APPENDIX

A PROOF OF THEOREM 1

Theorem 1 (Generalization Error Bound). *Let $\widetilde{\boldsymbol{x}}_k^T$ be a masked instance of $\boldsymbol{x}_k^T$ on an unseen domain T. Given an instance embedding $\boldsymbol{z}_k^T$ satisfies the composition of domain-specific $\boldsymbol{z}_k^{T\text{-}sh}$ and domain-sharing $\boldsymbol{z}_k^{T\text{-}sp}$, where $\hat{f}(\boldsymbol{x}_k^T) = \hat{c}(\boldsymbol{z}_k^T)$ be the predicted outcomes. The generalization error $\boldsymbol{GE} = \mathbf{E}_{\mathcal{X}}[\ \|f(\boldsymbol{x}_k^T), \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2\ ]$ of DISPEL framework can be bounded as:*

$$\boldsymbol{GE} \leq \mathbf{E}_{\mathcal{X}}[\|c(\boldsymbol{z}_k^{T\text{-}sh}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{T\text{-}sh})\|_2] + \mathbf{E}_{\mathcal{X}}[\|c(\boldsymbol{z}_k^{T\text{-}sp}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{T\text{-}sp})\|_2] \tag{5}$$

where $\widetilde{\boldsymbol{z}}_k^T = \boldsymbol{z}_k^T \odot m_k^T$ is composed of remained domain-specific embedding $\widetilde{\boldsymbol{z}}_k^{T\text{-}sp}$ and preserved domain-sharing embedding $\widetilde{\boldsymbol{z}}_k^{T\text{-}sh}$.

Proof. In order to estimate the generalization error of DISPEL on unseen domain $\mathcal{T}$, we calculate the expected values of distance between $f(\boldsymbol{x}_k^T)$ and $\hat{f}(\widetilde{\boldsymbol{x}}_k^T)$. Without loss of generality, we consider $\boldsymbol{\ell}_2$ norm to evaluate the distance. Hence, the estimated generalization error of $\mathcal{T}$ can be elaborated as:

$$\begin{aligned}\mathbf{GE} &= \mathbf{E}_{\mathcal{T}}[\ \|f(\boldsymbol{x}_k^T) - \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2\ ] \\ &= \int_{\mathcal{X}} \|f(\boldsymbol{x}_k^T) - \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2 P(\mathcal{T})\, d\mathcal{T}\end{aligned} \tag{6}$$

where $P(\mathcal{T})$ denotes probability density function of $\mathcal{T}$. As a lower generalized error **GE** represents better generalization capability, we can observe from Eq. 6 that the closer $\|f(\boldsymbol{x}_k^T) - \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2$ approaches zero, the better generalization capability is obtained.

In this manner, we now discuss the upper bound of $\|f(\boldsymbol{x}_k^T) - \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2$. This also ensure the upper bound of **GE**. Following the properties that each instance's embedding $\boldsymbol{z}_k^T$ can be composed of domain-specific $\boldsymbol{z}_k^{\text{T-sh}}$ and domain-sharing $\boldsymbol{z}_k^{\text{T-sp}}$, we consider upper bound as follows,

$$\begin{aligned}&\|f(\boldsymbol{x}_k^T) - \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2 \\ &= \|c(\boldsymbol{z}_k^T) - \hat{f}(\widetilde{\boldsymbol{z}}_k^T)\|_2 \\ &= \|c(\boldsymbol{z}_k^{\text{T-sh}} + \boldsymbol{z}_k^{\text{T-sp}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sh}} + \widetilde{\boldsymbol{z}}_k^{\text{T-sp}})\|_2\end{aligned} \tag{7}$$

Since $c(\cdot)$ is the linear predictor, we can now recast the Eq. 7 in the following,

$$\begin{aligned}&\|f(\boldsymbol{x}_k^T) - \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2 \\ &= \|c(\boldsymbol{z}_k^{\text{T-sh}} + \boldsymbol{z}_k^{\text{T-sp}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sh}} + \widetilde{\boldsymbol{z}}_k^{\text{T-sp}})\|_2 \\ &= \|c(\boldsymbol{z}_k^{\text{T-sh}}) + c(\boldsymbol{z}_k^{\text{T-sp}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sh}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sp}})\|_2 \\ &\leq \|c(\boldsymbol{z}_k^{\text{T-sh}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sh}})\|_2 + \|c(\boldsymbol{z}_k^{\text{T-sp}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sp}})\|_2\end{aligned} \tag{8}$$

Following the conclusion of Eq. 6 and Eq. 8, we have the upper bound of **GE** as follows:

$$\begin{aligned}&\mathbf{GE} = \mathbf{E}_{\mathcal{T}}[\ \|f(\boldsymbol{x}_k^T) - \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2\ ] \\ &= \int_{\mathcal{X}} \|f(\boldsymbol{x}_k^T) - \hat{f}(\widetilde{\boldsymbol{x}}_k^T)\|_2 P(\mathcal{T})\, d\mathcal{T} \\ &\leq \int_{\mathcal{X}} \Big[\|c(\boldsymbol{z}_k^{\text{T-sh}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sh}})\|_2 + \|c(\boldsymbol{z}_k^{\text{T-sp}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sp}})\|_2\Big] P(\mathcal{T}) d\mathcal{T} \\ &= \int_{\mathcal{X}} \|c(\boldsymbol{z}_k^{\text{T-sh}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sh}})\|_2\ P(\mathcal{T})\, d\mathcal{T} + \\ &\qquad \int_{\mathcal{X}} \|c(\boldsymbol{z}_k^{\text{T-sp}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sp}})\|_2\ P(\mathcal{T})\, d\mathcal{T} \\ &= \mathbf{E}_{\mathcal{X}}[\|c(\boldsymbol{z}_k^{\text{T-sh}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sh}})\|_2] + \mathbf{E}_{\mathcal{X}}[\|c(\boldsymbol{z}_k^{\text{T-sp}}) - \hat{c}(\widetilde{\boldsymbol{z}}_k^{\text{T-sp}})\|_2]\end{aligned}$$

□

B Related Works

There are two primary branches of research in the field of domain generalization: data manipulation and representation learning.

Data Manipulation. The data manipulation branch aims to reduce overfitting by increasing the diversity and quantity of available training data. This is typically achieved through the use of data augmentation methods or generative models (Tobin et al., 2017; Peng et al., 2018; Tremblay et al., 2018; Volpi et al., 2018; Zhang et al., 2017; Xu et al., 2020; Yan et al., 2020; Wang et al., 2020; Yu et al., 2023).

Representation Learning. Representation learning is another branch of methods that focuses on training an encoder that maps samples to a latent space where the embedding remains invariant to various domains (Arjovsky et al., 2019; Sagawa et al., 2019; Huang et al., 2020; Li et al., 2018a;b; Ganin et al., 2016; Cha et al., 2022). Alternative approaches for achieving invariant learning have been proposed, including techniques such as correlation alignment (Sun & Saenko, 2016), class-conditional adversarial learning (Li et al., 2018c), minimizing maximum mean discrepancy (Li et al., 2018d), and mutual information regularization (Cha et al., 2022) that doesn't require domain labels.

Ensemble Learning. There are some ensemble approaches for domain generalization, which train multiple models and then combine the predictions of these models at validation time to obtain a most generalization model. For instance, SWAD (Cha et al., 2021) aims to find a flatter minima and suffers less from overfitting than vanilla SWA (Izmailov et al., 2018) by a dense and overfit-aware stochastic weight sampling strategy; EoA (Arpit et al., 2022) finds that an ensemble of moving average models outperforms a traditional ensemble of unaveraged models.

C Datasets Details

To compare the efficacy of our proposed framework with existing algorithms, we conduct our experiments on 5 real-world benchmark datasets: PACS (Li et al., 2017), Office-Home (Venkateswara et al., 2017), VLCS (Fang et al., 2013), Terra Incognita (Beery et al., 2018), and DomainNet (Peng et al., 2019). Specifically, PACS includes four image styles (Photo, Art, Cartoon, and Sketch), which are considered 4 different domains, and each domain has 7 classes of images (Dog, Elephant, Giraffe, Horse, Person, Guitar, and House) for training and testing. It contains a total of $9,991$ instances in 4 domains. Office-Home consists of 65 classes of images for training and testing. These images belong to four image styles (Art, Clipart, Product, Real) being considered as 4 different domains. It contains a total of $15,588$ instances in 4 domains. VLCS includes images collected from 4 different datasets (Caltech101, LabelMe, SUN09, and VOC2007), which are considered 4 different domains, and each domain has 5 classes (Dog, Bird, Person, Car, and Chair) for training and testing. It contains a total of $10,729$ instances in 4 domains. Terra Incognita consists of 10 classes of photographs of wild animals taken at 4 different locations (Location 100, Location 38, Location 43, and Location 46), considered as 4 different domains. For our experiments, we use the downloader of DomainBed (Gulrajani & Lopez-Paz, 2020) to download the same version Terra Incognita dataset as theirs. It contains a total of $24,788$ instances in 4 domains. DomainNet includes 6 image styles (Clipart, Infograph, Painting, Quickdraw, Real, Sketch) considered as 6 different domains. In each domain, there are 345 classes for training and testing. It contains a total of $586,575$ instances in 6 domains. DomainNet can be considered a larger-scale dataset with a more difficult multi-classification task than the other 4 benchmarks.

D Baselines and Implementation Details

Baselines. To fairly compare our proposed framework with existing algorithms, we follow the settings of DomainBed (Gulrajani & Lopez-Paz, 2020) and DeepDG (Wang et al., 2022), using the best result between DomainBed, DeepDG, and the original literature. The comparisons include 12 baseline algorithms: ERM (Vapnik, 1999), IRM (Arjovsky et al., 2019), DRO (Sagawa et al., 2019), RSC (Huang et al., 2020), Mixup (Wang et al., 2020), MLDG (Li et al., 2018a), CORAL (Sun & Saenko, 2016), MMD (Li et al., 2018b), DANN (Ganin et al., 2016), C-DANN (Li et al., 2018d), DA-ERM (Dubey et al., 2021), and MIRO (Cha et al., 2022). Considering that domain labels can be leveraged as additional information for learning representations mitigating domain-specific features

Table 5: Hyper-parameters of DISPEL based on ERM.

	PACS	Office-Home	VLCS	TerraInc	DomainNet
DNN Architecture: ResNet-18					
Batch size	128	128	128	128	128
Learning rate	1×10^{-3}	1×10^{-3}	3×10^{-4}	1×10^{-4}	1×10^{-4}
τ	0.1	0.1	0.1	0.1	0.1
DNN Architecture: ResNet-50					
Batch size	64	64	64	64	64
Learning rate	5×10^{-5}	1×10^{-3}	1×10^{-3}	2×10^{-4}	1×10^{-4}
τ	0.1	0.1	0.1	0.1	0.1

projected to embedding space, we categorize the 12 baseline algorithms into two groups: **Group 1**: the algorithms requiring domain labels (Mixup, MLDG, CORAL, MMD, DANN, C-DANN, and DA-ERM); and **Group 2**: the algorithms without requiring domain labels (ERM, IRM, DRO, RSC, and MIRO).

Implementation. All the experimental results of the proposed DISPEL are implemented and performed based on the codebase of DeepDG (Wang et al., 2022). Unlike DomainBed (Gulrajani & Lopez-Paz, 2020), our implementation does not use any data augmentation during training. Regarding the setting of model selection, we use traditional *training-domain validation set* for our implementation, which does not require utilizing domain labels to split the desired validation set. For all the experimental results of DISPEL, we employ ERM algorithm to fine-tune the ResNet-18 and ResNet-50 as the fine-tuned model mentioned in Sec. 3.1. Concerning the use of the EMG, we utilize ResNet50 as the base model for EMG since the 5 domain generalization benchmarks we tested are image datasets.

DNN Architectures. The experimental results are all fine-tuned on the basis of ResNets. Since larger ResNets are known to have better generalization ability, we mainly conduct experiments with ResNet-50 models for all 5 benchmark datasets, and we also conduct the results of DISPEL based on ResNet-18 as a reference shown in Tab. 7. For both the two base network architectures, we both use the ResNet-18 and ResNet-50 pre-trained on ImageNet. As for the EMG component in DISPEL, we employ a ResNet50 pre-trained on ImageNet as the base model.

Table 6: Hyper-parameters of DISPEL for boosting other algorithms, where the DNN architecture is ResNet-50.

	DRO	CORAL	DANN	Mixup
Dataset: PACS				
Batch size	64	64	64	64
Learning rate	1×10^{-3}	1×10^{-3}	5×10^{-3}	5×10^{-4}
τ	0.1	0.1	0.1	0.1
Dataset: Office-Home				
Batch size	64	64	64	64
Learning rate	1×10^{-3}	1×10^{-3}	1×10^{-3}	1×10^{-3}
τ	0.1	0.1	0.1	0.1

D.1 Hyper-parameters of DISPEL

In the proposed DISPEL framework, the hyper-parameters are composed of batch size, learning rate, and τ in Eq. 1, where τ is the only hyper-parameter that is related to our algorithm. The hyper-parameters of DISPEL for each benchmark dataset are shown in Tab. 5.

D.2 Hyper-parameters of DISPEL for Boosting Other Algorithms

As shown in Sec. 4.4, we leverage our DISPEL to further improve the prediction performance on unseen test domain for four existing domain generalization algorithms on PACS and Office Home,

Table 7: Each unseen test domain accuracy of DISPEL.

Dataset: PACS						
	Art Painting	Cartoon	Photo	Sketch	-	-
DISPEL (ResNet-18)	83.6 ± 0.3	79.0 ± 0.2	97.0 ± 0.0	81.8 ± 0.0	-	-
DISPEL (ResNet-50)	87.1 ± 0.1	82.5 ± 0.0	98.0 ± 0.1	85.2 ± 0.1	-	-
Dataset: Office-Home						
	Art	Clipart	Product	Real	-	-
DISPEL (ResNet-18)	61.4 ± 0.0	53.9 ± 0.2	76.0 ± 0.1	77.8 ± 0.0	-	-
DISPEL (ResNet-50)	71.3 ± 0.5	59.4 ± 0.4	80.3 ± 0.3	82.1 ± 0.0	-	-
Dataset: VLCS						
	Caltech101	LabelMe	SUN09	VOC2007	-	-
DISPEL (ResNet-18)	97.2 ± 0.0	62.6 ± 0.1	75.0 ± 0.1	76.9 ± 0.1	-	-
DISPEL (ResNet-50)	98.3 ± 0.4	65.3 ± 0.1	77.2 ± 0.1	76.3 ± 0.1	-	-
Dataset: Terra Incognita						
	Location 100	Location 38	Location 43	Location 46	-	-
DISPEL (ResNet-18)	44.4 ± 0.4	49.6 ± 0.7	48.1 ± 0.2	37.3 ± 0.1	-	-
DISPEL (ResNet-50)	54.7 ± 0.3	48.1 ± 0.0	56.3 ± 0.3	42.3 ± 0.2	-	-
Dataset: DomainNet						
	Clipart	Infograph	Painting	Quickdraw	Real	Sketch
DISPEL (ResNet-18)	44.6 ± 0.0	14.2 ± 0.0	39.7 ± 0.0	10.3 ± 0.0	45.6 ± 0.0	40.8 ± 0.0
DISPEL (ResNet-50)	63.4 ± 0.0	20.1 ± 0.1	48.2 ± 0.0	14.2 ± 0.0	63.4 ± 0.0	54.9 ± 0.0

Table 8: Each unseen test domain accuracy comparisons of Terra Incognita (ResNet50).

	Location 100	Location 38	Location 43	Location 46
Group 1: algorithms requiring domain labels				
Mixup (Wang et al., 2020)	**60.6** ± 1.3	41.1 ± 1.8	58.5 ± 0.8	35.2 ± 1.1
MLDG (Li et al., 2018a)	48.5 ± 3.3	42.8 ± 0.4	56.8 ± 0.9	36.3 ± 0.5
CORAL (Sun & Saenko, 2016)	48.6 ± 0.9	42.2 ± 3.5	55.9 ± 0.6	38.7 ± 0.7
MMD (Li et al., 2018b)	52.2 ± 5.8	47.0 ± 0.6	**57.8** ± 1.3	40.3 ± 0.5
DANN (Ganin et al., 2016)	49.0 ± 3.8	46.3 ± 1.7	57.6 ± 0.8	40.6 ± 1.7
C-DANN (Li et al., 2018d)	49.5 ± 3.8	44.8 ± 1.0	57.3 ± 1.1	38.8 ± 1.7
Group 2: algorithms without requiring domain labels				
ERM (Vapnik, 1999)	50.8 ± 0.2	42.5 ± 0.2	**57.9** ± 1.3	37.6 ± 1.3
IRM (Arjovsky et al., 2019)	44.2 ± 2.7	41.3 ± 0.6	54.3 ± 0.2	36.0 ± 1.7
DRO (Sagawa et al., 2019)	31.8 ± 0.3	43.7 ± 1.2	58.0 ± 0.7	36.6 ± 1.3
RSC (Huang et al., 2020)	50.2 ± 2.2	39.2 ± 1.4	56.3 ± 1.4	40.8 ± 0.6
DISPEL	54.7 ± 0.3	**48.1** ± 0.0	56.3 ± 0.3	**42.3** ± 0.2

where all the DNN architectures are ResNet-50. The hyper-parameters of the DISPEL derivative models for the two datasets are shown in Tab. 6.

E Experimental Results of DISPEL

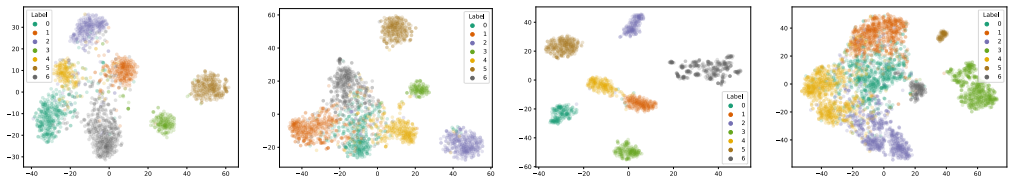
To closely investigate the fine-grained behavior of DISPEL in Sec. 4.3, we observe the prediction accuracy in each unseen test domain of all five domain generalization benchmark datasets. In Tab. 7, we show the experimental results of DISPEL on each unseen domain of five domain generalization benchmark datasets based on the two DNN architectures, ResNet-18 and ResNet-50. Based on the experimental results on each unseen domain, we conclude the **Observation 2: DISPEL possesses stable generalizing efficacy.** The results show that DISPEL maintains its stable efficacy in improving generalization ability over more different data distributions in more diverse classes of data. And these results reflect the purpose of the EMG module that considers each instance for fine-grained domain-specific feature masking.

F VISUALIZATION ANALYSIS VIA T-SNE

To illustrate how DISPEL improves generalization by blocking domain-specific features in the embedding space, we use t-SNE in the unseen test domains of all five benchmark datasets by comparing the embedding with and without DISPEL, as shown in Fig. 6 to Fig. 13. The key observation is that DISPEL aims to make each class more concentrated and separate them better. Taking PACS as an example, by drawing down more precise decision boundaries, the predictor can achieve better accuracy in the unseen *Art Painting* domain, in which DISPEL enhances the most accuracy among the 4 domains as shown in Tab. 2. Even in *Cartoon* domain where DISPEL only raises 0.7% accuracy, it shows the same intention to concentrate the embedding distribution for each class in Fig. 6-(b) and Fig. 7-(b). As for the unseen *Photo* domain, the base algorithm ERM has performed 96.7% accuracy, which means that Fig. 6-(c) reveals what a high-quality representation looks like. Compared to Fig. 7-(c), DISPEL follows the initial distribution and ameliorates the embedding to compress the distributions of each class.

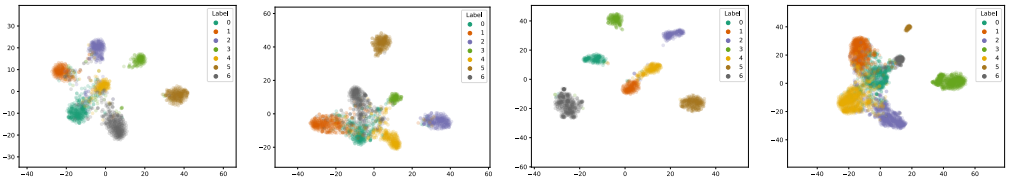
Investigating the embedding of Terra Incognita, a more difficult multi-class classification task dataset, we observe the coherent behavior of DISPEL to its manner in PACS. As shown in Fig. 12-(a)(b)(d) and Fig. 13-(a)(b)(d), DISPEL has the same effect as on *Cartoon* and *Sketch* domain of PACS, which is to reduce the length of decision boundaries between different classes by concentrating distribution of each class. In addition, as shown in Tab. 8, *Location 43* is the only domain in which DISPEL cannot improve its classification accuracy. However, the reason is that we cannot achieve our reproduced ERM the same performance as provided in DomainBed (Gulrajani & Lopez-Paz, 2020), and the accuracy of our reproduced ERM in the *Location 43* domain is 55.1%. Therefore, DISPEL actually improves the accuracy in this unseen test domain by 1.3%. As we can see in Fig. 12-(c) and Fig. 13-(c), each class's instance embedding is concentrated after employing DISPEL as in other domains.

Based on the t-SNE visualization analysis, we conclude **Observation 3: DISPEL concentrate the distribution of each class embedding.** The t-SNE analysis demonstrates the superiority of DISPEL, which improves the domain generalization ability of the fine-tuned ERM by concentrating the distribution of embeddings in the same class.



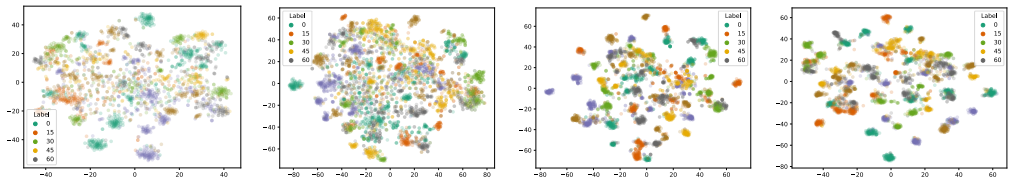
(a) Art Painting (b) Cartoon (c) Photo (d) Sketch

Figure 6: t-SNE visualization of ERM embedding in four unseen test domains of PACS.



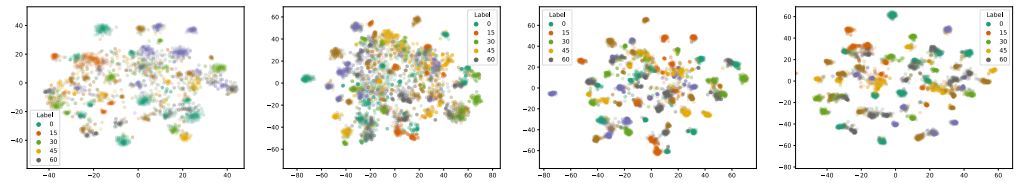
(a) Art Painting (b) Cartoon (c) Photo (d) Sketch

Figure 7: t-SNE visualization of DISPEL embedding in four unseen test domains of PACS.



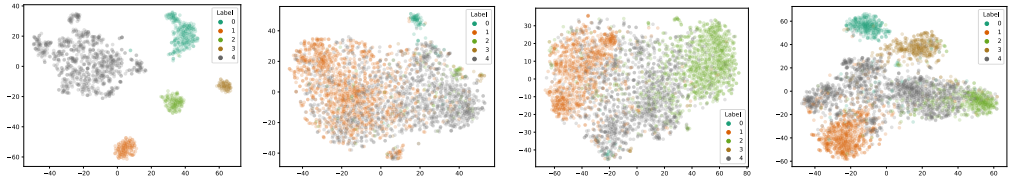
(a) Art (b) Clipart (c) Product (d) Real

Figure 8: t-SNE visualization of ERM embedding in four unseen test domains of Office-Home.



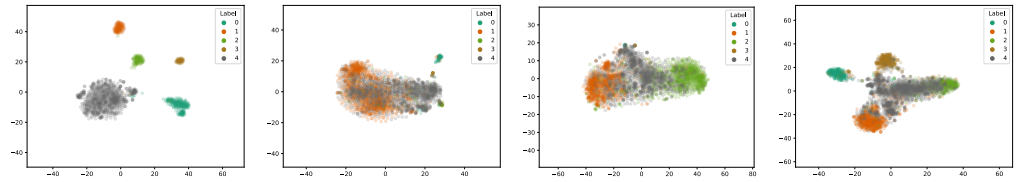
(a) Art (b) Clipart (c) Product (d) Real

Figure 9: t-SNE visualization of DISPEL embedding in four unseen test domains of Office-Home.



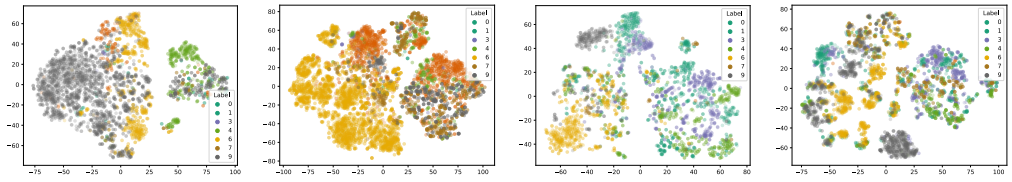
(a) Caltech101 (b) LabelMe (c) SUN09 (d) VOC2007

Figure 10: t-SNE visualization of ERM embedding in four unseen test domains of VLCS.



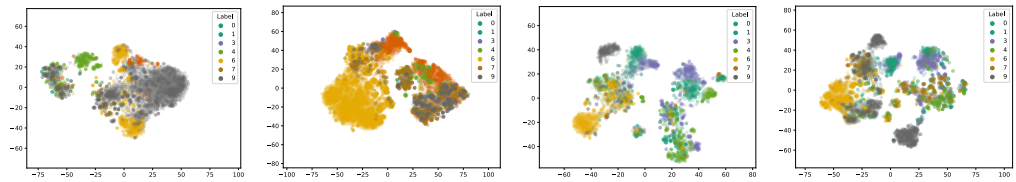
(a) Caltech101 (b) LabelMe (c) SUN09 (d) VOC2007

Figure 11: t-SNE visualization of DISPEL embedding in four unseen test domains of VLCS.



(a) Location 100 (b) Location 38 (c) Location 43 (d) Location 46

Figure 12: t-SNE visualization of ERM embedding in four unseen test domains of Terra Incognita.



(a) Location 100 (b) Location 38 (c) Location 43 (d) Location 46

Figure 13: t-SNE visualization of DISPEL embedding in four unseen test domains of Terra Incognita.