What's in a Latent? Leveraging Diffusion Latent Space for Domain Generalization

Xavier Thomas[1] Deepti Ghadiyaram[12*]
[1]Boston University [2]Runway
{xthomas, dghadiya}@bu.edu

Abstract

*We investigate how model architectures and pre-training objectives influence feature richness, and introduce a simple method to leverage these features for domain generalization. Given a pre-trained feature space, we first discover latent domain structures, referred to as pseudo-domains, that capture domain-specific variations in an unsupervised manner. We then augment classifiers with these complementary representations, improving generalization to diverse unseen domains. We also analyze how different pre-training feature spaces differ in terms of the granularity of domain-specific variances they capture. Our analysis reveals that diffusion models, in particular, effectively separate domains in their latent spaces. Across five datasets, our approach improves test accuracy by up to **4%** over the baseline Empirical Risk Minimization (ERM). Code is available at: xthomasbu.github.io/GUIDE.*

1. Introduction

It is now a common practice to use models pre-trained on billion-scale data [17, 19, 39, 44, 46, 49, 51] as defacto backbones for diverse downstream tasks [36, 57]. A variety of powerful pre-training strategies have been designed to make these models "foundational," offering rich feature representations. Some aim to eliminate the need for clean labeled data [7, 8, 11, 18, 66], some align visual and textual signals [25, 49], while others learn by predicting large hidden regions of images [19]. Despite such progress, what exactly is captured in the underlying latent space remains an open question, particularly in diffusion models due to their iterative denoising objective.

This work aims to understand the feature landscape learned from different pre-training models and objectives in the context of domain generalization. Robust generalization to unseen domains has been a long-standing goal [5, 41], particularly when collecting domain-specific information is infeasible. In such cases, models must learn to generalize without relying on explicit domain labels during training [33].

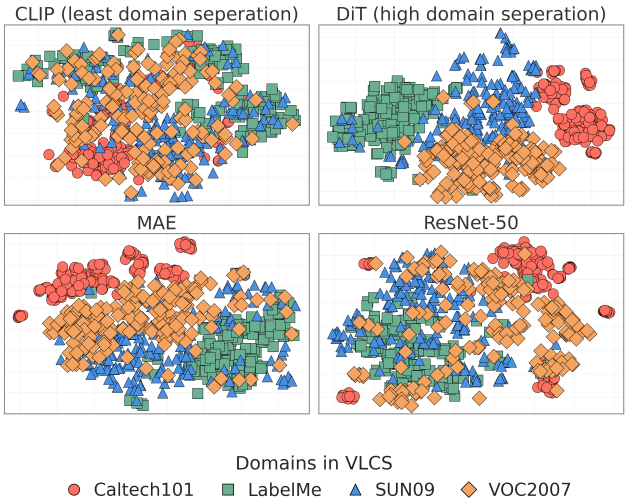

Figure 1. **T-SNE visualization of the latent space from different pre-training objectives:** CLIP [49], DiT [46], MAE [19], ResNet-50 [17] on the domain generalization benchmark VLCS [14]. VLCS is curated from 4 different datasets, thus dataset-specific biases like spatial composition and object size variations serve as different domains. Note how the diffusion features separate domains effectively, suggesting that latent domain structures can be captured without explicit supervision. Best viewed in color.

We posit that the first step to make fundamental progress towards designing foundational models is to examine and interpret how current state-of-the-art models structure visual information and uncover their strengths and limitations. For instance, how are object, scene, and domain-specific variations internally encoded in a latent space? Do domain-specific traits manifest in distinct regions of the latent space or are they engulfed along with low- to mid-level scene and object level information?

We study these questions in detail. Specific to the task of domain generalization, we analyze how different pre-training objectives and architectures influence the granularity of visual information captured in their feature space. Our key insight is that certain internal states of diffusion mod-

*Corresponding author.

els effectively capture abstract information such as photographic styles and camera angles. Building on this insight, we develop an unsupervised method for discovering latent domain structures. Next, we alter a standard domain generalization classification [58] pipeline with one key difference: we augment the classifier's representations with the discovered latent domain representations. We show through extensive empirical analysis that this simple tweak to the standard pipeline assists in training a model that generalizes well to unseen domains [4].

Our framework, **GUIDE** (**G**eneralization using **I**nferred **D**omains from Latent **E**mbeddings), offers a simple and effective method to "guide" a given feature space to adapt better to unseen domains.

Contributions:

- We propose a method of **unsupervised pseudo-domain discovery** from frozen pre-trained feature spaces and use them to improve a model's ability to generalize to diverse domains (Sec. 3).
- We **analyze different pre-training objectives and architectures** and investigate how they influence the structure of the feature latent landscape (Sec. 4.2).
- We shine light on **the ability of diffusion models to capture domain-specific information** (Fig. 1), and demonstrate their effectiveness to domain generalization (Sec. 4.3).

2. Related Work

Diffusion features for representation learning: Diffusion models [22, 52] have advanced image and video generation, with their intermediate features proving useful across tasks such as classification [29], segmentation [2, 63], depth estimation [62, 68], and visual reasoning [61]. Recent studies [27, 38, 60] demonstrate that features extracted across layers and timesteps encode rich semantic information, ranging from coarse patterns to fine-grained details. We study how these features encode class- and domain-specific signals and leverage them for domain generalization.

Domain generalization: Domain generalization aims to build models that perform well on unseen domains [5]. Various methods have been proposed to address this by learning domain-agnostic representations [24, 42], data or latent augmentation methods [23, 34, 37, 53], and meta-learning [1, 6]. Dubey et al. [13], Thomas et al. [56] explore techniques to incorporate pseudo-domain information into classifiers to make them generalizable to unseen domains. Our work differs from these prior arts in several crucial ways: we leverage pre-trained models instead of learning a separate domain prototype network as in [13], utilize a more domain-rich feature space compared to [56], and do not rely on domain labels as in [6, 13].

Diffusion models for domain generalization. Prior approaches [20, 67] use diffusion models for synthetic data augmentation, often requiring test-time access or model fine-tuning. In contrast, we propose a method to use frozen diffusion features in an unsupervised manner for domain generalization.

3. Approach

First, we introduce the preliminaries of diffusion models and the setting of domain generalization. Then, we present our two-step framework where we first learn pseudo-domain representations in an unsupervised manner and use them to adapt a classifier to unseen domains. We stress that we do not have domain label information during both training and test phases.

Diffusion models [22, 52] are generative models that learn data distributions by denoising progressively noised images $\{x_t\}_{t=1}^{T}$, where noise ϵ is added over T timesteps. A model θ is trained to predict the added noise $\epsilon_\theta(x_t, t)$ at each step.

Latent Diffusion Models (LDMs) [51] extend this by operating in a lower-dimensional latent space $z = E(x)$, using a Variational Autoencoder (VAE) [28] with an encoder E and decoder D. The model learns to denoise latent variables z_t via:

$$L_{\text{LDM}} = \mathbb{E}_{E(x),t,\epsilon\sim\mathcal{N}(0,1)}\|\epsilon - \epsilon_\theta(z_t, t)\|_2^2$$

Domain Generalization. Let X and Y denote input data and labels, and $\mathbf{\Phi}$ a feature extractor. In supervised learning, a predictor f maps features $\mathbf{\Phi}(x)$ to labels y, i.e., $f(\mathbf{\Phi}(x)) \rightarrow y$. Domain generalization extends this by assuming access to data from multiple training domains $\{P_d^{\text{tr}}\}_{d=1}^{d_{\text{tr}}}$, with the goal of generalizing to an unseen test domain P_d^{te} [5]. A common baseline is Empirical Risk Minimization (ERM) [58], which trains a domain-agnostic predictor by pooling all training data. However, this ignores inter-domain variations and may fail when the test domain differs significantly [13].

Learning and leveraging pseudo-domain representations. Inspired by prior work on leveraging domain-specific representations [6, 13, 40, 56], we augment features with complementary *pseudo-domain* information. Without access to domain labels, we uncover latent domain structure by clustering pre-trained features $\mathbf{\Psi}$ via K-Means++, treating the K resulting centroids as pseudo-domains. Each sample is assigned to its nearest centroid $\widehat{\mathbf{\Psi}}_k$. To improve generalization, we concatenate each input feature $\mathbf{\Phi}(x)$ with a transformed pseudo-domain vector $\mathcal{T}(\widehat{\mathbf{\Psi}}_k)$, where $\mathcal{T}$ is a Radial Basis Function (RBF) kernel ridge regressor that maps $\mathbf{\Psi}$ to $\mathbf{\Phi}$ to reduce feature drift. At test time, we extract $\mathbf{\Psi}(x)$, assign it to the nearest cluster, apply $\mathcal{T}$, and concatenate the result with $\mathbf{\Phi}(x)$ before passing it to the classifier. Our method (**GUIDE**) requires no domain supervision and makes no assumptions about the test domains.

Dataset	DiT	SD-2.1	MAE	CLIP	DINOv2	RN50
PACS	**0.85**	0.82	0.71	0.54	0.55	0.59
VLCS	**0.58**	0.26	0.20	0.01	0.05	0.22
TerraInc	0.22	**0.55**	0.21	0.01	0.01	0.25
OfficeHome	0.25	0.28	0.10	0.12	**0.38**	0.08
DomainNet	**0.54**	0.51	0.52	0.32	0.47	0.46

Table 1. **Comparison of domain NMI scores across datasets**. The highest domain NMI score depends both on the type of pre-training feature space and the underlying domain shifts in the dataset as noted in Sec 4.2. We note that inherent domain label noise can impact domain NMI scores. Thus, NMI is more valuable when used as a relative measure rather than an absolute indicator of domain separability.

4. Experiments

We outline the implementation details and training setup for GUIDE in Sec 4.1, followed by a detailed analysis of the capability of different feature extractors ($\mathbf{\Psi}$) in capturing domain-specific information to augment class-specific features ($\mathbf{\Phi}$) in Sec 4.2. We empirically show how our approach leads to a more domain generalizable classifier on unseen test domains in Sec. 4.3.

4.1. Implementation Details

We evaluate on five DomainBed [16] datasets: PACS [30], VLCS [14], TerraIncognita [3], OfficeHome [59], and DomainNet [47]. We provide details of the domain shifts in each dataset in Appendix B. We use the default DomainBed setup: batch size of 32 per domain, learning rate of 5×10^{-5}, 5001 steps, and no dropout or weight decay. Results are averaged over three seeds using leave-one-domain-out cross-validation. For the classifier backbone $\mathbf{\Phi}$, we use ResNet-50 pretrained with AugMix [21]. For latent features $\mathbf{\Psi}$, we compare ResNet-50 [17], CLIP [49], DINOv2 [44], MAE [19], Stable Diffusion 2.1 [51], and DiT [46]. The transformation $\mathcal{T}$ is an RBF kernel ridge regressor mapping each pseudo-domain centroid $\widehat{\mathbf{\Psi}}_k$ to the average $\mathbf{\Phi}$ feature vector of the samples assigned to that cluster. We set the number of clusters using $K = \max(\{1,3,5\} \times n_c, 200)$, where n_c is the number of classes.

To evaluate the *expressivity* [13] of pseudo-domains, we compute normalized mutual information (NMI) [40, 56] between discovered clusters and ground-truth domain or class labels. Given cluster assignments U and ground truth class / domain labels V, NMI is defined as $\text{NMI}(U,V) = \frac{2\cdot I(U,V)}{H(U)+H(V)}$, where I is mutual information and H is entropy. Higher domain NMI values reflect stronger domain separation in the latent space.

4.2. Effect of the Choice of Ψ on Domain Separation

Next, we study how different pre-training objectives affect the separation of domain-specific signals using **domain NMI** (↑) (introduced in Sec. 4.1), which measures how well domains are separated in the latent space. We acknowledge that all models are of varied architectural complexities, trained on very different datasets, thereby making it nonviable to concretely isolate the cause of performance discrepancies in domain separation. Nevertheless, we believe our below analysis is valuable to understand the semantic information captured by different pre-training objectives. The domain shifts in each dataset studied are presented in Appendix B.

ResNet-50 [17] (RN50) is pre-trained on ImageNet [12] using a cross-entropy loss, encouraging object-level discrimination. Consequently, the feature space evolves to aid object discrimination, making samples from the same class cluster together across domains. This results in low domain NMI but high class NMI across datasets (Table 1, and Table 4 in the Appendix); e.g., on PACS, class NMI is 0.29 vs. 0.08 for DiT. This makes RN50 well-suited for $\mathbf{\Phi}$ but less effective for modeling domain-sensitive features as $\mathbf{\Psi}$.

CLIP [49] is pre-trained on noisy image-text pairs using a contrastive loss that aligns images with textual descriptions in a joint embedding space. This prioritizes high-level semantic similarity, making CLIP's feature space reflect global context rather than object-specific details. Images of the same object may not form tight clusters if captions differ in contextual emphasis (e.g., "a dog on a beach" vs. "a golden retriever indoors"). Thus, CLIP, though rich in broader contextual semantics, yields low class and domain NMI scores across all datasets in (Tables 1 and 4).

DINOv2 [44] is a self-supervised vision transformer trained by aligning representations between a student and teacher network across global and local crops. This encourages the model to capture primarily low-level features, while also capturing global relationships to some extent [26, 44, 57]. These features are particularly effective for datasets like OfficeHome (domain NMI of 0.38 in Table 1), where domain shifts arise from low-level style differences such as bold outlines in "clipart" vs. soft, natural edges in "real" domain.

Masked Autoencoders [19] (MAEs) are pre-trained to reconstruct locally masked patches, which may induce a strong locality bias and hinder capturing global context, as studied in [35, 69]. We hypothesize that this limits their ability to offer complementary domain-specific representations (as seen in Table 1). However, MAEs achieve relatively high domain NMI on PACS (0.71) and DomainNet (0.52) by leveraging local details such as textures, shading, and brushstrokes. A similar trend is observed with DINOv2, suggesting both models perform better when domain shifts are driven by low-level visual variations. MAEs perform poorly on TerraIncognita, where domain separation likely requires both local and global spatial understanding (e.g., vegetation density, terrain patterns).

Diffusion models for domain separation

We now focus on diffusion models and examine how their architectural design impacts domain separation. As dis-

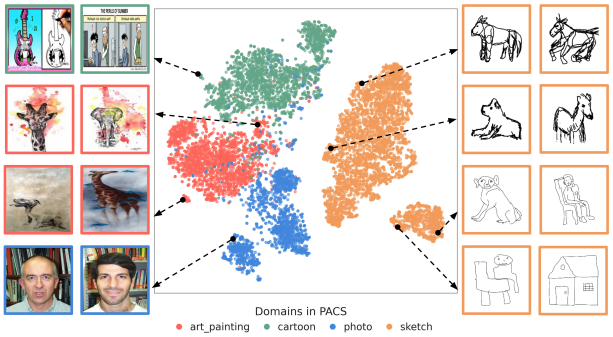

Figure 2. **t-SNE visualization of how pseudo-domains are clustered together in the latent space of DiT for PACS.** Note how the sketch domain forms distinct clusters, with light and dark pencil strokes mapped to separate regions in the latent space. Best viewed in color.

cussed in Sec. 3, the iterative denoising objective encourages models to first capture broad structures before refining details [45, 48]. We hypothesize that this implicit hierarchical learning, along with the absence of an explicit class-discriminative loss, allows domain-specific variations to emerge more prominently. This is reflected in Table 1, where diffusion features consistently yield higher domain NMI scores than non-diffusion counterparts. Figure 2 illustrates how pseudo-domains in diffusion latent space encode domain-specific structure. We compare two diffusion backbones: the transformer-based DiT [46] and the convolutional U-Net of SD-2.1 [51]. Both are trained on different datasets and exhibit complementary strengths. Following Kim et al. [27], we extract features at timestep t=50: for DiT, from the 14th transformer block; for SD-2.1, from the second upsampling layer (`up_ft:1`).

DiT's self-attention captures global context effectively, excelling on datasets with high-level semantic variation. It achieves the highest domain NMI on PACS (0.85) and on VLCS (0.58) (Table 1), where domains reflect dataset-specific biases. In contrast, SD-2.1 encodes fine-grained spatial detail, performing best on TerraIncognita (0.55), where domain shifts involve changes in foliage and terrain patterns.

Both models perform poorly on OfficeHome (DiT: 0.25, SD-2.1: 0.28) (Table 1), likely due to high inter-domain visual similarity (more details in Appendix). On DomainNet, DiT achieves the best score (0.54), though all models perform moderately, likely due to varied low- and high-level domain shifts (Appendix B).

4.3. Domain Generalization Performance

In this section, we compare GUIDE against prior domain generalization methods and examine the impact of different feature extractors (Ψ) in capturing domain-specific information to enhance classification performance.

Choice of Ψ on domain generalization: We evaluate different feature spaces for GUIDE on DomainBed [16]. As shown in Table 2, diffusion features (DiT, SD-2.1) consistently outperform non-diffusion counterparts across all datasets. GUIDE-DiT yields strong performance on VLCS (**+1.9%**) and PACS (**+3.3%**) over the baseline (ERM). GUIDE-SD-2.1 performs best on TerraIncognita (**+4.3%**). These results align with domain NMI trends from Table 1.

Dataset	DiT	SD-2.1	RN50	CLIP	DINOv2	MAE	ERM
VLCS	**78.5**	77.0	76.3	76.8	77.3	76.4	76.6
PACS	**87.1**	86.9	84.8	84.7	84.9	84.6	83.8
OH	68.4	**68.6**	65.7	64.6	68.3	65.2	67.2
TI	48.2	**51.3**	49.8	47.4	48.4	50.2	47.0
Avg	70.6	**71.0**	69.1	68.4	69.7	69.1	68.7

Table 2. **GUIDE performance on different Ψ spaces**. The pseudo-domain representations obtained from the latent space of diffusion models provide the highest gains in accuracy.

uses multi-layer features	uses domain labels	Algorithm	VLCS	PACS	OH	TI	DN	Avg
-	-	ERM [58]	76.6	83.8	67.2	47.0	44.1	63.7
✓	✓	MLDG [31]	77.2	84.9	66.8	47.7	41.2	63.6
✓	✓	MMD [32]	77.5	84.7	66.3	42.2	23.4	58.8
✓	✓	CORAL [54]	78.8	86.2	68.7	47.6	41.5	64.5
✓	✓	SagNet [43]	77.8	86.3	68.1	48.6	40.3	64.2
✗	✓	DANN [15]	78.6	83.6	65.9	46.7	38.3	62.6
✗	✓	Fishr [50]	77.8	85.5	67.8	47.4	41.7	64.0
✓	✗	MIRO [10]	**79.0**	85.4	**70.5**	50.4	44.3	65.9
✗	✗	Mixup [64, 65]	77.4	84.6	68.1	47.9	39.2	63.4
✗	✗	LatentDR (SA) [37]	78.7	85.8	69.0	49.9	45.1	65.7
✗	✗	LatentDR (Pool) [37]	78.0	86.3	68.4	49.5	43.9	65.2
✗	✓	DA-ERM ([13])	78.0	84.1	67.9	47.3	43.6	64.1
✗	✗	AdaClust ([56])	78.9	87.0	67.7	48.1	43.6	64.9
✗	✗	GUIDE-DiT (ours)	78.5	87.1	68.4	48.2	45.8	65.6
✗	✗	GUIDE-SD-2.1 (ours)	77.0	86.9	68.6	51.3	45.9	65.9
✗	✗	GUIDE-BEST (ours)	78.5	**87.1**	68.6	**51.3**	**45.9**	**66.3**

Table 3. **Comparison of GUIDE with prior domain generalization methods across 5 datasets** using the DomainBed test bed. Methods are grouped by (1) whether they use features from multiple intermediate layers, and (2) whether they require ground truth domain labels during training. The best performing method is underlined; the overall best is in **bold**. Cyan rows denote domain-adaptive classifiers (Sec. 3), among which GUIDE performs best. GUIDE-BEST reports the best performance among the two diffusion latent spaces (DiT and SD-2.1.

Comparison with prior art: In Table 3, we compare GUIDE with other state-of-the-art domain generalization algorithms[1] and note that GUIDE-BEST achieves the highest average performance of **66.3%** without using domain labels at any point. Compared to all methods, GUIDE-BEST shows the largest improvements on the PACS, TerraIncognita, and DomainNet datasets. Additional results incorporating enhanced training strategies from ERM++ [55] into GUIDE are presented in Appendix J.

5. Conclusion

We analyzed how pre-training objectives and architectures affect domain separation, showing that diffusion models naturally encode domain-specific variation. Building on this, GUIDE leverages latent domain structure in pre-trained feature spaces to improve generalization without requiring domain labels at train or test time.

[1] We compare against algorithms reported in [13, 37, 56].

References

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

[2] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021. 2

[3] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, 2018. 3, 8, 20

[4] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 2010. 2

[5] Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. *Advances in Neural Information Processing Systems*, 2011. 1, 2

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

[7] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, 2018. 1

[8] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 2020. 1

[9] 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*, 2021. 13

[10] Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. In *Proceedings of the European conference on computer vision (ECCV)*, 2022. 4

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 2020. 1

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2009. 3

[13] Abhimanyu Dubey, Vignesh Ramanathan, Alex Pentland, and Dhruv Mahajan. Adaptive methods for real-world domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021. 2, 3, 4

[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/CVF international conference on computer vision*, 2013. 1, 3, 8, 16

[15] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 2016. 4

[16] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021. 3, 4, 13

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2016. 1, 3

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 1

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022. 1, 3

[20] Sobhan Hemati, Mahdi Beitollahi, Amir Hossein Estiri, Bassel Al Omari, Xi Chen, and Guojun Zhang. Cross domain generative augmentation: Domain generalization with latent diffusion models. *arXiv preprint arXiv:2312.05387*, 2023. 2

[21] Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019. 3

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 2020. 2

[23] 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*, 2021. 2

[24] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *Proceedings of the European conference on computer vision (ECCV)*, 2020. 2

[25] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, 2021. 1

[26] Dongsheng Jiang, Yuchen Liu, Songlin Liu, Jin'e Zhao, Hao Zhang, Zhen Gao, Xiaopeng Zhang, Jin Li, and Hongkai Xiong. From clip to dino: Visual encoders shout in multi-modal large language models. *arXiv preprint arXiv:2310.08825*, 2023. 3

[27] Dahye Kim, Xavier Thomas, and Deepti Ghadiyaram. Revelio: Interpreting and leveraging semantic information in diffusion models. *arXiv preprint arXiv:2411.16725*, 2024. 2, 4, 11

[28] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013. 2

[29] Alexander C Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. Your diffusion model is secretly a zero-shot classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2206–2217, 2023. 2

[30] 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*, 2017. 3, 8, 14

[31] 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*, 2018. 4

[32] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2018. 4

[33] Jingjing Li, Zhiqi Yu, Zhekai Du, Lei Zhu, and Heng Tao Shen. A comprehensive survey on source-free domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 1

[34] 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 international conference on computer vision*, 2021. 2

[35] Feng Liang, Yangguang Li, and Diana Marculescu. Supmae: Supervised masked autoencoders are efficient vision learners. *arXiv preprint arXiv:2205.14540*, 2022. 3

[36] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 1

[37] Ran Liu, Sahil Khose, Jingyun Xiao, Lakshmi Sathidevi, Keerthan Ramnath, Zsolt Kira, and Eva L Dyer. Latentdr: Improving model generalization through sample-aware latent degradation and restoration. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2024. 2, 4

[38] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. *Advances in Neural Information Processing Systems*, 2024. 2

[39] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van Der Maaten. Exploring the limits of weakly supervised pretraining. In *Proceedings of the European conference on computer vision (ECCV)*, 2018. 1

[40] Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains. In *Proceedings of the AAAI conference on artificial intelligence*, 2020. 2, 3

[41] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, Bernhard Schölkopf, et al. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 2017. 1

[42] 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*, 2021. 2

[43] 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*, 2021. 4

[44] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1, 3

[45] Yong-Hyun Park, Mingi Kwon, Jaewoong Choi, Junghyo Jo, and Youngjung Uh. Understanding the latent space of diffusion models through the lens of riemannian geometry. *Advances in Neural Information Processing Systems*, 2023. 4

[46] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 1, 3, 4

[47] 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 international conference on computer vision*, 2019. 3, 22

[48] Yurui Qian, Qi Cai, Yingwei Pan, Yehao Li, Ting Yao, Qibin Sun, and Tao Mei. Boosting diffusion models with moving average sampling in frequency domain. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024. 4

[49] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 1, 3

[50] Alexandre Rame, Corentin Dancette, and Matthieu Cord. Fishr: Invariant gradient variances for out-of-distribution generalization. In *International conference on machine learning*, 2022. 4

[51] 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*, 2022. 1, 2, 3, 4

[52] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR, 2015. 2

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

[54] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Proceedings of the European conference on computer vision (ECCV)*, 2016. 4

[55] Piotr Teterwak, Kuniaki Saito, Theodoros Tsiligkaridis, Kate Saenko, and Bryan A Plummer. Erm++: An improved baseline for domain generalization. *arXiv preprint arXiv.2304.01973*, 2023. 4, 13

[56] Xavier Thomas, Dhruv Mahajan, Alex Pentland, and Abhimanyu Dubey. Adaptive methods for aggregated domain

generalization. *arXiv preprint arXiv:2112.04766*, 2021. 2, 3, 4

[57] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024. 1, 3

[58] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 1999. 2, 4

[59] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2017. 3, 8, 18

[60] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. p+: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023. 2

[61] Wenxuan Wang, Quan Sun, Fan Zhang, Yepeng Tang, Jing Liu, and Xinlong Wang. Diffusion feedback helps clip see better. *arXiv preprint arXiv:2407.20171*, 2024. 2

[62] Weijia Wu, Yuzhong Zhao, Hao Chen, Yuchao Gu, Rui Zhao, Yefei He, Hong Zhou, Mike Zheng Shou, and Chunhua Shen. Datasetdm: Synthesizing data with perception annotations using diffusion models. In *Advances in Neural Information Processing Systems*, 2023. 2

[63] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023. 2

[64] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. In *Proceedings of the AAAI conference on artificial intelligence*, 2020. 4

[65] Shen Yan, Huan Song, Nanxiang Li, Lincan Zou, and Liu Ren. Improve unsupervised domain adaptation with mixup training. *arXiv preprint arXiv:2001.00677*, 2020. 4

[66] Xueting Yan, Ishan Misra, Abhinav Gupta, Deepti Ghadiyaram, and Dhruv Mahajan. Clusterfit: Improving generalization of visual representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 1

[67] Runpeng Yu, Songhua Liu, Xingyi Yang, and Xinchao Wang. Distribution shift inversion for out-of-distribution prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023. 2

[68] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. In *Proceedings of the IEEE international conference on computer vision*, 2023. 2

[69] Xie Zhenda, Geng Zigang, Hu Jingcheng, Zhang Zheng, Hu Han, and Cao Yue. Revealing the dark secrets of masked image modeling. *arXiv preprint arXiv:2205.13543*, 2022. 3

Appendix:
What's in a Latent? Leveraging Diffusion Latent Space for Domain Generalization

A. Class NMI Scores

Dataset	DiT	SD-2.1	MAE	CLIP	DINOv2	RN50
PACS	0.08	0.08	0.11	0.05	0.15	0.29
VLCS	0.12	0.15	0.17	0.01	0.11	0.39
TerraInc	0.32	0.35	0.32	0.01	0.16	0.30
OfficeHome	0.16	0.22	0.28	0.10	0.23	0.59
DomainNet	0.16	0.20	0.22	0.13	0.19	0.36

Table 4. **Comparison of class NMI scores across datasets.** In order to choose auxiliary features for domain separation, a feature space that yields lower class NMI score along with high domain NMI is desirable, i.e. the latent space should favor grouping domains over object classes.

B. Underlying Domains in Each Dataset

We begin by summarizing the types of domain shifts present in the datasets we study. PACS [30] image dataset captures 7 object categories and 4 domains: real-world photos, art paintings, cartoons, and sketches. Thus, the domains have stark visual distinctions driven by both global and local changes such as shapes, colors, and edges. VLCS [14] is curated from different datasets, making dataset-specific biases such as spatial composition and object size variations as different domains. OfficeHome [59] similar to PACS also has images belonging to four domains: artistic, clip-art, product catalog, and real-world images. Thus, while there is some overlap in the underlying structural characteristics of the objects across domains, the domain shifts primarily involve style differences such as variations in texture, color, and outlines. TerraIncognita [3] consists of images taken from different camera trap locations, and each camera serves as a domain. Thus, the domain shifts are driven by physical environmental aspects such as variations in foliage density, terrain patterns, and spatial patterns of vegetation. DomainNet [**?**] is composed of six domains such as quick-draw, infographic, real images, and so on, and exhibits a broader range of domain shifts than PACS, spanning both coarse and fine-grained variations. For example, the "quickdraw" domain consists of simple, rough sketches, while "sketch" has more detailed drawings with shading and varied strokes, showing style differences. By contrast, "real" domain captures fully detailed images, indicating shifts of varied granularities between different domains.

C. Pseudo-domains Examples

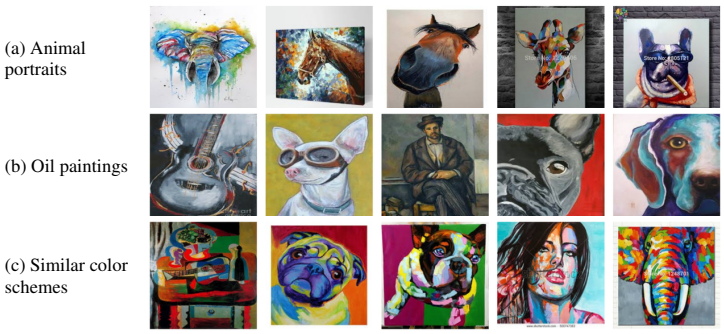

Figure 3. **Pseudo-domains captured in the diffusion latent space of DiT on PACS.** The clusters group images based on nuanced style-specific variances rather than class-specific variances.

D. Transformation Function

Transformation ($\mathcal{T}$)	**Acc**
ERM	83.8
Direct Concatenation (No Transformation)	84.3
Cluster-Based Replacement	84.6
Linear Regression	85.7
RBF Kernel Ridge Regression	**87.1**

Table 5. **Effect of $\mathcal{T}$ on Test Accuracy for PACS, using GUIDE-DiT.** We find that the RBF step (Sec 4.1) aids in classification performance on unseen domains.

Effect of the choice of $\mathcal{T}$:

As noted in Sec. 3, we apply a transformation function $\mathcal{T} : \mathbf{\Psi} \mapsto \mathbf{\Phi}$ to bring the latent manifold of Ψ closer to Φ and mitigate feature domain drift. To understand the role of $\mathcal{T}$, we explore the following alternatives to it:

- **(a) Direct concatenation**, i.e., appending pseudo-domain representations (from $\mathbf{\Psi}$) to the features (from $\mathbf{\Phi}$) without any transformation. While this introduces domain-specific information, lack of alignment between the two feature spaces led to a minimal improvement of $+0.5\%$ over ERM.
- **(b) Cluster-based replacement**, where pseudo-domains identified in the $\mathbf{\Psi}$ space are used to compute cluster centroids using features from $\mathbf{\Phi}$ space, i.e. cluster samples are averaged in $\mathbf{\Phi}$ space. This provides a slightly better alignment yielding an accuracy gain of $+0.8\%$ over the baseline.
- **(c) Linear regression**, where a linear mapping is learned between the pseudo-domain centroids and the centroids obtained in **(b)**. This helps in bridging differences between $\mathbf{\Psi}$ and $\mathbf{\Phi}$ better, leading to a larger improvement of $+1.4\%$.
- **(d) RBF kernel ridge regression**, where the linear regressor in **(c)** is replaced with an RBF kernel (Sec 4.1). We note that this achieves the highest accuracy gains of $+3.3\%$, highlighting its effectiveness of bridging feature domain drift while incorporating pseudo-domain information into the classifier.

These results underscore the necessity of a well-chosen transformation to fully leverage the pseudo-domain information.

E. Domain Predictability

Dataset	**DiT**	**SD-2.1**	**MAE**	**CLIP**	**DINOv2**	**RN50**
PACS	98.89	**98.95**	98.69	98.29	98.89	97.85
VLCS	**96.08**	92.72	94.03	83.87	81.86	88.48
TerraInc	**99.97**	99.94	99.91	99.83	99.87	99.79
OfficeHome	**89.16**	86.43	82.55	83.41	78.28	77.52
DomainNet	88.55	**89.58**	87.50	87.61	87.24	87.21
Synth-Artists	**100**	99.00	97.00	92.00	90.00	97.00
Synth-Photography	83.33	**87.50**	86.67	73.33	78.33	77.50

Table 6. **Comparison of Domain Predictability Scores Across Datasets.** Diffusion models consistently outperform other models in domain predictability scores, highlighting the effectiveness of encoding domain-specific information in their latent space.

Domain Predictability: To complement NMI, we evaluate domain predictability and predict domain labels from latent feature representations. Specifically, we use a single-layer MLP classifier, trained on an 80-20 train-test split. We report the mean test accuracy over 3 such random splits. While NMI measures alignment and variance across samples belonging to a domain, domain predictability directly assesses a latent representation's ability to learn to classify domain information. We observe in Table. 6 that diffusion models attain the highest domain predictability scores, highlighting their effectiveness in encoding domain-specific information.

F. Label Noise and Domain Inconsistencies

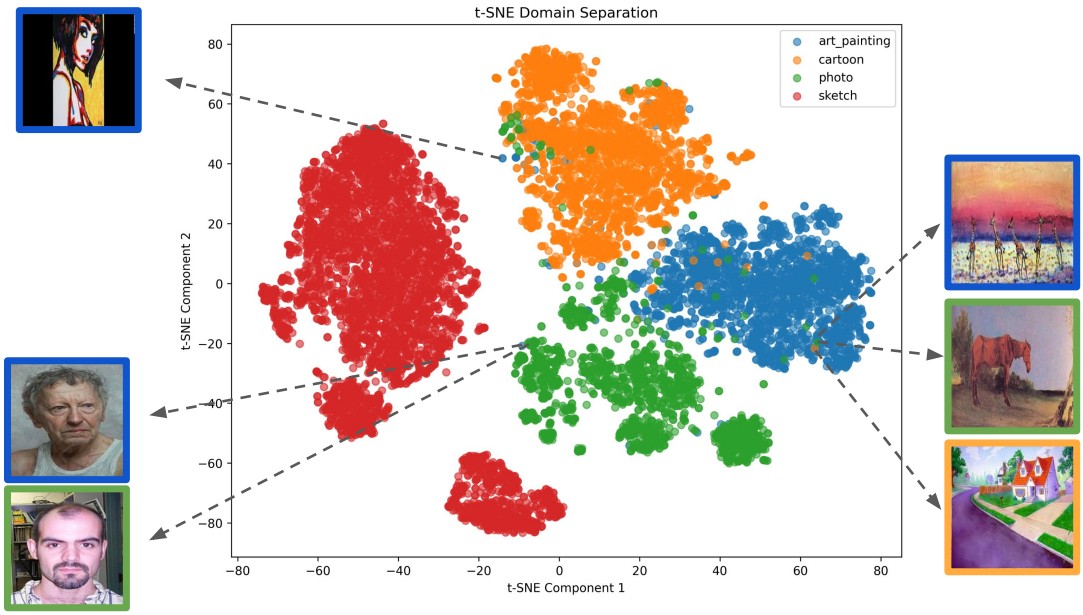

Figure 4. **Examples of inconsistent or confusing domain labels**. Given that most datasets in this study are web-scraped, we expect there to be label noise and domain inconsistencies which may impact the NMI scores. These examples from the PACS dataset and SD-2.1 feature space illustrate cases where domain assignments may be unclear or conflicting. The color of the border on the images denotes the ground truth domain label.

G. Effect of Text-Conditioning in SD-2.1 for Domain Separation

Dataset	Domain NMI		Domain Predictability	
	Empty Prompt	Prompt	Empty Prompt	Prompt
PACS	0.82	0.85	98.95	99.51
OfficeHome	0.22	0.24	86.43	92.91

Table 7. **Domain NMI and predictability scores for empty vs text conditioned prompts for SD-2.1 on PACS and OfficeHome.** For text conditioning we used the prompt: "A photo of an object in the style of {domain}". Similar to the findings of Kim et al. [27], text conditioning appears to activate more relevant features.

H. Effect of Layer and Timestep in Diffusion Models for Domain Separation (DiT vs SD-2.1) on PACS, and VLCS

Following Kim et al. [27], we choose a lower noise level at timestep (t=50), with a motivation to capture rich fine-grained visual information. We use t=50 for both DiT (at block 14) and SD-2.1 (at `up_ft:1`) for both class and domain NMI scores (in Tables 1, and 4), and to obtain the classification accuracies in Table 3. In Fig. 5, we observe that t=50 provides the highest domain NMI score for PACS using DiT. We also note that on VLCS, the `bottleneck` layer outperforms the domain NMI score obtained from `up_ft:1` in Fig. 6, likely due it's focus on coarse-grained features as noted in [27].

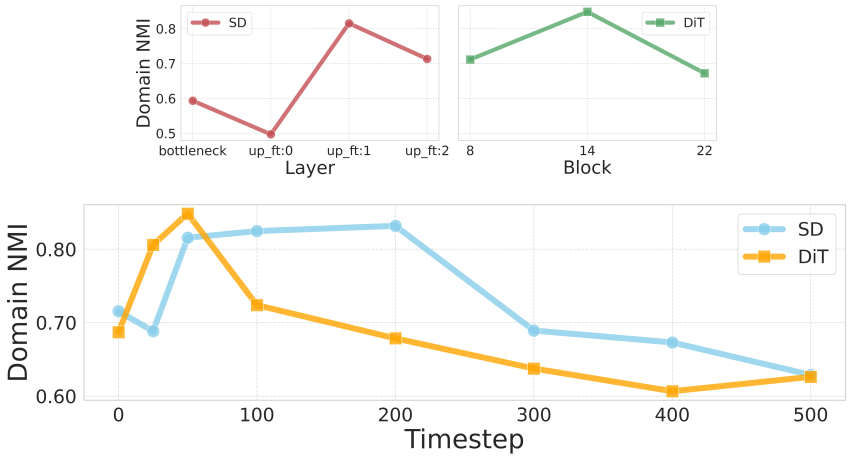

Figure 5. **Domain NMI comparison across layers and timesteps for PACS.** Top: Domain NMI scores for SD-2.1 layers (best: up_ft:1) and DiT blocks (best: block:14). Bottom: Domain NMI scores across various denoising timesteps for SD-2.1 and DiT on PACS.

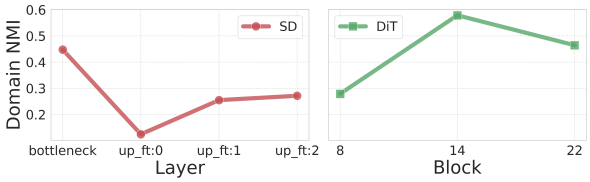

Figure 6. **Domain NMI comparison across layers for VLCS.** The Bottleneck Layer of Stable Diffusion (SD-2.1) which capture more coarse-grained features aids in separating high-level domain shifts in VLCS. However, DiT's superior capability to capture global context via self-attention outperforms the domain NMI scores at `bottleneck` and `up_ft:1`.

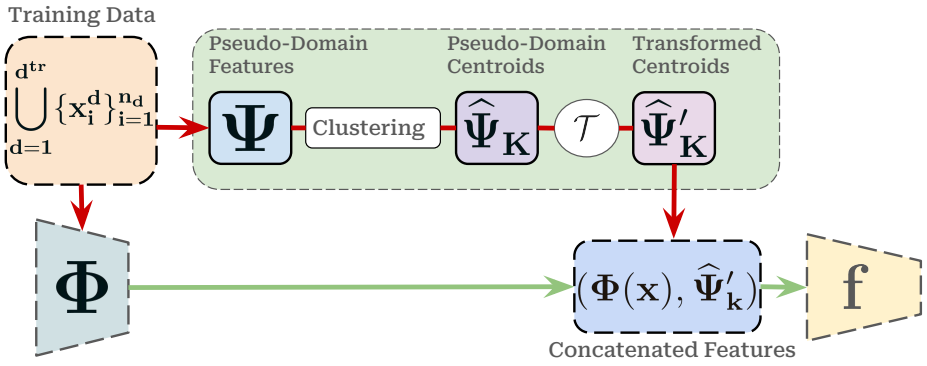

Figure 7. **Training Pipeline.** The green-shaded region represents the clustering and transformation step. Green solid arrows indicate gradient flow, while red arrows represent non-gradient operations. The feature extractor $\mathbf{\Psi}$ first clusters samples to compute the pseudo-domain centroids. The transformation function $\mathcal{T}$ then transforms these centroids to the latent space of $\mathbf{\Phi}$, producing transformed pseudo-domain centroids, which are concatenated with the features from $\mathbf{\Phi}$, and sent to the classifier.

I. GUIDE Pseudo-code

Algorithm 1 Training Pseudocode with RBF Kernel Ridge Regression

Input: Training data D_{tr}, transform schedule $T_{\text{transform}}$, K: #clusters
Output: $F_{\text{image}}(.;\omega)$, $F_{\text{MLP}}(.;\mathbf{W})$, mapping $\mathcal{T}$

Initialize: Compute feature representations Ψ, Φ, initialize model parameters $\omega_0, \mathbf{W}$.
$\{\psi_k\}, \{D_k\} \leftarrow$ CLUSTERING(Ψ, K)
for $t = 1$ to T **do**
  if $t \in T_{\text{transform}}$ **then**
    For each k: $\widehat{\Phi}_k = \frac{1}{|D_k|} \sum_{\mathbf{x} \in D_k} \Phi(\mathbf{x})$
    Compute pairwise distances $\|\psi_i - \psi_j\|_2, \forall i \neq j$
    $\gamma \leftarrow 1/(2 \cdot \text{median(pairwise distances)}^2)$ {using median heuristic}
    Fit $\mathcal{T}$ **via RBF Kernel Ridge Regression** using $\{\widehat{\psi}_k\} \mapsto \{\widehat{\Phi}_k\}$ and γ
    $\psi'_{\mathbf{x}} \leftarrow \mathcal{T}(\psi_{\mathbf{x}})$
  end if
  for batch $(\mathbf{x}, \psi_{\mathbf{x}}, y)$ in D_{tr} **do**
    $\Phi(\mathbf{x}) \leftarrow F_{\text{image}}(\mathbf{x}; \omega_t)$
    $\psi'_{\mathbf{x}} \leftarrow \mathcal{T}(\psi_{\mathbf{x}})$
    $\hat{y} \leftarrow F_{\text{MLP}}\big(\text{CONCAT}\big(\Phi(\mathbf{x}), \psi'_{\mathbf{x}}\big); \mathbf{W}_t\big)$
    Update $\omega_{t+1}, \mathbf{W}_{t+1}$ via SGD STEP on $\mathcal{L} = \text{CROSSENTROPY}(\hat{y}, y)$
  end for
end for
Return $F_{\text{image}}(.;\omega_T)$, $F_{\text{MLP}}(.;\mathbf{W}_T)$, and $\mathcal{T}$

Inference
Input: Test data D_{test}, transformation function $\mathcal{T}$, and centroids $\{\widehat{\psi}_k\}_{k=1}^K$
Output: Predicted labels $\hat{y}$

for $\mathbf{x} \in D_{\text{test}}$ **do**
  $\psi_{\mathbf{x}} \leftarrow$ NEARESTCENTROID$\big(\Psi, \mathbf{x}\big)$ {Find closest cluster in Ψ-space}
  $\psi'_{\mathbf{x}} \leftarrow \mathcal{T}(\psi_{\mathbf{x}})$ {Apply same RBF transform as in training}
  $\Phi(\mathbf{x}) \leftarrow F_{\text{image}}(\mathbf{x}; \omega_T)$
  $\hat{y} \leftarrow F_{\text{MLP}}\big(\text{CONCAT}(\Phi(\mathbf{x}), \psi'_{\mathbf{x}}); \mathbf{W}_T\big)$
end for
Return $\hat{y}$

J. Effect of enhanced training strategies

Dataset	ERM / ERM++	GUIDE / GUIDE++ (DiT)	GUIDE / GUIDE++ (SD-2.1)
PACS	83.8 / 88.0	87.1 / **89.2**	86.9 / 88.6
TI	47.0 / 50.7	48.2 / 52.7	51.3 / **53.6**

Table 8. **ERM++ [55] training strategies on GUIDE** boost performance.

We follow the ERM++ [55] implementation from DomainBed [16] which improves ERM by better utilization of training data, model parameter selection, and weight-space regularization techniques. From Table 8, ERM++ improves over standard ERM by **+4.2%** on PACS and **+3.7%** on TerraIncognita. Applying the same strategies to GUIDE, we achieve even greater improvements, with GUIDE++ outperforming ERM by **+5.4%** on PACS and **+6.6%** on TerraIncognita. These results show that GUIDE could benefit from any training optimizations proposed over ERM, such as SWAD [9].

K. Domain Shift Examples and Domain Separation in Feature Spaces

In this section, we provide:

- **Example images**, i.e. class samples across domains for each dataset.
- **Class vs Domain NMI scores** for each feature extractor (Ψ) studied in this work, on each dataset.
- **Feature space visualizations** for each feature extractor (Ψ) studied in this work, on the PACS, VLCS, OfficeHome, and TerraInognita datasets.

K.1. PACS [30]

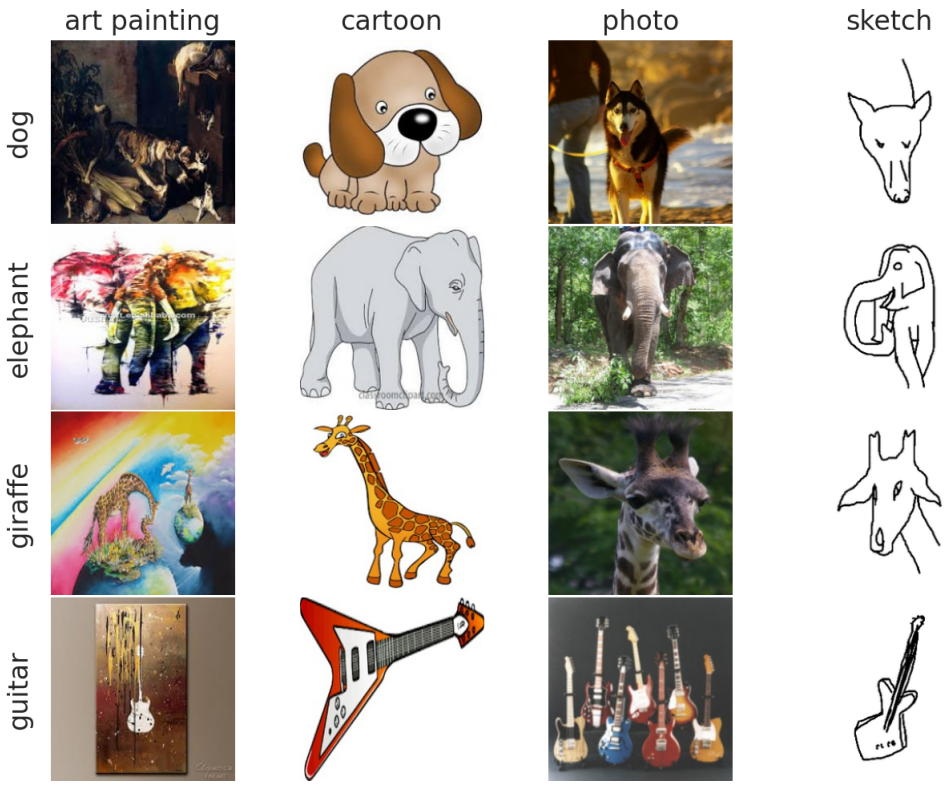

Figure 8. Class examples across domains in the PACS dataset. Each column represents a domain, and each row corresponds to a class.

Domains	Classes
art painting, cartoon, photo, sketch	dog, elephant, giraffe, guitar, horse, house, person

Table 9. 4 domains and 7 classes of the PACS dataset.

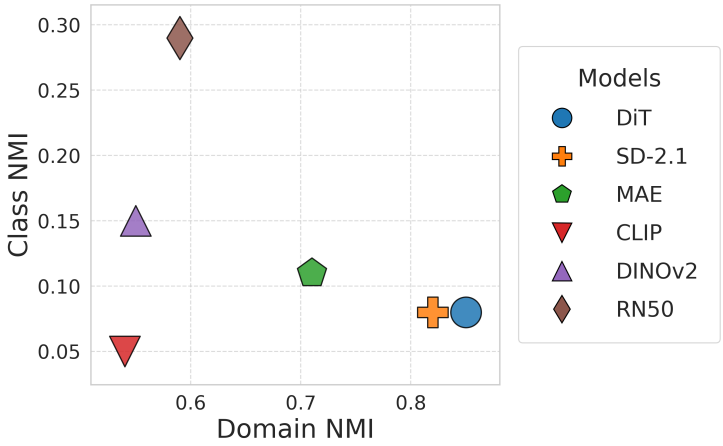

Figure 9. **Class vs Domain NMI scores for PACS.** Note how RN50 has the highest class NMI and diffusion models have low class NMI scores. Diffusion models also has the highest domain NMI scores, thereby capturing domain-specific class invariant structures.

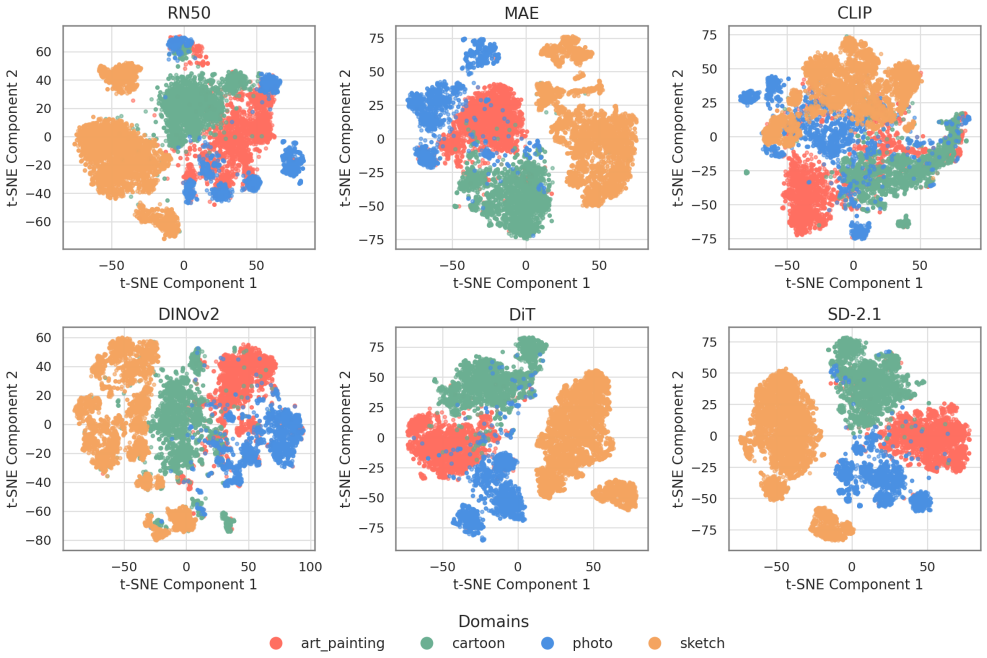

Figure 10. **T-SNE visualization of domain separation for PACS**. Each point represents a sample, colored by its domain. Notice how well separated the domains are when diffusion features are used compared to other models.

K.2. VLCS [14]

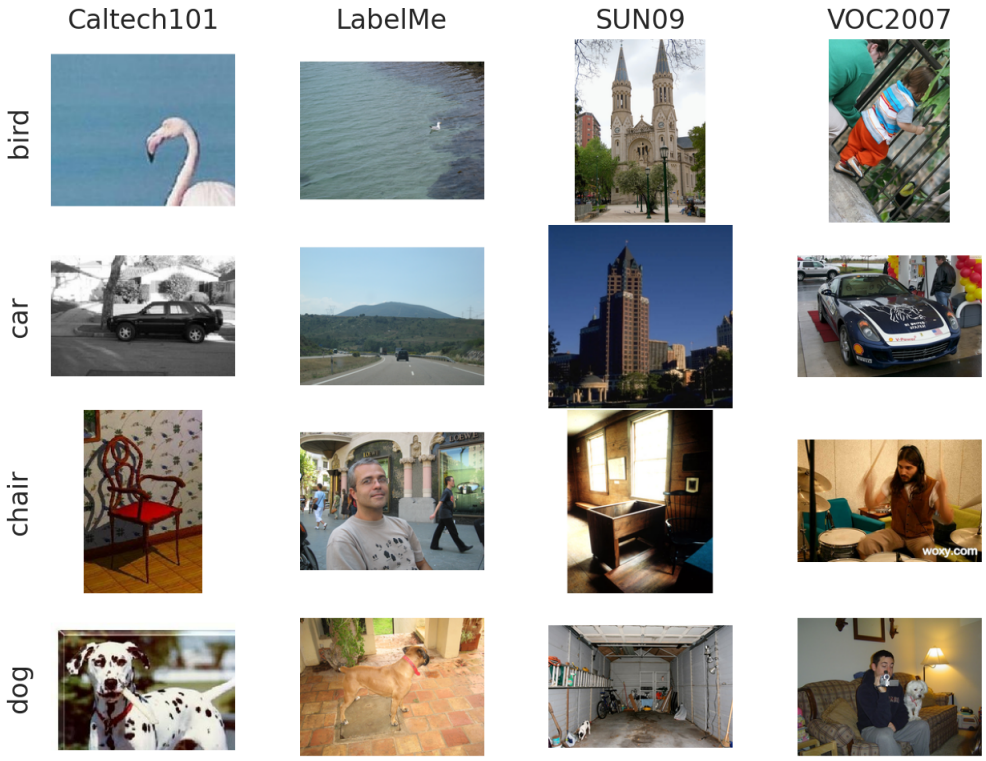

Figure 11. Class examples across domains in the VLCS dataset. Each column represents a domain, and each row corresponds to a class.

Domains	**Classes**
Caltech101, LabelMe, SUN09, VOC2007	bird, car, chair, dog, person

Table 10. 4 domains and 5 classes of the VLCS dataset.

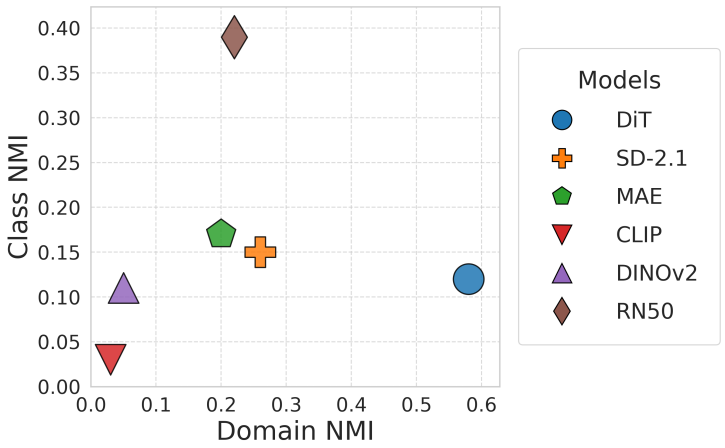

Figure 12. **Class vs Domain NMI scores for VLCS.** Note how RN50 has the highest class NMI score, and diffusion models have low class NMI scores. DiT has a much higher domain NMI score than SD-2.1, resulting from its stronger capability in capturing high-level dataset-specific biases, as discussed in Sec. 4.2.

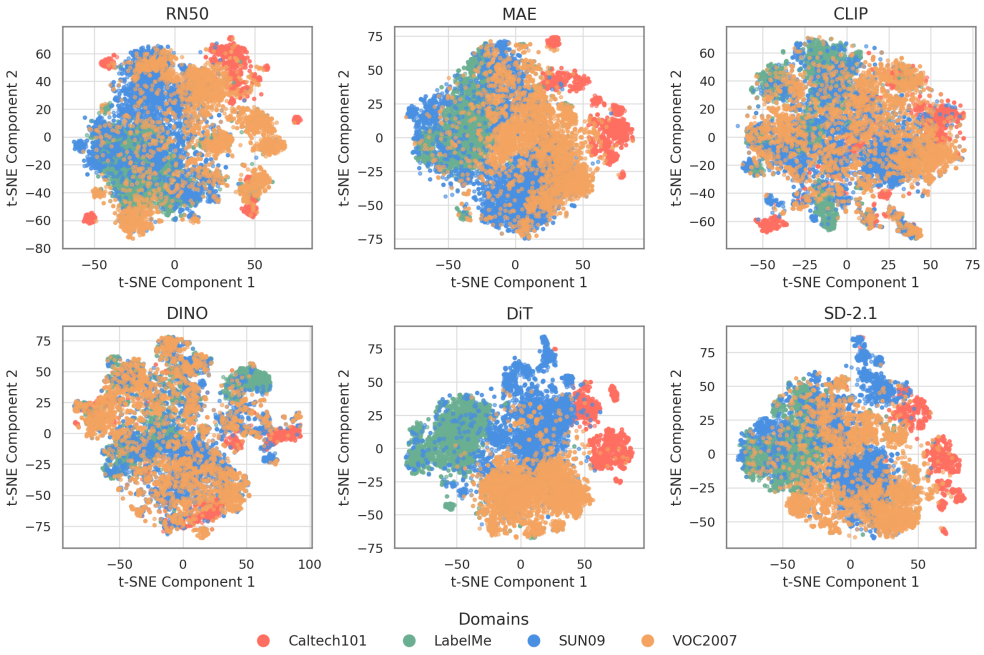

Figure 13. **T-SNE visualization of domain separation for VLCS**. Each point represents a sample, colored by its domain. Note how the DiT feature space best separate the domains.

K.3. OfficeHome [59]

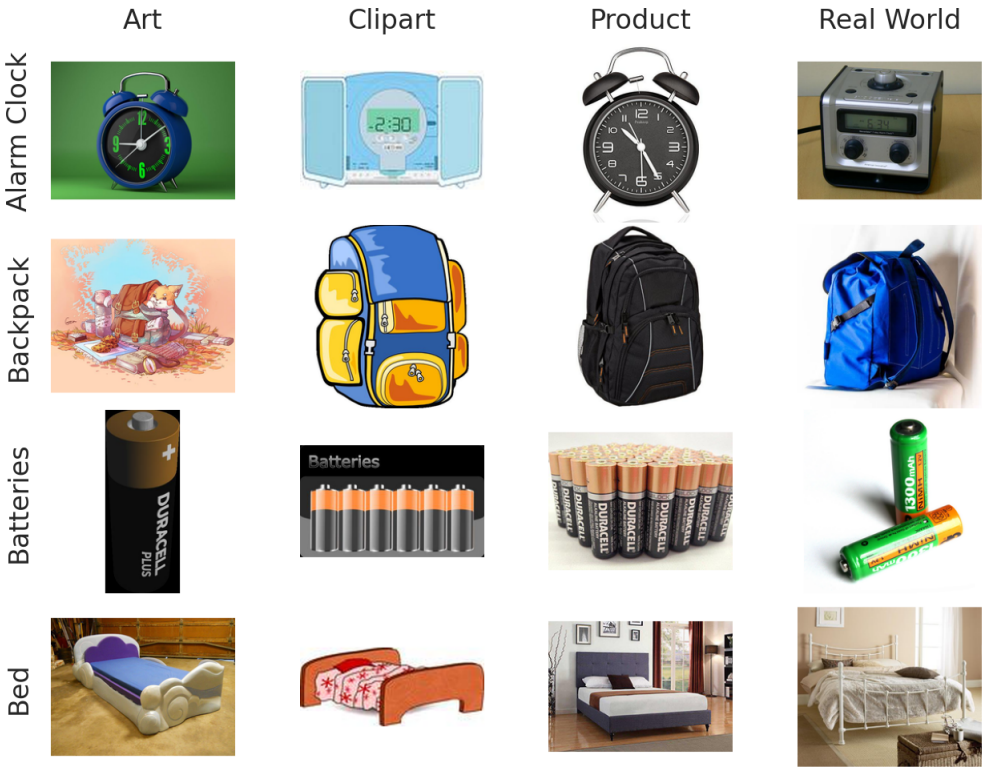

Figure 14. Class examples across domains in the OfficeHome dataset. Each column represents a domain, and each row corresponds to a class.

Domains	**Classes**
Art, Clipart, Product, Real World	Alarm Clock, Backpack, Batteries, Bed, Bike, Bottle, Bucket, Calculator, Calendar, Candles, Chair, Clipboards, Computer, Couch, Curtains, Desk Lamp, Drill, Eraser, Exit Sign, Fan, File Cabinet, Flipflops, Flowers, Folder, Fork, Glasses, Hammer, Helmet, Kettle, Keyboard, Knives, Lamp Shade, Laptop, Marker, Monitor, Mop, Mouse, Mug, Notebook, Oven, Pan, Paper Clip, Pen, Pencil, Post-it Notes, Printer, Push Pin, Radio, Refrigerator, Ruler, Scissors, Screwdriver, Shelf, Sink, Sneakers, Soda, Speaker, Spoon, TV, Table, Telephone, ToothBrush, Toys, Trash Can, Webcam.

Table 11. 4 domains and 65 Classes of the OfficeHome dataset.

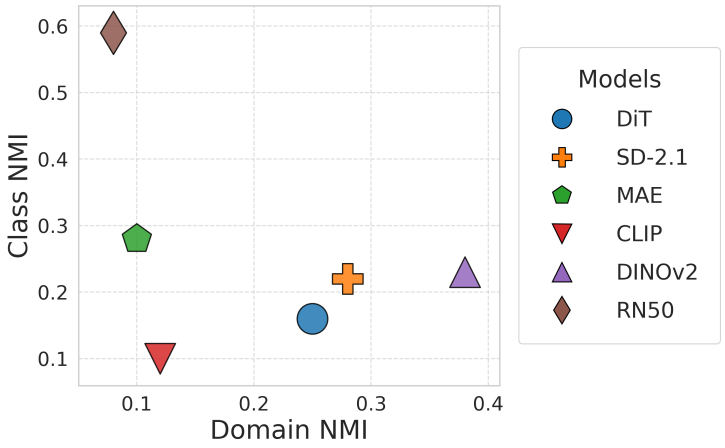

Figure 15. **Class vs Domain NMI scores for OfficeHome.** Note how RN50 has the highest class NMI score and DINOv2 has the highest domain NMI score, resulting form its stronger ability in capturing low-level style shifts, as discussed in Sec. 4.2. DiT and SD-2.1 have moderate domain NMI scores, with DiT having a lower class NMI score.

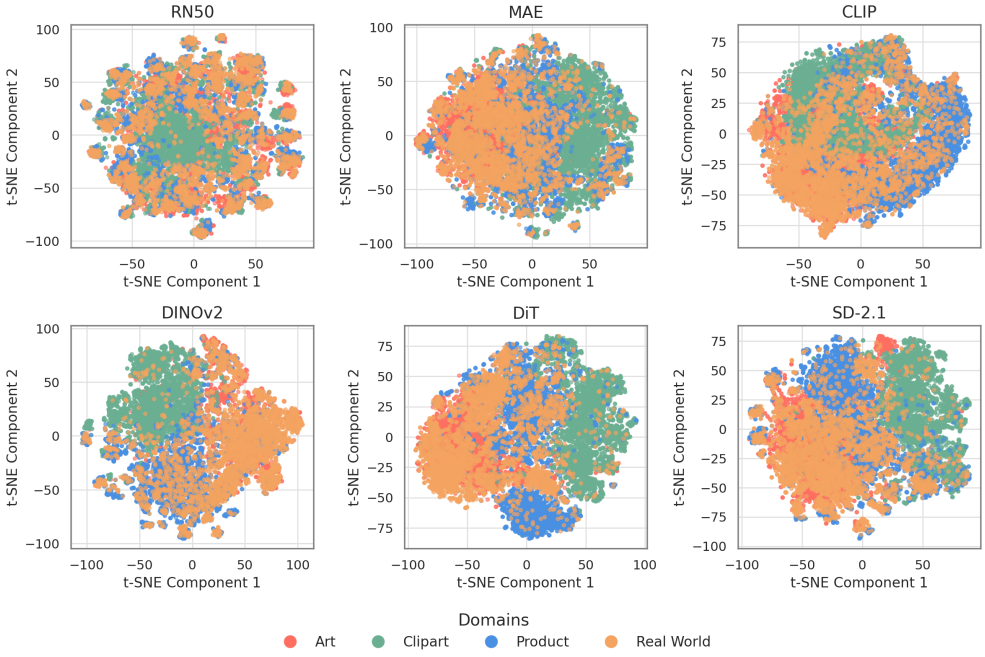

Figure 16. **T-SNE visualization of domain separation for OfficeHome.** Each point represents a sample, colored by its domain. All models struggle to separate the domains in this dataset. The "real" domain has considerable overlap with the other domains.

K.4. TerraIncognita [3]

Figure 17. Class examples across domains in the TerraIncognita dataset. Each column represents a domain, and each row corresponds to a class.

Domains	**Classes**
Location 100, Location 38, Location 43, Location 46	bird, bobcat, cat, coyote, dog, empty, opossum, rabbit, raccoon, squirrel

Table 12. 4 domains and 10 classes of the TerraIncognita dataset.

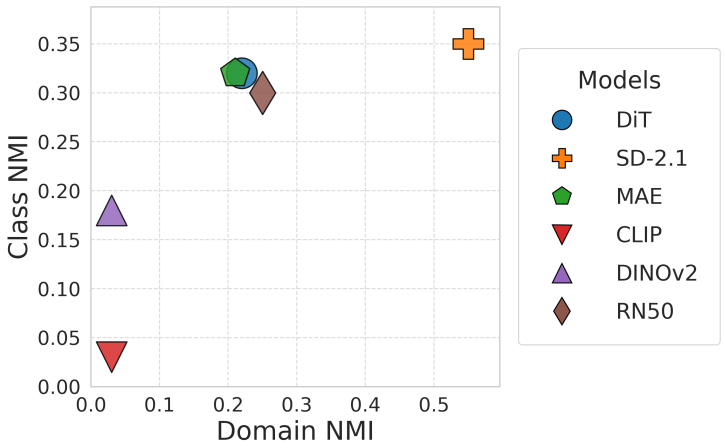

Figure 18. **Class vs Domain NMI scores for TerraIncognita.** Most models have a high class NMI score. SD-2.1 has the highest domain NMI score, resulting from its stronger capability in capturing spatial information, as discussed in Sec. 4.2.

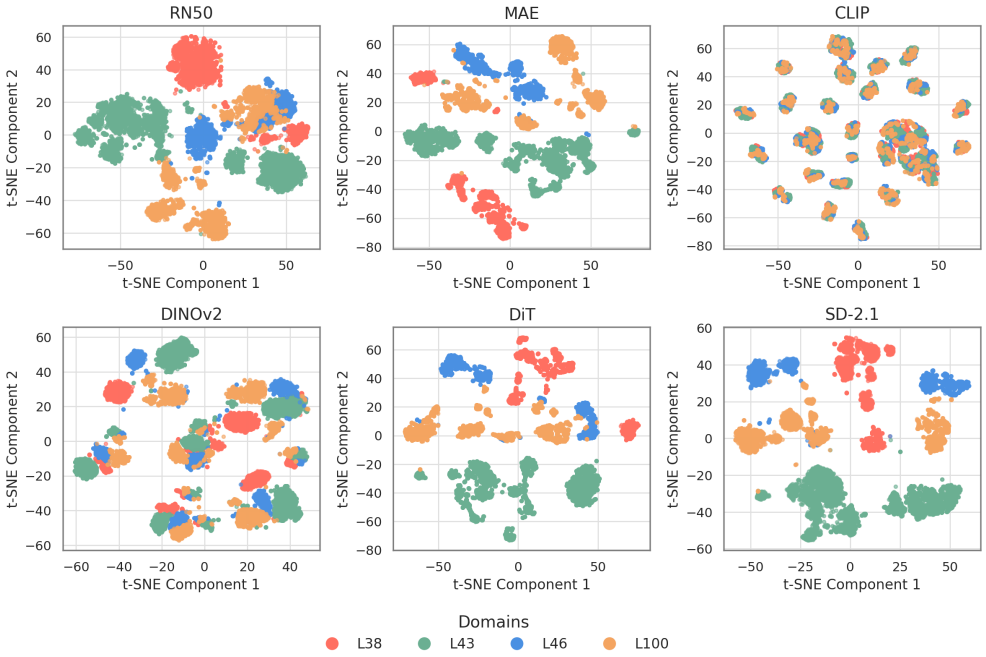

Figure 19. **T-SNE visualization of domain separation for TerraIncognita**. Each point represents a sample, colored by its domain. Note how the SD-2.1 feature space best groups samples from the same domain closer together, and separate from other domains.

K.5. DomainNet [47]

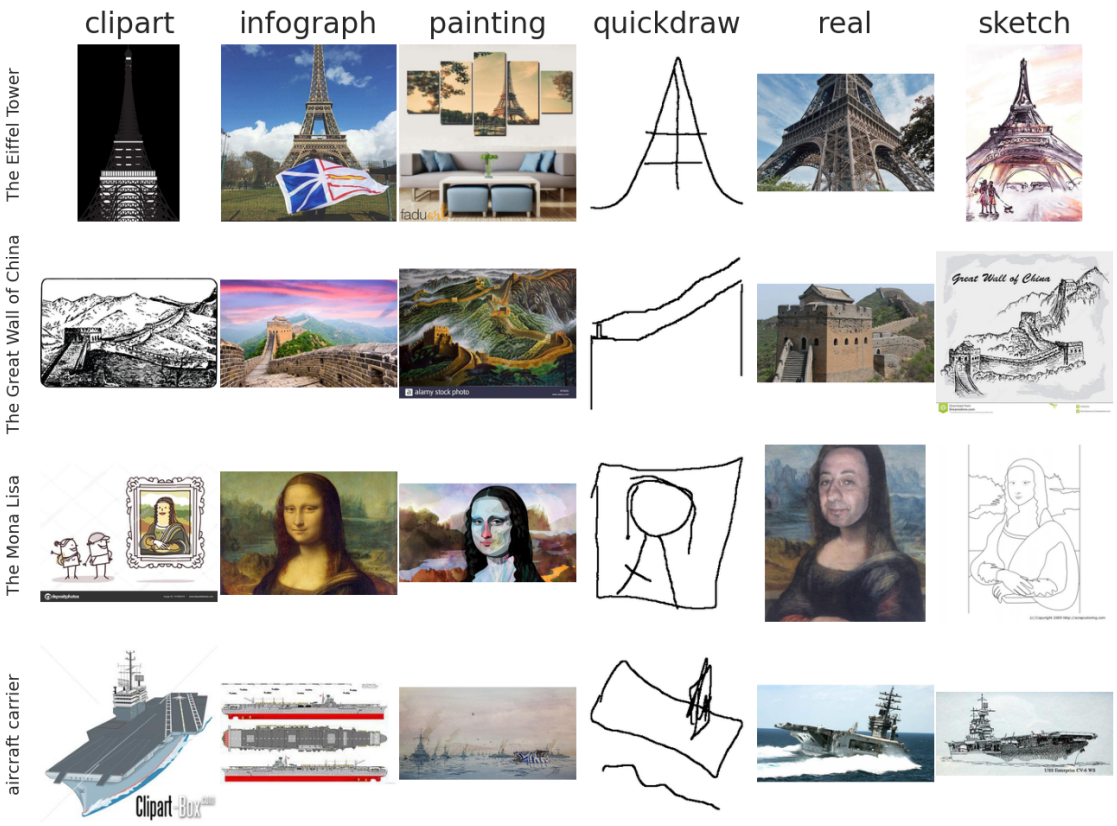

Figure 20. Class examples across domains in the DomainNet dataset. Each column represents a domain, and each row corresponds to a class.

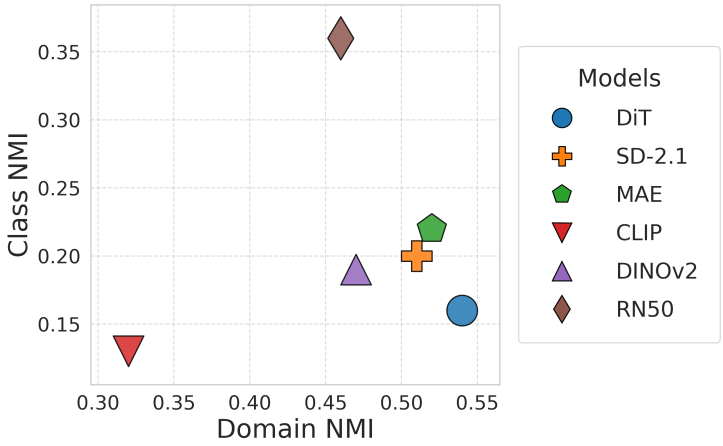

Figure 21. **Class vs Domain NMI scores for DomainNet.** Note how RN50 has the highest class NMI and diffusion models, and MAE have the highest domain NMI scores, with DiT having a lower class NMI score. All models except CLIP exhibit a moderate domain NMI score, likely due to the varied domain shifts inherent in the dataset, as discussed in Sec. 4.2.

Domains	Classes
clipart, infograph, painting, quickdraw, real, sketch	The Eiffel Tower, The Great Wall of China, The Mona Lisa, aircraft carrier, airplane, alarm clock, ambulance, angel, animal migration, ant, anvil, apple, arm, asparagus, axe, backpack, banana, bandage, barn, baseball, baseball bat, basket, basketball, bat, bathtub, beach, bear, beard, bed, bee, belt, bench, bicycle, binoculars, bird, birthday cake, blackberry, blueberry, book, boomerang, bottlecap, bowtie, bracelet, brain, bread, bridge, broccoli, broom, bucket, bulldozer, bus, bush, butterfly, cactus, cake, calculator, calendar, camel, camera, camouflage, campfire, candle, cannon, canoe, car, carrot, castle, cat, ceiling fan, cell phone, cello, chair, chandelier, church, circle, clarinet, clock, cloud, coffee cup, compass, computer, cookie, cooler, couch, cow, crab, crayon, crocodile, crown, cruise ship, cup, diamond, dishwasher, diving board, dog, dolphin, donut, door, dragon, dresser, drill, drums, duck, dumbbell, ear, elbow, elephant, envelope, eraser, eye, eyeglasses, face, fan, feather, fence, finger, fire hydrant, fireplace, firetruck, fish, flamingo, flashlight, flip flops, floor lamp, flower, flying saucer, foot, fork, frog, frying pan, garden, garden hose, giraffe, goatee, golf club, grapes, grass, guitar, hamburger, hammer, hand, harp, hat, headphones, hedgehog, helicopter, helmet, hexagon, hockey puck, hockey stick, horse, hospital, hot air balloon, hot dog, hot tub, hourglass, house, house plant, hurricane, ice cream, jacket, jail, kangaroo, key, keyboard, knee, knife, ladder, lantern, laptop, leaf, leg, light bulb, lighter, lighthouse, lightning, line, lion, lipstick, lobster, lollipop, mailbox, map, marker, matches, megaphone, mermaid, microphone, microwave, monkey, moon, mosquito, motorbike, mountain, mouse, moustache, mouth, mug, mushroom, nail, necklace, nose, ocean, octagon, octopus, onion, oven, owl, paint can, paintbrush, palm tree, panda, pants, paper clip, parachute, parrot, passport, peanut, pear, peas, pencil, penguin, piano, pickup truck, picture frame, pig, pillow, pineapple, pizza, pliers, police car, pond, pool, popsicle, postcard, potato, power outlet, purse, rabbit, raccoon, radio, rain, rainbow, rake, remote control, rhinoceros, rifle, river, roller coaster, rollerskates, sailboat, sandwich, saw, saxophone, school bus, scissors, scorpion, screwdriver, sea turtle, see saw, shark, sheep, shoe, shorts, shovel, sink, skateboard, skull, skyscraper, sleeping bag, smiley face, snail, snake, snorkel, snowflake, snowman, soccer ball, sock, speedboat, spider, spoon, spreadsheet, square, squiggle, squirrel, stairs, star, steak, stereo, stethoscope, stitches, stop sign, stove, strawberry, streetlight, string bean, submarine, suitcase, sun, swan, sweater, swing set, sword, syringe, t-shirt, table, teapot, teddy-bear, telephone, television, tennis racquet, tent, tiger, toaster, toe, toilet, tooth, toothbrush, toothpaste, tornado, tractor, traffic light, train, tree, triangle, trombone, truck, trumpet, umbrella, underwear, van, vase, violin, washing machine, watermelon, waterslide, whale, wheel, windmill, wine bottle, wine glass, wristwatch, yoga, zebra, zigzag

Table 13. 6 domains and 325 classes of the DomainNet dataset.