PURSUING FEATURE SEPARATION BASED ON NEURAL COLLAPSE FOR OUT-OF-DISTRIBUTION DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

In the open world, detecting out-of-distribution (OOD) data, whose labels are disjoint with those of in-distribution (ID) samples, is important for reliable deep neural networks (DNNs). To achieve better detection performance, one type of approach proposes to fine-tune the model with auxiliary OOD datasets to amplify the difference between ID and OOD data through a separation loss defined on model outputs. However, none of these studies consider enlarging the feature disparity, which should be more effective compared to outputs. The main difficulty lies in the diversity of OOD samples, which makes it hard to describe their feature distribution, let alone design losses to separate them from ID features. In this paper, we neatly fence off the problem based on an aggregation property of ID features named Neural Collapse (NC). NC means that the penultimate features of ID samples within a class are nearly identical to the last layer weight of the corresponding class. Based on this property, we propose a simple but effective loss called Separation Loss, which binds the features of OOD data in a subspace orthogonal to the principal subspace of ID features formed by NC. In this way, the features of ID and OOD samples are separated by different dimensions. By optimizing the feature separation loss rather than purely enlarging output differences, our detection achieves SOTA performance on CIFAR10, CIFAR100 and ImageNet benchmarks without any additional data augmentation or sampling, demonstrating the importance of feature separation in OOD detection. The code will be published.

1 INTRODUCTION

In the open world, deep neural networks (DNNs) encounter a diverse range of input images, including in-distribution (ID) data that shares the same distribution as the training data, and out-of-distribution (OOD) data, which has labels that are disjoint from those of the ID cases. Facing the complex input environment, a reliable network system must not only provide accurate predictions for ID data but also recognize unseen OOD data. This necessity gives rise to the critical problem of OOD detection (Cao et al., 2007; Liu et al., 2021), which has garnered significant attention in recent years, particularly in safety-critical applications.

A rich line of studies detect OOD samples by exploring the differences between ID and OOD data in terms of model outputs (Hendrycks & Gimpel, 2016; Liu et al., 2020), features (Sun et al., 2021; Zhu et al., 2022; Sun et al., 2022b), or gradients (Huang et al., 2021; Wu et al., 2023). However, it has been observed that models trained solely on ID data can make over-confident predictions on OOD data, and the features of OOD data intermingle with those of ID features (Hendrycks & Gimpel, 2016; Sun et al., 2022b). To develop more effective detection algorithms, a category of works focuse on the utilization of auxiliary OOD datasets, which significantly improves detection performance on unseen OOD data. One classical method, called Outlier Exposure (OE, Hendrycks et al. (2018)), employs a cross-entropy loss between the outputs of OOD data and uniformly distributed labels to fine-tune the model. Additionally, Energy method (Liu et al., 2020) proposes using the energy function as its training loss and designs an energy gap between ID and OOD data. Building on these proposed losses, recent works have concentrated on improving the quality of auxiliary OOD datasets through data augmentation (Wang et al., 2024; 2023; Zheng et al., 2024) or data sampling (Ming et al., 2022a; Chen et al., 2021; Jiang et al., 2023) to achieve better detection performance.

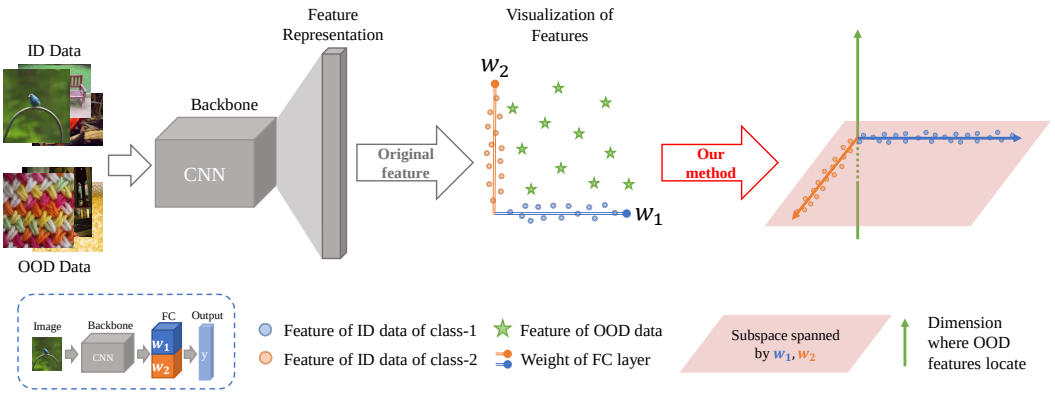


Figure 1: Overview of our method. An example of a well-trained binary classification network, where w_i denotes the i-th weight of the last fully connected layer. The features of ID samples within a class are nearly identical to the weight of the corresponding class, which is known as Neural Collapse phenomenon. Based on this property, we propose to constrain OOD features on dimensions orthogonal to FC weight subspace to explicitly separate the feature manifolds between ID and OOD data.

Existing losses designed for auxiliary OOD data primarily focus on increasing the output discrepancy between ID and OOD samples (Hendrycks et al., 2018; Liu et al., 2020). However, **NONE** of these approaches consider enhancing the separability in the feature space. Insights from knowledge distillation (Gou et al., 2021) and contrastive learning (Le-Khac et al., 2020) have demonstrated that optimizing compactness or dispersion in the feature space is equally or even more important than enforcing similar constraints in the output space. Furthermore, previous detection score function design has shown the importance of employing feature information (Sun et al., 2021; 2022b; Zhu et al., 2022), which can greatly improve detection performance. Therefore, when tackling the fine-tuning problem using auxiliary OOD data, we propose that it is crucial to **separate the features between ID and OOD data**, rather than merely enlarging their output differences.

Designing an effective feature separation loss for ID and OOD data is inherently challenging due to the diversity of OOD samples that belong to various categories. This diversity results in a dispersion of their features and difficulty in describing their feature distribution. Consequently, common feature separation losses, such as maximizing the distance between the average features of different classes (Ming et al., 2022b) or increasing the Kullback-Leibler divergence between ID and OOD feature distributions (Kullback, 1997), are not suitable in our cases. Despite the intricate distribution of OOD features posing a significant obstacle, in this paper, we derive solutions from the properties of ID features.

A recent observation named Neural Collapse (Papyan et al., 2020) gives us an inspiration, which reveals that the penultimate features of ID samples within a class are nearly identical to the last fully connected (FC) layer's weights of the corresponding class. Conversely, the features of OOD samples are scattered haphazardly throughout the feature space. A direct illustration [1] can be seen in Figure 1. Leveraging the property of ID features, we propose to constrain the features of OOD data on dimensions orthogonal to the subspace (denoted as $\mathbf{W}$) spanned by FC weights. The dimension of $\mathbf{W}$ equals to the number of ID categories, while the overall feature space dimension is significantly larger. Consequently, there are numerous redundant dimensions available for OOD features, indicating the feasibility of our method. To pursue this orthogonality, we introduce a loss function named Separation Loss (*ref.* Eq3), which calculates the absolute value of cosine similarity between OOD features and the weights of the final FC layer. By optimizing this simple yet effective loss to zero, we ensure that OOD features are distributed in entirely different dimensions from ID features, thereby enhancing their separability. Our approach utilizes the NC property of ID features, allowing us to avoid modeling OOD feature distributions while effectively segregating ID and OOD features. The overall method can be widely applied as a stronger baseline compared to OE (Hendrycks et al., 2018),

[1] For binary classification, NC indicates that the angle between w_1 and w_2 should be 180°. But in order to show the general case of higher dimensions, we depict an angle of 90° in the figure.

and seamlessly integrated with other approaches like ATOM(Chen et al., 2021), POEM(Ming et al., 2022a), etc(Wang et al., 2024; 2023) by replacing the OE loss with our separation loss.

We conduct extensive experiments over representative OOD detection setups, achieving the SOTA performance without any data augmentation or sampling algorithms (Ming et al., 2022a; Wang et al., 2023) on CIFAR10 (Krizhevsky et al., 2009b), CIFAR100 (Krizhevsky et al., 2009b) and ImageNet (Deng et al., 2009a) benchmarks. For example, on the CIFAR100 benchmark, by using our feature separation loss, we achieve the average FPR95 of 29.58% and AUROC of 94.01%, outperforming the traditional OE (Hendrycks et al., 2018) method by 8.19% on FPR95. Furthermore, our method exhibits a very stable performance while comparable methods like DAL (Wang et al., 2024) struggle with a high fluctuation, as Sec 4.5 shows. The contribution of our paper is summarized as follows:

- We are the first to propose the concept of feature separation when using auxiliary OOD data to fine-tune models, while previous works pay more attention to the output separation, providing new insights into the design of OOD data loss functions.
- To overcome the difficulty caused by OOD data diversity, we propose a feature separation loss based on the neural collapse property of ID features, which constrains OOD features to lie in dimensions where ID features are scarcely distributed.
- Our SOTA detection performance on representative OOD detection settings verify the effectiveness of our feature separation loss, implying that our loss can be a stronger baseline for future researches.

2 Related work

Post-hoc Detection. Given a model that is only trained by ID data, post-hoc detection approaches design score functions based on it to distinguish ID and OOD data. One type method named density-based (Lee et al., 2018; Kobyzev et al., 2020; Zisselman & Tamar, 2020; Kingma & Dhariwal, 2018; Jiang et al., 2021; Choi et al., 2018) is to explicitly model the ID data with some probabilistic models and flag test data in low-density regions as OOD samples. More popular approaches are to derive confidence score based on model outputs (Hendrycks & Gimpel, 2016; Liang et al., 2017; Liu et al., 2020), features (Sun et al., 2021; Zhu et al., 2022; Sun et al., 2022b; Lee et al., 2018; Ndiour et al., 2020; Cook et al., 2020; Ndiour et al., 2020; Cook et al., 2020; Wang et al., 2022) or gradients (Huang et al., 2021; Wu et al., 2023; Lee et al., 2023; Lust & Condurache, 2020; Sun et al., 2022a; Igoe et al., 2022). For example, the classical maximum softmax probability method (Hendrycks & Gimpel, 2016) utilizes the model output probability of the predicted class as a confidence score and then identifies samples with low scores as OOD data.

Contrastive Learning based Detection. Different from post-hoc methods based on vanilla-trained models, such methods generally apply contrastive losses defined on ID data in the model training process to obtain better feature representations for OOD detection. For example, KNN+ (Sun et al., 2022b) utilizes the SupCon loss (Khosla et al., 2020), which encourages alignment of features within a class and dispersion of features of different classes, to train a network to obtain greater differentiation between ID and OOD samples. Besides, CSI (Tack et al., 2020) contrasts original samples with their distributionally-shifted augmentations to improve detection performance. Recent advancements, such as CIDER (Ming et al., 2022b), combine a compactness loss to cluster samples near their class prototypes and a dispersion loss to maximize angular distances between different class prototypes, providing a more direct and clearer geometric interpretation for the disparity between ID and OOD samples.

Auxiliary OOD Data based Detection. With access to part of OOD data, previous works design training algorithms to utilize auxiliary OOD data for OOD detection. One type method is to propose unsupervised training loss functions (Hendrycks et al., 2018; Liu et al., 2020; Bai et al., 2023), such as the Kullback-Leibler divergence between OOD output probability and uniformly distributed label (Hendrycks et al., 2018), to fine-tune the model. Based on the proposed losses, another type is to select OOD data close to the decision boundary (Ming et al., 2022a) or conduct data augmentation through adversarial attack (Chen et al., 2021; Wang et al., 2024) and model perturbations (Wang et al., 2023) in the training process, which can tight the boundary so that pushing unseen OOD data far away from it. In general, using auxiliary OOD data in the training process can significantly improve detection performance, achieving better results compared with other detection approaches.

3 METHOD

3.1 PRELIMINARY

OOD Detection Problem. The framework for OOD detection is outlined as follows. We consider a classification problem involving C classes, where $\mathcal{X}$ represents the input space and $\mathcal{Y}$ denotes the label space. The joint data distribution over $\mathcal{X} \times \mathcal{Y}$ is referred to as $D_{\mathcal{XY}}$. Let $f_\theta : \mathcal{X} \mapsto \mathcal{Y}$ be a model trained on samples drawn independently and identically distributed (*i.i.d.*) from $D_{\mathcal{XY}}$ with parameters θ. Then, the distribution of ID data is the marginal distribution of $D_{\mathcal{XY}}$ over $\mathcal{X}$, denoted as D_{in}. Conversely, the distribution of OOD data is represented as D_{out}, whose label set does not intersect with $\mathcal{Y}$. The primary objective of OOD detection is to determine whether a test input x originates from D_{in} or D_{out}. Typically, this decision is made using a score function S as follows:

$$G_\lambda(x) = \begin{cases} \text{ID} & \text{if } S(x, f) \geq \lambda \\ \text{OOD} & \text{if } S(x, f) \leq \lambda \end{cases}$$

where λ is a threshold. Samples with scores higher than λ are classified as ID data. The threshold is usually set based on ID data to ensure that a high fraction of ID data (e.g., 95%) is correctly identified as ID samples.

Finetune Model with Auxiliary OOD Data. In this paper, we consider the task of using auxiliary OOD data to fine-tune the model (Hendrycks et al., 2018; Liu et al., 2020; Ming et al., 2022a; Wang et al., 2024), which can effectively enlarge the discrepancy between ID and unseen OOD data. Let's denote the auxiliary OOD dataset as $D_{\text{out}}^{\text{aux}}$, which is a subset of real OOD datasets but has different distributions from the test OOD datasets in the experiments for fair comparison. One classical method is the Outlier Exposure (OE, (Hendrycks et al., 2018)), which designs an outlier exposure loss that calculates the cross-entropy function between OOD outputs and uniformly distributed labels. The equation is as follows:

$$L_{\text{OE}}(x) = -\frac{1}{C}\sum_{j=1}^{C} \log f_j(x), \tag{1}$$

where $f_j(x)$ denotes the j-th element of the model output $f(x)$. The final training objective of OE is to simultaneously minimize cross-entropy loss on ID data and outlier exposure loss on OOD data, which can be formalized as:

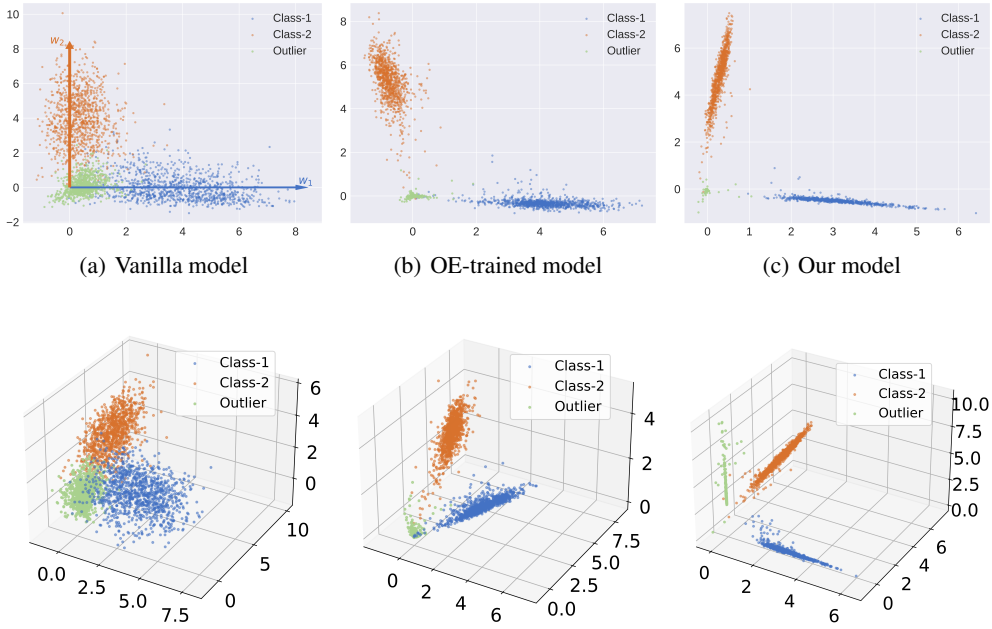
$$\min_f \mathbb{E}_{(x,y)\sim D_{\text{in}}} L_{\text{CE}}(x, y) + \lambda \mathbb{E}_{x\sim D_{\text{out}}^{\text{aux}}} L_{\text{OE}}(x) \tag{2}$$

where λ is a hyper-parameter. This optimization problem is regarded as a basic setting in auxiliary OOD data approaches. Most of subsequent methods adopt the same or similar loss functions that encourage ID and OOD data to differ in the output space. For example, POEM (Ming et al., 2022a) designs a data sampling algorithm for efficient training, and DAL (Wang et al., 2024) employs adversarial features to calculate the OE loss to minimize the generalization gap between auxiliary and real unseen OOD data.

3.2 MOTIVATION

Previous works have focused on increasing the discrepancy between ID and OOD data in the output space, while in this paper, we propose to explicitly enlarge the disparity of their features. Intuitively, separating features of ID and OOD data should be beneficial to OOD detection compared to solely augmenting the output differences. Existing feature separation functions in other fields, such as the dispersion loss that enlarges the distance between the average features of different classes (Ming et al., 2022b; Khosla et al., 2020), are not suitable for diverse OOD data since their features are dispersed instead of clustering around the mean. To design a separation loss that can handle the complicated distribution of OOD features, we delve into the property of ID features. A recent observation named NC (Papyan et al., 2020) gives us a new insight, which reveals that the penultimate features of ID samples within a class are nearly identical to the last layer weight of the corresponding class. This intriguing property has stimulated many fields of research, including low-dimensional characteristics of ID features (Garrod & Keating, 2024; Rangamani et al., 2023) and model generalization analysis (Kothapalli, 2022; Hui et al., 2022). Particularly, several works employ the principal component spaces identified by NC to design detection score functions (Liu & Qin, 2023; Zhang et al., 2024;

Haas et al., 2022; Ammar et al., 2023), which demonstrates the large potential of NC applied in OOD detection. We conduct empirical experiments on CIFAR10 to validate the NC property, as Figure 2(a) shows, where we plot the feature distribution of test ID samples under weight basis space. Furthermore, with the comparison of OOD features of vanilla and OE-trained model (Figure 2(a) *vs.* Figure 2(b)), we discover that although the outlier exposure loss only optimizes the output of training OOD samples, it implicitly changes the distribution of test unseen OOD features, making them more clustered and far way from test ID features. However, from the 3D visualization of features in Figure 2(e), it can be observed that the features of test unseen OOD data almost lie in the same subspace as ID features, without taking advantage of the new dimension (z-axis) to further widen the ID-OOD difference. Based on the above observations, we then design a feature separation loss without modeling OOD feature distributions as follows.



(a) Vanilla model (b) OE-trained model (c) Our model

(d) Vanilla model (e) OE-trained model (f) Our model

Figure 2: Visualization of features projected into the two-dimensional space consisted of w_1 and w_2 (*ref. Figure 1*) and the three-dimensional space consisted of w_1, w_2 and the principal eigenvector of OOD features on CIFAR10 benchmark. The Class-1 and Class-2 represent features of test ID samples of class-1 and class-2, and the Outlier means features of test unseen OOD data, i.e. SVHN. It can be observed that the feature separability between ID and OOD data gradually increases from left (Vanilla model) to right (Our model).

3.3 Feature Separation Loss

Our key idea is to confine the features of OOD data to dimensions where ID features are sparsely distributed. Considering that the principal subspace of ID features is C-dimensional, as determined by the NC property, while the overall feature space has a significantly larger dimensionality, there exist ample redundant dimensions that can accommodate OOD features. In pursuit of our goal, a straightforward condition arises: $z^T w_i = 0, i = 1, 2, ..., C$, where z denotes the normalized feature of OOD data, and w_i denotes the normalized fully connected layer weight for class i. According to this condition, we devise a Separation Loss for OOD data, which computes the average absolute value of the cosine similarity between z and w_i. The specific equation is as follows:

$$L_{\text{Sep}} = \frac{1}{C} \sum_{i=1}^{C} \left| z^T w_i \right| \tag{3}$$

Through minimizing the L_{Sep} loss, the OOD feature z tends to be distributed in the dimensions that are orthogonal to $w_i, i = 1, 2, ..., C$. Figures 2(c) and 2(f) illustrate the features of our model

fine-tuned using L_{Sep}. As observed, the features of outlier samples are indeed distributed in different dimensions from w_i, resulting in a larger discrepancy between ID and OOD features. Except for the L_{Sep} loss, we also propose an assistant loss that encourages ID features (denote as z_{ID}) within a class to align closely with the FC weight of their corresponding class (denote as w_y). We term this loss function as L_{Clu} as it promotes neural collapse phenomenon (Papyan et al., 2020), enabling ID features within a class more clustered. The formulation is as follows:

$$L_{\text{Clu}} = -z_{ID}^T w_y \tag{4}$$

Our empirical experiment in Sec 4.4 indicates that adding L_{Clu} in the training loss can further improve the detection performance. Combining the above two losses, the final optimization problem can be formulated as:

$$\min_f \mathbb{E}_{(x,y)\sim D_{in}}(L_{\text{CE}} + \alpha L_{\text{Clu}}) + \mathbb{E}_{x\sim D_{out}^{aux}}(\lambda L_{\text{OE}} + \beta L_{\text{Sep}}) \tag{5}$$

where α, λ and β are hyper-parameters. In our experiments, we use the common setting $\lambda = 0.5$ in previous works (Hendrycks et al., 2018) and set $\alpha = 1.0$ and $\beta = 1.0$ for simplicity.

3.4 Overall Framework

Train. Based on the optimization problem outlined in Eq.5, we detail the training procedure as follows. Depending on the model's initial state, we employ different training strategies. The critical factor is whether the model is well-trained on ID data. Notably, the neural collapse phenomenon occurs when the network is fully converged on ID data. Therefore, if the network is well-trained, we can directly apply Eq.5 to fine-tune the model; otherwise, we need to firstly utilize cross-entropy loss to train the model for convergence and then use our loss to fine-tune the model.

Test. After fine-tuning the model with our loss, we propose a new score function to detect OOD samples. Since our method simultaneously optimizes the outputs and features of OOD data, a more proper score function is the sum of the traditional MSP (Hendrycks & Gimpel, 2016) and the average cosine similarity between features and w_i. Mathematically, it can be expressed as:

$$S(x, f) = \max_i \frac{e^{y_i}}{\sum_j^C e^{y_j}} + \frac{1}{C}\sum_{i=1}^{C} \left| z^T w_i \right| \tag{6}$$

where y denotes the model output $f(x)$. Our experiments in Sec 4.5 compare the performance of using our score function with only using the MSP score. The result indicates that our method also performs well under MSP score, but slightly better under our proposed score function.

4 Experiments

In this section, we first conduct experiments on CIFAR10, CIFAR100 and ImageNet benchmarks to validate the superiority of our method in Sec 4.1. Then, we consider a variety of model architectures to further verify the effectiveness of our method in Sec 4.2. Subsequently, we study the hyper-parameter sensitivity of our method in Sec 4.3 and explore the contribution of each loss part in Sec 4.4. In the last part, we discuss the fluctuation of performance, the influence of different score functions, and the numerical result of feature separation degree in Sec 4.5. In our Appendix, we report more experimental results, including combining our loss with other output-based losses except for OE (Hendrycks et al., 2018) in Appendix A.1, detailed results on diverse networks in Appendix A.2, and performance on hard OOD detection settings proposed in CSI (Tack et al., 2020) in Appendix A.3. To begin with, we introduce our experiment setups as follows.

OOD Datasets. For CIFAR benchmarks, we randomly choose 300K samples from the 80 Million Tiny Images (Torralba et al., 2008) as our auxiliary OOD dataset. And we adopt five routinely used datasets as the test OOD datasets, including SVHN (Netzer et al., 2011), LSUN (Yu et al., 2015), iSUN (Xu et al., 2015), Texture (Cimpoi et al., 2014) and Places365 (Zhou et al., 2017), which have non-overlapping categories *w.r.t.* CIFAR datasets. For ImageNet benchmark, we use a validation subset of ImageNet-21k-p dataset as auxiliary OOD dataset, following the setting in DAL (Wang et al., 2024). And we adopt four commonly-used OOD datasets for evaluation, including iNaturalist (Van Horn et al., 2018), SUN (Xiao et al., 2010), Places (Zhou et al., 2017) and Textures (Cimpoi et al., 2014).

Pre-training Setups. For CIFAR benchmarks, we employ Wide ResNet-40-2 (Zagoruyko & Komodakis, 2016) trained for 200 epochs, with batch size 128, init learning rate 0.1, momentum 0.9, weight decay 0.0005, and cosine schedule. For ImageNet benchmarks, we directly use the pre-trained ResNet50 (He et al., 2016) model in Pytorch as the baseline network.

Fine-tuning Setups. For both CIFAR10 and CIFAR100 benchmarks, we adopt the model parameter of the 99th epoch in the pre-training process as our initial network parameters, and then add auxiliary OOD data to train the model for 50 epochs with ID batch size 128, OOD batch size 256, initial learning rate 0.07, momentum 0.9, weight decay 0.0005 and cosine schedule. This setting is aligned with experiments in DAL (Wang et al., 2024). Since the initial model is not sufficiently converged, we add our proposed L_{Sep} and L_{Clu} into the training loss after 25th epoch of the whole fine-tuning stage. For ImageNet benchmark, we use the pre-trained model in Pytorch as initial network, and then fine-tune the model for 5 epochs with ID/OOD batch size 64, initial learning rate $1e{-}4$, momentum 0.9, weight decay 0.0005 and cosine schedule.

Compared Methods. We compare our method with post-hoc approaches, contrastive learning based methods, and auxiliary OOD data based methods. The post-hoc methods include MSP (Hendrycks & Gimpel, 2016), Energy (Liu et al., 2020), Maha (Lee et al., 2018), and KNN (Sun et al., 2022b). The contrastive learning based methods include CSI (Tack et al., 2020), CIDER (Ming et al., 2022b), and KNN+ (Sun et al., 2022b). The auxiliary OOD data based methods include OE (Hendrycks et al., 2018), Energy-OE (Liu et al., 2020), POEM (Ming et al., 2022a), and DAL (Wang et al., 2024). For OE and Energy-OE, we adopt the same training setting as ours, since we have discovered that their recommended setting in the original paper performs much worse than our setting. For other methods, we adopt their suggested setups but unify the backbones for fairness.

Evaluation Metrics. We report two classical metrics in this paper: 1) FPR95: the false positive rate of OOD samples when the true positive rate of ID samples is at 95%. 2) AUROC: the area under the receiver operating characteristic curve. A lower FPR95 and a higher AUROC indicate better detection performance.

4.1 Main Results

The main results are shown in Table 1 and Table 2, where we report the FPR95 and AUROC across the considered real OOD datasets [2]. Compared to methods based on vanilla or contrastive learning models, whose training datasets only contain ID samples, incorporating auxiliary OOD data into the training process can significantly reduce the FPR95 and improve the AUROC, indicating that this direction is valuable to explore. Compared to the classical OE approach (Hendrycks et al., 2018), our method reduces the average FPR95 by 0.87% on CIFAR10, 8.30% on CIFAR100, and 2.93% on ImageNet, just by adding our Separation and Cluster losses into the training procedure. This result demonstrates the effectiveness of our proposed losses. In addition to the OE approach, we also compare our method with other advanced works, including the classical work that studies data sampling strategies (POEM, Ming et al. (2022a)) and the adversarial feature augmentation work that aims to mitigate the impact of OOD distribution discrepancy (DAL, Wang et al. (2024)). It is worth noticing that our method does not employ any data augmentation or selection algorithms, while exhibiting superior performances on CIFAR and ImageNet benchmarks. On CIFAR100, our method outperforms the best baseline DAL by 2.00% on FPR95 and 1.18% on AUROC. Based on the outstanding performance of our method, we suggest that the feature separation loss, which is simple yet effective, can be used as a basic training function like OE loss (Hendrycks et al., 2018) in the further works.

Table 1: Results on ImageNet-1k benchmark with auxiliary OOD data. The best result is in bold.

Method	iNaturalist		SUN		Places		Textures		Average		ID Acc↑
	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	
OEHendrycks et al. (2018)	48.30	88.91	61.40	83.09	70.36	80.78	58.60	82.78	59.66	83.89	76.04
DALWang et al. (2024)	47.92	89.12	61.20	83.22	70.55	80.79	57.91	83.02	59.39	84.04	75.94
Ours	43.01	90.17	60.11	83.56	68.46	81.31	55.35	83.45	**56.73**	**84.62**	**76.10**

[2]The symbol * in the table means the results are cited from DAL (Wang et al., 2024)

Table 2: Results on CIFAR10 and CIFAR100 benchmarks. The best result is in bold.

Method	SVHN		LSUN		iSUN		Textures		Places365		Average	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
CIFAR-10												
With vanilla training												
MSPHendrycks & Gimpel (2016)	44.22	93.61	27.56	96.12	69.62	85.29	60.02	88.53	65.68	86.25	53.42	89.96
EnergyLiu et al. (2020)	31.81	94.65	4.6	98.96	50.06	89.75	49.68	90.09	42.28	90.82	35.69	92.85
MahaLee et al. (2018)	42.67	90.71	18.96	96.46	28.86	93.76	26.22	92.81	86.78	69.14	40.70	88.58
KNNSun et al. (2022b)	44.76	92.55	27.38	95.34	43.84	91.24	37.64	92.82	49.23	87.89	40.57	91.97
With contrastive learning												
CSI*Tack et al. (2020)	17.37	97.69	6.75	98.46	12.58	97.95	25.65	94.70	40.00	92.05	20.47	96.17
CIDERMing et al. (2022b)	6.76	98.44	7.45	98.76	26.03	95.93	22.85	95.75	43.70	91.94	21.36	96.16
KNN+*Sun et al. (2022b)	3.28	99.33	2.24	98.90	17.85	97.65	10.87	97.92	30.63	94.98	12.97	97.32
With auxiliary OOD data												
OEHendrycks et al. (2018)	1.40	99.54	0.85	99.64	2.20	99.26	2.80	99.26	9.55	97.39	3.36	**99.02**
Energy-OELiu et al. (2020)	0.75	99.50	0.90	98.98	1.50	99.22	2.75	98.92	9.05	97.33	2.99	98.79
POEMMing et al. (2022a)	25.66	95.43	94.97	76.44	1.58	99.64	20.62	95.73	53.39	88.38	39.24	91.10
DALWang et al. (2024)	0.75	99.28	0.75	99.62	0.70	99.33	2.35	98.99	8.90	97.10	2.69	98.86
Ours	0.40	99.28	0.60	99.68	1.60	99.25	2.45	98.83	7.40	97.60	**2.49**	98.93
CIFAR-100												
With vanilla training												
MSPHendrycks & Gimpel (2016)	74.79	79.64	54.72	86.46	93.85	56.92	88.76	68.48	83.24	71.95	79.07	72.69
EnergyLiu et al. (2020)	70.18	87.15	17.15	97.05	91.37	65.50	84.77	76.72	78.91	75.77	62.75	80.44
MahaLee et al. (2018)	77.73	78.01	98.46	63.44	47.74	88.76	54.93	82.53	97.22	54.11	75.22	73.37
KNNSun et al. (2022b)	71.86	83.31	78.89	70.09	79.60	70.86	72.89	80.05	80.91	71.33	76.83	75.13
With contrastive learning												
CSI*Tack et al. (2020)	64.50	84.62	25.88	95.93	70.62	80.83	61.50	86.74	83.08	77.11	61.12	95.05
CIDERMing et al. (2022b)	16.47	96.23	45.45	81.64	66.01	82.21	49.79	87.48	82.66	68.39	52.08	83.19
KNN+*Sun et al. (2022b)	32.50	93.86	47.41	84.93	39.82	91.12	43.05	88.55	63.26	79.28	45.20	87.55
With auxiliary OOD data												
OEHendrycks et al. (2018)	38.70	92.90	18.30	96.67	36.35	92.59	43.05	91.00	52.45	87.86	37.77	92.21
Energy-OELiu et al. (2020)	17.75	96.94	34.00	94.82	60.75	87.32	45.70	90.09	53.50	89.08	42.34	91.65
POEMMing et al. (2022a)	45.41	90.70	3.01	99.24	18.60	95.79	51.37	83.85	84.13	73.93	40.5	88.87
DALWang et al. (2024)	16.45	96.10	17.00	96.52	36.95	90.88	38.40	91.72	48.55	88.91	31.47	92.82
Ours	17.95	96.52	12.50	97.64	27.00	93.85	41.70	91.37	48.20	90.64	**29.47**	**94.00**

4.2 DIFFERENT ARCHITECTURES

To further verify the effectiveness of our method, we evaluate and compare our performance with other approaches on more network architectures, including ResNet18 (He et al., 2016) and DenseNet121 (Huang et al., 2017). The results are shown in Table 3, where our method exhibits consistently superior performance across various architectures on CIFAR10 and CIFAR100 benchmarks. For instance, we reduce the FPR95 by 4.55% compared to DAL (Wang et al., 2024) with ResNet18 architecture on CIFAR100 benchmark. Detailed results can be seen in Appendix A.2.

Table 3: Results on different network architectures on CIFAR10 and CIFAR100 benchmarks. We report the average FPR95/AUROC across five OOD datasets. The best result is in bold.

Method	CIFAR-10			CIFAR-100		
	WRN-40-2	ResNet18	DenseNet-121	WRN-40-2	ResNet18	DenseNet121
OEHendrycks et al. (2018)	3.36/99.02	6.35/97.35	10.79/97.54	37.77/92.21	56.96/90.19	62.08/86.76
DALWang et al. (2024)	2.69/98.86	3.61/98.20	9.75/97.71	31.47/92.82	54.89/**90.95**	61.25/87.66
Ours	**2.49/98.93**	**3.52/98.75**	**8.90/97.74**	**29.47/94.00**	**50.34**/90.90	**59.13/88.45**

4.3 HYPER-PARAMETER SENSITIVITY

In this section, we study the influence of coefficient α and β in Eq. 5 on the detection performance. Specifically, we evaluate our method on CIFAR10 benchmark with $\alpha \in \{0.1, 0.5, 1.0, 2.0\}$ and $\beta \in \{0.1, 0.5, 1.0, 2.0\}$. Experiment results are shown in Table 4. Notably, our approach is not sensitive to the choice of hyper-parameters. Furthermore, we discover that using $\alpha = 0.1$ and $\beta = 0.1$ can achieve better performance of our method than our previous report.

Table 4: Influence of loss coefficient α and β. We report the average FPR95/AUROC across five OOD datasets on CIFAR10 benchmark. The best result is in bold, and the result for the parameter used in our main experiment is underlined.

CIFAR10	$\beta = 0.1$	$\beta = 0.5$	$\beta = 1.0$	$\beta = 2.0$
$\alpha = 0.1$	**2.21/99.13**	2.41/99.03	2.40/99.12	2.60/98.97
$\alpha = 0.5$	2.45/98.91	2.52/98.90	2.51/99.07	2.43/99.08
$\alpha = 1.0$	2.49/98.77	2.34/98.92	<u>2.49/98.93</u>	2.44/99.01
$\alpha = 2.0$	2.46/98.73	2.51/98.24	2.36/98.38	2.67/98.72

4.4 ABLATION STUDY

Considering our training objective loss contains four parts: L_{CE}, L_{Clu}, L_{OE}, and L_{Sep}, we explore the contribution of each part to the final detection performance in this section. The cross-entropy loss L_{CE} is used for ensuring ID accuracy, thus we skip it when discussing the detection performance. The rest three parts, one (L_{OE}) is for output discrepancy and the other two (L_{Clu} and L_{Sep}) is for feature separation. We firstly evaluate the performance of purely using cross-entropy loss (L_{CE}) and outlier exposure loss (L_{OE}), namely OE method (Hendrycks et al., 2018). And then we discuss three situations: 1) adding L_{Clu}; 2) adding L_{Sep}; 3) adding L_{Clu} and L_{Sep}. The results are shown in Table 5. Comparing the No.1 and No.3, it shows that our feature separation loss significantly improves detection performance compared to OE method. Additionally, only using cluster loss damages the performance but integrating it with separation loss can achieve the best result. The underlying reason is that the cluster loss only controls the property of ID features, but has negligible effect on enlarging the discrepancy between ID and OOD features when purely using it.

Table 5: Performance of our method under different training losses. The OE means using cross-entropy loss and outlier exposure loss. The No.2, No.3, No.4 settings are based on the OE setting, respectively adding cluster loss, separation loss, and cluster+separation losses.

No.	Method	CIFAR10		CIFAR100	
		FPR95↓	AUROC↑	FPR95↓	AUROC↑
1	OE Hendrycks et al. (2018)	3.36	**99.02**	37.77	92.21
2	+L_{Clu}	3.62	98.96	39.91	91.22
3	+L_{Sep}	2.65	99.00	33.30	93.42
4	+L_{Clu}+L_{Sep}	**2.49**	98.93	**29.47**	**94.00**

4.5 DISCUSSION

Cosine Similarity *vs.* Euclidean Distance. Our designed loss is calculated based on cosine similarity, which then induces dimensionality separation between ID and OOD features. In this part, we compare with an intuitive loss design, that is, maximizing the Euclidean distance between OOD features and weights of the last FC layer, with our orthogonality-based loss L_{Sep} to illustrate the importance of utilizing redundant dimensions to enlarge feature discrepancy. Since maximizing the Euclidean distance will cause its value to approach infinity, we instead use $\frac{1}{\|z-w_i\|}$ for OOD features and $\|z_{ID}-w_y\|$ for ID features as the training loss in the Euclidean distance setting, and then minimize this loss to fine-tune the model. The comparison results are presented in Table 6, where the cosine similarity loss significantly outperforms Euclidean distance loss. The underlying reason may be that our separation loss utilizes new dimensions to separate ID and OOD features. When faced with unseen OOD data, the feature variations tend to fall on the new dimension, resulting in minimal changes on the output.

Table 6: Comparison between using the Euclidean distance and cosine similarity (ours) as the training loss to separate ID-OOD features.

Method	SVHN		LSUN		iSUN		Textures		Places365		Average	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
CIFAR-10												
Euclidean	1.70	99.48	1.15	99.60	3.20	99.22	4.55	98.95	12.55	95.97	4.63	98.64
Ours	0.40	99.28	0.60	99.68	1.60	99.25	2.45	98.83	7.40	97.60	**2.49**	**98.93**
CIFAR-100												
Euclidean	51.95	85.97	19.95	96.13	42.35	87.45	44.80	88.18	56.95	85.29	43.20	88.60
Ours	17.95	96.52	12.50	97.64	27.00	93.85	41.70	91.37	48.20	90.64	**29.47**	**94.00**

Fluctuation in Detection Performance. We have observed considerable fluctuations in the performance of the DAL method (Wang et al., 2024) under repeated experiments with identical settings. Therefore, we evaluate the mean and variance of performance after repeating the same experiment five times. The results, shown in Table 7, indicate that our approach exhibits greater stability, particularly on the CIFAR100 benchmark. Notably, even the OE method shows the FPR95 variance of 1.137% on CIFAR100, whereas our method maintains a variance of only 0.027%.

Table 7: Fluctuation in detection performance of different methods.

Method	CIFAR10		CIFAR100	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑
OE Hendrycks et al. (2018)	$3.22_{\pm 0.0017}$	$\mathbf{99.07}_{\pm 0.0014}$	$36.29_{\pm 1.137}$	$92.31_{\pm 0.013}$
DAL Wang et al. (2024)	$2.87_{\pm 0.0234}$	$98.82_{\pm 0.0027}$	$30.44_{\pm 2.216}$	$93.07_{\pm 0.075}$
Ours	$\mathbf{2.49}_{\pm 0.0007}$	$98.92_{\pm 0.0033}$	$\mathbf{29.50}_{\pm 0.027}$	$\mathbf{93.98}_{\pm 0.014}$

Different Score Functions. Since we use our proposed score function in Eq 6 to detect OOD samples while other auxiliary OOD data based methods only employ MSP score (Hendrycks & Gimpel, 2016), in this part, we also evaluate our model using MSP score for a fair comparison. The results in Table 8 demonstrate that our approach also achieves commendable performance under the MSP score, with only a slight decline compared to using the proposed score function.

Table 8: Performance of adopting different score functions in our method.

Method	Score Function	CIFAR10		CIFAR100	
		FPR95↓	AUROC↑	FPR95↓	AUROC↑
Ours	MSP	2.77	98.76	29.96	93.27
	Eq. 6	**2.49**	**98.93**	**29.47**	**94.00**

Feature Separation Degree. In this part, we evaluate the degree of feature separation between ID and OOD data to validate the effectiveness of our proposed loss. Leveraging the NC property of ID features, we use the FC weight as the intermediary to design our measurement standards. Specifically, we propose three metrics, respectively defined by the Euclidean distance and cosine similarity between features and the weight of the predicted class, and the reconstruction error to the subspace spanned by FC weights. The results are shown in Table 9, where OOD means unseen test OOD data instead of auxiliary OOD samples. It can be seen that our model presents higher differences under the three metrics compared to vanilla and OE-trained models.

Table 9: Feature separation degree of different methods measured by three metrics. The higher difference (Diff) means better discrepancy between ID and OOD features.

Method	Euclidean Distance			Cosine Similarity			Reconstruction Error		
	ID	OOD	Diff↑	ID	OOD	Diff↑	ID	OOD	Diff↑
Vanilla	0.80	1.12	0.32	0.69	0.47	0.22	0.19	0.32	0.13
OE	0.86	1.21	0.35	0.79	0.32	0.47	0.41	0.83	0.42
Ours	0.69	1.16	**0.47**	0.75	$3e^{-5}$	**0.75**	0.43	0.86	**0.43**

5 Conclusion

In this paper, we propose a novel training loss to enhance the feature discrepancy between ID and OOD data during model fine-tuning with auxiliary OOD datasets. Given the inherent diversity and complexity of distributions of OOD features, we leverage the Neural Collapse property of ID features, which indicates that the penultimate features of ID samples within a class are nearly identical to the last layer weight of the corresponding class, to pursue dimensionality separation between ID and OOD features. Based on the idea, we introduce a separation loss that confines OOD features to dimensions that are orthogonal to the principal subspace of ID features formed by NC. Our extensive experiments validate the effectiveness of our feature separation loss, achieving SOTA performance on CIFAR10, CIFAR100 and ImageNet benchmarks. We believe that our study will inspire further researches into pursuing feature separation when employing auxiliary OOD data to fine-tune models in OOD detection.

References

Mouïn Ben Ammar, Nacim Belkhir, Sebastian Popescu, Antoine Manzanera, and Gianni Franchi. Neco: Neural collapse based out-of-distribution detection. *arXiv preprint arXiv:2310.06823*, 2023.

Haoyue Bai, Gregory Canal, Xuefeng Du, Jeongyeol Kwon, Robert D Nowak, and Yixuan Li. Feed two birds with one scone: Exploiting wild data for both out-of-distribution generalization and detection. In *International Conference on Machine Learning*, pp. 1454–1471. PMLR, 2023.

T Cao, CW Huang, DYT Hui, and JP Cohen. A benchmark of medical out of distribution detection. arxiv 2020. *arXiv preprint arXiv:2007.04250*, 2007.

Jiefeng Chen, Yixuan Li, Xi Wu, Yingyu Liang, and Somesh Jha. Atom: Robustifying out-of-distribution detection using outlier mining. In *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part III 21*, pp. 430–445. Springer, 2021.

Hyunsun Choi, Eric Jang, and Alexander A Alemi. Waic, but why? generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3606–3613, 2014.

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

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009a.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009b.

Connall Garrod and Jonathan P Keating. Unifying low dimensional observations in deep learning through the deep linear unconstrained feature model. *arXiv preprint arXiv:2404.06106*, 2024.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

Jarrod Haas, William Yolland, and Bernhard Rabus. Linking neural collapse and l2 normalization with improved out-of-distribution detection in deep neural networks. *arXiv preprint arXiv:2209.08378*, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

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

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. *Advances in Neural Information Processing Systems*, 34:677–689, 2021.

Like Hui, Mikhail Belkin, and Preetum Nakkiran. Limitations of neural collapse for understanding generalization in deep learning. *arXiv preprint arXiv:2202.08384*, 2022.

Conor Igoe, Youngseog Chung, Ian Char, and Jeff Schneider. How useful are gradients for ood detection really? *arXiv preprint arXiv:2205.10439*, 2022.

Dihong Jiang, Sun Sun, and Yaoliang Yu. Revisiting flow generative models for out-of-distribution detection. In *International Conference on Learning Representations*, 2021.

Wenyu Jiang, Hao Cheng, Mingcai Chen, Chongjun Wang, and Hongxin Wei. Dos: Diverse outlier sampling for out-of-distribution detection. *arXiv preprint arXiv:2306.02031*, 2023.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Ivan Kobyzev, Simon JD Prince, and Marcus A Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE transactions on pattern analysis and machine intelligence*, 43 (11):3964–3979, 2020.

Vignesh Kothapalli. Neural collapse: A review on modelling principles and generalization. *arXiv preprint arXiv:2206.04041*, 2022.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009a.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009b.

Solomon Kullback. *Information theory and statistics*. Courier Corporation, 1997.

Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *Ieee Access*, 8:193907–193934, 2020.

Jinsol Lee, Charlie Lehman, Mohit Prabhushankar, and Ghassan AlRegib. Probing the purview of neural networks via gradient analysis. *IEEE Access*, 11:32716–32732, 2023.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

Litian Liu and Yao Qin. Detecting out-of-distribution through the lens of neural collapse. *arXiv preprint arXiv:2311.01479*, 2023.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33:21464–21475, 2020.

Julia Lust and Alexandru Paul Condurache. Gran: An efficient gradient-norm based detector for adversarial and misclassified examples. *arXiv preprint arXiv:2004.09179*, 2020.

Yifei Ming, Ying Fan, and Yixuan Li. Poem: Out-of-distribution detection with posterior sampling. In *International Conference on Machine Learning*, pp. 15650–15665. PMLR, 2022a.

Yifei Ming, Yiyou Sun, Ousmane Dia, and Yixuan Li. How to exploit hyperspherical embeddings for out-of-distribution detection? *arXiv preprint arXiv:2203.04450*, 2022b.

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

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, pp. 7. Granada, Spain, 2011.

Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40): 24652–24663, 2020.

Akshay Rangamani, Marius Lindegaard, Tomer Galanti, and Tomaso A Poggio. Feature learning in deep classifiers through intermediate neural collapse. In *International Conference on Machine Learning*, pp. 28729–28745. PMLR, 2023.

Jingbo Sun, Li Yang, Jiaxin Zhang, Frank Liu, Mahantesh Halappanavar, Deliang Fan, and Yu Cao. Gradient-based novelty detection boosted by self-supervised binary classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8370–8377, 2022a.

Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. *Advances in Neural Information Processing Systems*, 34:144–157, 2021.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning*, pp. 20827–20840. PMLR, 2022b.

Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. *Advances in neural information processing systems*, 33:11839–11852, 2020.

Antonio Torralba, Rob Fergus, and William T Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 30(11):1958–1970, 2008.

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

Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4921–4930, 2022.

Qizhou Wang, Junjie Ye, Feng Liu, Quanyu Dai, Marcus Kalander, Tongliang Liu, Jianye Hao, and Bo Han. Out-of-distribution detection with implicit outlier transformation. *arXiv preprint arXiv:2303.05033*, 2023.

Qizhou Wang, Zhen Fang, Yonggang Zhang, Feng Liu, Yixuan Li, and Bo Han. Learning to augment distributions for out-of-distribution detection. *Advances in Neural Information Processing Systems*, 36, 2024.

Yingwen Wu, Tao Li, Xinwen Cheng, Jie Yang, and Xiaolin Huang. Low-dimensional gradient helps out-of-distribution detection. *arXiv preprint arXiv:2310.17163*, 2023.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 3485–3492. IEEE, 2010.

Pingmei Xu, Krista A Ehinger, Yinda Zhang, Adam Finkelstein, Sanjeev R Kulkarni, and Jianxiong Xiao. Turkergaze: Crowdsourcing saliency with webcam based eye tracking. *arXiv preprint arXiv:1504.06755*, 2015.

Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

Jiawei Zhang, Yufan Chen, Cheng Jin, Lei Zhu, and Yuantao Gu. Epa: Neural collapse inspired robust out-of-distribution detector. *arXiv preprint arXiv:2401.01710*, 2024.

Haotian Zheng, Qizhou Wang, Zhen Fang, Xiaobo Xia, Feng Liu, Tongliang Liu, and Bo Han. Out-of-distribution detection learning with unreliable out-of-distribution sources. *Advances in Neural Information Processing Systems*, 36, 2024.

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

Yao Zhu, YueFeng Chen, Chuanlong Xie, Xiaodan Li, Rong Zhang, Hui Xue, Xiang Tian, Yaowu Chen, et al. Boosting out-of-distribution detection with typical features. *arXiv preprint arXiv:2210.04200*, 2022.

Ev Zisselman and Aviv Tamar. Deep residual flow for out of distribution detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13994–14003, 2020.

A APPENDIX

A.1 COMBINATION WITH OTHER OUTPUT-BASED LOSS

In our main paper, we utilize our feature separation loss based on OE method (Hendrycks et al., 2018) since it is the most classical approach. In this section, we also combine our feature separation loss with another output-based loss to demonstrate our wide availability. We adopt Energy-OE approach (Liu et al., 2020) as our basic loss, which is a commonly-used output-based loss. The mathematical formula is as follows:

$$\min_{f} L_{\text{CE}} + \lambda L_{\text{energy}} \tag{7}$$

$$\begin{aligned} L_{energy} = \mathbb{E}_{(x_{\text{in}},y)\sim D_{\text{in}}}(\max(0, E(x_{\text{in}}) - m_{\text{in}}))^2 \\ + \mathbb{E}_{x_{\text{out}}\sim D_{\text{out}}^{\text{aux}}}(\max(0, m_{\text{out}} - E(x_{\text{out}})))^2 \end{aligned} \tag{8}$$

where $E(x) = -T \cdot \log \sum_i C e^{f_i(x)/T}$ is the energy score function, and λ, m_{in}, and m_{out} are hyper-parameters. We adopt the recommended setting in Energy-OE (Liu et al., 2020) to set the parameters and finetune our model. Combining our feature separation loss with the basic method, we obtain our training objective loss as follows:

$$\min_{f} L_{\text{CE}} + \lambda L_{\text{energy}} + \alpha L_{\text{Clu}} + \beta L_{\text{Sep}} \tag{9}$$

In our experiments, we still set $\alpha = 1.0$ and $\beta = 1.0$ for consistency with previous setting. The results are shown in Table 10, where our method significantly reduces the FPR95 by 6.35% on CIFAR100 benchmark compared to the basic Energy-OE approach, convincingly demonstrating our wide availability and effectiveness.

Table 10: Combination with Energy-OE loss on CIFAR10 and CIFAR100 benchmarks. The best result is in bold.

Method	SVHN		LSUN		iSUN		Textures		Places365		Average	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
CIFAR-10												
Energy-OE Liu et al. (2020)	0.75	99.50	0.90	98.98	1.50	99.22	2.75	98.92	9.05	97.33	2.99	98.79
Ours	1.20	99.33	0.65	99.14	1.75	99.33	2.20	99.08	7.55	97.88	**2.67**	**98.95**
CIFAR-100												
Energy-OE Liu et al. (2020)	17.75	96.94	34.00	94.82	60.75	87.32	45.70	90.09	53.50	89.08	42.34	91.65
Ours	11.05	97.65	21.35	96.40	52.95	88.63	42.65	91.14	51.95	88.81	**35.99**	**92.53**

A.2 DETAILED RESULTS ON DIFFERENT ARCHITECTURES

We report the detailed results with ResNet18 and DenseNet121 architectures on CIFAR10 and CIFAR100 benchmarks in Table 11.

Table 11: Detailed Results with ResNet18 and DenseNet121 architectures. The best result is in bold.

Method	SVHN		LSUN		iSUN		Textures		Places365		Average	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
CIFAR-10												
ResNet18												
OE	3.55	97.46	4.35	98.00	4.20	97.58	7.95	97.47	11.70	96.25	6.35	97.35
DAL	0.75	99.38	1.70	98.94	2.10	98.05	4.35	98.20	9.15	96.45	3.61	98.20
Ours	0.50	99.60	2.10	99.19	3.10	99.02	3.15	98.71	8.75	97.26	**3.52**	**98.75**
DenseNet121												
OE	4.40	98.51	4.40	98.68	22.80	96.19	5.55	98.52	16.80	95.78	10.79	97.54
DAL	3.10	98.65	3.10	99.05	21.40	96.56	5.55	98.46	15.60	95.85	9.75	97.71
Ours	4.45	98.73	4.45	98.86	13.90	97.41	4.90	98.76	16.80	95.94	**8.90**	**97.94**
CIFAR-100												
ResNet18												
OE	55.75	92.95	35.45	93.96	70.10	87.37	65.00	88.38	58.50	88.29	56.96	90.19
DAL	51.55	93.40	32.35	94.65	69.75	88.90	63.50	89.52	57.30	88.31	54.89	90.95
Ours	36.70	93.29	34.90	94.12	66.00	88.68	57.35	90.01	56.75	88.39	**50.34**	**90.90**
DenseNet121												
OE	59.40	90.35	48.70	90.15	70.65	83.28	66.90	85.36	64.75	84.65	62.08	86.76
DAL	47.00	92.81	60.05	87.88	55.20	88.65	65.40	86.90	78.60	82.07	61.25	87.66
Ours	67.40	90.49	37.15	92.77	65.15	85.68	63.00	87.13	62.95	86.18	**59.13**	**88.45**

A.3 HARD OOD DETECTION

In addition to testing on the regular OOD datasets, we further consider three hard OOD datasets proposed in Tack et al. (2020), which are considered more difficult to distinguish from ID samples. Following the same setting in (Tack et al., 2020; Sun et al., 2022b; Wang et al., 2024), we evaluate our detection performance on LSUN-Fix (Yu et al., 2015), ImageNet-Resize (Deng et al., 2009b) and CIFAR100 (Krizhevsky et al., 2009a) with CIFAR10 as the ID dataset. Specific results are shown in Table 12. As we can see, our method shows comparable performance with DAL (Wang et al., 2024) over three hard OOD datasets, outperforming the baseline OE method by 2.55% on FPR95 on the ImageNet-Resize dataset.

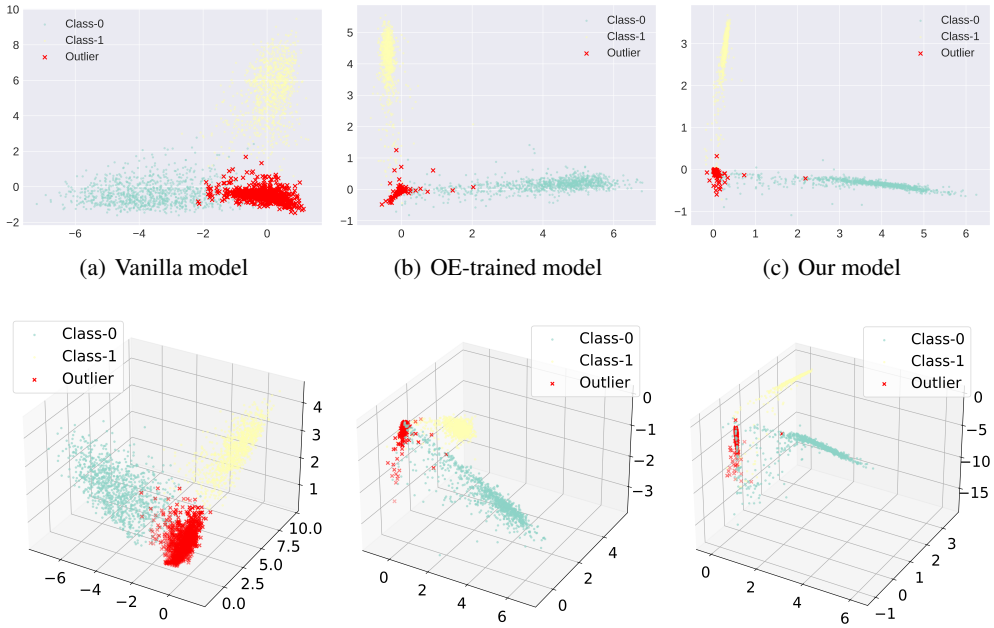
Table 12: Hard OOD detection on CIFAR10 benchmark.

Methods	LSUN-Fix		ImageNet-Resize		CIFAR-100	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
With contrastive learning						
CSI*	39.79	93.63	37.47	93.93	45.64	87.64
CIDER	8.98	98.56	43.45	93.82	55.84	90.0
KNN+*	24.88	95.75	30.52	94.85	40.00	89.11
With auxiliary OOD data						
OEHendrycks et al. (2018)	1.00	99.53	7.20	98.48	25.05	**94.86**
DALWang et al. (2024)	**0.65**	**99.59**	**3.75**	**98.63**	26.00	94.35
Ours	0.75	99.07	4.65	98.42	**24.60**	94.69

A.4 VISUALIZATION

In Figure 2, we visualize the random two class of samples of CIFAR10 dataset and the test unseen OOD samples of SVHN dataset. The low-dimensional visualization is calculated by linear projection into the subspace spanned by w_1, w_2 and principal components of OOD features, where w_i is the corresponding model parameters of the last fully connected layer. Specifically, denote features as $z \in \mathbb{R}^{n\times d}$, the reduction matrix as $M \in \mathbb{R}^{d\times 3}$, where n is the sample number and d is the feature dimension, then the coordinate in 3D space is computed as: $z_{3D} = zM, z_{3D} \in \mathbb{R}^{n\times 3}$.

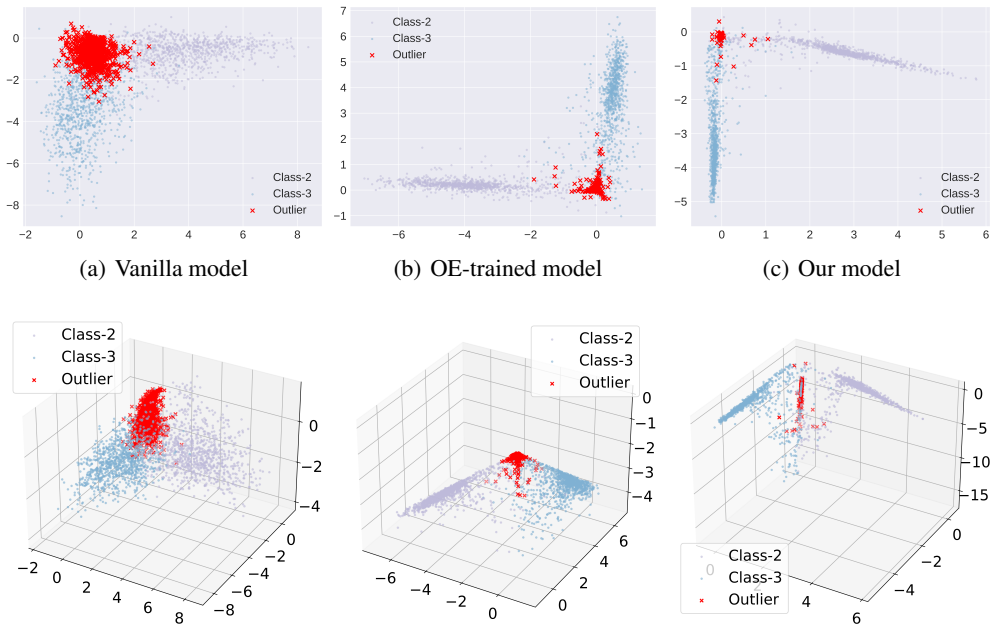
Using the above linear projection operation, we obtain the coordinate for both ID and OOD samples, then visualize them in 2D and 3D space. In the following, we respectively choose two class of ID samples and the test unseen OOD data to visualize their features in 2D and 3D space. In these figures, we can discover that, for all samples of the ten classes in CIFAR10, ID features rarely distribute on redundant dimensions, in contrast, OOD features almost locate on redundant dimensions while little component on model weight dimensions.



(a) Vanilla model (b) OE-trained model (c) Our model

(d) Vanilla model (e) OE-trained model (f) Our model

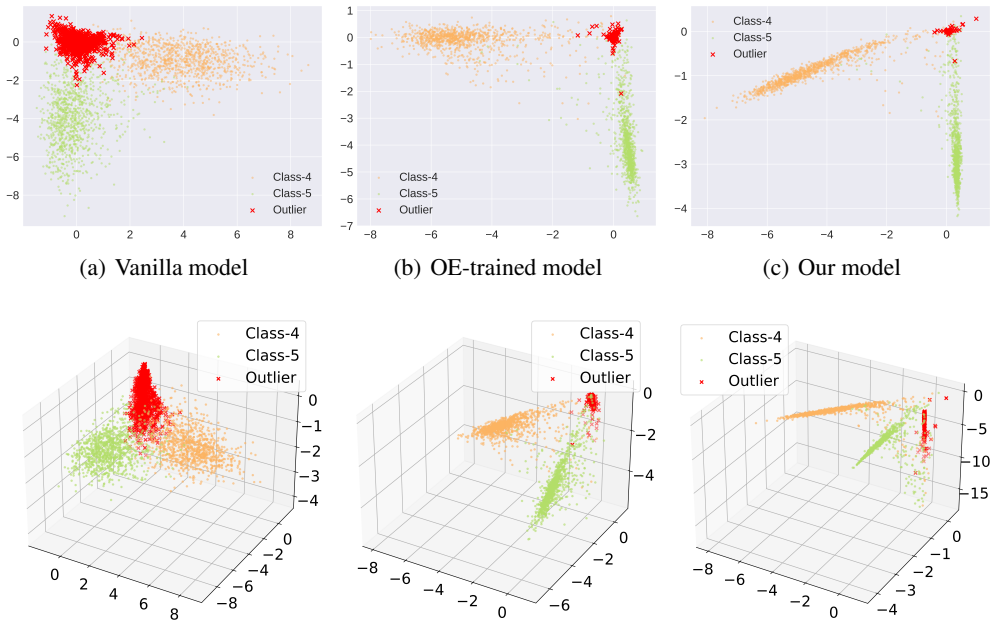
Figure 3: ID sample: Class-0 and Class-1, OOD sample: SVHN



(a) Vanilla model (b) OE-trained model (c) Our model

(d) Vanilla model (e) OE-trained model (f) Our model

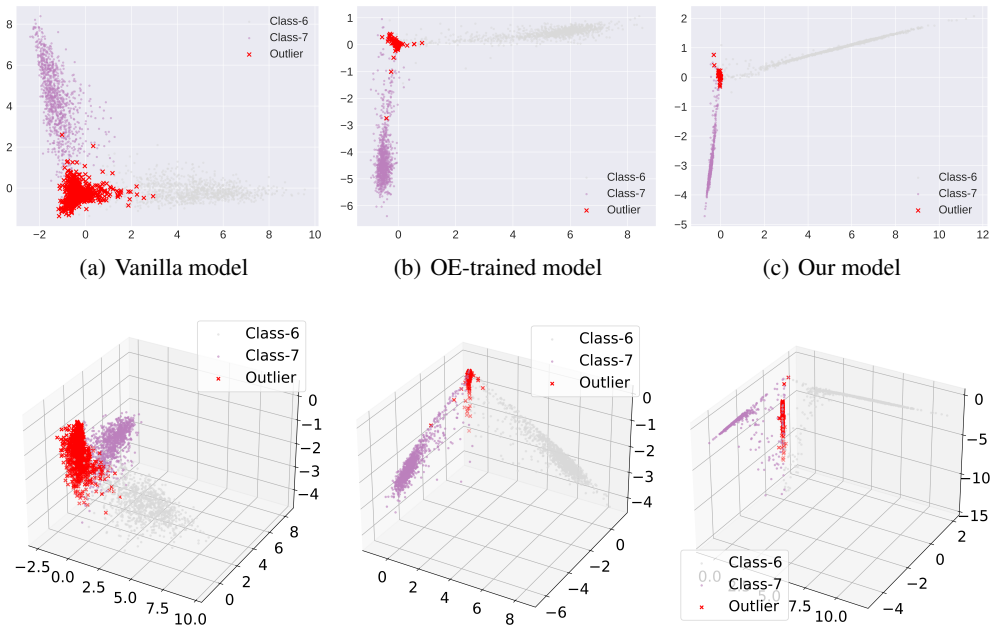
Figure 4: ID sample: Class-2 and Class-3, OOD sample: SVHN



(a) Vanilla model (b) OE-trained model (c) Our model

(d) Vanilla model (e) OE-trained model (f) Our model

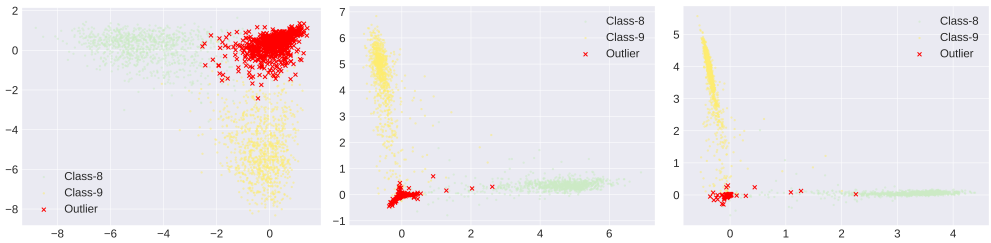
Figure 5: ID sample: Class-4 and Class-5, OOD sample: SVHN



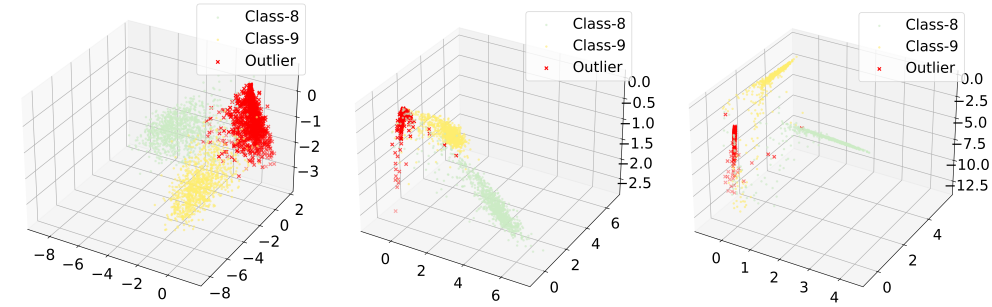
(a) Vanilla model (b) OE-trained model (c) Our model

(d) Vanilla model (e) OE-trained model (f) Our model

Figure 6: ID sample: Class-6 and Class-7, OOD sample: SVHN



(a) Vanilla model (b) OE-trained model (c) Our model



(d) Vanilla model (e) OE-trained model (f) Our model

Figure 7: ID sample: Class-8 and Class-9, OOD sample: SVHN