Neural Collapse Inspired Feature Alignment for Out-of-Distribution Generalization

Zhikang Chen[1] **Min Zhang**[2] **Sen Cui**[5] **Haoxuan Li**[† 3] **Gang Niu**[4]
Mingming Gong[6,8] **Changshui Zhang**[5] **Kun Zhang**[7,8]
[1] Tsinghua University [2] East China Normal University [3] Peking University [4] RIKEN
[5] Institute for Artificial Intelligence, Tsinghua University (THUAI)
Beijing National Research Center for Information Science and Technology (BNRist)
Department of Automation, Tsinghua University
[6] The University of Melbourne [7] Carnegie Mellon University
[8] Mohamed bin Zayed University of Artificial Intelligence (MBZUAI)

Abstract

The spurious correlation between the background features of the image and its label arises due to that the samples labeled with the same class in the training set often co-occurs with a specific background, which will cause the encoder to extract non-semantic features for classification, resulting in poor out-of-distribution generalization performance. Although many studies have been proposed to address this challenge, the semantic and spurious features are still difficult to accurately decouple from the original image and fail to achieve high performance with deep learning models. This paper proposes a novel perspective inspired by neural collapse to solve the spurious correlation problem through the alternate execution of environment partitioning and learning semantic masks. Specifically, we propose to assign an environment to each sample by learning a local model for each environment and using maximum likelihood probability. At the same time, we require that the learned semantic mask neurally collapses to the same simplex equiangular tight frame (ETF) in each environment after being applied to the original input. We conduct extensive experiments on four datasets, and the results demonstrate that our method significantly improves out-of-distribution performance.

1 Introduction

The out-of-distribution (OOD) problem refers to the fact that the training dataset and the test dataset have different distributions in different environments, after the training process is completed, what we want is to have correlation between semantic features and label, however, the presence of spurious feature(environmental information) can make false correlation between environmental information and labels. The semantic feature in an image refers to the object of the class, while the spurious feature in an image refers to the background or the environment. As shown in Figure 1, the background color for digits 0 and 2 is predominantly orange, while the background color for digit 1 is predominantly green. In this figure, the semantic feature includes digits 0, 1 and 2, whereas the spurious feature includes orange and green backgrounds.

Recently, many OOD methods have been proposed to learn invariant representations for different environments by introducing various regularization methods [Ajakan et al., 2014, Arjovsky et al., 2019, Ahmed et al., 2020, Ahuja et al., 2021, Krueger et al., 2021, Liu et al., 2021, Tong et al., 2023, Zhu et al., 2023a]. Although these methods achieve good OOD performance on test data with

[†]Corresponding author. Email: hxli@stu.pku.edu.cn

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

environment variations from the training dataset, on the one hand, they require a well-partitioned environment in advance, but real datasets usually do not have environment labels; on the other hand, under cross-environment conditions, these methods do not align feature prototypes well between the test environments and the training environments. In this paper, we equate semantic features with invariant features and spurious features with variable features.

We first explored the limitations of previous OOD learning methods from a new perspective and conducted a toy example experiment, as shown in Figure 1. The digits 0 and 2 are associated with orange background, and the digit 1 is associated with green background. The phenomenon of neural collapse, which means that after sufficient training, the categories will collapse to a simplex ETF such that the angles between the feature prototypes of two neighboring categories are equidistant, Thus, in Figure 1, after the training is completed, the feature prototypes of the digits 0, 1 and 2 occur in three equal parts. Here, to measure the extent of neural collapse, we propose to compute the Frobenius norm (F-norm) of the difference between the feature prototypes and the standard simplex ETF. A smaller F-norm indicates a closer proximity to the standard simplex ETF. The ERM approach [Vapnik et al., 1998] that leads to the failure of OOD generalization is due to the endogenous nature of the class features not being able to align the class prototypes after training is complete in different environments. Although IRM-based methods [Arjovsky et al., 2019] theoretically learn a feature representation such that the last layer of the feature extractor is similar across environments, we empirically found that IRM-based methods are not aligned very well, which motivates using neural collapse to align feature for OOD generation.

In this paper, in order to bridge this gap, we believe that a feature extractor for OOD generalization across environments should ensure that semantic features are aligned to the same simplex ETF, and we have verified from our experiments that better alignment can significantly improve OOD performance. Specifically, our method can be applied both with and without environment labels. In the absence of environment labels, we can automatically partition the environments and assign local models to different environments. By predicting the variable components of the input, we take the logits corresponding to the label (one-hot encoding) from the predictions of the local models in different environments, form a vector, and select the maximum value of the vector. The corresponding environment of this maximum value is assigned as the new environment for the input. When the new environment shows minimal changes compared to the old environment, we consider the environment partitioning to be complete. In Figure 3, we provide a detailed example to illustrate our environment partitioning method. When environment labels are available, we learn masks to extract semantic components, we firstly fix a simplex ETF classifier, and in different environments, for semantic features, we pull them all to the corresponding position of the same class prototypes, thus realizing the alignment operation.

The main contributions of this paper are summarized as follows:

- We explore the OOD problem from a new perspective, namely the use of neural collapse to guide feature alignment for OOD generalization.
- We explore the separation of semantic features and spurious features under conditions without environment labels, as well as with given environment labels.
- We conduct extensive experiments on four publicly available datasets to validate the effectiveness of the proposed methods.

2 Related Work

Out-of-Distribution Generalization. Out-of-distribution (OOD) generalization has been a topic of significant interest in the field of machine learning and computer vision. Researchers have explored various approaches and techniques to improve the robustness and generalization capabilities of models on unseen target domains. These OOD methods use different strategies to update the model, mainly including metalearning [Li et al., 2018a, Zhang et al., 2023], domain alignment [Ajakan et al., 2014, Li et al., 2018b], regularized training [Zhang et al., 2021, Krueger et al., 2021, Shi et al., 2021, Xu and Jaakkola, 2021], causal learning [Ahuja et al., 2021, Krueger et al., 2021, Koyama and Yamaguchi, 2020], etc. For example, IRM [Arjovsky et al., 2019], a representative OOD method, learns invariant representations with a classifier optimal to domain changes as a regularization term. SANDMask [Shahtalebi et al., 2021] regularizes model training by updating the parameters in the direction where the gradient components have consistent signs across domains. Although these

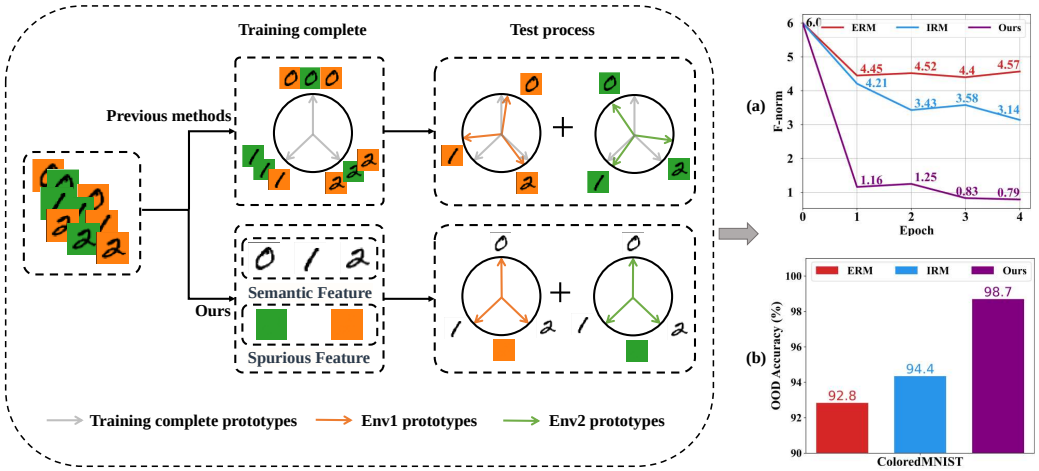


Figure 1: **Left Figure:** A comparison between our approach and previous methods(conclude ERM and IRM-based methods). **(a):** The comparison of our method, the ERM method, and the IRM method based on the F-norm metric. **(b):** The comparison of our method, the ERM method, and the IRM method in terms of OOD accuracy (%).

OOD methods have theoretically and experimentally demonstrated their effectiveness in learning domain-invariant features, most of them require the use of environment labels, which does not meet the requirements of real-world applications. Furthermore, we empirically found that the information extracted by the last layer of the feature extractor based on the IRM-based method in different environments cannot be well-aligned. In this paper, taking a step forward, we propose a novel method that aligns the semantic features extracted by the OOD feature extractor to the same simplex ETF under neural collapse. Note that our method can be applied both with and without environment labels.

Neural Collapse. The phenomenon known as neural collapse, first identified by [Papyan et al., 2020], refers to the observation that when the number of training samples across different classes is balanced, both the feature vectors in the final layer and the classifier vectors tend to converge to the simplex ETF upon the completion of training. Recently, several studies [Ji et al., 2021, Zhu et al., 2021, Tirer and Bruna, 2022, Zhu et al., 2023b, Li et al., 2023, Xie et al., 2023, Yang et al., 2023, Beaglehole et al., 2024, Fisher et al., 2024, Guo et al., 2024, Kothapalli et al., 2024, Súkeník et al., 2024] have utilized this phenomenon to guide the training process in imbalanced data sets. Among them, Yang et al. [2023] involves pre-allocating a fixed number of classes in a simplex ETF for continual learning, guiding the learning of minority classes in subsequent incremental steps, thereby ensuring convergence of intra-class features to specified positions and maximizing and uniformly separating inter-class features. Xie et al. [2023] addresses class imbalance by designing a novel loss function, namely the attraction-rejection balanced loss. Li et al. [2023] apples neural collapse into federated learning scenarios. Under distributed conditions, neural collapse is used to guide the alignment direction of each client, and the personality of each client model is maintained through fine-tuning. The aforementioned methods only consider collapsing the same class onto a single point in a fixed simplex ETF without accounting for the impact of intra-class spurious correlations. We are the first to utilize the concept of neural collapse to address spurious correlations within classes, thereby enhancing out-of-distribution (OOD) performance.

3 Preliminaries

3.1 Out-of-Distribution Generalization

In the given dataset $D := (\mathbf{X}, \mathbf{Y}) = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, typically, during the training process, the data set D is divided into a training set D^{tr} and a test set D^{te}, where D^{tr} is sampled from the training distribution $\mathcal{P}_{tr}(\mathbf{X}, \mathbf{Y})$, and D^{te} is sampled from the test distribution $\mathcal{P}_{te}(\mathbf{X}, \mathbf{Y})$. There is a model that can be divided into a feature extractor $f(\cdot; \omega_f)$ and a classifier $g(\cdot; \omega_g)$. For out-of-distribution (OOD) scenarios, the training distribution is not observable, and the training distribution differs from the test distribution, i.e., $\mathcal{P}_{tr}(\mathbf{X}, \mathbf{Y}) \neq \mathcal{P}_{te}(\mathbf{X}, \mathbf{Y})$. Specifically, according to previous work, we have multiple environments in the training set, that is, $\mathcal{E} = \{e_1, e_2, ..., e_E\}$. In different environments, for

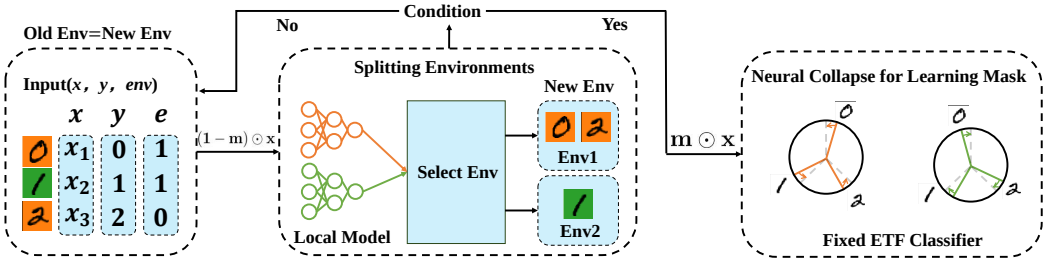


Figure 2: The overall framework of our method. **Left Figure:** Input information, including figure, label and environment. **Middle Figure:** In the scenario of unknown environments, the process of partitioning the environment. **Right Figure:** Utilizing neural collapse to guide learning of masks for extracting invariant components.

X in the training set, we can divide it into semantic parts and spurious parts, the correlation between these two is unstable, thus it is prone to generating spurious correlations.

3.2 Neural Collapse

Neural collapse refers to a phenomenon observed during the final stages of training on balanced data (post-zero training error) [Papyan et al., 2020]. It reveals a simplex ETF structure formed by the last-layer features and the classifier, which can be defined as:

Definition 1 (Simplex Equiangular Tight Frame)**.** *A simplex Equiangular Tight Frame (ETF) refers to a matrix that is composed of K vectors $\mathbf{v_i} \in R^d$, $d \geq K-1$ and satisfies:*

$$\mathbf{V} = \sqrt{\frac{K}{K-1}}\mathbf{U}\left(\mathbf{I}_K - \frac{1}{K}\mathbf{1}_K\mathbf{1}_K^T\right), \tag{1}$$

where $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_K] \in R^{d\times K}$, $\mathbf{U} \in R^{d\times K}$ allows a rotation and satisfies $\mathbf{U}^T\mathbf{U} = \mathbf{I}_K$, $\mathbf{I_K}$ represents identity matrix, and $\mathbf{1_K}$ stands for all one matrix.

For any $\mathbf{v}_i \in \mathbf{V}, i \in [1, K]$ and satisfies:

$$\mathbf{v}_i^T\mathbf{v}_j = \frac{K}{K-1}\delta_{i,j} - \frac{1}{K-1}, \quad \forall i, j \in [1, K], \tag{2}$$

where $\delta_{i,j}$ equals to 1 when $i = j$, otherwise it is equal to 0, for all $\mathbf{v}_i \in V, i = [1, K]$, there is the same L2 norm, and the inner product between any two vectors is $-\frac{1}{K-1}$.

Neural Collapse. The phenomenon of neural collapse is guaranteed by the following four findings: variability collapse (**NC1**), convergence to the simplex equiangular tight frame (**NC2**), convergence to self-duality (**NC3**), and simplification to the nearest class center (**NC4**).

NC1: During the final stages of model training, the final layer feature vectors of the same class will collapse to the class mean, i.e., for all k, $\sum_W^k \to 0$. $\Sigma_W^k = \frac{1}{n_k}\sum_{i=1}^{n_k}(\mathbf{f}_{k,i} - \mathbf{f}_k)(\mathbf{f}_{k,i} - \mathbf{f}_k)^T$, where n_k represents the number of samples in class k, $\mathbf{f}_{k,i}$ denotes the feature obtained from the i-th sample of class k, $\mathbf{f}_k$ represents the mean feature of class k.

NC2: The class means of all classes will converge to the vertices of a simplex ETF centered on the global mean, as defined in Definition 1, $\hat{\mathbf{f}}_k = (\mathbf{f}_k - \mathbf{f}_G) / \|\mathbf{f}_k - \mathbf{f}_G\|$, where $\mathbf{f}_G = \sum_{k=1}^K \sum_{i=1}^{N_k} \mathbf{f}_{k,i}$ represents the global mean of the feature.

NC3: The feature prototypes centered on the global mean will align with the corresponding classifier weights, which means that the classifier weights converge to the same simplex ETF, i.e, $\hat{\mathbf{f}}_k = \mathbf{v}_k / \|\mathbf{v}_k\|$, where $\mathbf{v}_k$ represents the classifier weight for class k.

NC4: Through the above points, it can be ensured that the network classifier converges to the nearest class center, i.e, $\arg\max_k \langle \mathbf{f}, \mathbf{v}_k \rangle = \arg\min_k \|\mathbf{f} - \mathbf{f}_k\|$, where $\mathbf{f}$ represents the features of the sample.

4 Methodology

In this section, we delve into a comprehensive discussion of our proposed method, NCFAL, along with its specific implementation. Section 4.1 outlines the framework of the proposed method for out-of-distribution based on neural collapse. We use a fixed ETF classifier to guide the alignment of invariant features across different environments, facilitating better learning of masks to separate invariant and variable features. In Section 4.2, we provide a detailed explanation of how environments are partitioned. In Section 4.3, we elaborate on how a fixed ETF classifier is used to guide mask learning. In Section 4.4, after obtaining invariant features using the trained mask, we input them into the neural network for subsequent training until convergence.

4.1 The framework of proposed method

Neural collapse reveals the optimal geometric structure of the classifier and feature prototypes after sufficient training (i.e., the simplex ETF). It inspires us to use a simplex ETF as a fixed classifier from the beginning, guiding the alignment of invariant components across different environments. Therefore, we propose a novel algorithm for the OOD generation inspired by neural collapse, NCFAL. Specifically, as described in Section 4.2, in real-world scenarios, environment information is often unknown. Hence, we need to automatically partition environments, and when certain conditions are satisfied, we consider the environment to be sufficiently well-partitioned. As described in Section 4.3, to improve model generalization, after obtaining environment information, we guide the alignment of invariant components across environments using a fixed ETF classifier. Given that class imbalance is likely across different environments, we employ a loss function to mitigate the impact of this imbalance. We alternate between Section 4.2 and Section 4.3, enabling the model to learn improved masks for partitioning invariant components. In Section 4.4, after learning the mask, we obtain semantic features for subsequent training of the predictive model.

Specifically, as shown in Figure 2, our methodological framework is illustrated. Initially, when $\mathbf{x}$ is input, it passes through $\mathbf{1} - \mathbf{m}$ for the separation of variable components, allowing for prediction using corresponding models in different environments. Subsequently, an environment selection process assigns the input to a designated environment, yielding a new environment label. When the difference between the new environment label and the previous one is less than a given threshold or a specified number of partitioning iterations is reached, the mask learning process begins. In this phase, $\mathbf{x}$ from different environments is input. Since invariant features are extracted, they should collapse onto the same simplex ETF in any environment. This approach guides the alignment of invariant features across variable environments.

4.2 The implementation of environment partitioning

This chapter focuses on environment partitioning in scenarios where environment labels are unknown. The partitioning process relies on the intermediate module depicted in Figure 2, which takes the input information and outputs a new set of environments $\mathcal{E}$, where each environment reflects a type of spurious correlation present in the input. We divide the input $\mathbf{x}$ into semantic and spurious features, using the invariant mask m to distinguish between them, resulting in semantic features $\Phi(\mathbf{x})$ and spurious features $\Psi(\mathbf{x})$: $\Phi(\mathbf{x}) = \mathbf{m} \odot \mathbf{x}$, $\Psi(\mathbf{x}) = (\mathbf{1} - \mathbf{m}) \odot \mathbf{x}$. Subsequently, the variable features $\Psi(\mathbf{x})$ are input into local models, which are associated with the environments, for identification. We designed a two-stage partitioning method to iterative partition environments until convergence.

Environment Local Model Learning Stage: Let $\mathbf{X}_e^{tr}$ be the interaction set in environment e. We aim to use local model to represent the environment e. Intuitively, the primary distinction between different environments lies in their interpretation of spurious correlations. Therefore, we model environments based on these spurious correlations. Specifically, for the interactions in environment e, we predict using a variable representation learning model $\Gamma^{(e)}$. In other words, to describe environment e, we learn the predictive model as follows:

$$\arg\min_{\omega_e} \mathcal{L}(\Gamma^{(e)}(\mathbf{x}, \Psi|\omega_e)|\mathbf{X}_e^{tr}), \tag{3}$$

where $\mathbf{X}_e^{tr}$ represents the training data $\mathbf{X}$ corresponding to environment e, ω_e represents model parameters corresponding to environment e.

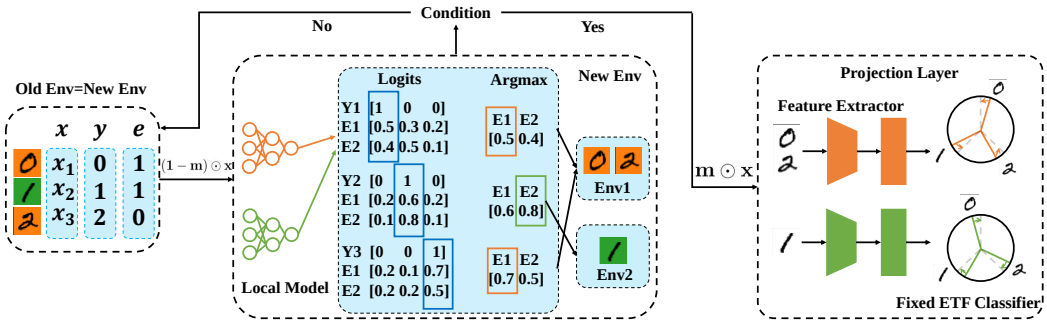


Figure 3: A comprehensive elucidation of Figure 2 is provided herein. **Middle Figure:** An illustrative example is presented to demonstrate the process of environment partitioning. **Right Figure:** An in-depth exposition of the process of learning invariant masks is provided.

Environment Partitioning Stage: In this stage, we have already developed environment models that can be used to assess the spurious correlations in different environments. Consequently, interactions should be separated based on these spurious correlations. To distinguish environments and the interactions within the input, we employ the following formula, which determines that an interaction belongs to the environment with the highest probability of identifying it. Note that predictions are based on the variable component $\Psi(\mathbf{x})$, which represents spurious information. The maximum value of this tensor is selected, indicating that the variable features of the input are more distinguishable in the environment corresponding to the maximum value. Thus, the input is assigned to that environment.

$$e(\mathbf{x}) = \arg\max_{e \in \mathcal{E}} \Gamma^{(e)}(\mathbf{x}, \Psi | \omega_e), \tag{4}$$

where for any environment $e \in \mathcal{E}$, we select the one with the maximum logit as the corresponding environment for input $\mathbf{x}$.

We illustrate this environment selection process with an example in Figure 3. Considering a logit 0 with orange background, when it is inputted into different environment models, the corresponding predictions are [0.5, 0.3, 0.2] in environment 1 and [0.4, 0.5, 0.1] in environment 2. By extracting the logits at the positions corresponding to the labels(one hot encoding), we obtain vectors [0.5, 0.4]. Taking the maximum value, we can determine the corresponding environment as environment 1.

4.3 The implementation of learning masks

This chapter primarily addresses OOD generalization by learning masks to separate invariant representations when environment labels are known. Our approach is guided by the perspective of neural collapse to align invariant representations across environments to a pre-fixed simplex ETF. As shown in Figure 3, once a suitable environment partitioning mask is learned, we apply it to the input x to obtain the invariant components. Thus, in different environments, for the semantic features $\Phi(\mathbf{x})$ in all $e_i, e_j \in supp(\mathcal{E})$, we have: $\mathcal{P}^{e_i}(\mathbf{Y}|\Phi(\mathbf{x})) = \mathcal{P}^{e_j}(\mathbf{Y}|\Phi(\mathbf{x}))$.

In different environments, we use corresponding models to align the invariant features. First, we fix a standard simplex ETF as the alignment direction. Given the potential issue of the number of different classes imbalance across different environments, we use a balanced loss to assign weights proportional to the quantity of each class for the training process:

$$\mathcal{L}_{mask} = -\log \frac{n_{e,y}^{\gamma} \exp(\beta \cdot \mathbf{v}_y^T \mathbf{f})}{\sum_k n_{e,k}^{\gamma} \exp(\beta \cdot \mathbf{v}_k^T \mathbf{f})}, k \in K, e \in \mathcal{E}, \tag{5}$$

$$\mathbf{f} = \hat{\mathbf{v}}/\|\hat{\mathbf{v}}\|, \hat{\mathbf{v}} = P(\mathbf{h}, \omega_p), \mathbf{h} = f(\Phi(\mathbf{x}), \omega_f),$$

where $n_{e,k}$ represents the number of samples of class k in environment e, $\mathbf{v}_k \in \mathbf{V}$ represents the feature vector corresponding to class k on the fixed simplex ETF $\mathbf{V}$, f represents the normalized feature $\hat{\mathbf{v}}$ obtained after projection mapping the original feature $\mathbf{h}$, β represents the learnable temperature, γ represents the sample balancing parameter, ω_f represents the parameters of the feature extraction module in the model, ω_p represents the parameters of the mapping layer in the model. When learning the mask, we add some random noise to initialize the mask:

$$m_i = \max\{0, \min\{1, m_i + \epsilon\}\}, \epsilon \sim N(0, \sigma^2), m_i \in \mathbf{m}.$$

After completing the above training steps Eq.(5), we perform a clipping operation on the mask to fix the range of m_i:

$$m_i = \max\{0, \min\{1, m_i + \epsilon\}\}, m_i \in \mathbf{m}.$$

4.4 The implementation of learning predictive model

In this chapter, we apply the environment partitioning and mask learning in Sections 4.2 and 4.3 respectively to the input data to extract semantic features. These features are then fed into the predictive neural network for training until the network converges:

$$\arg\min_{\omega^*} \mathcal{L}(\Gamma^*(\mathbf{x}, \Phi|\omega^*)|\mathbf{X}^{tr}), \tag{6}$$

where ω^* represents the parameters of the final predictive model.

Summarily, the overall training process is presented in Algorithm 1. The two steps in the **Splitting environments** process correspond to Section 4.2, while the **Learning mask** process corresponds to Section 4.3. After convergence, semantic features are used to train the predictive model, corresponding to Section 4.4, thereby achieving better OOD generalization.

Algorithm 1 The overall training process.

Data: X^{tr} for splitting environments, learning mask and training process.
Result: Predictive Model $\Gamma^*\left(\mathbf{x} \mid \omega^*, \Phi\right)$ for the final predictive process.
for $i \leftarrow 1$ *to* T **do**
  /* Splitting environments */
  if $\mathcal{E}$ *is unknown* **then**
    do
      for $e \in \mathcal{E}$ **do**
        Optimize $\Gamma^{(e)}$ via the local model training process on $\mathbf{X}_e^{tr}$ in Eq. (3);
      end
      for $e \in \mathcal{E}$ **do**
        Compute $\mathbf{X}_e^{tr}$ via the highest probability of identification as Eq. (4);
      end
    while *Converged*;
  end
  /* Learning mask */
  do
    Learn m via aligning feature prototypes to the fixed simplex ETF in Eq. (5);
  while *Converged*;
end
/* Training process */
Optimize $\Gamma^*\left(\mathbf{x} \mid \omega^*, \Phi\right)$ with semantic components $\Phi(\mathbf{x})$ via Eq. (6);

5 Experiments

5.1 Experiment Setup

Datasets. Following the work [Gulrajani and Lopez-Paz, 2020], we evaluate our method with baselines on benchmark datasets, using four datasets, namely ColoredMNIST, ColoredCOCO, COCOPlaces and NICO. **ColoredMNIST** is colorized on the MNIST dataset. The image dimensions were set to [2, 28, 28], the digits [0, 1, 2, 3, 4] are designated as category 0, while the digits [5, 6, 7, 8, 9] are designated as category 1. **ColoredCOCO** dataset is derived from the COCO dataset, which includes a selection of ten categories. Background color alterations were applied using ten different colors. All images are configured with dimensions of (3, 64, 64). **COCOPlaces** employs the same classes and settings as ColoredCOCO, with the distinction that we sample images from Places as spurious information. **NICO** dataset is a real-world dataset, including 10 subclasses for animals and 9 subclasses for vehicles. In total, our split consists of 4,080 samples of dimension (3, 224, 224) and 2 classes of the classification task. More details can be found in Appendix A.1. For each dataset, we partition it into two subsets, d_1 and d_2, based on the environment, with the ratio of sample quantities between d_1 and d_2 being 9:1. Subset d_1 from the training environment is used for

model training, while d_2 is employed for testing the models within the training environment. In the testing environment, d_1 is utilized to assess the out-of-distribution (OOD) performance of the models, while d_2 is utilized according to DomainBed standards for selecting the best model.

Architecture. On ColorMNIST, training is conducted using a 4-layer convolutional neural network. For the ColoredCOCO and COCOPlaces datasets, we adhere to the setup outlined in Ahmed et al. [2020], Gulrajani and Lopez-Paz [2020], employing ResNet8 for training. On the NICO dataset, training is performed using ResNet18.

Baselines. In order to demonstrate the advantages and effectiveness of our approach, we conduct comparative tests against different OOD learning methods. including (1) IID learning: ERM [Vapnik et al., 1998] (2) OOD learning (sixteen methods): IRM [Arjovsky et al., 2019], VREx [Krueger et al., 2021], ARM [Zhang et al., 2021], GroupDRO [Sagawa et al., 2020], MLDG [Li et al., 2018a], MMD [Li et al., 2018b], IGA [Koyama and Yamaguchi, 2020], SANDMask [Shahtalebi et al., 2021], Fish [Shi et al., 2021], CDANN [Li et al., 2018c], TRM [Xu and Jaakkola, 2021], IB_ERM [Ahuja et al., 2021], IB_IRM [Ahuja et al., 2021], CondCAD [Ruan et al., 2021], CausIRL_CORAL [Chevalley et al., 2022], MAP [Zhang et al., 2023].

5.2 The comparison of OOD accuracy (%) between our method and other approaches.

Table 1 presents a comparison between our method, the ERM algorithm, and 16 other OOD algorithms. It is evident that our approach outperforms all methods under the given environmental conditions. Moreover, in situations where the environment is unknown, our method demonstrates adaptive capabilities in partitioning the environment, achieving superior OOD performance. The reason behind this phenomenon, we believe, lies that sometimes manually partitioning the environment may not be optimal for neural networks to identify and discern spurious correlations, however, Utilizing neural networks to partition the environment based on predictions might be more adapt at separating semantic and spurious features. Particularly for COCOPlaces, due to the complexity of backgrounds, manual partitioning may not always achieve appropriate segmentation. Therefore, from the results, when the environment is known, the OOD accuracy is 33.3%; however, using our method for environment partitioning yields a result of 36.7%.

Table 1: Average accuracy (%) of OOD on three toy and one real datasets using different methods.

	ColoredMNIST	ColoredCOCO	COCOPlaces	NICO
ERM [Vapnik et al., 1998]	51.5 ± 0.1	45.4 ± 0.9	20.1 ± 0.7	73.6 ± 1.9
IRM [Arjovsky et al., 2019]	60.3 ± 2.8	49.2 ± 0.3	27.1 ± 0.9	75.8 ± 2.0
VREx [Krueger et al., 2021]	52.9 ± 1.2	48.8 ± 0.7	26.2 ± 0.7	76.9 ± 0.7
GroupDRO [Sagawa et al., 2020]	38.5 ± 1.5	49.1 ± 0.6	26.9 ± 0.6	74.6 ± 2.4
MLDG [Li et al., 2018a]	29.4 ± 0.6	11.9 ± 0.8	14.6 ± 0.5	68.4 ± 2.7
MMD [Li et al., 2018b]	50.6 ± 0.1	50.4 ± 0.8	26.3 ± 1.7	78.2 ± 1.2
IGA [Koyama and Yamaguchi, 2020]	50.5 ± 0.1	11.0 ± 0.6	10.8 ± 0.3	48.1 ± 1.3
SANDMask [Shahtalebi et al., 2021]	58.6 ± 6.5	49.2 ± 1.2	25.9 ± 1.4	72.8 ± 1.5
Fish [Shi et al., 2021]	28.0 ± 1.5	41.7 ± 0.5	19.3 ± 2.1	77.0 ± 1.2
CDANN [Li et al., 2018c]	41.7 ± 3.5	38.4 ± 1.5	19.4 ± 1.0	72.8 ± 1.8
TRM [Xu and Jaakkola, 2021]	44.2 ± 5.0	47.5 ± 0.6	24.8 ± 1.1	73.0 ± 0.9
IB_ERM [Ahuja et al., 2021]	50.2 ± 0.2	45.4 ± 1.1	20.2 ± 1.0	77.7 ± 1.9
CausIRL_CORAL [Chevalley et al., 2022]	28.7 ± 1.3	51.5 ± 1.1	26.1 ± 1.1	75.7 ± 0.9
CondCAD [Ruan et al., 2021]	49.2 ± 0.5	41.2 ± 0.7	20.8 ± 0.3	73.9 ± 1.4
IB_IRM [Ahuja et al., 2021]	53.8 ± 1.8	33.9 ± 0.6	14.8 ± 2.3	70.2 ± 2.2
ARM [Zhang et al., 2021]	28.1 ± 0.0	33.0 ± 0.6	25.1 ± 0.2	76.4 ± 1.6
MAP [Zhang et al., 2023]	52.6 ± 0.5	50.9 ± 1.3	26.9 ± 1.0	76.8 ± 1.4
Ours	$\mathbf{66.4 \pm 0.2}$	$\mathbf{58.0 \pm 0.2}$	$\mathbf{33.3 \pm 1.7}$	$\mathbf{85.4 \pm 0.6}$
Ours (w/o env)	$\mathbf{66.9 \pm 2.4}$	$\mathbf{56.9 \pm 1.1}$	$\mathbf{36.7 \pm 0.9}$	$\mathbf{85.9 \pm 0.2}$

5.3 Ablation Studies

The comparison between other methods employing masking techniques and our method. Table 2 presents the effectiveness of our proposed method for learning masks, we conducted comparisons with the IRM and REx methods, which incorporate regularization to facilitate learning of the mask through the gradient or variance. In contrast, our approach leverages the phenomenon of neural collapse to guide the optimization direction for semantic feature enhancement instead of regularization. Simultaneously, we compared our method with the ERM approach, confirming the superiority of using masks to acquire the invariant parts.

Table 2: The OOD accuracy (%) of our method and other regularization methods on three datasets.

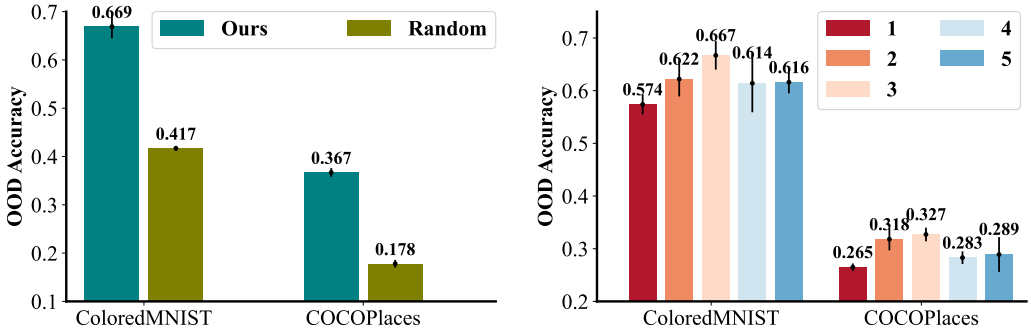
	ColoredMNIST	ColoredCOCO	COCOPlaces
Ours	**66.4 ± 0.2**	**58.0 ± 0.2**	**33.3 ± 1.7**
Ours (w/o env)	**66.9 ± 2.4**	**56.9 ± 1.1**	**36.5 ± 0.8**
REx	61.3 ± 3.0	56.0 ± 0.2	27.4 ± 1.3
REx (w/o env)	57.9 ± 2.0	52.9 ± 1.1	28.2 ± 1.7
IRM	64.4 ± 0.2	51.8 ± 0.8	32.2 ± 0.8
IRM (w/o env)	65.2 ± 0.4	52.8 ± 0.4	30.7 ± 1.7
ERM	51.5 ± 0.1	45.4 ± 0.9	20.1 ± 0.7

The comparison between mask is applied at the pixel level and the feature level. To validate the generality of our method, we conducted comparative experiments at both the pixel level and the feature level in Table 3. It was observed that sometimes extracting invariant features at the feature level could yield better results. In particular, improved performance was observed on the ColoredCOCO and COCOPlaces datasets. This phenomenon primarily stems from the fact that feature-level information is high-dimensional, capturing relationships that are difficult to discern.

Table 3: The OOD accuracy (%) of our method based on pixel and feature level on three datasets.

	ColoredMNIST	ColoredCOCO	COCOPlaces
Ours	**66.4 ± 0.2**	58.0 ± 0.2	33.3 ± 1.7
Ours (w/o env)	**66.9 ± 2.4**	56.9 ± 1.1	36.5 ± 0.8
Feature	58.7 ± 2.8	**63.9 ± 0.5**	**43.7 ± 0.7**
Feature (w/o env)	54.0 ± 0.3	**62.7 ± 1.8**	**44.1 ± 0.1**

The comparison between randomly and using maximum likelihood probability method to split environments. Compared with random environment splits in Figure 4(a), our method, which involves assigning different models to distinct environments and selecting maximum likelihood probabilities outputted by the models, was found to effectively partition environments.



(a) The method for selecting environments (b) The number of environments to partition

Figure 4: Comparative experiments for randomly and the different number of divided environments.

The comparison between different number of environment divisions. Compared with different number of environment splits in Figure 4(b), we can find that when the number of divided environments is closer to the actual number of given environments, better performance is achieved, thereby validating the effectiveness of our environment division method.

6 Conclusion

We explore a new perspective to understand the inherent limitations of ERM and IRM-based methods, which fail due to the inability to align semantic features across environments, resulting in reduced generalization performance. By leveraging the phenomenon of neural collapse to guide the alignment of semantic features across environments. Compared to other OOD methods, we have significantly improved OOD performance. Moreover, in real-world scenarios where environment labels are unknown, our method addresses this by training local models to different environments, automatically achieving environment partitioning. This method greatly reducing the cost of manual annotation and expanding the applicability of our method. Additionally, we will investigate and improve more

effective and efficient environment partitioning techniques in future work. We can also explore applying this method in other domains, such as segmentation or few-shot spurious correlation issues.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (No. 623B2002) and the National Natural Science Foundation of China (No. 62176132). MG was supported by ARC DE210101624, ARC DP240102088, and WIS-MBZUAI 142571.

References

Faruk Ahmed, Yoshua Bengio, Harm Van Seijen, and Aaron Courville. Systematic generalisation with group invariant predictions. In *International Conference on Learning Representations*, 2020.

Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Jean-Christophe Gagnon-Audet, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. Invariance principle meets information bottleneck for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 34:3438–3450, 2021.

Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand. Domain-adversarial neural networks. *arXiv preprint arXiv:1412.4446*, 2014.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Daniel Beaglehole, Peter Súkeník, Marco Mondelli, and Mikhail Belkin. Average gradient outer product as a mechanism for deep neural collapse. *arXiv preprint arXiv:2402.13728*, 2024.

Mathieu Chevalley, Charlotte Bunne, Andreas Krause, and Stefan Bauer. Invariant causal mechanisms through distribution matching. *arXiv preprint arXiv:2206.11646*, 2022.

Quinn Fisher, Haoming Meng, and Vardan Papyan. Pushing boundaries: Mixup's influence on neural collapse. *arXiv preprint arXiv:2402.06171*, 2024.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020.

Li Guo, Keith Ross, Zifan Zhao, Andriopoulos George, Shuyang Ling, Yufeng Xu, and Zixuan Dong. Cross entropy versus label smoothing: A neural collapse perspective. *arXiv preprint arXiv:2402.03979*, 2024.

Wenlong Ji, Yiping Lu, Yiliang Zhang, Zhun Deng, and Weijie J Su. An unconstrained layer-peeled perspective on neural collapse. *arXiv preprint arXiv:2110.02796*, 2021.

Vignesh Kothapalli, Tom Tirer, and Joan Bruna. A neural collapse perspective on feature evolution in graph neural networks. *Advances in Neural Information Processing Systems*, 36, 2024.

Masanori Koyama and Shoichiro Yamaguchi. When is invariance useful in an out-of-distribution generalization problem? *arXiv preprint arXiv:2008.01883*, 2020.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pages 5815–5826. PMLR, 2021.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018a.

Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5400–5409, 2018b.

Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 624–639, 2018c.

Zexi Li, Xinyi Shang, Rui He, Tao Lin, and Chao Wu. No fear of classifier biases: Neural collapse inspired federated learning with synthetic and fixed classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5319–5329, 2023.

Yong Lin, Hanze Dong, Hao Wang, and Tong Zhang. Bayesian invariant risk minimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16021–16030, 2022.

Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Heterogeneous risk minimization. In *International Conference on Machine Learning*, pages 6804–6814. PMLR, 2021.

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.

Yangjun Ruan, Yann Dubois, and Chris J Maddison. Optimal representations for covariate shift. *arXiv preprint arXiv:2201.00057*, 2021.

Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pages 8346–8356. PMLR, 2020.

Soroosh Shahtalebi, Jean-Christophe Gagnon-Audet, Touraj Laleh, Mojtaba Faramarzi, Kartik Ahuja, and Irina Rish. Sand-mask: An enhanced gradient masking strategy for the discovery of invariances in domain generalization. *arXiv preprint arXiv:2106.02266*, 2021.

Yuge Shi, Jeffrey Seely, Philip HS Torr, N Siddharth, Awni Hannun, Nicolas Usunier, and Gabriel Synnaeve. Gradient matching for domain generalization. *arXiv preprint arXiv:2104.09937*, 2021.

Peter Súkeník, Marco Mondelli, and Christoph H Lampert. Deep neural collapse is provably optimal for the deep unconstrained features model. *Advances in Neural Information Processing Systems*, 36, 2024.

Tom Tirer and Joan Bruna. Extended unconstrained features model for exploring deep neural collapse. In *International Conference on Machine Learning*, pages 21478–21505. PMLR, 2022.

Yunze Tong, Junkun Yuan, Min Zhang, Didi Zhu, Keli Zhang, Fei Wu, and Kun Kuang. Quantitatively measuring and contrastively exploring heterogeneity for domain generalization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2189–2200, 2023.

Vladimir Naumovich Vapnik, Vlamimir Vapnik, et al. Statistical learning theory. 1998.

Liang Xie, Yibo Yang, Deng Cai, and Xiaofei He. Neural collapse inspired attraction–repulsion-balanced loss for imbalanced learning. *Neurocomputing*, 527:60–70, 2023.

Yilun Xu and Tommi Jaakkola. Learning representations that support robust transfer of predictors. *arXiv preprint arXiv:2110.09940*, 2021.

Yibo Yang, Haobo Yuan, Xiangtai Li, Zhouchen Lin, Philip Torr, and Dacheng Tao. Neural collapse inspired feature-classifier alignment for few-shot class incremental learning. *arXiv preprint arXiv:2302.03004*, 2023.

Marvin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: Learning to adapt to domain shift. *Advances in Neural Information Processing Systems*, 34:23664–23678, 2021.

Min Zhang, Junkun Yuan, Yue He, Wenbin Li, Zhengyu Chen, and Kun Kuang. Map: Towards balanced generalization of iid and ood through model-agnostic adapters. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11921–11931, 2023.

Didi Zhu, Yinchuan Li, Yunfeng Shao, Jianye Hao, Fei Wu, Kun Kuang, Jun Xiao, and Chao Wu. Generalized universal domain adaptation with generative flow networks. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 8304–8315, 2023a.

Didi Zhu, Yinchuan Li, Min Zhang, Junkun Yuan, Jiashuo Liu, Kun Kuang, and Chao Wu. Bridging the gap: neural collapse inspired prompt tuning for generalization under class imbalance. *arXiv preprint arXiv:2306.15955*, 2023b.

Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34:29820–29834, 2021.

A Implementation Details

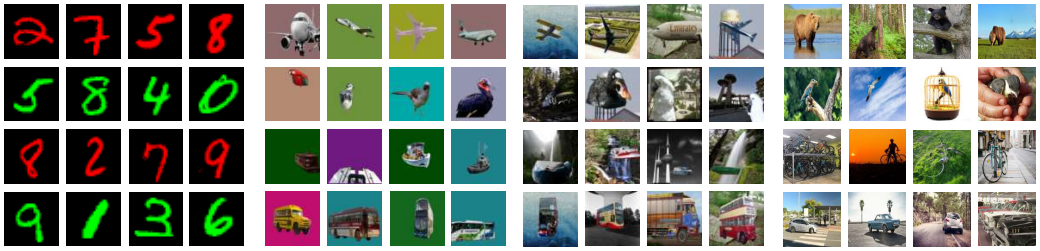
A.1 Datasets

ColoredMNIST, introduced by IRM for assessing spurious correlations in out-of-distribution (OOD) problems, follows the configuration outlined in [Zhang et al., 2023]. In accordance with this setup, we colorized the MNIST dataset with red and green hues to establish a strong correlation between digits and colors. The image dimensions were set to [2, 28, 28], and the correlation coefficients for two training sets and one test set were specified as (0.9, 0.8, 0.1), indicating the proportion of red and green within category 0. The digits [0, 1, 2, 3, 4] are designated as category 0, while the digits [5, 6, 7, 8, 9] are designated as category 1.

ColoredCOCO dataset is derived from the COCO dataset, which includes a selection of ten categories: airplane, bird, boat, bus, dog, horse, motorcycle, train, truck, and zebra. Background color alterations were applied using ten different colors. Their RGB values are [0, 100, 0], [188, 143, 143], [255, 0, 0], [255, 215, 0], [0, 255, 0], [65, 105, 225], [0, 225, 225], [0, 0, 255], [255, 20, 147] and [160, 160, 160]. The number of samples for each training environment is 400 for each class but the testing environment is 200 for each class. All images are configured with dimensions of (3, 64, 64).

COCOPlaces employs the same classes and settings as ColoredCOCO, with the distinction that we sample images from Places as spurious information [Liu et al., 2021], such as b/beach, c/canyon, b/building facade, s/staircase, d/desert/sand, c/crevasse, b/bamboo forest, f/forest/broadleaf, b/ball pit and o/oast house. Moreover, some random places are also used, i.e., k/kasbah, l/lighthouse, p/pagoda, r/rock arch, w/water tower, w/waterfall, z/zen garden.

NICO dataset is a real-world dataset including photos of animals and vehicles captured in a wide range of contexts (or backgrounds). There are 10 subclasses for animals and 9 subclasses for vehicles, with each subclass having 9 or 10 different contexts. Following [Lin et al., 2022], we select a subset of this dataset to simulate the spurious correlation of different contexts and classes (animal or vehicle), which is similar to the setting of ColoredMNIST. More specifically, we make use of both classes that appear in four overlapped contexts: “on snow”, “in forest”, “on beach” and “on grass” to construct two training environments and one testing environment. In total, our split consists of 4,080 samples of dimension (3, 224, 224) and 2 classes of the classiffcation task.



(a) ColoredMNIST (b) ColoredCOCO (c) COCOPlaces (d) NICO

Figure 5: Illustration of the ColoredMNIST, ColoredCOCO, COCOPlaces, and NICO datasets.

A.2 Impact Statements

We propose a new perspective for addressing the OOD generation problem by using neural collapse to align semantic features across different environments. Our method can be applied both when environments are given and when they are unknown, as it can autonomously partition environments. This approach enables the model to independently learn semantic features, thereby achieving better generalization in future practical applications.

A.3 Devices

In our experiments, we conduct all methods on a local Linux server that has two physical CPU chips (Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz) and 32 logical kernels. All methods are implemented using Pytorch framework and all models are trained on GeForce RTX 2080 Ti GPUs.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have done this work.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed the limitations of our work in conclusion part.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Our work studies the algorithm for Out-of-Distribution Generation, and provides empirical evidences.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have done this work in our manuscript.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have strictly adhered to the anonymity guidelines when uploading the code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We have provided the training and test details in main text.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We have done this work in our paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have provided computer resources in appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We are make sure to preserve anonymity.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discussed in boarder impact in appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This work focuses on addressing the OOD generation, which is not related to the misuse risk.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: We use public datasets in our experiments.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We don't publish new data assets in this work.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We conduce experiments on existing benckmark datasets.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This work is not related to human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.