al. 2023).

Yoga – A dataset of people performing different yoga poses from the internet (Sumanthvrao 2020).

4.5 Results

We report highlighted results in Tables 1, 2, and 3. These tables are a representative sample of the detailed results, see Appendix F. All methods have some level of difficulty in out-of-domain OOD detection, even though datasets like MNIST should not be challenging. On some in-distribution datasets, such as EuroSat, we observe that many methods obtain similar or better in-domain OOD detection performance compared to out-of-domain OOD detection performance. These results confirm the theoretical findings regarding domain feature collapse from Theorem 3.2.

It is also important to note that there is high variance for individual out-of-domain datasets when benchmarking. For example, when models are trained on the Colon ID dataset, they tend to struggle with detecting MNIST patterns as OOD. We observe that certain OOD sets are more problematic depending on the ID set. This suggests that only some OOD datasets contain features that could be similar to the ID class features, supporting the idea that domain feature collapse occurs from an over-reliance on class features and the inability to identify domain features.

In every case, adding a domain filter reduces the FPR@95 for out of domain OOD detection by a significant margin, sometimes trivializing the threat of out of domain OOD samples. For example, ReAct performs best in-domain on the Colon dataset, but suffers from extremely high FPR@95 of 61% on out of domain OOD samples. Adding a domain filter reduces this FPR rate to 0.7%, effectively eliminating the problem of out-of-domain OOD detection. In most cases, adding the domain filter reduces in domain performance by a marginal amount.

These results demonstrate that domain feature collapse is a real problem across a wide variety of datasets. We also ob-

serve that domain filtering is a generally applicable solution to address domain feature collapse.

5 Related Work

Out-of-Distribution Detection. Out-of-distribution (OOD) detection refers to identifying inputs that exhibit a semantic shift—namely, whose labels are not present during training (Yang et al. 2021). This capability is essential in high-stakes domains such as autonomous driving, medical imaging, and industrial systems (Huang et al. 2020). The baseline approach by Hendrycks and Gimpel (2016), which uses softmax confidence, sparked a wave of improved methods. For instance, ODIN introduced input perturbations and temperature scaling to better separate in- and out-distribution samples (Liang, Li, and Srikant 2017), while Lee et al. (2018) proposed Mahalanobis-distance scoring using intermediate network features. Liu et al. (2020) later introduced the energy score, offering a theoretically motivated alternative aligned with neural network logit functions. Subsequent innovations include MOS (Huang and Li 2021), utilizing logit margins, and Deep Adjacent Neighbors (Sun et al. 2022), which leverages feature-space neighborhood consistency using contrastive learning. Recent work by Liu and Qin (2025) investigates OOD detection through the lens of neural collapse, revealing that collapsed class means in deep networks produce highly discriminative directions that can distinguish OOD inputs. Their method leverages the geometry of learned representations, showing that deviations from the collapsed manifold signal OOD behavior. In a complementary direction, Xu et al. (2023) introduce SCALE, a simple and effective post-hoc technique that enhances OOD detection by scaling network activations. They further propose Intermediate Tensor Shaping (ISH) for training-time enhancement, jointly improving ID performance and OOD robustness with minimal computational overhead.

Single Domain Out-of-Distribution Detection. While most theoretical work in OOD detection is in the multi-domain setting, applied research in OOD detection often occurs in single domain settings. Some of these applications are in medical imaging (Narayanaswamy et al. 2023; Zhang, Delbrouck, and Rubin 2021; Cao et al. 2020), satellite imagery (Le Bellier and Audebert 2024; Gawlikowski et al. 2021), agriculture (Saadati et al. 2024; Li et al. 2023), and industrial systems (Kim, Cho, and Lee 2021; Kafunah et al. 2023).

Information Theory. Information theory has long played a foundational role in machine learning, providing theoretical tools for understanding generalization, compression, and representation learning. Shannon's entropy and mutual information are widely used for feature selection, regularization, and learning disentangled representations (Shannon 1948; Cover and Thomas 2006). The Information Bottleneck (IB) principle introduced by Tishby et al. (Tishby, Pereira, and Bialek 2000) has inspired various deep learning frameworks, such as the variational IB (Alemi et al. 2016), which approximates the trade-off between compression and prediction. Mutual information estimation techniques have also become critical in unsupervised and self-supervised learning, as in Deep InfoMax (Hjelm et al. 2018) and contrastive learning methods like CPC (Oord, Li, and Vinyals 2018). Moreover, recent work has connected generalization in deep networks to information-theoretic quantities, suggesting that flat minima and compression during training can explain generalization (Shwartz-Ziv and Tishby 2017; Achille and Soatto 2018).

6 Discussion: On the Limitations of Domain Filtering

Wide Domains. On some datasets, DinoV2 Domain Filtering has difficulty with outliers, resulting in a very large distance threshold $t_{\mathbf{d}}$ and poor domain filter performance. The Rock dataset (Hossain et al. 2021) would often set $t_{\mathbf{d}} \approx 1.78$, compared with the Colon dataset at $t_{\mathbf{d}} \approx 0.47$ and the Food dataset at $t_{\mathbf{d}} \approx 1.08$. By changing $p = 0.99 \rightarrow 0.98$, we can reduce $FPR@95 = 52.5 \rightarrow 27.9$ for the Rock dataset on out-of-domain OOD detection. One example of an outlier in the Rock dataset is an image of a marble countertop, as shown in Figure 9 in the Appendix. See Appendix F.3 for a more detailed analysis of percentiles.

Performance Cap. One major problem with domain filtering is the strict nature of its false positive rate. For in-domain data that is of a similar distribution to the training data, we expect a minimum false positive rate equal to $FPR = 1 - p$. We find that increasing p works well if the domain is narrow, but can significantly harm out-of-domain performance if there are outliers; see Appendix F.3.

Unseen Domains. Readers may question the viability of domain filtering when both the ID set and OOD set are unknown to the domain filtering model. In other words, since a pretrained DinoV2 model has seen such a wide variety of images, it may have already seen images similar to those in the out of domain OOD set. To address this concern, we included the chest Xray dataset (Yang et al. 2023) to show that a pretrained DinoV2 can filter between two unseen medical domains quite well (achieving 0.1 FPR@95 with the colon dataset as ID and Chest Xrays as OOD).

7 Conclusion

In this paper, for the problem of out-of-distribution (OOD) detection, we have theoretically proven the existence of a phenomenon that we label as domain feature collapse. Furthermore, we empirically demonstrated its existence through experimental simulation across a wide variety of single domain datasets. Notably, we introduced a new benchmark for evaluating OOD detectors in the under-explored single domain setting, including diverse data such as medical imaging, agriculture, and satellite imagery. We proposed domain filtering, a simple, easily-integrable, two-stage OOD detection process, as a potential counter-measure to domain feature collapse. We hope that this effort encourages further study into single domain out-of-distribution detection and improvements in AI safety.

Table 1: Summary OOD Performance Across All Datasets Reported As (In-Domain OOD Score)/(Out-of-Domain OOD Score). We exclude the Rock dataset from this summary as it is an outlier for reasons explained in Section 6. Best scores are in bold and second best are bold and italicized. The domain filter methods are italicized. SC Resnet is a not-compatible with OOD methods that uses logits. See Appendix F.1 for OOD detection method descriptions and Appendix F for more detailed results.

	FPR@95 (Lower is Better)			AUROC (Higher is Better)		
Method	CE DinoV2	CE Resnet	SC Resnet	CE DinoV2	CE Resnet	SC Resnet
PT KNN	79.7 / **0.9**	79.7 / **0.9**	79.7 / **0.9**	65.1 / **99.6**	65.1 / **99.6**	65.1 / **99.6**
MSP	65.4 / 43.0	61.8 / 38.9	NA	75.1 / 82.0	78.3 / 87.4	NA
Energy	65.0 / 37.3	65.3 / 41.4	NA	75.3 / 85.6	78.0 / 87.6	NA
Mahalanobis	***62.5*** / 18.5	**59.9** / 16.2	62.3 / 34.7	***75.9*** / 93.4	**78.4** / 94.4	**78.9** / 87.6
Scale	65.0 / 37.3	65.3 / 41.4	NA	75.3 / 85.6	78.0 / 87.6	NA
NCI	66.7 / 35.3	74.5 / 36.1	NA	74.1 / 86.6	73.3 / 88.5	NA
KNN	61.9 / 25.4	64.4 / 25.8	**61.5** / 32.9	75.8 / 91.0	76.1 / 91.1	78.0 / 87.8
ReAct	64.2 / 36.4	71.9 / 47.7	NA	75.9 / 86.3	74.4 / 84.9	NA
DF + KNN	65.2 / 3.2	64.3 / ***2.5***	63.8 / ***3.2***	73.9 / ***99.0***	75.8 / ***99.2***	76.2 / ***99.0***
DF + ReAct	64.3 / 3.7	72.3 / 4.1	NA	75.7 / 98.8	73.8 / 99.1	NA
DF + MDS	**62.1** / ***2.9***	***60.0*** / 11.8	***62.4*** / 11.6	**76.1** / ***99.0***	**78.6** / 96.3	***78.1*** / 95.2

Table 2: Summary FPR@95 OOD Performance Across All Datasets for Selected ID Datasets Reported As (In-Domain OOD Score)/(Out-of-Domain OOD Score). Best scores are in bold and second best are bold and italicized. See Appendix F.1 for OOD detection method descriptions and Appendix F for more detailed results.

method	Colon	Eurosat	Food	Garbage	Rock	Tissue
PT KNN	67.4 / **0.0**	69.1 / **0.3**	80.0 / **0.6**	87.2 / ***0.4***	91.9 / **6.6**	89.3 / **0.0**
MSP	59.1 / 53.0	**41.3** / 49.8	74.9 / 63.7	68.0 / 42.0	85.8 / 71.8	84.2 / 76.6
Energy	61.0 / 70.7	***42.5*** / 50.1	75.2 / 62.9	78.7 / 54.3	86.7 / 71.2	84.4 / 79.2
Mahalanobis	40.8 / 12.5	51.4 / 13.7	76.8 / 52.1	**59.8** / 13.9	***83.1*** / 44.2	91.4 / 3.8
Scale	61.0 / 70.7	***42.5*** / 50.1	75.2 / 62.9	78.7 / 54.3	86.7 / 71.2	84.4 / 79.2
NCI	74.5 / 24.8	72.7 / 57.1	80.4 / 65.4	74.2 / 31.4	75.9 / 64.0	84.5 / 35.7
KNN	40.0 / 13.2	48.4 / 31.3	**73.3** / 62.7	77.9 / 33.3	77.3 / 61.8	92.6 / 31.6
ReAct	**39.0** / 61.2	55.5 / 54.4	85.9 / 71.1	82.9 / 58.6	84.7 / 75.0	**81.7** / 48.0
DF + KNN	41.5 / ***0.2***	49.6 / ***1.5***	***73.5*** / 2.3	76.5 / 2.1	75.1 / 52.5	92.2 / ***0.4***
DF + ReAct	40.6 / 0.7	65.2 / 4.3	86.4 / ***2.2***	82.9 / ***1.8***	84.9 / 61.0	***81.9*** / 0.7
DF + MDS	***40.4*** / 6.9	51.4 / 10.4	76.6 / 39.1	**61.6** / 12.7	**82.0** / ***39.8***	91.3 / 0.9

Table 3: Detailed FPR@95 OOD Detection Performance for the Colon Dataset. See Appendix F.1 for OOD detection method descriptions. Best scores are in bold and second best are bold and italicized.

OOD Dataset Method	In Domain (Adjacent)	Chest	Cifar10	Cifar100	Mnist	Place365	Svhn	Texture	Tin
PT KNN	67.4	**0.1**	**0.0**	**0.0**	**0.0**	**0.0**	**0.1**	**0.0**	**0.0**
MSP	59.1	3.3	46.8	66.9	38.2	36.9	63.5	96.1	72.4
Energy	61.0	3.1	89.1	92.1	41.1	75.5	77.4	98.5	89.3
Mahalanobis	40.8	16.7	8.7	9.0	27.7	6.1	15.6	13.2	3.1
Scale	61.0	3.1	89.1	92.1	41.1	75.5	77.4	98.5	89.3
NCI	74.5	11.0	24.2	24.5	14.4	29.2	42.4	28.5	24.4
KNN	40.0	12.4	11.9	11.6	26.7	4.3	16.0	17.8	5.0
ReAct	**39.0**	41.0	62.6	64.2	45.7	52.4	77.2	74.2	72.3
DF + KNN	41.5	***0.2***	***0.2***	***0.2***	***0.3***	***0.1***	***0.2***	***0.3***	***0.1***
DF + ReAct	40.6	0.4	0.8	0.8	0.5	0.6	0.9	0.9	0.8
DF + MDS	***40.4***	8.9	4.6	4.7	15.5	3.3	8.6	7.3	1.8

References

Achille, A.; and Soatto, S. 2018. Emergence of invariance and disentanglement in deep representations. *The Journal of Machine Learning Research*, 19(1): 1947–1980.

AIPlanet. 2023. Data Sprint 107 – Butterfly Image Classification [Dataset]. https://aiplanet.com/challenges/325/butterfly_identification/overview/about. Accessed: 2025-05-09.

Alemi, A. A.; Fischer, I.; Dillon, J. V.; and Murphy, K. 2016. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.

Bakator, M.; and Radosav, D. 2018. Deep learning and medical diagnosis: A review of literature. *Multimodal Technologies and Interaction*, 2(3): 47.

Bossard, L.; Guillaumin, M.; and Van Gool, L. 2014. Food-101 – Mining Discriminative Components with Random Forests. In *European Conference on Computer Vision*.

Cao, T.; Huang, C.-W.; Hui, D. Y.-T.; and Cohen, J. P. 2020. A benchmark of medical out of distribution detection. *arXiv preprint arXiv:2007.04250*.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; and Vedaldi, A. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3606–3613.

Cover, T. M.; and Thomas, J. A. 2006. *Elements of Information Theory*. John Wiley & Sons.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Drummond, N.; and Shearer, R. 2006. The open world assumption. In *eSI Workshop: The Closed World of Databases meets the Open World of the Semantic Web*, volume 15, 1.

Ekim, B.; Tadesse, G. A.; Robinson, C.; Hacheme, G.; Schmitt, M.; Dodhia, R.; and Ferres, J. M. L. 2024. Distribution shifts at scale: Out-of-distribution detection in earth observation. *arXiv preprint arXiv:2412.13394*.

Esmaeilpour, S.; Liu, B.; Robertson, E.; and Shu, L. 2022. Zero-shot out-of-distribution detection based on the pre-trained model clip. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 6568–6576.

Federici, M.; Dutta, A.; Forré, P.; Kushman, N.; and Akata, Z. 2020. Learning robust representations via multi-view information bottleneck. *arXiv preprint arXiv:2002.07017*.

Fort, S.; Ren, J.; and Lakshminarayanan, B. 2021. Exploring the limits of out-of-distribution detection. *Advances in Neural Information Processing Systems*, 34: 7068–7081.

Gawlikowski, J.; Saha, S.; Kruspe, A.; and Zhu, X. X. 2021. Out-of-distribution detection in satellite image classification. *arXiv preprint arXiv:2104.05442*.

Gerry. 2023. Cards Image Dataset-Classification.

Helber, P.; Bischke, B.; Dengel, A.; and Borth, D. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Hendrycks, D.; and Gimpel, K. 2016. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*.

Hjelm, R. D.; Fedorov, A.; Lavoie-Marchildon, S.; Grewal, K.; Bachman, P.; Trischler, A.; and Bengio, Y. 2018. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*.

Hossain, S.; Uddin, J.; Nahin, R.; and Ibne Eunus, S. 2021. Rock Classification Dataset.

Huang, R.; and Li, Y. 2021. Mos: Towards scaling out-of-distribution detection for large semantic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8710–8719.

Huang, X.; Kroening, D.; Ruan, W.; Sharp, J.; Sun, Y.; Thamo, E.; Wu, M.; and Yi, X. 2020. A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review*, 37: 100270.

Hughes, D. P.; and Salathé , M. 2015. An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. *CoRR*, abs/1511.08060.

Kafunah, J.; Verma, P.; Ali, M. I.; and Breslin, J. G. 2023. Out-of-Distribution Data Generation for Fault Detection and Diagnosis in Industrial Systems. *IEEE Access*, 11: 135061–135073.

Kim, Y.; Cho, D.; and Lee, J.-H. 2021. Wafer defect pattern classification with detecting out-of-distribution. *Microelectronics Reliability*, 122: 114157.

Krizhevsky, A.; Nair, V.; and Hinton, G. 2009. CIFAR-10 and CIFAR-100 datasets.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Lakshminarayanan, B.; Pritzel, A.; and Blundell, C. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30.

Le Bellier, G.; and Audebert, N. 2024. Detecting Out-Of-Distribution Earth Observation Images with Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 481–491.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Lee, K.; Lee, K.; Lee, H.; and Shin, J. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31.

Li, D.; Yin, Z.; Zhao, Y.; Zhao, W.; and Li, J. 2023. MLFAnet: A Tomato Disease Classification Method Focusing on OOD Generalization. *Agriculture*, 13(6): 1140.

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

Liu, L.; and Qin, Y. 2025. Detecting Out-of-Distribution through the Lens of Neural Collapse. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liu, W.; Wang, X.; Owens, J.; and Li, Y. 2020. Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33: 21464–21475.

Liu, Z.; Zhou, J. P.; Wang, Y.; and Weinberger, K. Q. 2023. Unsupervised out-of-distribution detection with diffusion inpainting. In *International Conference on Machine Learning*, 22528–22538. PMLR.

McCloskey, M.; and Cohen, N. J. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, 109–165. Elsevier.

Narayanaswamy, V.; Mubarka, Y.; Anirudh, R.; Rajan, D.; and Thiagarajan, J. J. 2023. Exploring inlier and outlier specification for improved medical ood detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4589–4598.

Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; and Ng, A. Y. 2011. Reading digits in natural images with unsupervised feature learning.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Ramanagopal, M. S.; Anderson, C.; Vasudevan, R.; and Johnson-Roberson, M. 2018. Failing to learn: Autonomously identifying perception failures for self-driving cars. *IEEE Robotics and Automation Letters*, 3(4): 3860–3867.

Robert, M. 1952. Fano. Class Notes for MIT Course 6.574: Transmission of Information. *MIT, Cambridge, MA*, 8: 33.

Saadati, M.; Balu, A.; Chiranjeevi, S.; Jubery, T. Z.; Singh, A. K.; Sarkar, S.; Singh, A.; and Ganapathysubramanian, B. 2024. Out-of-distribution detection algorithms for robust insect classification. *Plant Phenomics*, 6: 0170.

Sehwag, V.; Chiang, M.; and Mittal, P. 2021. Ssd: A unified framework for self-supervised outlier detection. *arXiv preprint arXiv:2103.12051*.

Shannon, C. E. 1948. A mathematical theory of communication. *The Bell system technical journal*, 27(3): 379–423.

Shwartz-Ziv, R.; and LeCun, Y. 2023. To Compress or Not to Compress–Self-Supervised Learning and Information Theory: A Review. *arXiv preprint arXiv:2304.09355*.

Shwartz-Ziv, R.; and Tishby, N. 2017. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*.

Single, S.; Iranmanesh, S.; and Raad, R. 2023. Realwaste: A novel real-life data set for landfill waste classification using deep learning. *Information*, 14(12): 633.

Sumanthvrao. 2020. Yoga Poses. Version 6.

Sun, Y.; Guo, C.; and Li, Y. 2021. React: Out-of-distribution detection with rectified activations. *Advances in neural information processing systems*, 34: 144–157.

Sun, Y.; Ming, Y.; Zhu, X.; and Li, Y. 2022. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning*, 20827–20840. PMLR.

Tishby, N.; Pereira, F. C.; and Bialek, W. 2000. The information bottleneck method. *arXiv preprint physics/0004057*.

Tishby, N.; and Zaslavsky, N. 2015. Deep learning and the information bottleneck principle. In *2015 ieee information theory workshop (itw)*, 1–5. Ieee.

Wang, M.; and Deng, W. 2021. Deep face recognition: A survey. *Neurocomputing*, 429: 215–244.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. *CoRR*, abs/1708.07747.

Xu, C.; Yu, F.; Xu, Z.; Inkawhich, N.; and Chen, X. 2024a. Out-of-Distribution Detection via Deep Multi-Comprehension Ensemble. In Salakhutdinov, R.; Kolter, Z.; Heller, K.; Weller, A.; Oliver, N.; Scarlett, J.; and Berkenkamp, F., eds., *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 55465–55489. PMLR.

Xu, K.; Chen, R.; Franchi, G.; and Yao, A. 2023. Scaling for training time and post-hoc out-of-distribution detection enhancement. *arXiv preprint arXiv:2310.00227*.

Xu, K.; Chen, R.; Franchi, G.; and Yao, A. 2024b. Scaling for Training Time and Post-hoc Out-of-distribution Detection Enhancement. In *The Twelfth International Conference on Learning Representations*.

Yang, H.; Yu, Q.; and Desell, T. 2025. Can We Ignore Labels in Out of Distribution Detection? In *The Thirteenth International Conference on Learning Representations*.

Yang, J.; Shi, R.; Wei, D.; Liu, Z.; Zhao, L.; Ke, B.; Pfister, H.; and Ni, B. 2023. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1): 41.

Yang, J.; Wang, P.; Zou, D.; Zhou, Z.; Ding, K.; Peng, W.; Wang, H.; Chen, G.; Li, B.; Sun, Y.; et al. 2022. Openood: Benchmarking generalized out-of-distribution detection. *Advances in Neural Information Processing Systems*, 35: 32598–32611.

Yang, J.; Zhou, K.; Li, Y.; and Liu, Z. 2021. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*.

Zhang, J.; Yang, J.; Wang, P.; Wang, H.; Lin, Y.; Zhang, H.; Sun, Y.; Du, X.; Li, Y.; Liu, Z.; et al. 2023. Openood v1. 5: Enhanced benchmark for out-of-distribution detection. *arXiv preprint arXiv:2306.09301*.

Zhang, O.; Delbrouck, J.-B.; and Rubin, D. L. 2021. Out of distribution detection for medical images. In *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Perinatal Imaging, Placental and Preterm Image Analysis: 3rd International Workshop, UNSURE 2021, and 6th International Workshop, PIPPI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, October 1, 2021, Proceedings 3*, 102–111. Springer.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6): 1452–1464.

Zhou, Y. 2022. Rethinking reconstruction autoencoder-based out-of-distribution detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7379–7387.

Appendix

A Properties of Mutual Information and Entropy

In this Section we enumerate some of the properties of mutual information that are used to prove the theorems reported in this work, initially proposed by (Shannon 1948). For any random variables $\mathbf{w}, \mathbf{x}, \mathbf{y}$ and $\mathbf{z}$:

(P_1) Positivity:

$$I(\mathbf{x};\mathbf{y}) \geq 0, I(\mathbf{x};\mathbf{y} \mid \mathbf{z}) \geq 0$$

(P_2) Chain rule:

$$I(\mathbf{xy};\mathbf{z}) = I(\mathbf{y};\mathbf{z}) + I(\mathbf{x};\mathbf{z} \mid \mathbf{y})$$

(P_3) Chain rule (Multivariate Mutual Information):

$$I(\mathbf{x};\mathbf{y};\mathbf{z}) = I(\mathbf{y};\mathbf{z}) - I(\mathbf{y};\mathbf{z} \mid \mathbf{x})$$

(P_4) Positivity of discrete entropy: For discrete $\mathbf{x}$

$$H(\mathbf{x}) \geq 0, H(\mathbf{x} \mid \mathbf{y}) \geq 0$$

(P_5) Entropy and Mutual Information

$$H(\mathbf{x}) = H(\mathbf{x} \mid \mathbf{y}) + I(\mathbf{x};\mathbf{y})$$

(P_6) Conditioning a variable cannot increase its entropy

$$H(\mathbf{y}|\mathbf{z}) \leq H(\mathbf{y})$$

(P_7) A variable knows about itself as much as any other variable can

$$I(\mathbf{x};\mathbf{x}) \geq I(\mathbf{x};\mathbf{y})$$

(P_8) Symmetry of Mutual Information

$$I(\mathbf{x};\mathbf{y}) = I(\mathbf{y};\mathbf{x})$$

(P_9) Entropy and Conditional Mutual Information (This is simply P_5 conditioned on $\mathbf{z}$)

$$I(\mathbf{x};\mathbf{y}|\mathbf{z}) = H(\mathbf{x}|\mathbf{z}) - H(\mathbf{x}|\mathbf{yz})$$

(P_{10}) Functions of Independent Variables Remain Independent

$$I(\mathbf{x};\mathbf{y}) = 0 \rightarrow I(f(\mathbf{x});\mathbf{y}) = 0$$

B Main Theorems and Proofs

We ignore cases where the determined variable has an entropy of 0. Generally, if $H(\mathbf{y}|\mathbf{x}) = 0 \rightarrow H(\mathbf{y}) > 0$. Also, we only consider cases where the random variables have more than zero entropy.

Note that $R_{\mathbf{x}}$ represents the support of random variable $\mathbf{x}$ such that $R_{\mathbf{x}} = \{\boldsymbol{x} \in \mathbb{R} : P(\boldsymbol{x}) > 0\}$.

B.1 Lower Bound of Mutual Information for Sufficiency

Lemma B.1. *Let $\mathbf{x}$ and $\mathbf{y}$ be random variables with joint distribution $p(\mathbf{x},\mathbf{y})$. Let $\mathbf{z}$ be a representation of $\mathbf{x}$ that is sufficient, as per definition 2.1. Then $I(\mathbf{x};\mathbf{z}) \geq I(\mathbf{z};\mathbf{y})$ and $I(\mathbf{x};\mathbf{z}) \geq I(\mathbf{x};\mathbf{y})$.*

Hypothesis:

(H_1) *$\mathbf{z}$ is a representation of $\mathbf{x}$* : $I(\mathbf{y};\mathbf{z} \mid \mathbf{x}) = 0$

(H_2) *$\mathbf{z}$ is a sufficient representation of $\mathbf{x}$* : $I(\mathbf{x};\mathbf{y}|\mathbf{z})) = 0$

Thesis:

(T_1) $\forall \mathbf{z}. I(\mathbf{x};\mathbf{z}) \geq I(\mathbf{z};\mathbf{y}), I(\mathbf{x};\mathbf{z}) \geq I(\mathbf{x};\mathbf{y})$

Proof. By Construction

$$\begin{aligned} I(\mathbf{xy}|\mathbf{z})) &\overset{(H_2)}{=} 0 \\ &\overset{(P_2)}{=} I(\mathbf{zy};\mathbf{x}) - I(\mathbf{z};\mathbf{x}) \\ &\overset{(P_2)}{=} I(\mathbf{x};\mathbf{y}) + I(\mathbf{x};\mathbf{z}|\mathbf{y}) - I(\mathbf{z};\mathbf{x}) \\ &\overset{(PropB1)}{=} I(\mathbf{z};\mathbf{y}) + I(\mathbf{x};\mathbf{z}|\mathbf{y}) - I(\mathbf{z};\mathbf{x}) \\ I(\mathbf{z};\mathbf{x}) &= I(\mathbf{z};\mathbf{y}) + I(\mathbf{x};\mathbf{z}|\mathbf{y}) \\ I(\mathbf{z};\mathbf{x}) &\overset{(P_1)}{\geq} I(\mathbf{z};\mathbf{y}) \end{aligned}$$

Note that $I(\mathbf{z};\mathbf{y}) = I(\mathbf{x};\mathbf{y})$ for all sufficient representations, as per proposition C.1.

This supports our intuition that the information in the representation consists of relevant information $I(\mathbf{z};\mathbf{y})$ and irrelevant information $I(\mathbf{x};\mathbf{z}|\mathbf{y})$. By definition of sufficiency, there must be enough information for $I(\mathbf{z};\mathbf{y})$ in $I(\mathbf{x};\mathbf{z})$, which is to say that the size of the encoding cannot be smaller than the minimum size to encode all of $I(\mathbf{x};\mathbf{y})$.

□

B.2 Factorization of Bottleneck Loss

Lemma B.2. *Let $\mathbf{x}$ be a random variable with label $\mathbf{y}$ such that $H(\mathbf{y}|\mathbf{x}) = 0$ and $\mathbf{z}$ is a sufficient representation of $\mathbf{x}$ for $\mathbf{y}$. The loss function $\mathcal{L} = I(\mathbf{x};\mathbf{z}) - \beta I(\mathbf{z};\mathbf{y})$ is equivalent to $\mathcal{L} = H(\mathbf{z}) - \beta I(\mathbf{z};\mathbf{y})$, with β as some constant.*

Hypothesis:

(H_1) *$\mathbf{z}$ is fully determined by $\mathbf{x}$* : $H(\mathbf{z}|\mathbf{x}) = 0$

Thesis:

(T_1) $I(\mathbf{x};\mathbf{z}) - \beta I(\mathbf{z};\mathbf{y}) = H(\mathbf{z}) - \beta I(\mathbf{z};\mathbf{y})$

Proof. By Construction.

$$\begin{aligned} I(\mathbf{x};\mathbf{z}) - \beta I(\mathbf{z};\mathbf{y}) &\overset{(P_5)}{=} H(\mathbf{z}) - H(\mathbf{z}|\mathbf{x}) - \beta I(\mathbf{z};\mathbf{y}) \\ &\overset{(H_1)}{=} H(\mathbf{z}) - \beta I(\mathbf{z};\mathbf{y}) \end{aligned}$$

Due to the relationship between $\mathbf{x}$ and $\mathbf{z}$, we can create an intuitive factorization of the bottleneck loss function. Effectively, we want to maximize $I(\mathbf{z};\mathbf{y})$ while minimizing the information content of $\mathbf{z}$

□

B.3 Conditional Mutual Information of Noise

Lemma B.3. *Let* $\mathbf{x}$ *and* $\mathbf{y}$ *be independent random variables and* $\mathbf{z}$ *be a function of* $\mathbf{x}$ *with joint distribution* $p(\mathbf{x}, \mathbf{y}, \mathbf{z})$*. The conditional mutual information* $I(\mathbf{x}; \mathbf{z}|\mathbf{y})$ *is always equal to the mutual information* $I(\mathbf{x}; \mathbf{z})$*. As in the information content is unchanged when adding noise.*

Hypothesis:

(H_1) *Independence of* $\mathbf{x}$ *and* $\mathbf{y}$ *:* $I(\mathbf{x}; \mathbf{y}) = 0$

(H_2) $\mathbf{z}$ *is fully determined by* $\mathbf{x}$ *:* $H(\mathbf{z}|\mathbf{x}) = 0$

Thesis:

(T_1) $I(\mathbf{x}; \mathbf{z}|\mathbf{y}) = I(\mathbf{x}; \mathbf{z})$

Proof. By Construction.

(C_1) Demonstrates that $H(\mathbf{z}|\mathbf{xy}) = 0$

$$0 \overset{(P_4)}{\leq} H(\mathbf{z}|\mathbf{xy}) \overset{(P_6)}{\leq} H(\mathbf{z}|\mathbf{x})$$
$$H(\mathbf{z}|\mathbf{xy}) \overset{(H_2)}{\leq} 0$$

(C_2) Demonstrates that $I(\mathbf{z}; \mathbf{y}) = 0$

$$\begin{aligned} I(\mathbf{z}; \mathbf{y}) &\overset{(H_2)}{=} I(f(\mathbf{x}); \mathbf{y}) \\ &\overset{(P_{10})}{=} I(\mathbf{x}; \mathbf{y}) \\ &\overset{(H_1)}{=} 0 \end{aligned}$$

Thus

$$\begin{aligned} I(\mathbf{x}; \mathbf{z}|\mathbf{y}) &\overset{(P_9)}{=} H(\mathbf{z}|\mathbf{y}) - H(\mathbf{z}|\mathbf{xy}) \\ &\overset{(C_1)}{=} H(\mathbf{z}|\mathbf{y}) - 0 \\ &\overset{(P_5)}{=} H(\mathbf{z}) - I(\mathbf{z}; \mathbf{y}) \\ &\overset{(C_2)}{=} H(\mathbf{z}) - 0 \\ &\overset{(H_2)}{=} H(\mathbf{z}) - H(\mathbf{z}|\mathbf{x}) \\ &\overset{(P_5)}{=} I(\mathbf{x}; \mathbf{z}) \end{aligned}$$

This supports the intuition that if one added a random noise channel it will not change the mutual information.

□

B.4 Domain Feature Collapse

Theorem B.4. *Let* $\mathbf{x}$ *come from a distribution.* $\mathbf{x}$ *is composed of two independent variables* $\mathbf{x_d}$ *and* $\mathbf{x_y}$*, where* $\mathbf{x_d}$ *is a set of domain features as per definition 3.1. Let* $\mathbf{d}$ *be a domain label generated from* $f_{\mathbf{d}}(\mathbf{x_d}) = \mathbf{d}_1$*, where* $\mathbf{d}_1$ *is a constant value for all* $\mathbf{x}$*. Let* $\mathbf{y}$ *be a class label generated from* $f_{\mathbf{y}}(\mathbf{x_d}, \mathbf{x_y}) = \mathbf{y}$*. Let* $\mathbf{z}$ *be any sufficient representation of* $\mathbf{x}$ *for* $\mathbf{y}$ *that satisfies the sufficiency definition 2.1 and minimizes the loss function* $\mathcal{L} = I(\mathbf{x_d x_y}; \mathbf{z}) - \beta I(\mathbf{z}; \mathbf{y})$*. The possible* $\mathbf{z}$ *that minimizes* $\mathcal{L}$ *and is sufficient must meet the condition* $I(\mathbf{x_d}; \mathbf{z}) = 0$*.*

Hypothesis:

(H_1) $\mathbf{z}$ *is fully determined by* $\mathbf{x}$ *:* $H(\mathbf{z}|\mathbf{x}) = 0$

(H_2) $\mathbf{z}$ *is a representation of* $\mathbf{x}$ *:* $I(\mathbf{y}; \mathbf{z} \mid \mathbf{x}) = 0$

(H_3) $\mathbf{z}$ *is a sufficient representation of* $\mathbf{x}$ *:* $I(\mathbf{x}; \mathbf{y}|\mathbf{z}) = 0$

(H_4) $\mathbf{x}$ *is composed of two independent variables* $\mathbf{x_d}, \mathbf{x_y}$ *:* $\mathbf{x} = \mathbf{x_d}, \mathbf{x_y}, I(\mathbf{x_y}; \mathbf{x_d}) = 0$

(H_5) $\mathbf{y}$ *and* $\mathbf{d}$ *are fully determined by* $\mathbf{x_y}$ *and* $\mathbf{x_d}$*, respectively:* $H(\mathbf{y}|\mathbf{x_y}) = 0$*,* $H(\mathbf{d}|\mathbf{x_d}) = 0$*.*

Thesis:

(T_1) $\forall \mathbf{z}. I(\mathbf{x_d}, \mathbf{z}) = 0$

Proof. By Construction

(C_1) demonstrates that $\mathcal{L} = H(\mathbf{z}) - \beta I(\mathbf{z}; \mathbf{y})$ via factoring $I(\mathbf{x_d x_y}; \mathbf{z})$. Alternatively, Theorem B.2 creates the same result.

$$\begin{aligned} I(\mathbf{x_d x_y}; \mathbf{z}) &\overset{(P_2)}{=} I(\mathbf{x_y}; \mathbf{z}) + I(\mathbf{x_d}; \mathbf{z}|\mathbf{x_y}) \\ &\overset{(P_5)}{=} H(\mathbf{z}) - H(\mathbf{z}|\mathbf{x_y}) + I(\mathbf{x_d}; \mathbf{z}|\mathbf{x_y}) \\ &\overset{(P_9)}{=} H(\mathbf{z}) - H(\mathbf{z}|\mathbf{x_y}) + H(\mathbf{z}|\mathbf{x_y}) - H(\mathbf{z}|\mathbf{x_y x_d}) \\ &\overset{(H_1)}{=} H(\mathbf{z}) - H(\mathbf{z}|\mathbf{x_y}) + H(\mathbf{z}|\mathbf{x_y}) - 0 \\ \mathcal{L} &= H(\mathbf{z}) - \beta I(\mathbf{z}; \mathbf{y}) \end{aligned}$$

(C_2) Demonstrates that $I(\mathbf{z}; \mathbf{y}) = I(\mathbf{x}; \mathbf{y})$ as per Theorem C.1.

(C_3) Demonstrates that $I(\mathbf{z}; \mathbf{y})$ is a constant across all sufficient representations because Theorem C.1 applies.

(C_4) Demonstrates that for all possible $\mathbf{z}$ satisfying (H_3), their loss can be compared using only $\mathcal{L}_z = H(\mathbf{z})$ for comparing across $\mathbf{z}$

$$\begin{aligned} \frac{d\mathcal{L}}{d\mathbf{z}} &\overset{(C_1)}{=} \frac{H(\mathbf{z})}{d\mathbf{z}} - \frac{\beta I(\mathbf{z}; \mathbf{y})}{d\mathbf{z}} \\ &\overset{(C_3)}{=} \frac{H(\mathbf{z})}{d\mathbf{z}} - 0 \end{aligned}$$

(C_5) Demonstrates that the value of $H(\mathbf{z})$ at all possible $\mathbf{z}$ that minimizes $\mathcal{L}$ is the same. Even for different minimal $\mathbf{z}$, they must have the same $H(\mathbf{z})$ to all be minimal. When comparing possible minimal solutions to $\mathcal{L}$, $H(\mathbf{z})$ is constant across all minimal solutions.

(C_6) Demonstrates that any $\mathbf{z}$ that satisfies sufficiency must satisfy $I(\mathbf{z}; \mathbf{x}) \geq I(\mathbf{z}; \mathbf{y})$ and $I(\mathbf{z}; \mathbf{x}) \geq I(\mathbf{x}; \mathbf{y})$ as per Theorem B.1.

(C_7) Demonstrates that minima(s) exists only where $H(\mathbf{z}) = I(\mathbf{z}; \mathbf{y})$ and $H(\mathbf{z}|\mathbf{x}) = 0$. Note that $H(\mathbf{z}) = I(\mathbf{x}; \mathbf{y}) = I(\mathbf{z}; \mathbf{y})$ is the most compact representation size that is sufficient.

$$\begin{aligned} I(\mathbf{z}; \mathbf{x}) &\overset{(C_6)}{\geq} I(\mathbf{z}; \mathbf{y}) \\ H(\mathbf{z}) - H(\mathbf{z}|\mathbf{x}) &\overset{(P_5)}{\geq} I(\mathbf{z}; \mathbf{y}) \\ \forall \mathbf{z} \mid C_6 \wedge H_3 \wedge I(\mathbf{z}; \mathbf{x}) &> I(\mathbf{z}; \mathbf{y}). \\ \exists \mathbf{z}' \mid \mathbf{z}' = f(\mathbf{z}) \wedge I(\mathbf{z}; \mathbf{x}) &> I(\mathbf{z}'; \mathbf{x}) \wedge C_6 \wedge H_3 \end{aligned}$$

From (C_7) there exists only 3 types of minimas, separated by their dependence on the variables $\mathbf{x_y}, \mathbf{x_d}$. As per (H_1), any $\mathbf{z}$ must follow one of the 3 types.

1. Dependent only on $\mathbf{x_y}$: $\forall \mathbf{z} | H(\mathbf{z}|\mathbf{x_y}) = 0 \rightarrow I(\mathbf{x_d}; \mathbf{z}) = 0$

2. Dependent only on $\mathbf{x_d}$: $\forall \mathbf{z} | H(\mathbf{z}|\mathbf{x_d}) = 0 \rightarrow I(\mathbf{x_d};\mathbf{z}) > 0$
3. Dependent on both $\mathbf{x_y}\mathbf{x_d}$: $\forall \mathbf{z} | H(\mathbf{z}|\mathbf{x_y},\mathbf{x_d}) = 0 \wedge h(\mathbf{z}|\mathbf{x_y}) > 0 \wedge H(\mathbf{z}|\mathbf{x_d}) > 0 \rightarrow I(\mathbf{x_d};\mathbf{z}) > 0$

From here we will show that all type 2 and type 3 minimas always fail (H_3) or have greater $\mathcal{L}$ than any type 1 minima.

Type 1 $\mathbf{x_y}$: $\forall \mathbf{z} | H(\mathbf{z}|\mathbf{x_y}) = 0 \rightarrow I(\mathbf{x_d};\mathbf{z}) = 0$

(C_8) Demonstrates that there exists $H(\mathbf{z}) = I(\mathbf{z};\mathbf{y}) = I(\mathbf{x_y};\mathbf{z})$ and it is a set of minimas satisfying (C_7). This also establishes an upper bound for solutions to $\mathcal{L}$ due to (C_5). Therefore, any solution for type 1, type 2, and type 3 must satisfy $I(\mathbf{z};\mathbf{y}) \leq I(\mathbf{x_y};\mathbf{z})$ to be sufficient and $I(\mathbf{z};\mathbf{y}) = I(\mathbf{x_y};\mathbf{z})$ to be minimal.

$$
\begin{aligned}
I(\mathbf{z};\mathbf{y}) &\overset{(C_6)}{\leq} I(\mathbf{z};\mathbf{x}) \\
&\overset{(H_4)}{\leq} I(\mathbf{x_y},\mathbf{x_d};\mathbf{z}) \\
&\overset{(P_2)}{\leq} I(\mathbf{x_d};\mathbf{z}) + I(\mathbf{x_y};\mathbf{z}|\mathbf{x_d}) \\
&\overset{(Type1)}{\leq} 0 + I(\mathbf{x_y};\mathbf{z}|\mathbf{x_d}) \\
&\overset{(TheorumB.3)}{\leq} I(\mathbf{x_y};\mathbf{z}) \\
&\overset{(P_5)}{\leq} H(\mathbf{z}) - H(\mathbf{z}|\mathbf{x_y}) \\
\exists \mathbf{z} | I(\mathbf{x_y};\mathbf{z}) &= I(\mathbf{z};\mathbf{y}) = I(\mathbf{x};\mathbf{z}) = I(\mathbf{x};\mathbf{y})
\end{aligned}
$$

(C_9) Demonstrates that there exists no $H(\mathbf{z}') < H(\mathbf{z})$ that satisfies sufficiency if $\mathbf{z}$ satisfies (C_8) and is also $I(\mathbf{z};\mathbf{x_d}) = 0$.

$$
\begin{aligned}
C_8 &\rightarrow I(\mathbf{x_y};\mathbf{z}) = I(\mathbf{x};\mathbf{y}) \\
H(\mathbf{z}') < H(\mathbf{z}) &\rightarrow I(\mathbf{x_y};\mathbf{z}') < I(\mathbf{x_y};\mathbf{z}) \\
&\rightarrow \neg(C_2) : I(\mathbf{x_y};\mathbf{z}') < I(\mathbf{x_y};\mathbf{z}) = I(\mathbf{y};\mathbf{z}) = I(\mathbf{x};\mathbf{y})
\end{aligned}
$$

Type 2 $\mathbf{x_d}$: $\forall \mathbf{z} | H(\mathbf{z}|\mathbf{x_d}) = 0 \rightarrow I(\mathbf{x_d};\mathbf{z}) > 0$

(C_{10}) Demonstrates that no type 2 minima can exist, simply because it would contain no information regarding $\mathbf{x_y}$, thus failing to satisfy (H_3). This is because $\mathbf{z}$ cannot contain any information about $\mathbf{x_y}$, otherwise we would not satisfy $H(\mathbf{z}|\mathbf{x_d}) = 0$. If the representation $\mathbf{z}$ contains no information about $\mathbf{y}$, then it is not sufficient.

$$
\begin{aligned}
H(\mathbf{z}|\mathbf{x_d}) = 0 \rightarrow \mathbf{z} &= f(\mathbf{x_d}) \\
0 &\overset{(H_4)}{=} I(\mathbf{x_y};\mathbf{x_d}) \\
&\overset{(P_{10})}{=} I(f(\mathbf{x_y});\mathbf{x_d}) \\
&\overset{(H_5)}{=} I(\mathbf{y};\mathbf{x_d}) \\
&\overset{(P_{10})}{=} I(\mathbf{y};f(\mathbf{x_d})) \\
0 &= I(\mathbf{y};\mathbf{z})
\end{aligned}
$$

Type 3 $\mathbf{x_y},\mathbf{x_d}$: $\forall \mathbf{z} | H(\mathbf{z}|\mathbf{x_y},\mathbf{x_d}) = 0 \wedge H(\mathbf{z}|\mathbf{x_y}) > 0 \wedge H(\mathbf{z}|\mathbf{x_d}) > 0 \rightarrow I(\mathbf{x_d};\mathbf{z}) > 0$

(C_{11}) Demonstrates that any $\mathbf{z}$ that could be minimal must also satisfy (C_8) for sufficiency. Note that (C_8) implies that any $I(\mathbf{x_y};\mathbf{z}) > I(\mathbf{z};\mathbf{y})$ is not minimal.

$$
\begin{aligned}
I(\mathbf{z};\mathbf{y}) &\overset{(C_6)}{\leq} I(\mathbf{z};\mathbf{x}) \\
&\overset{(H_4)}{\leq} I(\mathbf{x_y}\mathbf{x_d};\mathbf{z}) \\
I(\mathbf{z};\mathbf{y}) &\overset{(P_2)}{\leq} I(\mathbf{x_y};\mathbf{z}) + I(\mathbf{x_d};\mathbf{z}|\mathbf{x_y}) \\
&(C_8) \rightarrow I(\mathbf{x_y};\mathbf{z}) = I(\mathbf{z};\mathbf{y})
\end{aligned}
$$

(C_{12}) Demonstrates that any $\mathbf{z}'$ where $I(\mathbf{z}';\mathbf{x_d}) > I(\mathbf{z};\mathbf{x_d})$ and $I(\mathbf{z};\mathbf{x_d}) = 0$ that maintains $H(\mathbf{z}') = H(\mathbf{z})$ results in solutions that are not sufficient as required by (H_3) because we know that the size of the representation must be at least $I(\mathbf{x};\mathbf{y})$ as defined in (C_6).

$$
\begin{aligned}
&C_8 \rightarrow H(\mathbf{z}) \text{ is constant across all minima} \\
&C_8 \rightarrow H(\mathbf{z}) = H(\mathbf{z}') \text{ for } \mathbf{z}' \text{ to be minimal} \\
&C_8 \rightarrow I(\mathbf{x_y};\mathbf{z}) = I(\mathbf{x};\mathbf{y}) \\
I(\mathbf{x_d};\mathbf{z}) = 0 &\rightarrow H(\mathbf{z}|\mathbf{x_y}) = 0 \\
\forall \mathbf{z}' | I(\mathbf{x_d};\mathbf{z}') > 0 &: H(\mathbf{z}'|\mathbf{x_y}) > H(\mathbf{z}|\mathbf{x_y}) \\
H(\mathbf{z}'|\mathbf{x_y}) &> H(\mathbf{z}|\mathbf{x_y}) \\
\rightarrow H(\mathbf{z}') - H(\mathbf{z}'|\mathbf{x_y}) &< H(\mathbf{z}) - H(\mathbf{z}|\mathbf{x_y}) \\
&\overset{(P_5)}{\rightarrow} I(\mathbf{x_y};\mathbf{z}') < I(\mathbf{x_y};\mathbf{z}) \\
&\rightarrow \neg(C_6) : I(\mathbf{x_y};\mathbf{z}') < I(\mathbf{x};\mathbf{y})
\end{aligned}
$$

(C_{13}) Demonstrates that combining (C_{11}) and (C_{12}), there is no type 3 solution that has an equal $\mathcal{L}$ to the minimal type 1 solution that also maintains sufficiency (H_3) and (C_6). This confirms the definition of entropy, in that encoding more independent information requires more bits or nats.

This means that only a type 1 solution can be both minimal and sufficient, which proves the thesis.

To summarize this proof, we can compare the losses of all sufficient solutions with $\mathcal{L} = H(\mathbf{z})$. Of those sufficient solutions, the one that minimizes $\mathcal{L}$ is the one with the smallest $H(\mathbf{z})$. The minimal sufficient representation is $\mathbf{z}$ that captures only all of $I(\mathbf{x_y};\mathbf{y})$ and nothing else. Thus the minimal $\mathbf{z}$ cannot have $I(\mathbf{x_d};\mathbf{z}) > 0$ because such $\mathbf{z}$ would encode information outside of $I(\mathbf{x_y};\mathbf{y})$.

□

C Theorems and Proofs of Previous Work

This Section contains the supporting theorems and proofs provided by previous work (Federici et al. 2020).

When random variable $\mathbf{z}$ is defined to be a representation of another random variable $\mathbf{x}$, we state that $\mathbf{z}$ is conditionally independent from any other variable in the system once $\mathbf{x}$ is observed. This does not imply that $\mathbf{z}$ must be a deterministic function of $\mathbf{x}$, but that the source of stochasticity for $\mathbf{z}$ is independent of the other random variables. As a result whenever $\mathbf{z}$ is a representation of $\mathbf{x}$:

$$I(\mathbf{z};\mathbf{a} \mid \mathbf{xb}) = 0,$$

for any variable (or groups of variables) $\mathbf{a}$ and $\mathbf{b}$ in the system. This condition is accounts for the randomness experienced in training neural networks and the error expected from human labelers. This condition applies to this and the following sections.

C.1 Sufficiency

Proposition C.1. *Let* $\mathbf{x}$ *and* $\mathbf{y}$ *be random variables from joint distribution* $p(\mathbf{x}, \mathbf{y})$*. Let* $\mathbf{z}$ *be a representation of* $\mathbf{x}$*, then* $\mathbf{z}$ *is sufficient for* $\mathbf{y}$ *if and only if* $I(\mathbf{x}; \mathbf{y}) = I(\mathbf{y}; \mathbf{z})$

Hypothesis:

(H_1) $\mathbf{z}$ *is a representation of* $\mathbf{x}$: $I(\mathbf{y}; \mathbf{z} \mid \mathbf{x}) = 0$

Thesis:

(T_1) $I(\mathbf{x}; \mathbf{y} \mid \mathbf{z}) = 0 \iff I(\mathbf{x}; \mathbf{y}) = I(\mathbf{y}; \mathbf{z})$

Proof.

$$I(\mathbf{x}; \mathbf{y} \mid \mathbf{z}) \overset{(P_3)}{=} I(\mathbf{x}; \mathbf{y}) - I(\mathbf{x}; \mathbf{y}; \mathbf{z})$$
$$I(\mathbf{x}; \mathbf{y}) - I(\mathbf{x}; \mathbf{y}; \mathbf{z}) \overset{(P_3)}{=} I(\mathbf{x}; \mathbf{y}) - I(\mathbf{y}; \mathbf{z}) + I(\mathbf{y}; \mathbf{z} \mid \mathbf{x})$$
$$\overset{(H_1)}{=} I(\mathbf{x}; \mathbf{y}) - I(\mathbf{y}; \mathbf{z})$$

Since both $I(\mathbf{x}; \mathbf{y})$ and $I(\mathbf{y}; \mathbf{z})$ are non-negative (P_1), $I(\mathbf{x}; \mathbf{y} \mid \mathbf{z}) = 0 \iff I(\mathbf{y}; \mathbf{z}) = I(\mathbf{x}; \mathbf{y})$

□

D Two Stage Domain Filter

Algorithm 1: Two-Stage Domain Filter for OOD Detection

Input:
$\mathbf{x}$: Input sample
$\mathbb{X}_{train}$: Training dataset
k: Number of neighbors (default=50)
$\mathbf{t_d}$: Domain threshold (99th percentile)
Output:
OOD decision $\in \{\text{True}, \text{False}\}$
procedure DOMAINFILTER($\mathbf{x}, \mathbb{X}_{train}, \mathbf{t_d}$)
    $d_k \leftarrow$ KNN-Distance$(\mathbf{x}, \mathbb{X}_{train}, k)$ ▷ k^{th} neighbor distance
    if $d_k > \mathbf{t_d}$ **then**
        return True ▷ Out-of-Domain
    else
        return False ▷ In-Domain
    end if
end procedure
procedure TWOSTAGEDETECTION(x)
    // Stage 1: Domain Filtering
    if DOMAINFILTER($\mathbf{x}, \mathbb{X}_{train}, \mathbf{t_d}$) **then**
        return True ▷ Reject as OOD (Avoids Domain Feature Collapse)
    end if
    // Stage 2: In-Distribution OOD Detection
    $s \leftarrow$ OOD-Score$(\mathbf{x})$ ▷ Using preferred OOD detector
    if $s > \tau$ **then** ▷ τ is OOD threshold
        return True
    else
        return False
    end if
end procedure
Threshold Calibration:
$\mathbf{t_d} \leftarrow$ Percentile$(\{f_{knn}(\mathbf{x}_i) | \mathbf{x}_i \in \mathbb{X}_{train}\}, 99\%)$

E Detailed Experimental Setup

E.1 Adjacent OOD Construction

For each seed, we randomly select $1/3$ of ID classes to be treated as in domain OOD classes. This is repeated 5 times per dataset, such that all 3 training methods use the same 5 seeds for their experiments.

E.2 Cross Entropy Resnet50

We train the Cross Entropy Resnet50 using the baseline training pipeline from OpenOOD. This pipeline uses an SGD optimizer with an initial LR of 0.1, momentum of 0.9, and a weight decay of 0.0005. We use a cosine annealing schedule for the learning rate. We train with a 256 batch size and an image size of 64. We use the OpenOOD base preprocesser for augmentations, which only includes a center crop, horizontal flip, and random crop. The ResNet50 is initialized with the default Torchvision weights, derived from Imagenet.

The model with the best accuracy on the validation set is selected for OOD evaluation.

We use the OpenOOD OODEvaluator class to evaluate OOD performance. Hyper parameters are selected using the ID validation set and the Tiny ImageNet validation set. Hyperparameters are selected using the configurations provided by OpenOOD. We limit the domain filter's possible k values to $[50, 100, 200]$.

E.3 Cross Entropy DinoV2

We train the Cross Entropy DinoV2 Vit-S14 using the baseline training pipeline from OpenOOD. We modify the pipeline to use an Adam optimizer with an initial LR of 0.00001 and a weight decay of 0.0005. We use a cosine annealing schedule for the learning rate. We train with a 128 batch size and an image size of 224. We use the OpenOOD base preprocesser for augmentations, which only includes a center crop, horizontal flip, and random crop.

The model with the best accuracy on the validation set is selected for OOD evaluation.

We use evaluation process as the Cross Entropy Resnet50.

E.4 Supervised Contrastive Learning ResNet50

We implement a Supervised Contrastive Learning pipeline in OpenOOD by following the implementation by (Sehwag, Chiang, and Mittal 2021). This pipeline uses an SGD optimizer with an initial LR of 0.5, momentum of 0.9, a weight decay of 0.0005, and a SimCLR temperature of 0.5. The model trains for 10 warm up epochs using a cyclic LR scheduler followed by 500 epoches using a cosine annealing LR scheduler. Preprocessing follows (Sehwag, Chiang, and Mittal 2021), where two augmented copies of an image are generated for contrastive learning, using RandomResizeCrop, RandomHorizontalFlip, ColorJitter, and GrayScale.

The model with the best accuracy on the validation set is selected for OOD evaluation, with accuracy established using a KNN fitted on the learned representations.

We use evaluation process as the Cross Entropy Resnet50, except all logit based OOD methods are not evaluated (due to the lack of a classification head).

F Detailed Experimental Results

F.1 OOD Method References

PT KNN refers to a KNN OOD detector (Sun et al. 2022) using only a pretrained DinoV2. DF + KNN refers to the two stage domain filter combined with an KNN OOD detector (Sun et al. 2022) and likewise with DF + ReAct (Sun, Guo, and Li 2021). Other listed methods are MSP (Hendrycks and Gimpel 2016), Energy (Liu et al. 2020), Mahalanobis (Lee et al. 2018), Scale (Xu et al. 2024b), NCI (Liu and Qin 2025), and KNN (Sun et al. 2022).

F.2 Experimental Results by ID Dataset and OOD Method

We provide FPR@95 and AUROC scores for each ID dataset and OOD detection method, across the 3 models. These results can be found in tables 4, 5, 6, 7, 8, and 9.

Table 4: FPR@95 Performance by Method and ID Dataset For Supervised Cross Entropy Trained Resnet50

method	Butterfly	Cards	Colon	Eurosat	Fashion	Food	Garbage	Plant	Rock	Tissue	Yoga
PT KNN	93.0 / 0.0	91.9 / 2.9	67.4 / 0.0	69.1 / 0.3	65.6 / 1.8	80.0 / 0.6	87.2 / 0.4	62.7 / 0.0	91.9 / 6.6	89.3 / 0.0	83.7 / 1.7
MSP	50.7 / 15.4	57.3 / 23.0	59.1 / 53.0	41.3 / 49.8	75.6 / 19.7	74.9 / 63.7	68.0 / 42.0	30.1 / 8.6	85.8 / 71.8	84.2 / 76.6	77.1 / 36.9
Energy	60.8 / 13.5	64.7 / 19.9	61.0 / 70.7	42.5 / 50.1	77.2 / 15.8	75.2 / 62.9	78.7 / 54.3	31.5 / 11.3	86.7 / 71.2	84.4 / 79.2	76.7 / 36.4
Mahalanobis	57.6 / 6.7	45.5 / 15.1	40.8 / 12.5	51.4 / 13.7	75.8 / 6.3	76.8 / 52.1	59.8 / 13.9	27.1 / 1.0	83.1 / 44.2	91.4 / 3.8	72.3 / 36.9
Scale	60.8 / 13.5	64.7 / 19.9	61.0 / 70.7	42.5 / 50.1	77.2 / 15.8	75.2 / 62.9	78.7 / 54.3	31.5 / 11.3	86.7 / 71.2	84.4 / 79.2	76.7 / 36.4
NCI	75.7 / 10.7	74.9 / 14.3	74.5 / 24.8	72.7 / 57.1	65.9 / 38.6	80.4 / 65.4	74.2 / 31.4	58.7 / 48.0	75.9 / 64.0	84.5 / 35.7	83.3 / 34.8
KNN	67.0 / 20.5	52.3 / 13.8	40.0 / 13.2	48.4 / 31.3	79.0 / 14.9	73.3 / 62.7	77.9 / 33.3	29.6 / 2.4	77.3 / 61.8	92.6 / 31.6	83.3 / 33.7
ReAct	81.7 / 40.5	71.2 / 21.7	39.0 / 61.2	55.5 / 54.4	82.0 / 37.4	85.9 / 71.1	82.9 / 58.6	63.0 / 44.4	84.7 / 75.0	81.7 / 48.0	75.8 / 39.2
DF + KNN	66.5 / 0.5	52.3 / 5.5	41.5 / 0.2	49.6 / 1.5	77.5 / 1.4	73.5 / 2.3	76.5 / 2.1	29.6 / 0.1	75.1 / 52.5	92.2 / 0.4	84.0 / 11.4
DF + ReAct	82.2 / 0.3	70.5 / 6.2	40.6 / 0.7	65.2 / 4.3	71.0 / 1.6	86.4 / 2.2	82.9 / 1.8	62.7 / 0.3	84.9 / 61.0	81.9 / 0.7	79.4 / 22.8
DF + MDS	56.7 / 4.7	45.5 / 13.0	40.4 / 6.9	51.4 / 10.4	76.3 / 3.2	76.6 / 39.1	61.6 / 12.7	26.7 / 0.8	82.0 / 39.8	91.3 / 0.9	73.0 / 26.1

Table 5: AUROC Performance by Method and ID Dataset For Supervised Cross Entropy Trained Resnet 50

method	Butterfly	Cards	Colon	Eurosat	Fashion	Food	Garbage	Plant	Rock	Tissue	Yoga
PT KNN	53.0 / 99.9	52.8 / 98.8	73.0 / 99.9	73.4 / 99.9	78.4 / 99.2	67.5 / 99.6	62.4 / 99.9	81.4 / 99.9	58.2 / 98.0	58.7 / 99.9	57.8 / 99.3
MSP	86.3 / 94.9	73.7 / 92.2	85.0 / 90.9	89.3 / 89.1	73.8 / 93.4	66.4 / 74.1	76.9 / 84.4	95.3 / 98.1	64.2 / 66.2	64.2 / 71.3	71.8 / 85.5
Energy	86.2 / 96.4	73.9 / 94.0	84.7 / 86.7	89.6 / 89.6	73.2 / 96.0	66.9 / 75.5	73.7 / 81.8	95.1 / 97.8	64.0 / 68.6	65.5 / 71.5	71.3 / 86.6
Mahalanobis	86.8 / 98.5	78.3 / 94.7	89.3 / 96.3	88.4 / 97.3	75.3 / 98.5	66.8 / 84.2	77.5 / 96.1	96.0 / 99.8	63.7 / 85.1	58.0 / 99.2	68.0 / 79.8
Scale	86.2 / 96.4	73.9 / 94.0	84.7 / 86.7	89.6 / 89.6	73.2 / 96.0	66.9 / 75.5	73.7 / 81.8	95.1 / 97.8	64.0 / 68.6	65.5 / 71.5	71.3 / 86.6
NCI	83.7 / 96.9	70.3 / 95.8	68.3 / 95.1	80.1 / 84.2	76.0 / 90.6	65.7 / 73.3	72.7 / 90.0	83.5 / 87.1	67.4 / 71.9	65.2 / 84.8	67.7 / 86.8
KNN	81.5 / 94.1	76.2 / 93.7	89.4 / 96.8	88.3 / 92.9	75.1 / 96.3	66.6 / 74.2	65.2 / 88.4	95.4 / 99.3	65.0 / 73.8	59.2 / 89.2	64.2 / 86.1
ReAct	74.0 / 88.1	71.8 / 93.9	86.3 / 78.6	86.0 / 87.9	67.1 / 89.8	64.8 / 74.8	68.2 / 81.3	85.4 / 89.8	63.8 / 65.2	68.3 / 79.5	71.8 / 85.7
DF + KNN	80.6 / 99.8	76.0 / 98.1	88.6 / 99.9	87.8 / 99.7	75.4 / 99.6	66.8 / 99.1	64.4 / 99.5	94.6 / 99.9	64.2 / 84.5	59.4 / 99.9	65.0 / 96.7
DF + ReAct	73.8 / 99.9	72.0 / 98.6	84.7 / 99.8	79.9 / 99.3	71.7 / 99.6	64.4 / 99.1	68.6 / 99.5	85.6 / 99.9	63.1 / 80.3	68.6 / 99.7	69.0 / 95.2
DF + MDS	86.9 / 99.0	78.6 / 95.6	89.3 / 97.9	88.4 / 97.9	75.7 / 99.2	66.9 / 89.5	77.4 / 96.3	96.0 / 99.8	63.0 / 87.4	58.3 / 99.8	68.3 / 87.8

Table 6: FPR@95 Performance by Method and ID Dataset For Supervised Cross Entropy Trained DinoV2

method	Butterfly	Cards	Colon	Eurosat	Fashion	Food	Garbage	Plant	Rock	Tissue	Yoga
PT KNN	93.0 / 0.0	91.9 / 2.9	67.4 / 0.0	69.1 / 0.3	65.6 / 1.8	80.0 / 0.6	87.2 / 0.4	62.7 / 0.0	91.9 / 6.6	89.3 / 0.0	83.7 / 1.7
MSP	64.4 / 44.6	64.0 / 38.1	58.9 / 11.4	57.8 / 19.4	65.7 / 52.2	69.6 / 53.5	81.2 / 60.5	21.2 / 5.2	87.2 / 87.0	83.6 / 77.6	85.6 / 62.6
Energy	69.7 / 39.5	62.5 / 26.4	54.8 / 16.2	54.6 / 15.8	70.9 / 46.9	67.9 / 41.1	81.9 / 55.8	20.4 / 3.5	87.3 / 85.8	81.6 / 64.3	84.0 / 59.2
Mahalanobis	74.8 / 36.7	68.2 / 18.6	37.6 / 0.3	27.5 / 6.4	67.7 / 4.9	76.5 / 37.3	75.8 / 27.5	26.0 / 2.1	90.8 / 62.6	87.6 / 0.2	76.4 / 49.1
Scale	69.7 / 39.5	62.5 / 26.4	54.8 / 16.2	54.6 / 15.8	70.9 / 46.9	67.9 / 41.1	81.9 / 55.8	20.4 / 3.5	87.3 / 85.8	81.6 / 64.3	84.0 / 59.2
NCI	68.4 / 38.8	62.5 / 26.9	55.9 / 13.3	54.2 / 19.8	71.1 / 32.9	67.7 / 42.2	83.7 / 48.1	21.5 / 3.9	85.6 / 76.7	93.0 / 65.5	86.2 / 58.6
KNN	77.6 / 48.7	69.7 / 21.7	31.2 / 1.6	29.8 / 23.4	68.5 / 8.4	70.8 / 48.4	79.2 / 45.8	19.4 / 3.1	90.0 / 72.1	87.1 / 1.4	79.2 / 51.2
ReAct	69.4 / 39.7	64.2 / 26.2	48.9 / 19.6	51.2 / 15.8	72.4 / 40.5	68.1 / 40.3	82.1 / 55.2	19.8 / 3.4	87.3 / 84.5	81.4 / 61.7	81.8 / 57.6
DF + KNN	87.5 / 0.8	82.6 / 8.8	33.3 / 0.1	33.1 / 0.8	68.9 / 2.7	72.4 / 2.1	79.7 / 1.4	19.8 / 0.1	91.5 / 57.9	87.4 / 0.1	81.1 / 14.3
DF + ReAct	69.3 / 0.7	63.7 / 8.8	49.1 / 0.3	52.6 / 0.8	71.9 / 5.0	67.8 / 1.9	81.3 / 1.9	20.5 / 0.1	87.3 / 74.8	81.5 / 0.8	83.2 / 16.1
DF + MDS	75.2 / 0.7	67.5 / 7.1	37.9 / 0.0	29.8 / 0.8	67.4 / 2.5	76.4 / 2.1	76.1 / 1.5	25.8 / 0.1	86.8 / 50.5	87.7 / 0.0	76.9 / 14.1

Table 7: AUROC Performance by Method and ID Dataset For Supervised Cross Entropy Trained DinoV2

method	Butterfly	Cards	Colon	Eurosat	Fashion	Food	Garbage	Plant	Rock	Tissue	Yoga
PT KNN	53.0 / 99.9	52.8 / 98.8	73.0 / 99.9	73.4 / 99.9	78.4 / 99.2	67.5 / 99.6	62.4 / 99.9	81.4 / 99.9	58.2 / 98.0	58.7 / 99.9	57.8 / 99.3
MSP	77.2 / 82.2	73.1 / 82.9	85.1 / 97.2	87.3 / 94.3	73.6 / 80.1	69.2 / 78.6	68.3 / 75.4	95.7 / 98.6	61.3 / 53.6	58.7 / 60.0	65.1 / 73.5
Energy	78.3 / 85.5	72.8 / 89.0	84.8 / 96.6	87.9 / 96.1	70.0 / 84.6	71.2 / 86.2	67.2 / 77.4	96.2 / 99.2	66.5 / 52.3	61.0 / 67.9	66.4 / 75.8
Mahalanobis	74.5 / 86.6	66.1 / 92.2	90.0 / 99.9	93.7 / 98.8	79.3 / 98.7	66.8 / 88.0	69.2 / 93.0	93.6 / 99.5	58.4 / 74.2	60.8 / 99.9	68.2 / 78.1
Scale	78.3 / 85.5	72.8 / 89.0	84.8 / 96.6	87.9 / 96.1	70.0 / 84.6	71.2 / 86.2	67.2 / 77.4	96.2 / 99.2	66.5 / 52.3	61.0 / 67.9	66.4 / 75.8
NCI	78.4 / 86.0	73.0 / 88.8	86.1 / 96.8	86.6 / 95.1	71.5 / 90.3	71.2 / 85.4	65.7 / 80.9	96.0 / 99.1	64.8 / 62.5	50.9 / 69.6	64.0 / 75.5
KNN	72.4 / 82.2	68.2 / 91.4	91.8 / 99.6	92.6 / 94.7	80.1 / 97.7	69.4 / 83.1	64.1 / 85.5	96.0 / 99.2	57.8 / 66.0	60.0 / 99.7	66.4 / 78.0
ReAct	78.3 / 85.4	72.6 / 89.1	86.4 / 95.0	88.5 / 96.2	72.2 / 87.6	71.2 / 86.8	66.7 / 78.1	96.2 / 99.2	66.9 / 53.9	61.8 / 71.2	67.3 / 76.5
DF + KNN	66.7 / 99.7	61.5 / 96.5	90.4 / 99.9	91.3 / 99.8	79.7 / 99.2	68.8 / 99.2	63.2 / 99.5	95.3 / 99.9	56.5 / 82.8	60.1 / 99.9	66.0 / 96.0
DF + ReAct	77.8 / 99.7	72.7 / 96.9	86.2 / 99.9	88.1 / 99.8	73.0 / 98.3	71.4 / 99.3	67.3 / 99.4	95.5 / 99.9	65.7 / 75.0	61.5 / 99.6	66.3 / 94.9
DF + MDS	74.1 / 99.6	66.1 / 97.4	89.7 / 99.9	93.0 / 99.8	79.4 / 99.3	67.1 / 99.3	69.7 / 99.6	93.2 / 99.9	60.1 / 82.9	60.9 / 99.9	67.9 / 95.4

Table 8: FPR@95 Performance by Method and ID Dataset For Supervised Constrastive Learning Trained Resnet50

method	Butterfly	Cards	Colon	Eurosat	Fashion	Food	Garbage	Plant	Rock	Tissue	Yoga
KNN	72.0 / 38.1	80.9 / 23.5	28.1 / 21.8	34.1 / 20.6	59.6 / 23.2	65.7 / 44.0	84.6 / 70.2	41.2 / 3.7	88.3 / 84.0	90.2 / 61.8	70.9 / 36.1
DF + KNN	79.5 / 0.7	81.6 / 8.3	32.0 / 0.4	41.7 / 1.2	56.9 / 3.1	65.9 / 1.6	85.9 / 3.2	42.2 / 0.1	87.3 / 71.0	89.5 / 0.6	73.4 / 11.0
Mahalanobis	64.9 / 23.8	74.3 / 15.5	38.5 / 57.1	26.1 / 17.5	69.1 / 38.8	73.0 / 49.6	77.2 / 45.6	51.1 / 15.5	97.0 / 72.1	94.6 / 64.4	69.2 / 34.2
DF + MDS	64.7 / 21.6	73.2 / 15.0	37.1 / 1.0	25.1 / 4.2	70.2 / 5.9	72.6 / 12.5	77.0 / 29.0	48.1 / 0.4	96.5 / 64.6	91.5 / 2.2	66.9 / 24.2
PT KNN	93.0 / 0.0	91.9 / 2.9	67.4 / 0.0	69.1 / 0.3	65.6 / 1.8	80.0 / 0.6	87.2 / 0.4	62.7 / 0.0	91.9 / 6.6	89.3 / 0.0	83.7 / 1.7

Table 9: AUROC Performance by Method and ID Dataset For Supervised Constrastive Learning Trained Resnet50

method	Butterfly	Cards	Colon	Eurosat	Fashion	Food	Garbage	Plant	Rock	Tissue	Yoga
KNN	75.3 / 85.4	70.9 / 91.6	93.4 / 93.7	92.1 / 94.8	83.2 / 92.3	72.9 / 83.4	59.3 / 72.3	93.1 / 99.1	51.0 / 60.4	55.6 / 74.3	74.6 / 85.2
DF + KNN	65.8 / 99.7	69.6 / 97.6	91.7 / 99.9	88.0 / 99.7	83.9 / 99.1	72.9 / 99.2	58.2 / 99.3	92.2 / 99.9	49.3 / 79.2	57.2 / 99.8	74.1 / 96.3
Mahalanobis	83.5 / 90.9	73.3 / 93.0	89.7 / 80.6	94.3 / 95.2	79.8 / 86.2	70.8 / 81.8	67.2 / 84.4	90.6 / 96.6	47.0 / 68.0	51.3 / 74.8	75.4 / 86.0
DF + MDS	83.4 / 91.3	73.4 / 93.1	90.1 / 99.5	94.3 / 98.2	79.9 / 97.1	70.8 / 95.5	67.9 / 88.1	91.3 / 99.8	47.8 / 70.6	52.7 / 99.0	75.6 / 90.8
PT KNN	53.0 / 99.9	52.8 / 98.8	73.0 / 99.9	73.4 / 99.9	78.4 / 99.2	67.5 / 99.6	62.4 / 99.9	81.4 / 99.9	58.2 / 98.0	58.7 / 99.9	57.8 / 99.3

F.3 Discussion On Percentile for Domain Filtering

The effectiveness of the domain filter can be negatively impacted if the in domain distribution is wider than desired. In particular, the Rock dataset (Hossain et al. 2021) would often set $t_{\mathbf{d}} \approx 1.78$, compared with the Colon dataset at $t_{\mathbf{d}} \approx 0.47$ and the Food dataset at $t_{\mathbf{d}} \approx 1.08$. By changing $p = 0.99 \rightarrow 0.98$, we can reduce $FPR@95 = 52.5 \rightarrow 27.9$ for the Rock dataset on out-of-domain OOD detection. However, reducing the percentile p will inevitably result in more false positive rejections for in domain data.

In these situations, it may be more appropriate to investigate why the initial assumptions do not hold. Namely, we may want to consider whether or not our dataset is truly a narrow domain dataset and whether or not outliers within the ID data may have an larger than expected influence on the calculation of $t_{\mathbf{d}}$.

A comparison table of domain filtering at different percentiles for the rock dataset can be found in Table 10 and the average for all datasets excluding rock can be found in Table 11. Sample images from the Rock dataset are shown in Figure 9, which shows that these images can contain close up shots of rock patterns, but also rock formations in the wild. Interestingly, the dataset creators decided to include what appears to be a marble counter top as a member of the marble class.

F.4 Variance Analysis

We use a Wilcoxon Signed Rank test to determine the whether or not the improvement offered by domain filtering is statistically consistent, in Table 12. Due to the non normal distribution of FPR@95 values across seeds, we use a non parametric test, as opposed to the paired t test. We show that for each ID dataset, the average far (out of domain) OOD performance improves with domain filtering in each seed. This is a consistent result implying the domain filtering never harms far (out of domain) OOD performance.

G Single Domain Dataset Details

G.1 Butterfly

This is a dataset hosted by Kaggle originating from (AIPlanet 2023). It consists of 75 classes of Butterflies. It contains 2786 images. See Figure 1 for sample images.

G.2 Cards

This is a playing card classification dataset by rank and suit (Gerry 2023). This dataset is hosted on Kaggle and consists of 7624 images split into 53 categories. See Figure 2.

G.3 Colon

This is a colon pathology dataset with different diseases labeled (Yang et al. 2023). This dataset consists of 89,996 images in 9 different classes of colon disease. See Figure 3.

G.4 Eurosat

This is a satellite images dataset for classifying different types of land use (Helber et al. 2019). It contains 27,000 labeled images and only RGB images were used from the dataset. See Figure 4.

Table 10: Performance for the Rock Dataset across different OOD methods and training methods, considering differing percentiles for the domain filter. DF98 means domain filter at the 98th percentile and DF99.9 means domain filter at the 99.9th percentile. The default domain filter used in this paper is at the 99th percentile.

	FPR@95			AUROC		
	CE DinoV2	CE Resnet	SC Resnet	CE DinoV2	CE Resnet	SC Resnet
method						
PT KNN	91.9 / 6.6	91.9 / 6.6	91.9 / 6.6	58.2 / 98.0	58.2 / 98.0	58.2 / 98.0
MSP	87.2 / 87.0	85.8 / 71.8	NA	61.3 / 53.6	64.2 / 66.2	NA
Energy	87.3 / 85.8	86.7 / 71.2	NA	66.5 / 52.3	64.0 / 68.6	NA
Mahalanobis	90.8 / 62.6	83.1 / 44.2	97.0 / 72.1	58.4 / 74.2	63.7 / 85.1	47.0 / 68.0
Scale	87.3 / 85.8	86.7 / 71.2	NA	66.5 / 52.3	64.0 / 68.6	NA
NCI	85.6 / 76.7	75.9 / 64.0	NA	64.8 / 62.5	67.4 / 71.9	NA
KNN	90.0 / 72.1	77.3 / 61.8	88.3 / 84.0	57.8 / 66.0	65.0 / 73.8	51.0 / 60.4
ReAct	87.3 / 84.5	84.7 / 75.0	NA	66.9 / 53.9	63.8 / 65.2	NA
DF + KNN	91.5 / 57.9	75.1 / 52.5	87.3 / 71.0	56.5 / 82.8	64.2 / 84.5	49.3 / 79.2
DF + ReAct	87.3 / 74.8	84.9 / 61.0	NA	65.7 / 75.0	63.1 / 80.3	NA
DF + MDS	86.8 / 50.5	82.0 / 39.8	96.5 / 64.6	60.1 / 82.9	63.0 / 87.4	47.8 / 70.6
DF98 + KNN	90.4 / 34.7	76.1 / 27.9	94.3 / 44.5	56.2 / 92.2	63.5 / 93.7	47.0 / 90.9
DF99.9 + KNN	91.9 / 71.1	76.0 / 60.1	87.9 / 82.2	56.7 / 71.5	64.5 / 77.2	50.1 / 66.0

Table 11: Performance Average for all Dataset (Excluding Rock) across different OOD methods and training methods, considering differing percentiles for the domain filter. DF98 means domain filter at the 98th percentile and DF99.9 means domain filter at the 99.9th percentile. The default domain filter used in this paper is at the 99th percentile.

	FPR@95			AUROC		
	CE DinoV2	CE Resnet	SC Resnet	CE DinoV2	CE Resnet	SC Resnet
method						
PT KNN	79.7 / 0.9	79.7 / 0.9	79.7 / 0.9	65.1 / 99.6	65.1 / 99.6	65.1 / 99.6
MSP	65.4 / 43.0	61.8 / 38.9	NA	75.1 / 82.0	78.3 / 87.4	NA
Energy	65.0 / 37.3	65.3 / 41.4	NA	75.3 / 85.6	78.0 / 87.6	NA
Mahalanobis	62.5 / 18.5	59.9 / 16.2	62.3 / 34.7	75.9 / 93.4	78.4 / 94.4	78.9 / 87.6
Scale	65.0 / 37.3	65.3 / 41.4	NA	75.3 / 85.6	78.0 / 87.6	NA
NCI	66.7 / 35.3	74.5 / 36.1	NA	74.1 / 86.6	73.3 / 88.5	NA
KNN	61.9 / 25.4	64.4 / 25.8	61.5 / 32.9	75.8 / 91.0	76.1 / 91.1	78.0 / 87.8
ReAct	64.2 / 36.4	71.9 / 47.7	NA	75.9 / 86.3	74.4 / 84.9	NA
DF + KNN	65.2 / 3.2	64.3 / 2.5	63.8 / 3.2	73.9 / 99.0	75.8 / 99.2	76.2 / 99.0
DF + ReAct	64.3 / 3.7	72.3 / 4.1	NA	75.7 / 98.8	73.8 / 99.1	NA
DF + MDS	62.1 / 2.9	60.0 / 11.8	62.4 / 11.6	76.1 / 99.0	78.6 / 96.3	78.1 / 95.2
DF98 + KNN	64.8 / 2.4	63.8 / 2.2	62.7 / 3.3	74.0 / 99.0	75.8 / 99.2	77.1 / 98.9
DF99.9 + KNN	63.6 / 9.0	64.8 / 7.7	63.2 / 7.7	75.2 / 97.5	75.9 / 98.0	77.5 / 98.0

Table 12: Wilcoxon signed-rank test p-values comparing FPR@95 in Far (Out of Domain) OOD detection using regular KNN and domain filtered KNN across models and ID datasets. Low p-values indicate statistically significant improvements from filtering. Note that the p value of 0.03125 indicates that in all 5 seeds, the domain filtered KNN method achieved lower FPR@95 than the regular KNN method. This is also the lowest possible p value with a paired sample size of 5.

	Tissue	Plant	Yoga	Colon	Garbage	Food	Fashion	Rock	Eurosat	Cards	Butterfly
CE DinoV2	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03
CE Resnet	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03
SC Resnet	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03

Figure 1: Sample images for the Butterfly dataset.

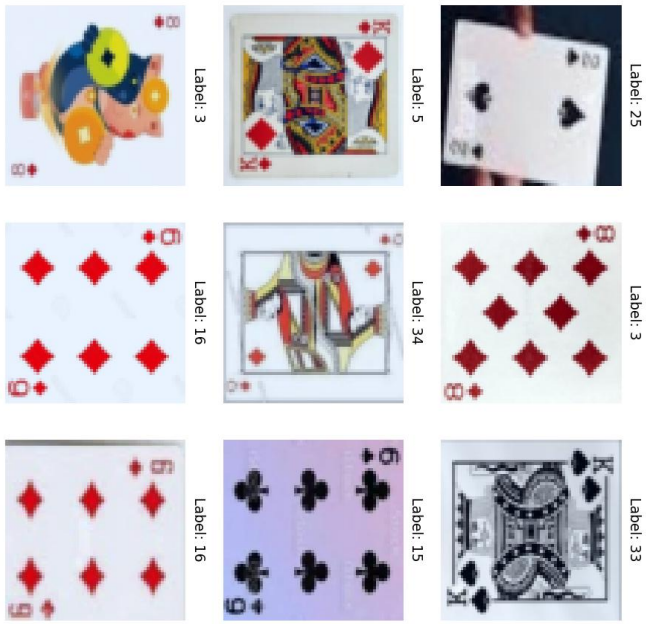

Figure 2: Sample images for the Cards dataset.

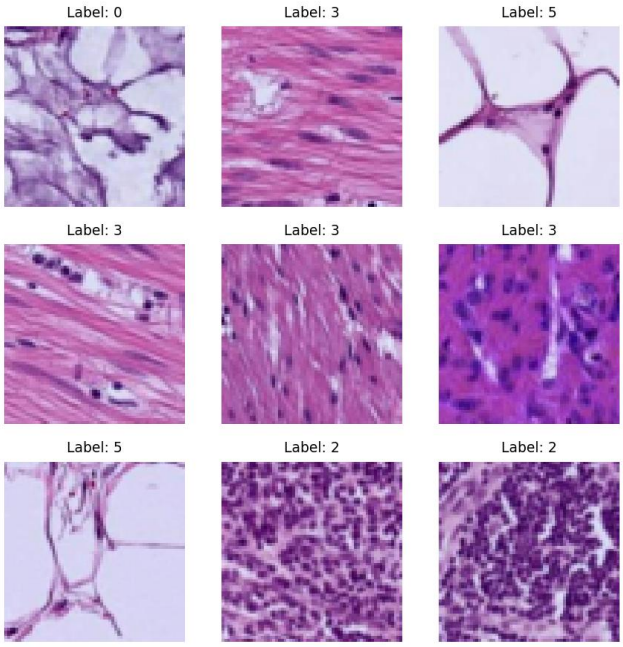

Figure 3: Sample images for the Colon dataset.

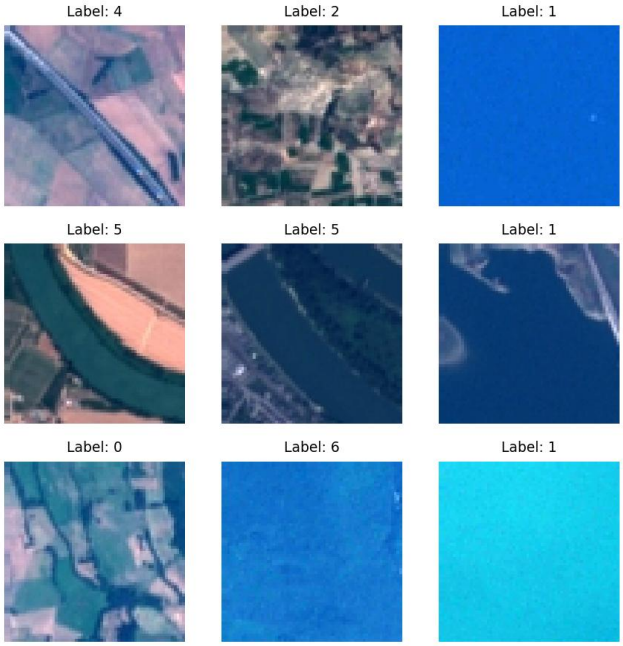

Figure 4: Sample images for the Eurosat dataset.

G.5 Fashion

The FashionMNIST dataset describing different articles of clothing (Xiao, Rasul, and Vollgraf 2017). It consists of 70,000 grey scale images labeld into 10 classes. See Figure 5.

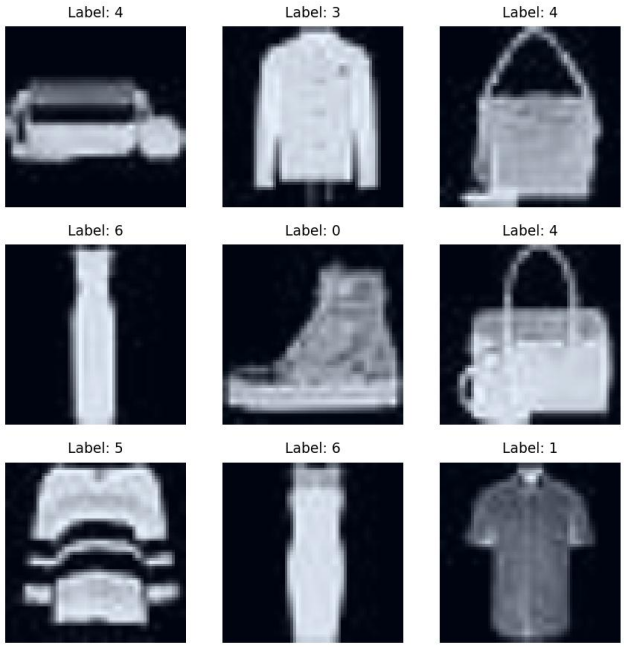

Figure 5: Sample images for the Fashion dataset.

G.6 Food

The Food101 dataset (Bossard, Guillaumin, and Van Gool 2014) contains 101 classes of different types of food. It consists of 101,000 images with 1000 images per class.

See figure 6.

G.7 Garbage

This is a dataset to classify the material of different waste objects (Single, Iranmanesh, and Raad 2023). The dataset is split into 9 classes with more than 4000 images and at least 300 images per class. See Figure 7.

G.8 Plant

This is a plant leaves dataset detailing different types of disease (Hughes and Salathé 2015). There are over 50,000 images across 38 classes. Each variety of plant contains a set of healthy leaf images and one more diseased leaf images. See Figure 8.

G.9 Rock

This is a dataset of different types of rocks and minerals (Hossain et al. 2021). It consists of 7 different classes across more than 2000 images. See Figure 9.

Figure 6: Sample images for the Food dataset.

Figure 7: Sample images for the Garbage dataset.

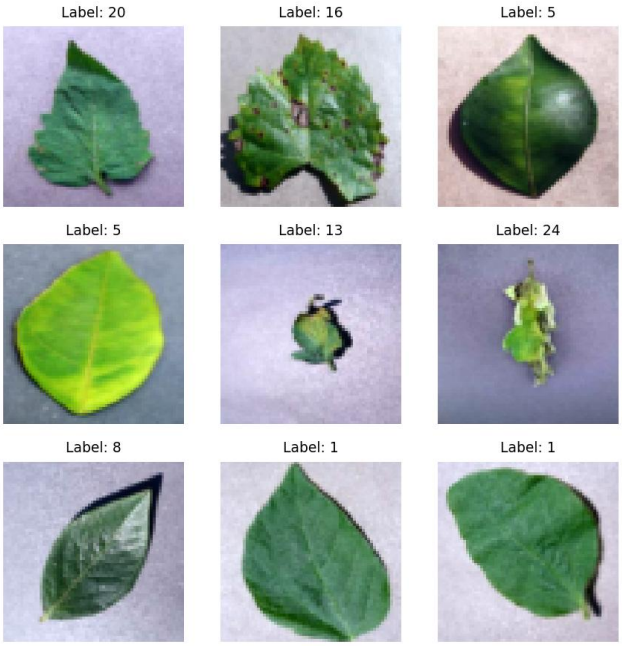

Figure 8: Sample images for the Plant dataset.

Figure 9: Sample images for the Rock dataset. Note that this dataset appears to contain images in multiple domains, such as the kitchen image of a marble countertop.

G.10 Tissue

This is a kidney cortex microscope dataset with various types of tissue labeled (Yang et al. 2023). It consists of over 200,000 images across 8 different classes. See Figure 10

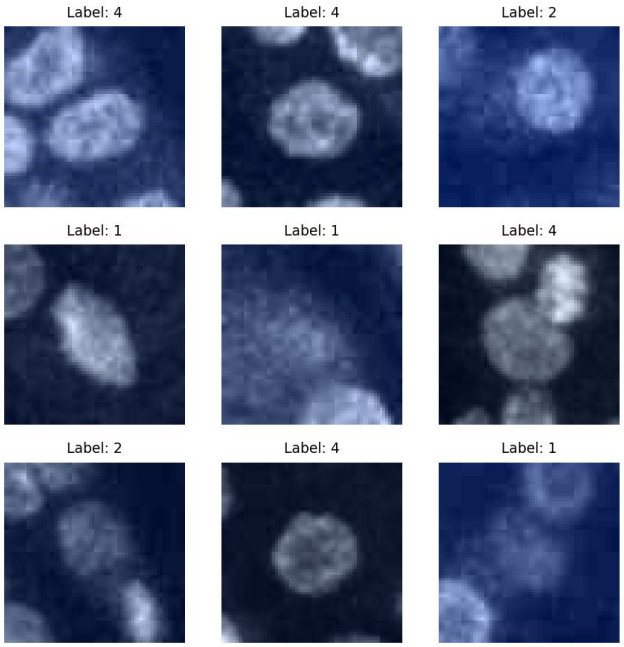

Figure 10: Sample images for the Tissue dataset.

G.11 Yoga

This is a dataset of people performing different yoga poses from the internet (Sumanthvrao 2020). It consists of 2,964 images across 6 classes. See Figure 11.

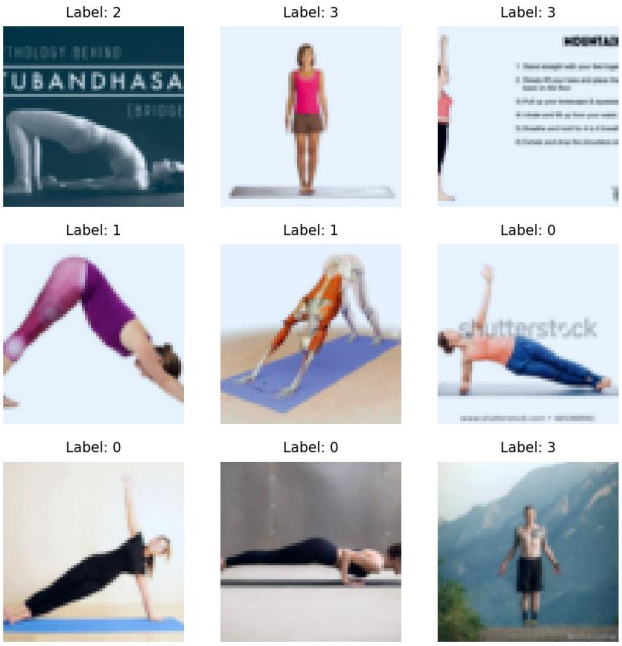

Figure 11: Sample images for the Yoga dataset.