

# Domain Prompt Learning for Efficiently Adapting CLIP to Unseen Domains

Xin Zhang,<sup>1</sup> Shixiang Shane Gu<sup>1,2</sup> Yutaka Matsuo,<sup>1</sup> Yusuke Iwasawa,<sup>1</sup>

<sup>1</sup> The University of Tokyo

<sup>2</sup> Google Research, Brain Team

## Abstract

Domain generalization (DG) is a difficult transfer learning problem aiming to learn a generalizable model for unseen domains. Recent foundation models (FMs) are robust to many distribution shifts and, therefore, should substantially improve the performance of DG. In this work, we study generic ways to adopt CLIP, a Visual-Language Foundation Model, for DG problems in image classification. While ERM greatly improves the accuracy with bigger backbones and training datasets using standard DG benchmarks, fine-tuning FMs is not practical in many real-world situations. We propose DPL (Domain Prompt Learning) as a novel approach for domain inference in the form of conditional prompt generation. DPL achieved a significant accuracy improvement with only training a lightweight prompt generator (a three-layer MLP), whose parameter is of equivalent scale to the classification projector in the previous DG literature. Combining DPL with CLIP provides surprising performance, raising the accuracy of zero-shot CLIP from 73.7% to 79.3% on several standard datasets, namely PACS, VLCS, OfficeHome, and TerraIncognita. We hope the simplicity and success of our approach lead to broader adoption and analysis of foundation models in the domain generalization field. Our code is available at <https://github.com/shogi880/DPLCLIP>

## 1 Introduction

Pre-training large vision models using web-scale images is an essential ingredient of recent success in computer vision. Fine-tuning pre-trained models, such as ResNet (He et al. 2015) and Vision Transformer (ViT) (Dosovitskiy et al. 2020) is the most popular paradigm for many downstream tasks. However, domain shifts pose a substantial challenge in real-world scenarios for successfully transferring models. Over the past decade, various studies on domain generalization (DG) have sought a systematic way to narrow the gap between source and target domains (Zhou et al. 2021a; Wang et al. 2021; Shen et al. 2021) aiming to build a model that generalizes to unseen domains. Despite the significant work on this front, machine learning systems are still vulnerable to domain shifts even after using DG methods (Gulrajani and Lopez-Paz 2020).

Large pre-trained vision-language models like Contrastive Language-Image Pre-Training (CLIP) are an emerging category of models showing great potential in learning

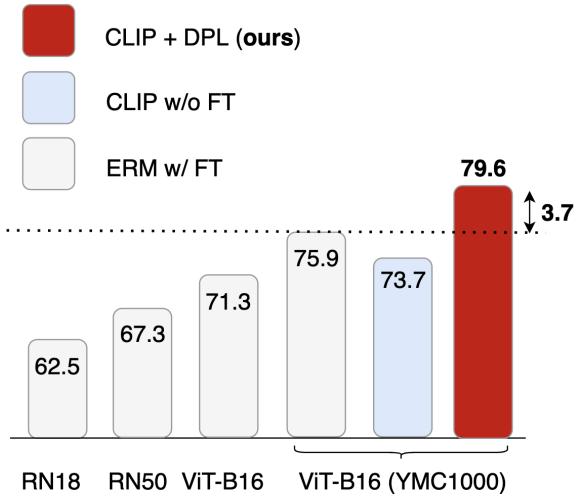


Figure 1: Bigger backbone (from ResNet18 to ViT-B16) and bigger pre-train dataset (from ImageNet to YMC1000) improve the performance of ERM on VLCS, PACS, OfficeHome, TerraIncognita. Even without fine-tuning the image encoder, our DPL (Domain Prompt Learning) effectively improves the performance of CLIP and outperforms the baseline ERM by a large margin (3.7%). The CLIP w/o FT use the template prompt, such as ‘a photo of a {class name}’.

transferable representation across many vision tasks. At the core of CLIP is to learn image representations by contrasting them with the representations of text description of the image, such as ‘a photo of a {class name}’. The text description is often called *prompt*, and its design is vital in enhancing CLIP performance. Notably, CLIP can handle unseen classes without fine-tuning them by adequately changing the text description using the target class name.

This paper investigates the robustness of CLIP against various distribution shifts using DomainBed (Gulrajani and Lopez-Paz 2020), a recently proposed benchmark for DG setup. While prior works test various DG methods in the benchmark, the most studied only focused on medium-scale pre-trained models, such as ResNet18 or ResNet50. There are two naïve approaches to leveraging CLIP in the DG setup Figure 2. The first approach is fine-tuning the image

encoder trained by CLIP, similar to the other vision models such as ResNet and ViT. We show that the backbone networks trained by CLIP substantially outperform many backbone networks trained solely on images, such as ResNet, big transfer (Kolesnikov et al. 2020), and vision transformer (Dosovitskiy et al. 2020). At the same time, however, fine-tuning sometimes degraded the performance on some domains, suggesting that fine-tuning possibly distorts good properties of pre-trained features (Kumar et al. 2022). Another naïve approach is designing the template prompt, such as ‘a photo of a {class name}’. The clear merit of this approach is that it does not require optimizing any network and, therefore, keeps the representations learned via pre-training. Despite its simplicity, we show that zero-shot CLIP is still more robust on many DG benchmarks than the vision backbones (e.g., ResNet18, ResNet50, ViT-B16) fine-tuned on source domains, while it is inferior to fine-tuning vision backbone trained by CLIP.

Based on the observations, we propose Domain Prompt Learning (DPL), a simple yet effective extension of CLIP in the DG setup. A natural way to adapt the model is to add domain-specific features to the prompt template. However, manually designing a prompt template is challenging in many cases due to its ambiguity. Instead, we propose DPL for automatically generating a prompt that estimates domain-specific features given unlabeled examples from each distribution. More specifically, DPL trains a lightweight prompt generator using source domains, which outputs fixed-length continuous domain prompts given input images of each distribution while freezing other networks. During test-time, the prompt generator generates domain prompt given input images from the target distribution and adds them to the label prompts. Since the entire networks are frozen, the core properties of the pre-training would remain in DPL and are expected to improve CLIP performance in DG stably, as shown in our experiments.

It is worth noting our work is not the first attempt to tune the prompt of CLIP. For example, (Gao et al. 2021; Zhou et al. 2021b) have proposed optimizing continuous prompts on the target datasets, effectively improving CLIP performance. CoCoOp (Zhou et al. 2022), as a contemporary work, trains a meta-net to generate a meta token for adapting to each instance. CoCoOp focuses on unseen classes and demonstrates its performance by transferring from ImageNet to the four specially designed ImageNet variants. This work focuses on the robustness of CLIP against distribution shifts, and proposes a generic way to extract a domain-specific features and improve performance on the target domain at test-time.

We conduct experiments on four standard datasets included in DomainBed to evaluate DPL, following the experiment setup in (Gulrajani and Lopez-Paz 2020; Iwasawa and Matsuo 2021), such as parameter tuning and model selection. We show that CLIP with DPL outperforms the strong baselines by a large margin, raising the accuracy from 73.7% to 79.6% (Table 1). Moreover, since DPL can be seen as a kind of Test-Time Adaptation (TTA) method, we compare it with a series of SoTA TTA methods and demonstrate the efficiency of DPL (Table 2). And lastly, through various ab-

lation studies, we surprisingly found that frozen backbone outperforms fine-tuning on OfficeHome datasets for all of ResNet, DeiT (Touvron et al. 2021), HViT, and ViT-B16 (Table 4). These results prove that DPL is effective, and more importantly, they provide many insights for future works that apply CLIP on DG.

In summary, our main contributions are:

1. We introduce CLIP to standard DG benchmark DomainBed via prompt learning.
2. We propose Domain Prompt Learning (DPL), a novel approach of domain inference, to effectively help domain generalization by utilizing domain-specific features.
3. We demonstrate the impressive empirical performance of DPL by comparing with strong DG baselines and a series of state-of-the-art (SoTA) TTA methods.

## 2 Related Work

### 2.1 Domain Generalization

Over the past decade, various approaches have been proposed to solve DG. Most prior works have focused on regularizing the model using the knowledge from multiple source domains. For example, domain-invariant representation learning (Ganin et al. 2016a) is a major branch of domain generalization, aiming to reduce the domain gaps in the space of latent representations. There are many different approaches to measure the domain gaps, including adversarial classifier (Li et al. 2018; Ganin and Lempitsky 2015; Ganin et al. 2016a), kernel mapping (Blanchard, Lee, and Scott 2011; Grubinger et al. 2015), metric learning (Motiian et al. 2017; Jin et al. 2020), and invariant risk minimization (Arjovsky et al. 2020). Similarly, several researchers have sought to generate samples with diverse styles so that models can learn domain-invariant features through them (Shankar et al. 2018; Zhou et al. 2020; Borlino, D’Innocente, and Tommasi 2021). Other methods use meta learning to learn how to regularize the model to improve robustness (Dou et al. 2019; Li et al. 2017b).

Our work investigates the importance of the CLIP (Radford et al. 2021) in DG, and proposes a lightweight way to adapt the CLIP for unseen domains. There are several recent observations to motivate us to benchmark CLIP in the DG setup. First, (Gulrajani and Lopez-Paz 2020) shows that many prior approaches do not provide significant improvement compared to simple supervised learning. The results imply that regularizing the model is not sufficient to achieve high performance in DG. Secondly, despite significant related works, most studies have focused on medium-scale pre-trained models, such as ResNet18 or ResNet50, although very large-scale models often lead to substantial improvements. Notably, the latest work (Iwasawa and Matsuo 2021) compares more large-scale backbone networks, including big transfer (Kolesnikov et al. 2020) (BiT-MR50x3, BiT-M-R101x3, and BiT-M-R152x4), vision transformer (ViTB16 and ViT-L16 (Dosovitskiy et al. 2020), Hybrid ViT, DeiT (Touvron et al. 2021)), and MLP-Mixer (Tolstikhin et al. 2021) (Mixer-L16), and shows that the selection of backbone networks is important in DG. In contrast

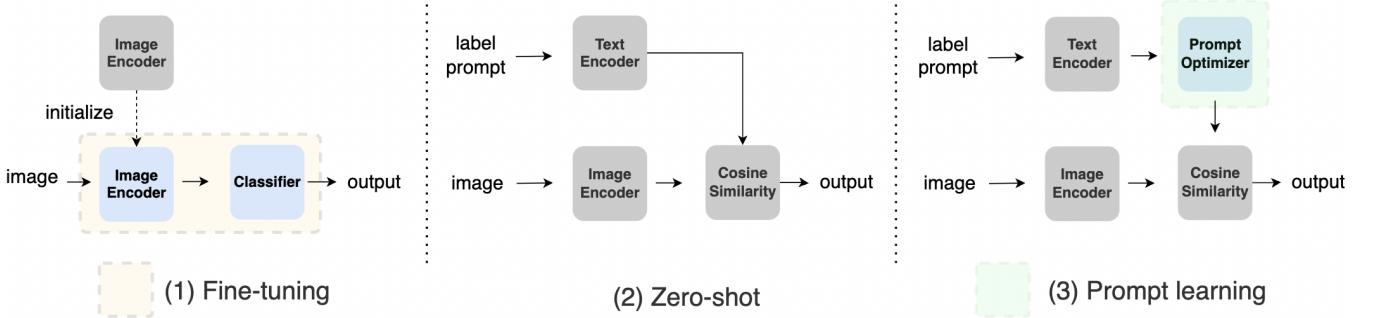


Figure 2: The concept illustration of three approaches to apply CLIP in DG. (1) Fine-tuning updates the CLIP’s image encoder with a trainable classifier. (2) Zero-shot CLIP contrastive prediction with hand-craft prompts at the test time without updating parameters on the train domains. (3) Prompt learning trains a prompt optimizer then utilize the optimized prompts to prediction. Our DPL is categorized to (3) Prompt learning, which trains a prompt generator in train phase and infers unseen domain to generate a domain-specific prompt.

with (Iwasawa and Matsuo 2021), we herein demonstrate that CLIP performs surprisingly well without fine-tuning the entire model in source domains, which is time-consuming in practice.

From the methodological perspective, our work relates to several prior works that have attempted leveraging domain features rather than discarding them (Ganin et al. 2016a; Zhou et al. 2020; Borlino, D’Innocente, and Tommasi 2021). While these works focused on the standard vision backbone, we propose a CLIP-specific approach to leverage the domain features by combining these features with prompt tuning.

## 2.2 Test Time Adaptation

Regarding the problem setup, our work can also be seen as Test-Time Adaptation (TTA). The concept of TTA is updating a part of networks to minimize the prediction entropy for adapting the model to an unseen domain robustly at the test time. Pseudo Label (Lee et al. 2013) updates entire networks and Tent (Wang et al. 2020) updates the BN parameters. SHOT(Liang, Hu, and Feng 2020) update feature extractor and minimizes a diversity regularizer and pseudo-label loss, not only prediction entropy. Instead of minimizing prediction entropy at the test time, we infer domain information and generate a domain-specific prompt to adapt CLIP to an unseen target domain.

Our work also relates to (Iwasawa and Matsuo 2021) in that both approaches modulate their prediction given the unlabeled data available at test time. Specifically, (Iwasawa and Matsuo 2021) proposes T3A that replaces the linear classifier using pseudo-labeling and prototypical classification and shows that it stably improves the performance in unseen domains. However, T3A cannot be directly applied to CLIP, as it assumes a simple linear classifier that CLIP does not employ.

## 2.3 Prompt Learning

The success of GPT-3 demonstrated the importance of prompt tuning. There are various prompting strategies, such as discrete natural language prompts and continuous

prompts (Liu et al. 2021a). PADA (Ben-David, Oved, and Reichart 2021) proposed a domain adaptation algorithm that trains T5 (Raffel et al. 2019), a language foundation model, to generate unique domain-relevant features for each input. PADA uses discrete prompts for the NLP applications and differs from our DPL with continuous prompts in computer vision. On the other hand, many recent works (Li and Liang 2021; Lester, Al-Rfou, and Constant 2021) directly tuning prompts in continuous vector forms, and P-Tuning v2 (Liu et al. 2021b) showed that continuous prompt tuning achieves the same performance as fine-tuning in various settings.

Because of the successful applications of CLIP, prompt tuning is also of great interest in computer vision. Context Optimization (CoOp (Zhou et al. 2021b)) demonstrated that CLIP performance is susceptible to prompts and that a suitable prompt can improve performance for the image recognition task. CLIP-Adapter (Gao et al. 2021) was proposed to learn with an additional adapter network. (Ge et al. 2022) adapts CLIP using contrastive learning in the Unsupervised Domain Adaption setup. Unlike these works, which need to access the image or class labels in the target domain, we adapt CLIP to an unseen domain with a generated domain prompt inferred from input images.

## 3 Method

In this section, we first introduce the notations and definitions of DG following (Wang et al. 2021). Then, we explain how to use CLIP in DG and introduce Domain Prompt Learning to enhance CLIP performance in DG.

### 3.1 Problem Setup of DG

Let  $\mathcal{X}$  denote an input space and  $\mathcal{Y}$  an output space. A domain is composed of data that has been sampled from a distribution. We denote the datasets from distribution as  $\mathcal{S}^i = (\mathbf{x}_j^i, y_j^i)_{j=1}^{n_i} \sim \mathcal{P}_{XY}^i$ , where  $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^d$  is an input image,  $y \in \mathcal{Y}$  denotes the class associated with  $\mathbf{x}$ , and  $\mathcal{P}_{XY}^i$  denotes the joint distribution of the sample and output label in the domain  $i$ .  $X, Y$  denote the corresponding random variables.

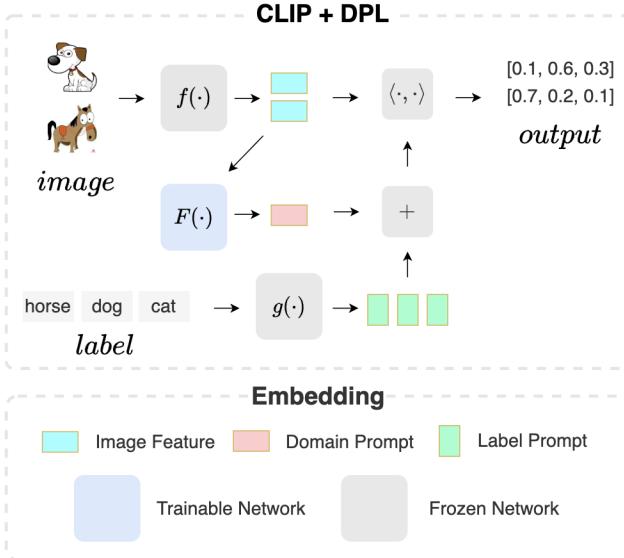


Figure 3: The architecture of CLIP + DPL. The only one network we trained is the prompt generator  $F(\cdot)$ , which is colored in blue. First, the input images are encoded to obtain image features with the frozen CLIP’s image encoder  $f(\cdot)$ . The image features are fed into the domain prompt generator  $F(\cdot)$  to generate a domain prompt. Simultaneously, all of labels are encoded using the frozen CLIP’s text encoder  $g(\cdot)$  to obtain the label prompt embeddings. Secondly, the domain prompt embeddings are added to the label prompt embeddings for calculating the similarity. Finally, to obtain the prediction output in probability, the cosine similarity  $\langle \cdot, \cdot \rangle$  are calculated with image embeddings and domain prompt embeddings.

In DG, we are interested in predictor  $h$  performance on data from an unseen domain  $\mathcal{P}_{XY}^t \neq \mathcal{P}_{XY}^i$  for all  $i$ . Prior works fine-tuned a pre-trained image encoder  $f$  (usually ResNet18 or ResNet50) in conjunction with a randomly initialized classification head  $g$  (linear classifier), using data from multiple different datasets to achieve the goal. Specifically, given  $M$  datasets  $\mathcal{S}^i$  collected from various domains  $i \in \{1, \dots, M\}$ ,  $f$  and  $g$  are updated by

$$\min_{f,g} \frac{1}{M} \sum_{i=1}^M \frac{1}{n_i} \sum_{j=1}^{n_i} \ell(g \circ f(x_j^i), y_j^i), \quad (1)$$

where  $\ell(\cdot)$  is a loss function. In the simplest case,  $\ell$  is a simple cross-entropy loss, and minimizing eq. 2 is called empirical risk minimization (ERM). As discussed in Section ??, different methods in DG use other loss functions by designing regularization terms to prevent overfitting specific domains. These datasets are frequently referred to as source domains, and they are distinguished as target domains where we want the model to perform well.

### 3.2 Naïve Approaches for Using CLIP in DG

CLIP consists of two parts: an image encoder  $f_{clip}$  and a language model  $g_{clip}$ . CLIP classifies the image features based

on the similarity between embedding of a text prompt  $p$ , such as ‘dog’ or ‘a photo of a class label,’ rather than initially using the classification head trained from scratch. Specifically, given an image  $x$  and  $K$  class prompt  $p_k$ , CLIP output a prediction using both  $f_{clip}$  and  $g_{clip}$ :

$$\hat{y}_{clip} = \arg \max_k \langle f_{clip}(x), g_{clip}(p_k) \rangle \quad (2)$$

where  $K$  is the number of categories and  $\langle \cdot, \cdot \rangle$  is cosine similarity.

To demonstrate how powerful the representation of massively pre-trained models (CLIP) for DG setup, we tested following two naïve approaches to use CLIP in DG setups: fine-tuning and zero-shot. Firstly, we evaluated CLIP in a zero-shot manner; i.e., we freeze both the image encoder and the language model, and substitute the class labels used in each dataset for the text prompt  $p$ .

Secondly, we can use the image encoder  $f_{clip}$  as an alternative to the standard image backbones, such as ResNet and ViT. In this setup, we train  $f_{clip}$  by using the datasets  $\mathcal{S}^i$  from multiple source domains  $i$ , similar to the standard DG setup. We can use any algorithms tailored for DG setup, such as DANN and CORAL during fine-tuning. While it is powerful as shown in experiments, it requires additional computational costs to re-train such large models entirely. Besides, good properties of massive pre-training might be distorted during fine-tuning, as highlighted by the performance degradation compared to zero-shot approach.

In summary, zero-shot approach is computationally effective yet less expressive, and fine-tuning can leverage the knowledge of source datasets but it is computationally heavy and possibly distort good representations learned during pre-training. Based on the observation, we propose a novel approach to design the prompt  $p$  to improve the performance in an unseen domain without fine-tuning the entire model.

### 3.3 Domain Prompt Learning for CLIP in DG

As discussed in Section 2.3, designing a prompt is a powerful approach to improve the performance of the transformer-based models. It is powerful and should also be easier to train because the dimension of prompts is significantly smaller than the entire parameters of  $f$  and  $g$ . For example, supposing we can access a supervised dataset from the target domain, we can optimize a prefix vector  $p_{pre}$  by simple supervised loss:

$$\min_{p_{pre}} \mathbb{E}_{x,y \sim \mathcal{S}} \ell(\hat{y}_{clip*}, y), \quad (3)$$

where  $\hat{y}_{clip*}$  is

$$\hat{y}_{clip*} = \arg \max_k \langle f_{clip}(x), g_{clip}(p_k^*) \rangle, \quad (4)$$

where  $p_k^*$  is a concatenation of trainable parameters  $p_{pre}$  and  $p_k$ . Particularly,  $g_{clip}$  outputs the fixed length vector regardless of the input dimension (i.e., size of  $p_k$ ). The size of  $p_k$  is a hyperparameter.

Unfortunately, this labeled training data for the target domain is unavailable in DG. Instead, we proposed DPL to replace the optimization process of  $p_{pre}$  in each domain by training novel prompt generators  $F(\cdot)$  that generate a

<b>DomainBed</b>	category	<b>VLCS</b>	<b>PACS</b>	<b>OfficeHome</b>	<b>Terra</b>	<b>Avg</b>
ERM (CLIP)	Fine-tuning	$82.7 \pm 0.3$	$92.9 \pm 1.9$	$78.1 \pm 2.1$	$50.2 \pm 1.7$	75.9
CORAL	Fine-tuning	$82.0 \pm 0.2$	$93.2 \pm 1.1$	$78.9 \pm 1.9$	$\mathbf{53.5 \pm 0.7}$	76.9
DANN	Fine-tuning	$\underline{83.2 \pm 1.2}$	$93.8 \pm 1.3$	$78.8 \pm 1.1$	$52.2 \pm 2.0$	77.0
CLIP	Zero-shot	$76.6 \pm 0.0$	$95.8 \pm 0.1$	$79.9 \pm 0.1$	$36.4 \pm 0.1$	72.2
CLIP (template prompt)	Zero-shot	$82.3 \pm 0.1$	$96.1 \pm 0.1$	$\underline{82.3 \pm 0.2}$	$34.1 \pm 0.1$	73.7
<b>CLIP + DPL (ours)</b>	Prompt learning	$\mathbf{84.3 \pm 0.4}$	$\mathbf{97.3 \pm 0.2}$	$\mathbf{84.2 \pm 0.2}$	$52.6 \pm 0.6$	<b>79.6</b>

Table 1: Comparison experiments on VLCS, PACS, OfficeHome, and TerraIncognita. The best results are in bold, and the second-best results are underlined. CLIP (standard template) indicates using ‘a photo of a {class name}’ prompt. Following the experiment setup in (Iwasawa and Matsuo 2021), reported results are the mean and std with seed={1, 2, 3}.

prompt  $\mathbf{p}_{pre}$  given small unlabeled images from a distribution. Specifically, we use a fully connected network  $F(\cdot)$  to generate a prompt  $p$  from input images:

$$\mathbf{p}_{ap}^i = \frac{1}{N} \sum_{j=1}^N F(f(x_j^i)), \quad (5)$$

where  $N$  is the batch size for each domain, and  $x_j^i$  denotes the images from the  $i$ -th distribution. Given a batch of data from multiple source distributions, we use the following loss function to optimize  $F$ :

$$\min_F \frac{1}{M} \sum_{i=1}^M \frac{1}{n_i} \sum_{j=1}^{n_i} \ell(\hat{y}_{ap}^i, y_j^i), \quad (6)$$

and

$$\hat{y}_{ap}^i = \arg \max_k \langle f_{clip}(\mathbf{x}^i), g_{clip}(\mathbf{p}_k^i) \rangle, \quad (7)$$

where  $\mathbf{p}_k^i$  is a concatenation of pre-defined  $\mathbf{p}_k$  and  $\mathbf{p}_{ap}^i$ . The architecture of CLIP + DPL is depicted in Figure 3.

## 4 Experiment

In this section, we experimentally demonstrate the effectiveness of DPL. First, we clarify the important DG settings, including the datasets, hyperparameters, model selection strategy, and other implementation details. Second, we show CLIP + DPL outperforms the strong DG baselines and several SoTA TTA methods on DomainBed benchmark. Finally, our ablation experiments, including variants backbone comparison and different prompt strategies study, provide meaningful insights of applying CLIP + DPL to DG.

**Datasets** Following (Iwasawa and Matsuo 2021), we selected four real-world datasets from DomainBed benchmark, including VLCS (Fang, Xu, and Rockmore 2013), PACS (Li et al. 2017a), OfficeHome (Venkateswara et al. 2017), TerraIncognita (Beery, van Horn, and Perona 2018). More details are provided in Appendix A.

**Hyperparameters and model selection.** We set up experiments on DomainBed<sup>1</sup>, and implemented DPL based

on CLIP<sup>2</sup>. We strictly followed the basic selection criterion (Gulrajani and Lopez-Paz 2020) and selected the hyperparameters using standard training-domain validation. First, we split the data of each domain into 80% and 20% for the training model and select hyperparameters. Then, we ran 20 trials at random across a joint distribution of all hyperparameters. Next, we ran three trials of each hyperparameter setting, reserving one domain for testing and the rest for training. Finally, we selected the hyperparameters that maximize validation accuracy across the training domains and reported overall accuracy averaged across all three trials.

**Detail of the implements** As shown as in Figure 3, we only trained a three-layer MLP as the domain prompt generator. We used stochastic gradient descent (Bottou 2012) with momentum as an optimizer. Refer to our source code for implement details.

### 4.1 Comparison with existing DG methods

**Baselines** We compared our method to domain generalization algorithms, which fine-tune image features, and the handcrafted prompt for CLIP. For DG, we trained CLIP image features (ViT-B16) using ERM, CORAL (Sun and Saenko 2016), and DANN (Ganin et al. 2016b). Note that, as (Gulrajani and Lopez-Paz 2020) pointed out, ERM is a strong DG baseline when the experiments are fairly performed. For handcrafted prompt, we adopted three types prompt for CLIP including ‘{class name}’, template prompt ‘a photo of a {class name}.’, and Domain Prompt Learning ‘ $v_1, v_2, \dots, v_n$  {class name}.’.

All experiments listed in Table 1 are based on the CLIP ViT-B16 backbone. We observed that zero-shot CLIP could achieve an average of 72.2% accuracy and an average of 73.7% by using a template prompt. Notably, DPL improves CLIP performance to 79.6% and outperforms all baselines, although ERM, CORAL, and DANN are fine-tuning their image encoder. Based on this result, we infer that DPL should be an effective method in DG.

Surprisingly, we found that fine-tuning the backbones hurts performance on PACS and OfficeHome. We consider that fine-tuning causes the model to overfit in the source domain in the case where the pre-training domain is big enough to cover the target domain. On the other hand, the

<sup>1</sup><https://github.com/facebookresearch/DomainBed>

<sup>2</sup><https://github.com/openai/CLIP>

Methods	VLCS	PACS	OfficeHome	Terra	Avg
ERM	$81.4 \pm 0.3$	$91.9 \pm 0.7$	$78.4 \pm 1.1$	$47.8 \pm 3.1$	74.9
+T3A	<b><math>82.2 \pm 0.1</math></b>	$88.2 \pm 0.0$	$76.9 \pm 0.9$	$48.2 \pm 3.2$	73.9
+Pseudo Label	$81.1 \pm 1.0$	$92.1 \pm 0.4$	$78.0 \pm 2.5$	<b><math>50.5 \pm 4.4</math></b>	<u>75.4</u>
+Pseudo Label*	<u><math>82.1 \pm 0.4</math></u>	$87.6 \pm 0.0$	$76.6 \pm 1.9$	$46.9 \pm 3.1$	73.3
+Tent*	<b><math>82.2 \pm 0.4</math></b>	$87.8 \pm 0.0$	$76.5 \pm 1.2$	$46.7 \pm 3.2$	73.3
+SHOT	$80.4 \pm 0.4$	<u><math>93.8 \pm 0.8</math></u>	<u><math>80.7 \pm 0.8</math></u>	$40.5 \pm 1.7$	73.9
+SHOT**	$80.3 \pm 0.5$	$91.2 \pm 0.0$	$79.4 \pm 1.1$	$40.6 \pm 1.8$	72.9
CORAL	$81.2 \pm 0.3$	$91.1 \pm 1.9$	$78.7 \pm 0.8$	$48.6 \pm 2.9$	74.9
+T3A	$80.8 \pm 0.5$	$91.2 \pm 1.9$	$79.1 \pm 0.9$	$49.0 \pm 3.0$	75.0
+Pseudo Label	$80.0 \pm 1.4$	$93.1 \pm 2.0$	$79.8 \pm 1.2$	$44.5 \pm 3.3$	74.4
+Pseudo Label*	$81.4 \pm 0.1$	$91.2 \pm 1.9$	$78.8 \pm 0.9$	$48.6 \pm 2.9$	75.0
+Tent*	$81.3 \pm 0.2$	$91.2 \pm 1.9$	$78.6 \pm 0.8$	$48.7 \pm 2.6$	75.0
+SHOT	$78.7 \pm 1.9$	$93.0 \pm 1.2$	<u><math>80.7 \pm 0.9</math></u>	$41.9 \pm 2.0$	73.6
+SHOT**	$78.5 \pm 2.0$	$93.1 \pm 1.1$	<u><math>80.7 \pm 0.9</math></u>	$41.9 \pm 2.0$	73.5
<b>CLIP + DPL (ours)</b>	$81.0 \pm 1.1$	<b><math>95.9 \pm 0.0</math></b>	<b><math>82.3 \pm 0.7</math></b>	$49.4 \pm 1.1$	<b>77.2</b>

Table 2: Comparison with TTA methods. Here, \* indicates updating the linear classifier, and \*\* indicates updating the feature extractor to minimize entropy reported in table 3 of the T3A paper. The best results are in bold, and second-best results are underlined. All the experiments listed in this table are run on a cluster of A100 GPUs. The numbers of ERM, CORAL, and CLIP + DPL are different from Table 1 because of the use of half precision floating point on A100.

models perform better with fine-tuning on Terra, with a high likelihood of being not covered by the CLIP’s pre-training dataset. It is worth noting that our DPL can effectively trade-off well between both cases.

## 4.2 Comparison with existing TTA methods

DPL is generated by extracting domain features from a batch of input images. As discussed in subsection 2.2, DPL can be considered as a TTA method. Therefore, we performed a fair comparison with several TTA algorithms to validate DPL.

**Baselines** Following (Iwasawa and Matsuo 2021), we adopted the baselines including Pseudo Label, SHOT, Tent, and T3A with batch size equal to 64 during the test time. We trained all models with the same CLIP ViT-B16 backbone. All the experiments follow the model selection, hyperparameter selection strategy, and evaluation method proposed in T3A and DomainBed.

As shown in Table 2, DPL beats the most effective TTA methods on four datasets. The result demonstrates that DPL can consistently improve the model’s generalization performance at test time. We believe this is sufficient evidence that the central concept of DPL, extracting unseen domain features to help model adapting at the test time, is practical.

## 4.3 Backbone Ablation

**Different Backbones** Many proposed DG methods are evaluated using the standard ResNet backbones. However, more and more large models are being studied, and their validity is being experimentally demonstrated (Bommasani et al. 2021; Wang et al. 2022). Therefore, we reported the performance of ResNet18 and ResNet50, Mixer-16 (Tolstikhin et al. 2021), Vision Transformer (ViT) (Dosovitskiy et al. 2020) and several variations of ViT, such as BiT (Kolesnikov et al. 2020), DeiT (Touvron et al. 2021), HViT, and Mutual Information Regularization with Oracle (MIRO) (Cha et al. 2022) in Table 3.

As a result, we discovered that the CLIP ViT-B16 backbone trained on YFCC100M (Thomee et al. 2016) performs as well as HViT. Moreover, CLIP + DPL surpassed most of the backbones, including HViT and MIRO. Notably, DPL only trains a three-layer MLP, in contrast to others fine-tuning their backbones. We observed that the SoTA performance is provided by MIRO using RegNetY-16GF backbone with SWAG pre-training and combined with Stochastic Weight Averaging Densely (MIRO + SWAG (Singh et al. 2022) + SWAD (Cha et al. 2021)). The simple DPL can achieve close performance (difference of 1.9%). Although comparing with different pre-training datasets and different parameters is unfair, this result demonstrates that Domain Prompt Learning can efficiently adapt CLIP to unseen domains.

**Frozen Backbone** Fine-tuning a large model like CLIP or other Foundation Models necessitates much computing power. DPL also aims to adapt CLIP to the target domain with minimum computing. We wondered if simply training an MLP classifier with the frozen backbone could aid model transfer and conducted the ablation experiments with five different backbones.

From Table 4, we surprisingly found that Frozen ERM outperforms the standard ERM in OfficeHome with all the backbones. In VLCS, the performance of Frozen ERM is also unexpected. These results show that fine-tuning hurts the model more than expected on specific datasets. On the other hand, DPL steadily improving the performance on all datasets demonstrates the robustness of DPL.

Backbone Model	VLCS	PACS	OfficeHome	Terra	Avg
ResNet18 <sup>†</sup>	73.2 ± 0.9	80.3 ± 0.4	55.7 ± 0.2	40.7 ± 0.3	62.5
ResNet50 <sup>†</sup>	75.5 ± 0.1	83.9 ± 0.2	64.4 ± 0.2	45.4 ± 1.2	67.3
Mixer-L16 <sup>†</sup>	76.4 ± 0.2	81.3 ± 1.0	69.4 ± 1.6	37.1 ± 0.4	66.1
BiT-M-R50x3 <sup>†</sup>	76.7 ± 0.1	84.4 ± 1.2	69.2 ± 0.6	52.5 ± 0.3	70.7
BiT-M-R101x3 <sup>†</sup>	75.0 ± 0.6	84.0 ± 0.7	67.7 ± 0.5	47.8 ± 0.8	68.6
BiT-M-R152x2 <sup>†</sup>	76.7 ± 0.3	85.2 ± 0.1	71.3 ± 0.6	51.4 ± 0.6	71.1
ViT-B16 <sup>†</sup>	79.2 ± 0.3	85.7 ± 0.1	78.4 ± 0.3	41.8 ± 0.6	71.3
ViT-L16 <sup>†</sup>	78.2 ± 0.5	84.6 ± 0.5	78.0 ± 0.1	42.7 ± 1.9	70.9
DeiT <sup>†</sup>	79.3 ± 0.4	87.8 ± 0.5	76.6 ± 0.3	50.0 ± 0.2	73.4
HViT <sup>†</sup>	79.2 ± 0.5	89.7 ± 0.4	80.0 ± 0.2	51.4 ± 0.9	75.1
MIRO*	79.0 ± 0.0	85.4 ± 0.4	70.5 ± 0.4	50.4 ± 1.1	71.3
MIRO + SWAD*	79.6 ± 0.2	88.4 ± 0.1	72.4 ± 0.1	52.9 ± 0.2	73.3
MIRO + SWAG*	79.9 ± 0.6	<b>97.4 ± 0.2</b>	80.4 ± 0.2	<u>58.9 ± 1.3</u>	79.2
MIRO + SWAD + SWAG*	81.7 ± 0.1	96.8 ± 0.2	<u>83.3 ± 0.1</u>	<b>64.3 ± 0.3</b>	<u>81.5</u>
CLIP ViT-B16	82.7 ± 0.3	92.9 ± 1.9	78.1 ± 2.1	50.2 ± 1.7	75.9
<b>CLIP+DPL</b>	<b>84.3 ± 0.4</b>	<u>97.3 ± 0.2</u>	<b>84.2 ± 0.2</b>	52.6 ± 0.6	<u>79.6</u>

Table 3: Results of ERM with various backbone networks on DG benchmark. <sup>†</sup> indicates that the numbers are taken from Table 2 in (Iwasawa and Matsuo 2021). \* indicates the numbers are taken from MIRO (Cha et al. 2022). The best scores are bolded, and the second-best scores are underlined.

A similar phenomenon, fine-tuning does not constantly improve performance in DG, is also observed in subsection 4.1. Due to computing resource constraints, we only evaluated several backbones of varying sizes in this work. We found several recent studies analyzing the same phenomenon, the effect of pre-training datasets and backbones on DG and Out-of-Distribution settings(Kim et al. 2022; Wenzel et al. 2022).

## 5 Conclusions

We introduce CLIP to DG on DomainBed. For this purpose, we proposed a novel approach called Domain Prompt Learning (DPL) for efficiently adapting CLIP to an unseen domain. By generating the domain prompt conditional on input images, CLIP + DPL brings substantial improvements over strong DG baselines and several effective TTA methods on DomainBed. Then, we conducted ablation experiments with various backbones and Frozen ERM. We verified that DPL can stabilize performance and present meaningful insights about existing datasets and fine-tuning strategy of backbones. We hope that our research will broaden and inspire the roles of prompt learning in domain transfer learning.

### 5.1 Limitation

**Interpretability of Domain Prompt** To better perform, our DPL is directly represented in a continuous vector form, which lacks interpretability. However, improving interpretability is an important research direction in both FM applications and Domain Generalization. We consider producing discrete semantically informative prompts by some means is an exciting extension of DPL, even with some loss of precision.

**Label Shift** From the technical perspective, DPL cannot capture the domain shift outside of the images because DPL uses domain features extracted from only images. As a

Backbone	VLCS	PACS	OfficeHome	Terra	Avg
(1) Frozen	76.0 ± 0.3	66.0 ± 0.7	61.7 ± 0.5	25.5 ± 1.8	57.3
(2) ResNet18 <sup>†</sup>	73.2 ± 0.9	80.3 ± 0.4	55.7 ± 0.2	40.7 ± 0.3	62.5
(2) - (1)	<b>-2.8</b>	+14.3	<b>-6.0</b>	+15.2	+4.2
(1) Frozen	77.4 ± 0.3	67.2 ± 0.4	68.0 ± 0.3	35.4 ± 1.5	62.0
(2) ResNet50 <sup>†</sup>	75.5 ± 0.1	83.9 ± 0.2	64.4 ± 0.2	45.4 ± 1.2	67.3
(2) - (1)	<b>-1.9</b>	+16.7	<b>-3.6</b>	+10.0	+5.3
(1) Frozen	77.5 ± 0.4	74.3 ± 0.3	77.4 ± 0.2	43.4 ± 0.3	68.2
(2) DeiT <sup>†</sup>	79.3 ± 0.4	87.8 ± 0.5	76.6 ± 0.3	50.0 ± 0.2	73.4
(2) - (1)	+1.8	+13.5	<b>-0.8</b>	+6.6	+5.2
(1) Frozen	79.2 ± 0.1	76.6 ± 0.4	81.1 ± 0.2	35.7 ± 0.7	68.1
(2) HViT <sup>†</sup>	79.2 ± 0.5	89.7 ± 0.4	80.0 ± 0.2	51.4 ± 0.9	75.1
(2) - (1)	<b>-0.0</b>	+13.1	<b>-1.1</b>	+15.7	+7.0
(1) Frozen	82.6 ± 0.3	96.9 ± 0.1	83.2 ± 0.2	46.5 ± 2.1	77.3
(2) CLIP ViT-B16	82.7 ± 0.3	92.9 ± 1.9	78.1 ± 2.1	50.2 ± 1.7	75.9
(2) - (1)	+0.1	<b>-4.0</b>	<b>-5.1</b>	+3.7	<b>-1.4</b>
(3) DPL (ours)	<b>84.3 ± 0.4</b>	<b>97.3 ± 0.2</b>	<b>84.2 ± 0.2</b>	<b>52.6 ± 0.6</b>	<b>79.6</b>
(3) - (1)	+1.7	+0.4	1.0	+6.1	+2.3

Table 4: The results of frozen backbone ablation with ERM. Each block represents a backbone. Frozen means using the frozen backbone. <sup>†</sup> indicates that the numbers are taken from Table 2 in (Iwasawa and Matsuo 2021). The highlighted numbers indicate the Frozen ERM outperforms the standard ERM with fine-tuning backbones. (3) refers to DPL, which scores beat all others and are bolded.

result, DPL has no idea how to record such non-visual domain shift. Unfortunately, the label shift exists in the actual-world applications (Azizzadenesheli et al. 2019). An innovative question is whether adding appropriate information to the Domain Prompt can help solve the label shift problem, such as a detailed textual description of the target domain.

**Social impact perspective** Many images and text descriptions of web data are used directly to train CLIP. Though CLIP benefits from low-cost data that do not require manual labeling, it inevitably includes a lot of bias and privacy in CLIP and other foundation models (Bommasani et al. 2021). This requires us to spend more time paying attention to the opportunities and risks of Foundation Models.

### 5.2 Future Work

First and foremost, interpretability is critical in both Domain Transfer Learning and the Foundation Model. As discussed in subsection 5.1, DPL introduce the possibility of using a large language model in DG in the form of prompt. We will investigate this direction in our future work.

There are two simple and critical approaches to improving the performance of DG. One is to apply visual prompt tuning (Jia et al. 2022) on the pure visual backbones, which can be used to more previous methods. Another is focusing on a data-centric approach since we observe uneven data quality on the widely used datasets.

Finally, several recent studies systematically analyze the performance and shortcomings of large-scale pre-train models in the Out-of-Distribution generalization (Cha et al. 2022; Wenzel et al. 2022). We hope that our results will inspire more research in this direction.

## References

- Arjovsky, M.; Bottou, L.; Gulrajani, I.; and Lopez-Paz, D. 2020. Invariant Risk Minimization. *arXiv:1907.02893*.
- Azizzadenesheli, K.; Liu, A.; Yang, F.; and Anandkumar, A. 2019. Regularized learning for domain adaptation under label shifts. *arXiv preprint arXiv:1903.09734*.
- Beery, S.; van Horn, G.; and Perona, P. 2018. Recognition in Terra Incognita. *arXiv:1807.04975*.
- Ben-David, E.; Oved, N.; and Reichart, R. 2021. PADA: A Prompt-based Autoregressive Approach for Adaptation to Unseen Domains. *arXiv:2102.12206*.
- Blanchard, G.; Lee, G.; and Scott, C. 2011. Generalizing from several related classification tasks to a new unlabeled sample. *Advances in neural information processing systems*, 24: 2178–2186.
- Bommasani, R.; Hudson, D. A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M. S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.
- Borlino, F. C.; D’Innocente, A.; and Tommasi, T. 2021. Rethinking Domain Generalization Baselines. *arXiv:2101.09060*.
- Bottou, L. 2012. Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, 421–436. Springer.
- Cha, J.; Chun, S.; Lee, K.; Cho, H.-C.; Park, S.; Lee, Y.; and Park, S. 2021. Swad: Domain generalization by seeking flat minima. *Advances in Neural Information Processing Systems*, 34: 22405–22418.
- Cha, J.; Lee, K.; Park, S.; and Chun, S. 2022. Domain Generalization by Mutual-Information Regularization with Pre-trained Models. *arXiv preprint arXiv:2203.10789*.
- Choi, M. J.; Lim, J. J.; Torralba, A.; and Willsky, A. S. 2010. Exploiting hierarchical context on a large database of object categories. In *2010 IEEE computer society conference on computer vision and pattern recognition*, 129–136. IEEE.
- Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
- Dou, Q.; Coelho de Castro, D.; Kamnitsas, K.; and Glocker, B. 2019. Domain generalization via model-agnostic learning of semantic features. *Advances in Neural Information Processing Systems*, 32: 6450–6461.
- Everingham, M.; and Winn, J. 2009. The PASCAL visual object classes challenge 2007 (VOC2007) development kit.
- Fang, C.; Xu, Y.; and Rockmore, D. N. 2013. 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*, 1657–1664.
- Fei-Fei, L.; Fergus, R.; and Perona, P. 2006. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4): 594–611.
- Ganin, Y.; and Lempitsky, V. 2015. Unsupervised Domain Adaptation by Backpropagation. *arXiv:1409.7495*.
- Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016a. Domain-Adversarial Training of Neural Networks. *arXiv:1505.07818*.
- Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016b. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1): 2096–2030.
- Gao, P.; Geng, S.; Zhang, R.; Ma, T.; Fang, R.; Zhang, Y.; Li, H.; and Qiao, Y. 2021. CLIP-Adapter: Better Vision-Language Models with Feature Adapters. *arXiv:2110.04544*.
- Ge, C.; Huang, R.; Xie, M.; Lai, Z.; Song, S.; Li, S.; and Huang, G. 2022. Domain Adaptation via Prompt Learning. *arXiv preprint arXiv:2202.06687*.
- Grubinger, T.; Birlutiu, A.; Schöner, H.; Natschläger, T.; and Heskes, T. 2015. Domain generalization based on transfer component analysis. In *International Work-Conference on Artificial Neural Networks*, 325–334. Springer.
- Gulrajani, I.; and Lopez-Paz, D. 2020. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep residual learning for image recognition. *ComputerScience*.
- Iwasawa, Y.; and Matsuo, Y. 2021. Test-Time Classifier Adjustment Module for Model-Agnostic Domain Generalization. In *Thirty-Fifth Conference on Neural Information Processing Systems*.
- Jia, M.; Tang, L.; Chen, B.-C.; Cardie, C.; Belongie, S.; Hariharan, B.; and Lim, S.-N. 2022. Visual prompt tuning. *arXiv preprint arXiv:2203.12119*.
- Jin, X.; Lan, C.; Zeng, W.; and Chen, Z. 2020. Feature Alignment and Restoration for Domain Generalization and Adaptation. *arXiv:2006.12009*.
- Kim, D.; Wang, K.; Sclaroff, S.; and Saenko, K. 2022. A Broad Study of Pre-training for Domain Generalization and Adaptation. *arXiv preprint arXiv:2203.11819*.
- Koh, P. W.; Sagawa, S.; Marklund, H.; Xie, S. M.; Zhang, M.; Balsubramani, A.; Hu, W.; Yasunaga, M.; Phillips, R. L.; Gao, I.; et al. 2021. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, 5637–5664. PMLR.
- Kolesnikov, A.; Beyer, L.; Zhai, X.; Puigcerver, J.; Yung, J.; Gelly, S.; and Houlsby, N. 2020. Big Transfer (BiT): General Visual Representation Learning. *arXiv:1912.11370*.
- Kumar, A.; Raghunathan, A.; Jones, R.; Ma, T.; and Liang, P. 2022. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*.
- Lee, D.-H.; et al. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, 896.
- Lester, B.; Al-Rfou, R.; and Constant, N. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. *arXiv:2104.08691*.

- Li, D.; Yang, Y.; Song, Y.-Z.; and Hospedales, T. M. 2017a. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, 5542–5550.
- Li, D.; Yang, Y.; Song, Y.-Z.; and Hospedales, T. M. 2017b. Learning to Generalize: Meta-Learning for Domain Generalization. arXiv:1710.03463.
- Li, H.; Pan, S. J.; Wang, S.; and Kot, A. C. 2018. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5400–5409.
- Li, X. L.; and Liang, P. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.
- Liang, J.; Hu, D.; and Feng, J. 2020. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, 6028–6039. PMLR.
- Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2021a. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.
- Liu, X.; Ji, K.; Fu, Y.; Du, Z.; Yang, Z.; and Tang, J. 2021b. P-Tuning v2: Prompt Tuning Can Be Comparable to Fine-tuning Universally Across Scales and Tasks. arXiv:2110.07602.
- Motian, S.; Piccirilli, M.; Adjeroh, D. A.; and Doretto, G. 2017. Unified Deep Supervised Domain Adaptation and Generalization. arXiv:1709.10190.
- Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020.
- Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.
- Russell, B. C.; Torralba, A.; Murphy, K. P.; and Freeman, W. T. 2008. LabelMe: a database and web-based tool for image annotation. *International journal of computer vision*, 77(1-3): 157–173.
- Shankar, S.; Piratla, V.; Chakrabarti, S.; Chaudhuri, S.; Jyothi, P.; and Sarawagi, S. 2018. Generalizing Across Domains via Cross-Gradient Training. arXiv:1804.10745.
- Shen, Z.; Liu, J.; He, Y.; Zhang, X.; Xu, R.; Yu, H.; and Cui, P. 2021. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*.
- Singh, M.; Gustafson, L.; Adcock, A.; de Freitas Reis, V.; Gedik, B.; Kosaraju, R. P.; Mahajan, D.; Girshick, R.; Dollár, P.; and van der Maaten, L. 2022. Revisiting Weakly Supervised Pre-Training of Visual Perception Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 804–814.
- Sun, B.; and Saenko, K. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, 443–450. Springer.
- Thomee, B.; Shamma, D. A.; Friedland, G.; Elizalde, B.; Ni, K.; Poland, D.; Borth, D.; and Li, L.-J. 2016. YFCC100M: The new data in multimedia research. *Communications of the ACM*, 59(2): 64–73.
- Tolstikhin, I.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Steiner, A.; Keyser, D.; Uszkoreit, J.; Lucic, M.; and Dosovitskiy, A. 2021. MLP-Mixer: An all-MLP Architecture for Vision. arXiv:2105.01601.
- Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, 10347–10357. PMLR.
- Venkateswara, H.; Eusebio, J.; Chakraborty, S.; and Panchanathan, S. 2017. Deep Hashing Network for Unsupervised Domain Adaptation. arXiv:1706.07522.
- Wang, D.; Shelhamer, E.; Liu, S.; Olshausen, B.; and Darrell, T. 2020. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*.
- Wang, H.; Ma, S.; Dong, L.; Huang, S.; Zhang, D.; and Wei, F. 2022. DeepNet: Scaling Transformers to 1,000 Layers. *arXiv preprint arXiv:2203.00555*.
- Wang, J.; Lan, C.; Liu, C.; Ouyang, Y.; Zeng, W.; and Qin, T. 2021. Generalizing to Unseen Domains: A Survey on Domain Generalization. *arXiv preprint arXiv:2103.03097*.
- Wenzel, F.; Dittadi, A.; Gehler, P. V.; Simon-Gabriel, C.-J.; Horn, M.; Zietlow, D.; Kernert, D.; Russell, C.; Brox, T.; Schiele, B.; et al. 2022. Assaying Out-of-Distribution Generalization in Transfer Learning. *arXiv preprint arXiv:2207.09239*.
- Zhou, K.; Liu, Z.; Qiao, Y.; Xiang, T.; and Loy, C. C. 2021a. Domain Generalization in Vision: A Survey. arXiv:2103.02503.
- Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2021b. Learning to Prompt for Vision-Language Models. *arXiv preprint arXiv:2109.01134*.
- Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16816–16825.
- Zhou, K.; Yang, Y.; Hospedales, T.; and Xiang, T. 2020. Deep domain-adversarial image generation for domain generalisation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 13025–13032.

## 6 Appendix

### 6.1 Datasets

Following (Iwasawa and Matsuo 2021), we selected four real-world datasets from DomainBed benchmark, including VLCS (Fang, Xu, and Rockmore 2013), PACS (Li et al. 2017a), OfficeHome (Venkateswara et al. 2017), TerraIncognita (Beery, van Horn, and Perona 2018).

VLCS (Fang, Xu, and Rockmore 2013) gathers four photographic datasets  $d \in \{\text{Caltech101} (\text{Fei-Fei, Fergus, and Perona 2006}), \text{LabelMe} (\text{Russell et al. 2008}), \text{SUN09} (\text{Choi et al. 2010}), \text{VOC2007} (\text{Everingham and Winn 2009})\}$ , containing 10,729 samples of 5 classes. PACS (Li et al. 2017a) comprises four domain datasets  $d \in \{\text{art, cartoons, photos, sketches}\}$ , with 9,991 samples and 7 classes. OfficeHome (Venkateswara et al. 2017) includes domains  $d \in \{\text{art, clipart, product, real}\}$ , with 15,588 samples and 65 classes. TerraIncognita (Beery, van Horn, and Perona 2018) includes photos of wild animals taken by a camera at different locations. Following (Gulrajani and Lopez-Paz 2020), we used datasets of  $d \in \{\text{Location 100, Location 38, Location 43, Location 46}\}$ , with a total of 24,788 samples and classes. We show random samples from each dataset.

VLCS includes four photo datasets, so many objects unrelated to class are captured together. We conjecture that training in the source domain can help the model capture the correspondence between images and labels.

However, from the PACS dataset, we can find that the object corresponding to each image is straightforward and clear. This would be a relatively simple task for a CLIP trained on large-scale data. However, the shift of each Domain is evident, and if the large model is trained on the source domain, it will lead to performance degradation.

The dataset characteristics of OfficeHome resemble those of PACS in general, which explains our experimental resultsFigure 4 that fine-tuning hurts CLIP performance on PACS and OfficeHome. Through the t-SNE visualization<sup>3</sup> of zero-shot CLIP’s embeddingsFigure 5, we can find that zero-shot CLIP has a specific feature separation on PACS and OfficeHome. In contrast, VLCS and Terra have various color classes overlapping each other, which need to be trained. We hope this additional analysis can give a better understanding of our results.

Finally, we found that the models perform well if fine-tuning on Terra. This is because there is no way for Terra’s image-label correspondence to be learned during pre-training. It is worth noting that the SoTA method MIRO (Cha et al. 2022) not only uses a more advanced backbone than CLIP but also fine-tunes the backbone and adds the SWAD technique. These factors lead to the fantastic result of MIRO reaching 64.3% on Terra.

The real-world domain generalization is similar to the case of Terra (Koh et al. 2021). Therefore, we believe that similar to the MIRO, and it is essential to study FM in the DG domain.

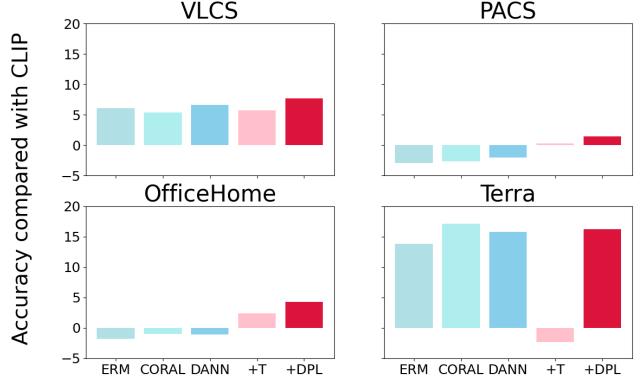


Figure 4: The visualization of the results compared with CLIP. The methods fine-tuning backbones are colored in blue, and freezing backbones in red. +T indicates using template prompt for CLIP.

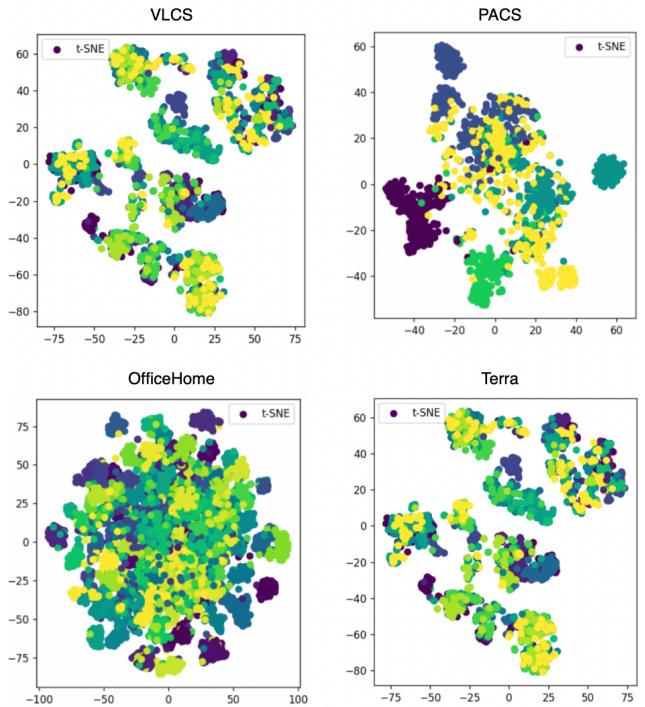


Figure 5: The t-SNE visualization of zero-shot CLIP on each dataset. We sampled 100 images for each class randomly. If the number of the class is less than 100, we sampled all of them. We used default numbers for all parameter of t-SNE, like number of componentsint is 2.

<sup>3</sup><https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html>

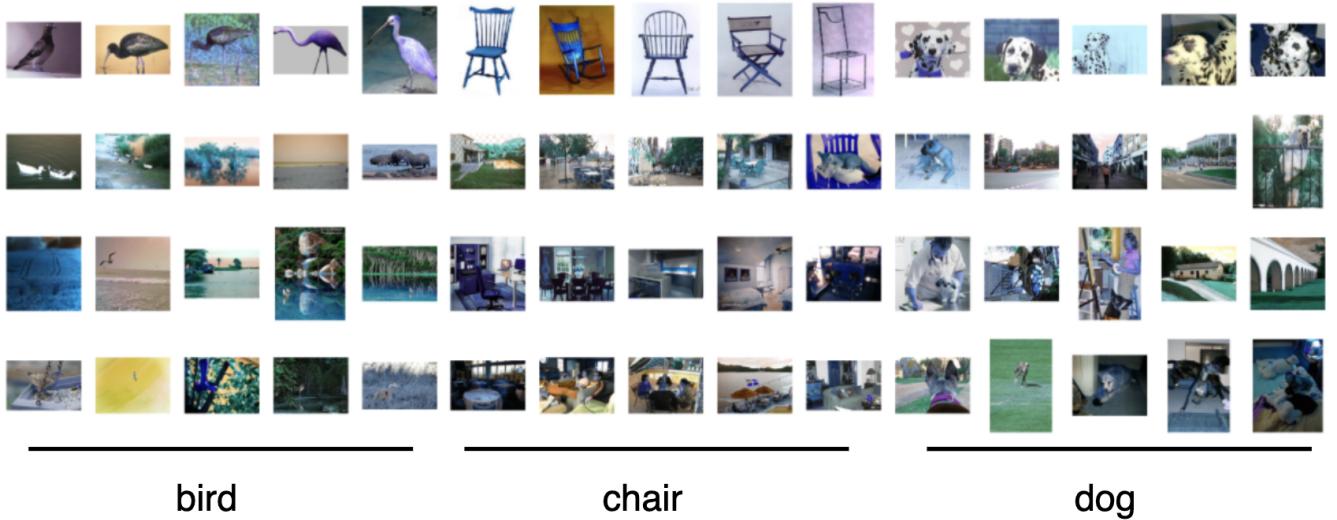


Figure 6: The image examples in VLCS.

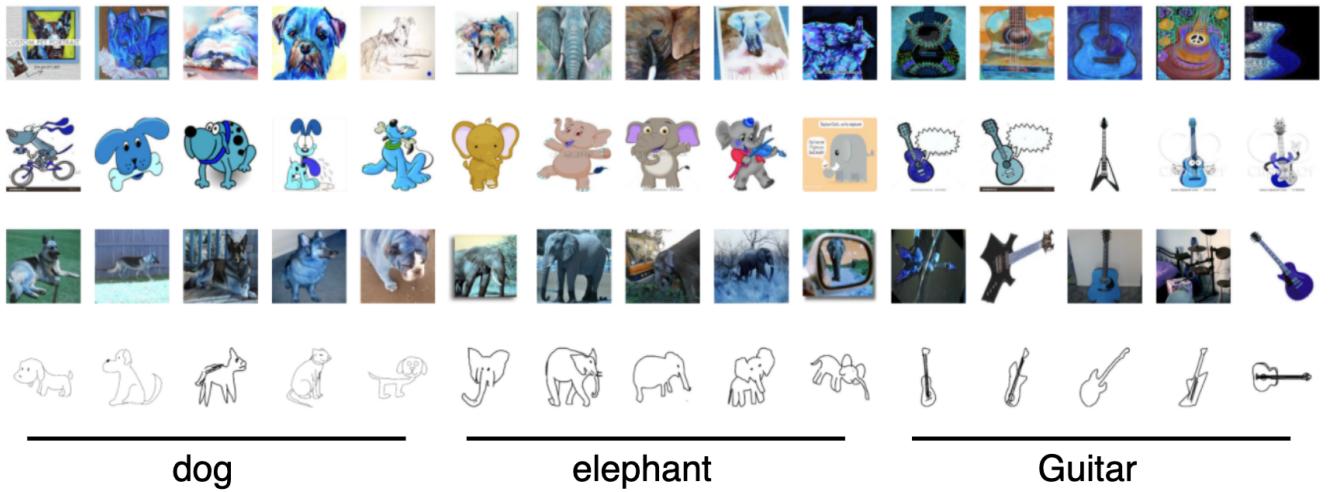


Figure 7: The image examples in PACS. From the first row to the fourth row are ‘art painting’, ‘cartoons’, ‘photos’, and ‘sketches’.



Figure 8: The image examples in OfficeHome. From the first row to the fourth row are ‘art’, ‘product’, ‘real’, and ‘clipart’

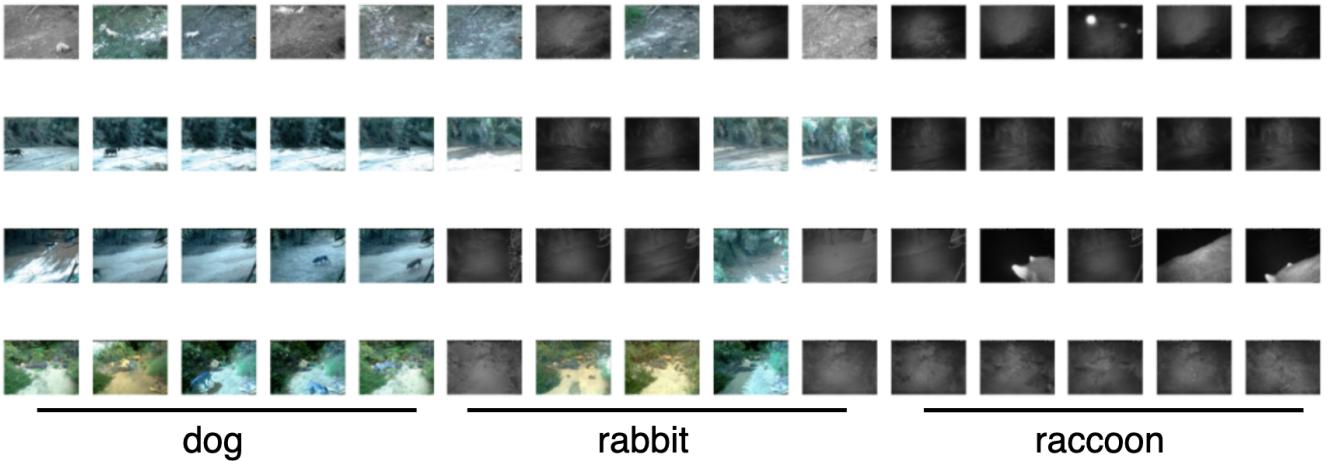


Figure 9: The image examples in TerraIncognita. From the first row to the fourth row are different location domains. From the dataset we can conjecture that Terra is a difficult dataset to benefit from pre-training backbone. This is because his domain is very specific and difficult to classify.