LatentDR: Improving Model Generalization Through Sample-Aware Latent Degradation and Restoration

Ran Liu, Sahil Khose, Jingyun Xiao, Lakshmi Sathidevi, Keerthan Ramnath, Zsolt Kira, Eva L. Dyer
Georgia Institute of Technology, Atlanta, GA, 30332

Abstract

Despite significant advances in deep learning, models often struggle to generalize well to new, unseen domains, especially when training data is limited. To address this challenge, we propose a novel approach for distribution-aware latent augmentation that leverages the relationships across samples to guide the augmentation procedure. Our approach first degrades the samples stochastically in the latent space, mapping them to augmented labels, and then restores the samples from their corrupted versions during training. This process confuses the classifier in the degradation step and restores the overall class distribution of the original samples, promoting diverse intra-class/cross-domain variability. We extensively evaluate our approach on a diverse set of datasets and tasks, including domain generalization benchmarks and medical imaging datasets with strong domain shift, where we show our approach achieves significant improvements over existing methods for latent space augmentation. We further show that our method can be flexibly adapted to long-tail recognition tasks, demonstrating its versatility in building more generalizable models. https://github.com/nerdslab/LatentDR.

1. Introduction

In machine learning, it is often challenging to train a model that generalizes well when tested on domains that it was not exposed to during training [7]. This is especially true when the available training data is limited, as is often the case in real-world applications. Various methods have been proposed to address this challenge, including feature alignment approaches [15,51], meta-learning [2,46], and data augmentations [57,67]. However, achieving robust out-of-domain (OOD) generalization remains a challenging problem that requires further research [16].

Among existing strategies, one emerging technique for addressing the problem of OOD generalization is *latent augmentation*. Similar to data augmentation methods, the overall objective of latent augmentation is to increase the diversity of the source data so that the model is encouraged to learn domain-agnostic representations [56]. However, different from data augmentation methods which directly manipulate the input data, latent augmentation modifies the hidden data representations or feature space of a model, which avoids the usage of generative networks [67], additional classifiers [47], or adversarial training [57].

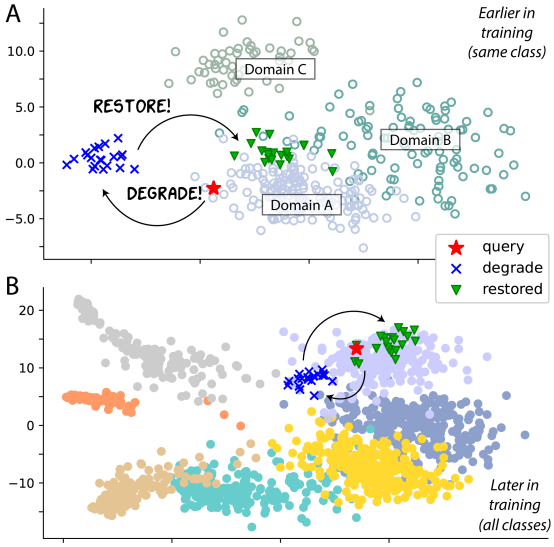

Figure 1. *Visualization of latent degradation and restoration steps during training.* In **(A)**, when it is earlier in training, latents from different domains are separated for the same class, which hurts model generalization. To tackle this issue, for each latent query (★), `LatentDR` first degrades (×) them to a point that is far away from the existing samples, and restores (▼) them back to the existing distribution to diversify existing domains and improve model generalization. In **(B)**, we visualize the latents for all classes at a later point in training and color data by classes. More details and visualizations are in Appendix D.

Many existing latent augmentation methods typically assume that linear combinations [56], the mixture of style statistics [17, 49], or additional randomization [33] of hidden representations can provide the desired diversity for generalization. However, when dealing with highly diverse domains, using simple assumptions (like linear mixing) is often insufficient [10], and generating robust latent augmentations remains an outstanding challenge. Here we ask

if we can go beyond pairwise mixing and use sample-to-sample relationships across points in the latent space to build sample-aware latent augmentations that can allow us to transport across domains.

To build sample-aware latent augmentations, we take inspiration from recent work where batch-level relationships are learned through an attention mechanism to "reconstruct" the original latent samples [18]. However, rather than trying to reconstruct the original sample, we consider a bi-directional objective where the goal is to both create a *degraded* sample that confuses a downstream classifier and also create a *restored* sample that recovers the class information after this transformation. In both steps, we use relationships across many samples within mini-batches to help build augmentations. The idea is shown in Figure 1(A): The degradation process aims to map a query latent to a point where its class cannot be easily distinguished by a downstream classifier; the restoration process uses a cross-attention mechanism [22, 54] to (only) recover the class information from the degraded latents by conditioning on the original latents. By combining degradation and restoration, `LatentDR` encourages the model to capture the relationship across samples from different classes and domains, and thus guides the model to generalize better to unseen sources.

To test our approach in diverse settings, we conduct experiments on: (i) five standardized domain generalization benchmarks [10, 16], (ii) five medical imaging classification datasets that suffer from domain shift [26, 61], and (iii) a long-tail (LT) recognition task where our latent augmentation technique can improve generalization in the presence of strong data imbalance. In all of these cases, we provide impressive boosts over other augmentation-based approaches on all of the datasets tested, and competitive performance with state-of-the-art methods that use more complex losses and domain information. To better understand the effectiveness of our approach on learning, we use two measures of representation quality [58] and nearest neighbor visualization to measure the impact of our approach.

In summary, the contributions of our work include:

- We propose a novel approach for latent augmentation that aims to build connections across samples from different domains by both *degrading* and *restoring* the samples from the latent distribution.
- We conduct empirical studies to demonstrate that our approach improves class-level alignment of features, as well as the uniformity or spread of the data distribution. These two properties are shown to be strongly correlated with downstream task performance [58].
- We extensively evaluate the robustness of our method on domain generalization benchmarks and medical imaging datasets with strong domain shifts, where we provide various ablations to better understand different components of the method. Finally, we further demonstrate the versatility of our augmentation method by applying it to a long-tail recognition task.

2. Background and Related Work

2.1. Domain generalization (DG)

Various methods have been proposed to address the challenge of domain generalization, including applying explicit domain alignment [15, 51], learning domain-agnostic representations [21, 40, 65], meta-learning [2, 9, 46], enforcing domain invariance by adjusting direction of the gradients [43, 47, 48], and data or latent augmentation methods [17, 33, 49]. An orthogonal line of research aims to develop better optimization approaches to alleviate the overfitting issue [1, 10, 28], which often happens in DG settings. More recently, a line of work focuses on taming large-scale pre-trained models to improve domain generalization performance through oracle methods [11, 36] or prompts [41, 66]. Despite the numerous proposed approaches, domain generalization remains a challenging problem, with only a few methods consistently outperforming Empirical Risk Minimization (ERM) with properly designed data augmentation and training protocols [16].

2.2. Data and latent augmentation for DG

Augmentation approaches have been extensively used to improve the generalization performance of models [3, 37]. Commonly used image-space data augmentation methods for DG include general-purpose operations [12, 25, 63, 64], and DG-specific methods [46, 47, 57, 67, 68]. Due to the additional complexity and high computation demand of operating directly on data space, recent works focus on exploring the use of latent space augmentation methods [56], which aim to manipulate data in the feature space. For instance, Zhou et al. [69] and Somavarapu et al. [49] proposed to perform linear mixing of style statistics with AdaIN module [20], while Li et al. [33] and Wang et al. [60] introduced randomization modules to augment the latent space. However, many of these models operate on within-domain datasets and few studies have demonstrated their scalability in large-scale DG experiments [16].

Sample-aware approaches. Augmentation methods typically rely on sample-to-sample relationships to improve model generalization, such as Mixup which interpolates between pairs of samples. Recently, non-convex methods have been proposed to learn more sophisticated sample-to-sample relationships. For example, Batch-Graph [59] models relationships using a graph, while BatchFormer [18, 19] uses self-attention to encourage gradient flow between samples in a batch. Building on these works, `LatentDR` also models sample-to-sample relationships using attention, but with the goal of degrading and restoring the latents using information from other samples in the batch. By leveraging

the relationships across samples, our method aims to confuse the classifier during degradation and restore the original class distribution during restoration, ultimately improving the model's ability to generalize to new domains.

2.3. Transformers

While transformers were initially proposed in the context of natural language processing applications for learning sequences [54], the use of transformers and related components is becoming commonplace in vision applications [13, 39], augmentation [50], and conditioning [45].

Transformer with self-attention. A typical transformer layer processes a sequence of inputs $X \in \mathbb{R}^{N\times d}$ through the multi-head self-attention (MSA) mechanism to learn the relationship among N tokens of d-dimensions. For each token, we derive its queries Q, keys K, and values V using linear projections, and perform the following operations:

$$Q = XW_Q, K = XW_K, V = XW_V \tag{1}$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(QK^T/\sqrt{d_k}\right)V, \tag{2}$$

where $\text{softmax}(\cdot)$ gives row-wise softmax normalization.

The transformer architecture divides self-attention into multiple different heads, and combines the attention mechanism (MSA), MLP blocks (FF), and layerwise normalization (LN). A transformer layer consists of below operations:

$$\begin{aligned} Z'_{\ell+1} &= \text{LN}(Z_\ell + \text{MSA}(Z_\ell)) \\ Z_{\ell+1} &= \text{LN}(Z'_{\ell+1} + \text{FF}(Z'_{\ell+1})),\ 0 \le \ell \le L-1 \end{aligned} \tag{3}$$

where Z_ℓ is the output of the l-th layer of the transformer.

Study correlation across tokens with spatial operations. Recent works have shown that self-attention can be replaced with a spatial MLP [52], or even a Pooling operation [62], to study the spatial correlations across tokens. In the case of vision transformers [13], where each token represents a patch in the image, the pooling operator simply takes an average or max pooling over subsets of embeddings for spatially-correlated (nearby) patches.

Conditioning with cross-attention. Whereas self-attention considers the pairing of queries and keys across tokens in the same sequence, cross-attention (CA) can be used to compute attention between two different input sources or modalities [22, 29, 45, 54]: Given $X_1 \in \mathbb{R}^{N\times d_1}$ and $X_2 \in \mathbb{R}^{N\times d_2}$, the queries Q, keys K, and values V are computed from different input sources:

$$Q = X_1W_Q, K = X_2W_K, V = X_2W_V \tag{4}$$

The permutation-invariant property of the cross attention mechanism allows the generated token to be dependent on the queries (X_1) while keeping the information from the conditions (X_2).

3. Method

3.1. Overview of our method

To capture relationships across samples and build robustness into the model, our latent augmentation operation is learned through a *degradation* (D) step and a *restoration* (R) step, hence the name `LatentDR`. We visualize both steps in Figure 1 and provide an overview of the architecture in Figure 2. The method does not require layerwise manipulations, does not rely on domain information, and can be implemented easily within the mini-batch training process and removed during inference.

Consider (x, y) as an original datapoint and label pair from our data distribution $(\mathcal{X}, \mathcal{Y})$. Let $z = f(x)$ denote the embedding of x after the encoder $f : \mathcal{X} \to \mathbb{R}^d$, and $g : \mathbb{R}^d \to \mathcal{Y}$ as the classifier that produces a final prediction. The overall objective of the method is to learn a latent degradation operator $D_\mu(;\xi)$ that produces stochastic (ξ) embeddings that have the potential to 'confuse' a downstream classifier; and a latent restoration operator R_θ that would 'denoise' the prediction of the labels.

This high-level objective can be specified as:

$$\begin{aligned} \textit{degrade:} \quad & \min_{f,g,\mu} \ell(g(D_\mu(z;\xi)), \tilde{y}) \\ \textit{restore:} \quad & \min_{f,g,\theta,\mu} \ell(g(R_\theta(D_\mu(z;\xi))), y) \end{aligned} \tag{5}$$

where ℓ is a classification loss, $\tilde{y}$ is a distribution-aware label that is constructed to guide mixing in latent space (detailed in Section 3.2), θ and μ denotes the weights of restoration and degradation operators, and ξ denotes the stochasticity or randomness in the degradation process. Hence, while the restoration operator learns to predict the original distribution of labels, the degradation operator aims to introduce non-trivial perturbation to the latent variables to alter the classifier's prediction. We provide more motivations in Appendix A. Note that, different from other methods [32], our recovered latent $z_r = R_\theta(D_\mu(z;\xi))$ does not necessarily need to be close to its original position z as long as the classifier's prediction is correct, which encourages the latents to go across domains for domain generalization.

3.2. LatentDR

3.2.1 Latent degradation with soft-labels

In order to encourage the datapoints from different sources to mix with each other, we rely on *sample-to-sample relationships* between points to construct the latent degradation operator $D_\mu(;\xi)$ and the corresponding soft-labels $\tilde{y}$.

Let $(\mathcal{S}_X, \mathcal{S}_Y) = \{(x_i, y_i)\}_{i=1}^B$ denote a set of samples from $(\mathcal{X}, \mathcal{Y})$. Let $Z_{set} \in \mathbb{R}^{B\times d}$ denote a matrix containing the embeddings of the samples in $\mathcal{S}_X$ and let $z = f(x)$ be a *query* latent vector. In practice, one convenient way is to draw a query from the batch and use

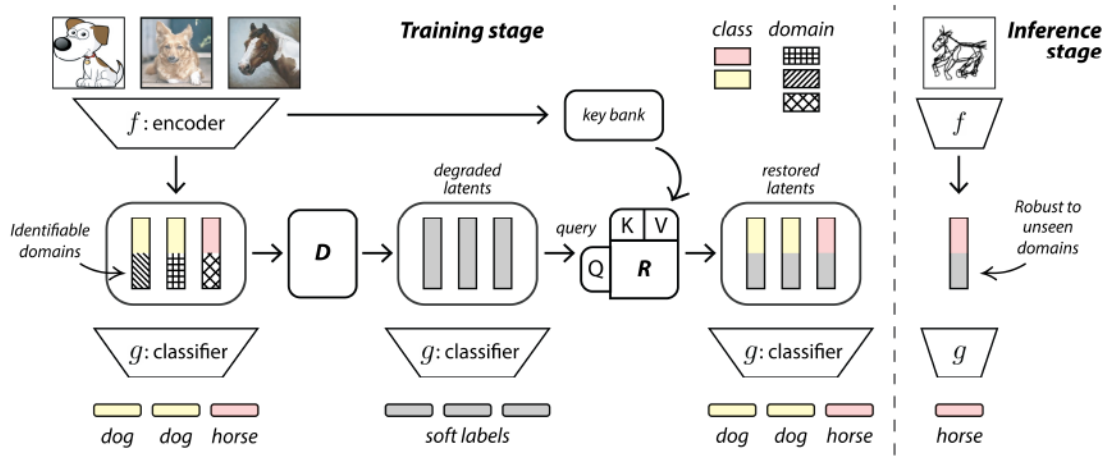

Figure 2. *Detailed architecture of LatentDR.* **(A)** Without latent augmentation, latents produced by encoders would contain identifiable information about domains, which hurts the generalization ability of the model. To address this issue, `LatentDR` first uses a degradation operator to produce degraded latents that are mapped to constructed soft labels. Subsequently, we fed degraded latents as queries and original latents as keys to a cross-attention transformer, which restores latents to assorted domains. **(B)** Applying augmentations during training improves the model's ability to generalize, resulting in latent features that are robust to unseen domains, even when the degradation and restoration operators are removed during inference.

the remaining samples as Z_{set}. The goal is to generate $(\text{Mix}_{\text{data}}(z, Z_{set}), \text{Mix}_{\text{label}}(y, Y_{set}))$ where Mix is a general (and potentially nonlinear) mixing operator that combines samples and labels within the set to build an augmentation of the query z and its label y. We can consider latent Mixup and other pairwise mixture schemes as a special instance where Z_{set} consists of only one sample and the combination operator is (typically) linear for both data and label.

To create non-linear and non-trivial sample-aware augmentations, we rely on transformers similar as in [18, 19]. Specifically, in this work, we tested two variants of transformers, where the first uses self-attention (SA) and the other a pooling mechanism (Pool). Thus, we can generate the degraded augmentation z_d of a query z as follows:

$$\begin{aligned} (SA) \quad & z'_d = z + \text{AttN}(Z_{set}), \\ (Pool) \quad & z'_d = z + \text{Pool}(\Omega(Z_{set})), \\ & z_d = z'_d + \text{MLP}(z'_d) \end{aligned} \tag{6}$$

where $\Omega(\cdot)$ is a spatial selection operator, and $\text{Pool}(\cdot)$ takes an average pooling over the selected subset. In practice, we also apply normalization (see Appendix A). The stochasticity (parameterized by ξ in our degradation operator) is introduced through a high dropout rate of 50% in both operations, and additionally by taking a different subset of samples in the Pooling variant.

To encourage the degraded latent to be apart from the query, we define the mixing operator for our labels as:

$$\widetilde{y} = \sum_{y_i \in \mathcal{S}_Y} w_i y_i. \tag{7}$$

where we take $w_i = 1/B$ for simplicity. Note that Z_{set} fits perfectly into mini-batch training, such that it can just be a batch of data that produces degraded latent samples Z_d.

3.2.2 Latent restoration with cross-attention

To encourage the model to preserve critical semantic information during the degradation process, we perform *latent restoration* to recover the latents back to their original classes. Our model leverages the cross-attention mechanism that computes queries Q from the corrupted latents Z_d and the keys K and values V from the original latents Z_{set}:

$$Q = Z_d W_Q, K = Z_{set} W_K, V = Z_{set} W_V, \tag{8}$$

where we use them to produce the restored latent z_r:

$$\begin{aligned} z'_r &= z_d + \text{softmax}(QK^T)V \\ z_r &= z'_r + \text{MLP}(z'_r) \end{aligned} \tag{9}$$

As the attention mechanism is permutation-invariant to the ordering of the keys, the output is decided based on the ordering of the queries, which enforces the corrupted latents to contain the information that is sufficient to recover the class information of the original latents. Thus, intuitively, the restoration operator generates a new latent z_r by having z_d to select similar samples from the original latent set Z_{set}, which provides assistance in the restoration process.

3.2.3 Regularization with classifier guidance

Thus, for each sample x, we created three latent and label pairs: (z, y) as the original ones, $(z_d, \tilde{y})$ as the degraded ones, and (z_r, y) as the restored ones. We perform a relaxed regularization with classifier guidance, where our model

	(1)	(2)	Model	PACS	VLCS	OfficeHome	TerraInc	DomainNet	Avg.
Algorithms	✓	✓	MMD† [32]	84.7	77.5	66.4	42.2	23.4	58.8
	✗	✓	DANN† [35]	83.7	78.6	65.9	46.7	38.3	62.6
	✓	✓	SagNet† [40]	**86.3**	77.8	68.1	48.6	40.3	64.2
	✓	✓	CORAL† [51]	86.2	78.8	68.7	47.7	41.5	64.5
	✓	✓	MLDG† [30]	84.9	77.2	66.8	47.7	41.2	63.6
	✗	✓	Fishr† [43]	85.5	77.8	67.8	47.4	41.7	64.0
	✓	✗	MIRO† [11]	85.4	**79.0**	**70.5**	**50.4**	**44.3**	**65.9**
Augmentations	-	-	ERM	84.1	76.7	64.8	47.0	41.9	62.9
	✗	✗	+ Mixup [64]	83.1	76.9	67.0	48.0	43.0	63.6
	✗	✗	+ CutMix [63]	80.4	74.9	66.9	**52.1**	43.2	63.5
	✓	✗	+ M-Mixup [56]	84.2	77.8	67.4	46.3	43.0	63.7
	✓	✓	+ MixStyle† [69]	85.2	77.9	60.4	44.0	34.0	60.3
	✗	✗	+ BatchFormer [18]	82.7	76.7	65.8	48.6	42.8	63.3
	✗	✗	+ `LatentDR` (SA)	85.8 (↑1.7)	**78.7** (↑2.0)	**69.0** (↑4.2)	49.9 (↑2.9)	**45.1** (↑3.2)	**65.7** (↑2.8)
	✗	✗	+ `LatentDR` (Pool)	**86.3** (↑2.2)	78.0 (↑1.3)	68.4 (↑3.6)	49.5 (↑2.5)	43.9 (↑2.0)	65.2 (↑2.3)

Table 1. *Results on domain generalization benchmarks.* All models use a ResNet-50 backbone pre-trained on ImageNet. We use † to denote numbers that are reported from [10] and [43], while the rest are reproduced. See Appendix B for breakdown numbers and model variance. We categorize methods on the left into algorithms and augmentations, and use (1) to denote methods that need to be applied in many intermediate layers of the network; and use (2) to denote methods that require explicit domain information to guide learning. We use cyan to denote the improvement over the robust ERM baseline [16], and use bold and underline to highlight the highest number and the second highest number in both sections. Note that training MIRO requires guidance from pre-trained models.

aims to optimize the below combined loss:

$$\mathcal{L} = \underbrace{\ell(g(z), y)}_{\text{original}} + \underbrace{\ell(g(D_\mu(z), \tilde{y})}_{\text{degraded}} + \underbrace{\ell(g(R_\theta(D_\mu(z)), y)}_{\text{restored}} \tag{10}$$

where ℓ is a classification loss function (e.g. CrossEntropy).

Crucially, our method relies on the classifier to both guide the corrupted latents towards degradation and back to their correct labels after restoration. This provides a soft constraint that encourages the latent embeddings to be flexible enough to mix information across domains. Additionally, both operations use sample-to-sample relationships to create distribution-aware degraded and restored samples, which can further encourage the training set to be diversified. `LatentDR` fits perfectly into mini-batch training, as shown in pseudocode in Appendix Alg. 1.

3.3. Assessing representation quality with alignment and uniformity metrics

To assess the quality of representations learned through our model, we use two metrics shown to have good correlation with accuracy of classification from pre-trained models [58]. The first metric is the *alignment score*, a measure of the 'closeness' of features from the same class. The second metric is the *uniformity score*, which measures the distribution of the (normalized) features on the hypersphere and captures how well the representations span the space.

The alignment score is defined as:

$$\mathcal{L}_{\text{align}}(f) \triangleq \mathbb{E}_{(x_1,x_2)\sim p_{\text{cls}}} \left[\|f(x_1) - f(x_2)\|_2^2\right], \tag{11}$$

where $(x_1, x_2) \sim p_{\text{cls}}$ draws samples from the same class.

The uniformity score is defined as:

$$\mathcal{L}_{\text{uniform}}(f) \triangleq \log \mathbb{E}_{x_1,x_2 \overset{\text{i.i.d.}}{\sim} p_{\text{data}}} \left[e^{-2\|f(x_1)-f(x_2)\|_2^2}\right], \tag{12}$$

where in this case the samples (x_1, x_2) are across the entire dataset. Uniformity demonstrates the generalization ability of the method, as it measures how much information the model preserves from the training data.

4. Experimental Results

To examine the performance of `LatentDR` on diverse datasets, we study our approach on: (i) domain generalization benchmarks, and (ii) medical imaging datasets with strong domain shifting. To further demonstrate its potential in different applications, we also test our method on (iii) long-tail recognition with imbalanced classes.

4.1. Domain generalization

4.1.1 Experiment setup

Dataset. Following [16], we comprehensively evaluate our method on the standardized DomainBed benchmarks. Specifically, five datasets are used in our experiments: (1) PACS [31] (9,991 images, 7 classes, and 4 domains), (2) VLCS [14] (10,729 images, 5 classes, and 4 domains), (3) Office-Home [55] (15,588 images, 65 classes, and 4 domains), (4) TerraIncognita [6] (24,788 images, 10 classes, and 4 domains), (5) DomainNet [42] (586,575 images, 345 classes, and 6 domains).

Evaluation and experimental details. For a fair comparison, we follow the leave-one-domain-out model training and evaluation protocol as in prior works [10, 11]. Unless specified otherwise, the same dataset splits (80/20%), hyperparameter sets, and model optimizer in [10] are used, and a ResNet-50 pre-trained on ImageNet is used for weight initialization. For each training step, we construct a minibatch containing 32 images from each training domain. For `LatentDR`, we perform a fixed learning rate adjustment due to our high loss value. We performed hyperparameter search on the validation sets of PACS and used the fixed hyperparameter sets on other experiments. All experiments are repeated five times with different random seeds except for DomainNet which has six subdomains. More experimental details can be found in Appendix A.

4.1.2 Results on DomainBed

In Table 1, we evaluate our approach against a diverse set of algorithms that include general-purpose data augmentation techniques such as Mixup [64] and Manifold-Mixup [56], domain-specific data augmentation techniques like MixStyle [17], as well as other DG algorithms that leverage domain information to build domain-invariant representations [43, 51]. On the left, we provide a breakdown of model assumptions used in the different models tested to highlight the flexibility of our approach and limited use of additional assumptions.

Our approach outperforms the robust ERM baseline [16] and shows significantly more stable and robust performance compared to other augmentation methods like BatchFormer [18] and MixStyle [69], resulting in a 2.7% improvement over ERM when averaged over all 5 datasets in DomainBed and a ≈2.0% improvement over other augmentation methods. Our approach shows particularly robust performance on DomainNet, the largest DG dataset with the most classes, highlighting the scalability of our approach. Overall, we achieve performance on par with the state-of-the-art method MIRO that regularizes features at many intermediate layers of the encoder, and requires an unaltered pre-trained encoder to provide guidance throughout training. The breakdown of accuracies for each dataset reported across domains and model variance across random seeds are in Appendix B.

4.1.3 Ablations and visualizations

To further understand the robustness of the proposed method, we perform model ablation, latent representation evaluation, and nearest neighbor visualization with our model on the PACS dataset.

Ablations. Our approach consists of two main components, a step to degrade samples and a step to restore them. Thus, we wanted to understand how both steps contribute towards the overall performance. In Table 2, we compare the performance of our method when we use both D+R vs. when we only apply degradation (D-only) or restoration (R-only). Across all tested domains in PACS, we see that the combination of both D+R is consistently the best. However, when we remove restoration and use D-only, we also obtain good performance, suggesting that sample-aware degradation can also be a good augmentation on its own. [1]

		A	C	P	S	Avg.
Reference	ERM	84.9	80.6	95.9	75.0	84.1
Ours	D-only	85.7	78.6	96.9	78.2	<u>84.8</u>
	R-only	82.6	79.3	95.4	73.5	82.7
	D+R	86.3	82.6	97.1	79.2	**86.3**
Gaussian	D-only	85.0	78.1	96.5	75.6	83.8
	R-only	86.4	80.0	97.4	76.5	<u>85.1</u>
	D+R	87.4	79.7	97.1	77.1	**85.3**

Table 2. *Ablations of degradation and restoration on PACS*. We compare the performance of `LatentDR` using the degradation loss (D-only) or restoration loss (R-only) alone vs. combining them (D+R), for our sample-aware degradation approach (top) vs. additive Gaussian noise (below).

To test whether a simpler degradation mechanism could be used, we tested a variant of our model where we remove the sample-aware mixing operation used in our degradation step and replace it with i.i.d. Gaussian noise $\mathcal{N}(0, 1)$. When no restoration is used (D-only), we find that the Gaussian noise augmentation does not help; However, when we couple Gaussian noise with our restoration procedure (R-only, D+R), we find that we can obtain good performance. Overall, we observe that combining both degradation and restoration achieves the best overall performance, and our sample-aware approach provides a flexible and robust way to augment latents.

Latent quality evaluation. To understand how our proposed approach shapes the representations of the data, we examined the uniformity and alignment metrics for different models as shown in Figure 4 (see Appendix D for direct visualizations). We confirmed that our method gives both the highest uniformity score, which demonstrates that it preserves the maximum amount of information inside the training data and the lowest alignment score, which demonstrates that it encourages the closeness of latents within the same class. Interestingly, we find that Mixup [64] does improve alignment but has very little improvement in uniformity when compared with ERM. On the other hand, BatchFormer [18] produces good diversity of latents by considering nonconvex sample-to-sample relationships but poorer alignment. Our method, instead, combines the advantages at both ends by encouraging the diversity of the generated samples while maintaining good alignment by separating

[1] If there is no degradation, the cross-attention operation becomes a self-attention operation, and thus our method converges back to [18].

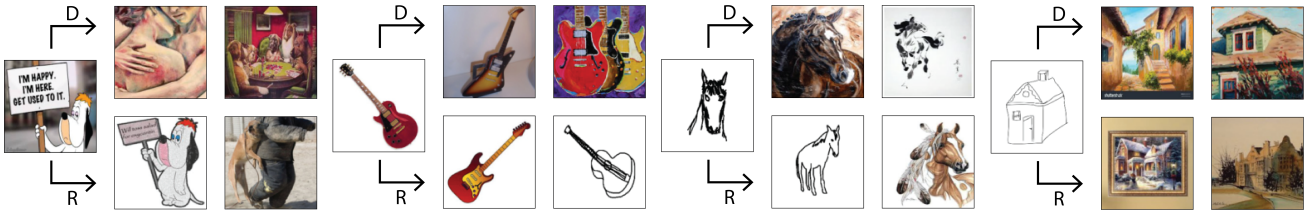

Figure 3. *Nearest neighbors visualizations.* We show the top two nearest neighbors for a query sample after degradation (top row) and restoration (bottom row). In the leftmost example, we find a case where degradation places a sample near points from a different class but brings it back to diverse domains from its original class. On the right, we show examples where degradation maps the latent next to samples of the same class that are different in color or background, and restores to samples that are similar but might be from another domain.

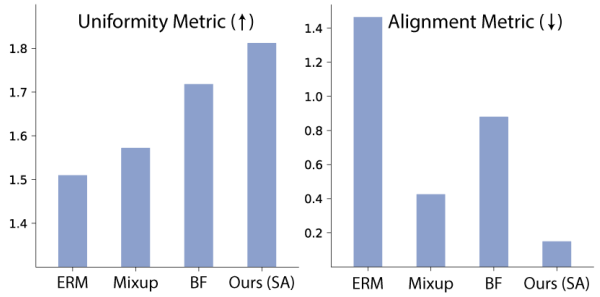

Figure 4. *Latent quality evaluation.* We measure the uniformity and alignment of the latent spaces for ERM, Mixup [64], BatchFormer [18] (BF), and `LatentDR` (Ours-SA).

the different classes.

Nearest neighbor visualization. To observe how the degradation and restoration steps remap different query points, we examined the nearest neighbors of the augmented latents for both operations (Figure 3). We observed that indeed the latents produced from the latent restoration process would retrieve nearest neighbors to samples from the same class but different domains (both the testing domain and another training domain). Interestingly, in the degradation step, although the produced samples would often confuse the classifier, the nearest neighbors are often from the same class but with dissimilar colors or backgrounds. We conjecture that it is possible that this classifier guidance may encourage certain content/style separation in the latent space.

4.2. Experiments on medical imaging datasets

To further demonstrate the use of our method across different tasks and images, we applied the method to multiple medical imaging datasets where domain shift occurs and generalization is difficult due to small sample sizes.

4.2.1 Experiment setup

Datasets. We evaluate our method on five medical imaging datasets: Derma [53] (10,015 samples, 7 classes, 2D), OrganS [8] (25,221 samples, 11 classes, 2D), OCT [24] (109,309 samples, 4 classes, 2D), Fracture [23] (1,370 samples, 3 classes, 3D), as well as Camelyon17 [4] (302,436 samples, 2 classes, 2D). For the first four datasets, we follow the preprocessing and data splitting protocol as in [61]; while for Camelyon17 we followed [26]. More details about datasets and their domain shifts are in Appendix A.

Evaluation and experimental details. For the first four datasets, we follow [61] and use a ResNet-18 for 2D image classification and a ResNet-18-based 3D backbone for 3D image classification. We replace the Adam optimizer with an SGD optimizer with lr=0.001 and momentum=0.9 for better performance. For Camelyon17, we follow the model and training setup in [48] and use a DenseNet-121 backbone. More information about the training and model optimization process is in Appendix A. For `LatentDR`, we performed the same learning rate adjustment, and applied the same set of hyperparameters as defined in previous experiments, demonstrating the stability our method across various choices of hyperparameters. All experiments are repeated three times with different random seeds.

4.2.2 Results: Medical imaging datasets

In Table 3 and 4, we compare our model with other baseline and competitor models. For the first four datasets where we do not have explicitly defined different domains, we benchmark our method with other augmentation meth-

Model	Derma	OrganS	OCT	Fracture	Avg.
ERM†	75.4	77.8	76.3	50.8	70.1
ERM	75.8	79.8	76.4	49.6	70.4
+ Mixup	70.0	80.8	78.9	48.3	69.5
+ M-Mixup	76.4	80.1	76.3	49.6	70.6
+ BF	76.8	82.7	77.8	51.7	72.3
+ Ours (SA)	78.1	**83.3**	79.1	52.1	73.2
+ Ours (Pool)	**78.4**	82.9	**79.4**	**55.0**	**73.9**

Table 3. *Medical image classification without explicit domains.* We follow the setup in [61], where † are the classification accuracies reported by them with the same ResNet-18 backbone model.

Model	Without Aug	Targeted Aug	Avg.
CORAL†	59.5	–	
Fish [48]	74.7†	76.9	75.8
ERM†	70.8	82.0	76.4
ERM	72.3	84.3	78.3
+ Mixup†	63.5	–	
+ Ours (SA)	76.4	**91.0**	83.7
+ Ours (Pool)	**81.3**	89.9	**85.6**

Table 4. *Medical image classification on Camelyon17*. We follow the setup in [48], where † are the classification accuracies reported in [26, 48] with the same DenseNet-121 backbone model.

ods [18, 56, 64]. We find that augmentation on image space (Mixup) does not always improve performance, as blending the color or the shape of medical objects might undermine the classification of specific features. In terms of latent augmentation methods, we observed that BatchFormer [18], which considers sample-to-sample relationships, also has good performance. However, our method outperforms all other methods, demonstrating the effectiveness of considering both sample-to-sample relationships and classifier-guided latent degradation.

When testing on Camelyon17, we compare models under two different settings: (i) standard evaluation setting without custom augmentations [26], and (ii) with targeted augmentations that use underlying knowledge about the type of domain shift in the dataset. In the first setting, our model outperforms many reported algorithms by a large margin and improves over the ERM baseline by nearly 10% (from 72.3% to 81.3%), where the performance of our method almost reaches the performance of ERM with targeted augmentation (84.3%). When testing on targeted augmentations, we find an impressive boost of almost 7% over the ERM baseline that is competitive with state-of-the-art for this task. In contrast, Fish [48] remains at around 75% in both the original case and in targeted augmentations. This application shows the great potential of `LatentDR` when applied to medical imaging datasets where data augmentation requires domain knowledge to define.

4.3. Application to long-tail recognition

Augmentation methods are shown to be effective in improving model generalization when datasets exhibit class imbalance [34, 38]. To examine if `LatentDR` can also enhance generalization in the presence of strong data imbalance, we applied our method to a long-tail recognition task.

Experiment setup. We perform our experiments on CIFAR-100-LT [27] with an imbalance ratio of 100. `LatentDR` is applied on top of BalancedSoftmax (BALMS [44]) and Balanced contrastive learning (BCL [70]), which both have explicit CrossEntropy term in their loss functions. We followed their hyperparameter and training settings, and for our method we performed the same learning rate adjustment as in domain generalization experiments. More information is in Appendix A.

	Many	Medium	Few	Avg.
BCL [70]	69.1	52.4	**30.5**	51.7
+ BF	**69.5**	51.7	27.9	50.8
+ Ours	68.9	**53.9**	**30.5**	**52.1**
BALMS [44]	67.9	50.7	32.2	51.1
+ BF†	68.4	49.3	**34.3**	51.7
+ Ours	**69.8**	**51.3**	34.2	**52.7**

Table 5. *Long-tail recognition results on CIFAR-100-LT*. We reproduced all numbers except for BALMS+BF, where we used their reported numbers [18] as it is higher than what we reproduced.

Results. As shown in Table 5, our method surpasses both the original method and [18], demonstrating its potential for being a versatile plugin module. Interestingly, different from [18] that argues sample-to-sample relationship mostly improves performance on tail classes, our method gains the largest performance improvements on Medium classes, which might be attributed to our soft-label construction process. Additionally, we observed that our method performs better when combined with a simpler method (BALMS).

5. Conclusion

In this paper, we proposed a novel latent augmentation method `LatentDR` to improve domain generalization. The idea behind our approach is to learn to *degrade* and *restore* latents at the batch level, using information across samples to build augmentations for a latent. We use a classifier to guide both steps: first to obtain degraded latents that are identified to a constructed soft-label and second, to get the latents restored to the original class. The proposed method can be easily integrated into existing deep learning methods and used with different encoders without any modifications, achieving significant improvements.

This work opens up a number of interesting future directions including the integration of our method with domain adaptation approaches, as well as exploring its use in semi-supervised learning and self-supervised learning. It would be interesting to study the connections between our method and generative modeling frameworks that use cycle consistency and cold diffusion [5].

Acknowledgement

This project is supported by NIH award 1R01EB029852-01, NSF award IIS-2039741, NSF award IIS-2212182, NSF CAREER award IIS-2146072, the NSF Graduate Research Fellowship Program (GRFP) for MD, as well as generous gifts from the Alfred Sloan Foundation, the McKnight Foundation, the CIFAR Azrieli Global Scholars Program.

References

[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. 2

[2] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. *Advances in Neural Information Processing Systems*, 31, 2018. 1, 2

[3] Randall Balestriero, Leon Bottou, and Yann LeCun. The effects of regularization and data augmentation are class dependent. *arXiv preprint arXiv:2204.03632*, 2022. 2

[4] Peter Bandi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. *IEEE Transactions on Medical Imaging*, 38(2):550–560, 2018. 7

[5] Arpit Bansal, Eitan Borgnia, Hong-Min Chu, Jie S Li, Hamid Kazemi, Furong Huang, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Cold diffusion: Inverting arbitrary image transforms without noise. *arXiv preprint arXiv:2208.09392*, 2022. 8

[6] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 456–473, 2018. 5

[7] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2010. 1

[8] Patrick Bilic, Patrick Christ, Hongwei Bran Li, Eugene Vorontsov, Avi Ben-Cohen, Georgios Kaissis, Adi Szeskin, Colin Jacobs, Gabriel Efrain Humpire Mamani, Gabriel Chartrand, et al. The liver tumor segmentation benchmark (lits). *Medical Image Analysis*, 84:102680, 2023. 7

[9] Manh-Ha Bui, Toan Tran, Anh Tran, and Dinh Phung. Exploiting domain-specific features to enhance domain generalization. *Advances in Neural Information Processing Systems*, 34:21189–21201, 2021. 2

[10] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. *Advances in Neural Information Processing Systems*, 34:22405–22418, 2021. 1, 2, 5, 6

[11] Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIII*, pages 440–457. Springer, 2022. 2, 5, 6

[12] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 2

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[14] Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1657–1664, 2013. 5

[15] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 1, 2

[16] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020. 1, 2, 5, 6

[17] Minui Hong, Jinwoo Choi, and Gunhee Kim. Stylemix: Separating content and style for enhanced data augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14862–14870, 2021. 1, 2, 6

[18] Zhi Hou, Baosheng Yu, and Dacheng Tao. Batchformer: Learning to explore sample relationships for robust representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7256–7266, 2022. 2, 4, 5, 6, 7, 8, 1, 3

[19] Zhi Hou, Baosheng Yu, Chaoyue Wang, Yibing Zhan, and Dacheng Tao. Batchformerv2: Exploring sample relationships for dense representation learning. *arXiv preprint arXiv:2204.01254*, 2022. 2, 4

[20] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017. 2

[21] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 124–140. Springer, 2020. 2

[22] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021. 2, 3

[23] Liang Jin, Jiancheng Yang, Kaiming Kuang, Bingbing Ni, Yiyi Gao, Yingli Sun, Pan Gao, Weiling Ma, Mingyu Tan, Hui Kang, et al. Deep-learning-assisted detection and segmentation of rib fractures from ct scans: Development and validation of fracnet. *EBioMedicine*, 62:103106, 2020. 7

[24] Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *cell*, 172(5):1122–1131, 2018. 7

[25] Jang-Hyun Kim, Wonho Choo, and Hyun Oh Song. Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In *International Conference on Machine Learning*, pages 5275–5285. PMLR, 2020. 2

[26] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021. 2, 7, 8

[27] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 8

[28] 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. 2

[29] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pages 3744–3753. PMLR, 2019. 3

[30] 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, 2018. 5

[31] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5542–5550, 2017. 5

[32] 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, 2018. 3, 5

[33] Pan Li, Da Li, Wei Li, Shaogang Gong, Yanwei Fu, and Timothy M Hospedales. A simple feature augmentation for domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8886–8895, 2021. 1, 2

[34] Shuang Li, Kaixiong Gong, Chi Harold Liu, Yulin Wang, Feng Qiao, and Xinjing Cheng. Metasaug: Meta semantic augmentation for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5212–5221, 2021. 8

[35] Ya Li, Mingming Gong, Xinmei Tian, Tongliang Liu, and Dacheng Tao. Domain generalization via conditional invariant representations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018. 5

[36] Ziyue Li, Kan Ren, Xinyang Jiang, Bo Li, Haipeng Zhang, and Dongsheng Li. Domain generalization using pretrained models without fine-tuning. *arXiv preprint arXiv:2203.04600*, 2022. 2

[37] Chi-Heng Lin, Chiraag Kaushik, Eva L Dyer, and Vidya Muthukumar. The good, the bad and the ugly sides of data augmentation: An implicit spectral regularization perspective. *arXiv preprint arXiv:2210.05021*, 2022. 2

[38] Jialun Liu, Yifan Sun, Chuchu Han, Zhaopeng Dou, and Wenhui Li. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2970–2979, 2020. 8

[39] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. 3

[40] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8690–8699, 2021. 2, 5

[41] Hongjing Niu, Hanting Li, Feng Zhao, and Bin Li. Domain-unified prompt representations for source-free domain generalization. *arXiv preprint arXiv:2209.14926*, 2022. 2

[42] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1406–1415, 2019. 5

[43] Alexandre Rame, Corentin Dancette, and Matthieu Cord. Fishr: Invariant gradient variances for out-of-distribution generalization. In *International Conference on Machine Learning*, pages 18347–18377. PMLR, 2022. 2, 5, 6

[44] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in Neural Information Processing Systems*, 33:4175–4186, 2020. 8

[45] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 3

[46] Swami Sankaranarayanan and Yogesh Balaji. Meta learning for domain generalization. In *Meta-Learning with Medical Imaging and Health Informatics Applications*, pages 75–86. Elsevier, 2023. 1, 2

[47] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. *arXiv preprint arXiv:1804.10745*, 2018. 1, 2

[48] 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. 2, 7, 8

[49] Nathan Somavarapu, Chih-Yao Ma, and Zsolt Kira. Frustratingly simple domain generalization via image stylization. *arXiv preprint arXiv:2006.11207*, 2020. 1, 2

[50] Thomas Stegmüller, Behzad Bozorgtabar, Antoine Spahr, and Jean-Philippe Thiran. Scorenet: Learning non-uniform attention and augmentation for transformer-based histopathological image classification. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6170–6179, 2023. 3

[51] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*, pages 443–450. Springer, 2016. 1, 2, 5, 6

[52] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34:24261–24272, 2021. 3

[53] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data*, 5(1):1–9, 2018. 7

[54] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017. 2, 3

[55] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017. 5

[56] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447. PMLR, 2019. 1, 2, 5, 6, 8

[57] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *Advances in Neural Information Processing Systems*, 31, 2018. 1, 2

[58] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020. 2, 5

[59] Xixi Wang, Bo Jiang, Xiao Wang, and Bin Luo. Rethinking batch sample relationships for data representation: A batch-graph transformer based approach. *arXiv preprint arXiv:2211.10622*, 2022. 2

[60] Yue Wang, Lei Qi, Yinghuan Shi, and Yang Gao. Feature-based style randomization for domain generalization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(8):5495–5509, 2022. 2

[61] Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023. 2, 7

[62] Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10819–10829, 2022. 3

[63] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019. 2, 5

[64] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 2, 5, 6, 7, 8, 4

[65] Shanshan Zhao, Mingming Gong, Tongliang Liu, Huan Fu, and Dacheng Tao. Domain generalization via entropy regularization. *Advances in Neural Information Processing Systems*, 33:16096–16107, 2020. 2

[66] Zangwei Zheng, Xiangyu Yue, Kai Wang, and Yang You. Prompt vision transformer for domain generalization. *arXiv preprint arXiv:2208.08914*, 2022. 2

[67] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13025–13032, 2020. 1, 2

[68] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *European Conference on Computer Vision*, pages 561–578. Springer, 2020. 2

[69] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. *arXiv preprint arXiv:2104.02008*, 2021. 2, 5, 6

[70] Jianggang Zhu, Zheng Wang, Jingjing Chen, Yi-Ping Phoebe Chen, and Yu-Gang Jiang. Balanced contrastive learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6908–6917, 2022. 8