STYLIP: Multi-Scale Style-Conditioned Prompt Learning for CLIP-based Domain Generalization

Shirsha Bose[1*]   Ankit Jha[2*]   Enrico Fini[3]   Mainak Singha[2*]   Elisa Ricci[3]   Biplab Banerjee[2]

[1]Technical University of Munich, Germany   [2]Indian Institute of Technology Bombay, India

[3]University of Trento, Italy

shirshabosecs@gmail.com, ankitjha16@gmail.com, enrico.fini@unitn.it

mainaksingha.iitb@gmail.com, e.ricci@unitn.it, getbiplab@gmail.com

Abstract

Large-scale foundation models, such as CLIP, have demonstrated impressive zero-shot generalization performance on downstream tasks, leveraging well-designed language prompts. However, these prompt learning techniques often struggle with domain shift, limiting their generalization capabilities. In our study, we tackle this issue by proposing STYLIP, a novel approach for Domain Generalization (DG) that enhances CLIP's classification performance across domains. Our method focuses on a domain-agnostic prompt learning strategy, aiming to disentangle the visual style and content information embedded in CLIP's pre-trained vision encoder, enabling effortless adaptation to novel domains during inference. To achieve this, we introduce a set of style projectors that directly learn the domain-specific prompt tokens from the extracted multi-scale style features. These generated prompt embeddings are subsequently combined with the multi-scale visual content features learned by a content projector. The projectors are trained in a contrastive manner, utilizing CLIP's fixed vision and text backbones. Through extensive experiments conducted in five different DG settings on multiple benchmark datasets, we consistently demonstrate that STYLIP outperforms the current state-of-the-art (SOTA) methods.

1. Introduction

Advancements in large-scale vision and language models, such as CLIP [50] and ALIGN [18], have made remarkable progress in computer vision tasks. These models employ contrastively trained vision and text encoders to capture semantically meaningful concepts in a shared embedding space. They demonstrate impressive zero-shot generalization performance using text prompts like A photo of a [CLS]. However, designing an optimal prompt is
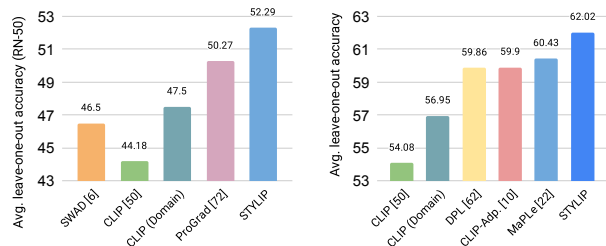

Figure 1. The performance of different DG techniques for the DomainNet [48] dataset for both the RN-50 and ViT backbones. Zero-shot CLIP falls behind SOTA traditional DG models like SWAD [6], indicating poor generalization. However, incorporating domain identifiers in prompts boosts baseline CLIP's performance. Finally, STYLIP outperforms the previous best prompting techniques [10, 22, 74] substantially, highlighting the importance of style and content disentanglement in the prompts for DG tasks.

challenging, and recent studies focus on data-driven prompt optimization [69]. Despite their success, prompt learning is limited to the training data distribution and is susceptible to domain shift [9]. Domain shift, common in real-world applications, poses challenges as deep learning models are sensitive to differences between training and test data distributions [17]. To tackle this, researchers explore Domain Generalization (DG) [28, 67, 71], which aims to learn a domain-agnostic representation from multiple datasets sourced from different domains for application to novel target domains. Traditional DG techniques rely on vision encoders trained exclusively on image data [70, 72, 73]. Recent efforts combine foundation models with prompt engineering [50, 69] to bridge the semantic gap, but their practical applicability in DG settings requires further exploration.

In this paper, we focus on a more challenging setting where significant visual variations exist across different domains, unlike existing prompting methods [38, 68, 69, 74] that evaluate CLIP's generalization capabilities on datasets with limited domain shift (e.g., variants of ImageNet [24]). Fig. 1 illustrates the average multi-source DG performance

---

on DomainNet [48], where zero-shot CLIP [50] underperforms compared to the best traditional DG model, SWAD [6], by approximately 2.5%. Using a domain-conditional prompt (A [Domain] of a [CLS]) boosts CLIP's accuracy by nearly 3%, highlighting the importance of a representative prompt for DG. *However, domain-level annotations are not always available, and the static domain name may not capture the style properties characterizing the domains [42].* Existing specialized prompt-tuning techniques [64, 68, 74] improve CLIP's performance (Fig. 1), but their effectiveness for DG is uncertain as prompts refined from random vectors may not effectively encode domain knowledge [68, 74]. Zhang et al. [64] propose a domain-prompt initialization strategy based on batch statistics of visual features but overlook important lower-level style characteristics and consider domain-level supervision. Another recent approach [45] learns prompts from CLIP without utilizing visual samples from the source domains but incorporates textual domain knowledge.

These discussions highlight a research gap in learning prompts that account for unknown domain shifts without explicit domain identifiers. We argue that leveraging visual features in such scenarios is crucial, along with dynamically incorporating object-level variations into the prompts to aid in cross-domain generalization tasks [68]. Motivated by these considerations, our research question is *whether we can utilize CLIP's vision backbone to encode image style and content information for learning domain and instance-aware prompts to address DG.*

Our proposed STYLIP: We introduce STYLIP, a novel generic prompt tuning strategy for CLIP that addresses these challenges. Our approach aims to enhance the prompts' understanding of class concepts by conditioning them on domain and content information derived from CLIP's visual space. To achieve this, we leverage the hypothesis that instance-wise feature statistics from intermediate layers of an image encoder capture the visual domain information [35]. We extract mean and standard deviation values from CLIP's intermediate feature map outputs and utilize a set of STYLE PROJECTORS to learn domain-specific tokens in the prompts. Unlike existing models such as [68, 69, 74] that learn prompt token embeddings from ad-hoc sentences, our approach benefits from using style features at different scales, which leads to improved domain-aware prompts and better prompt initialization.

Additionally, we propose to incorporate image content information into the prompt embeddings to capture object-level variations and avoid overfitting to the training classes, which is particularly important in DG scenarios concerning disjoint training and test classes. While Zhou et al. [68] addresses this issue by adding high-level semantic features from CLIP's vision encoder to prompt tokens, we consider a DG setting where the distributions of training and test classes differ [40]. To achieve this, we combine visual feature responses from different layers of CLIP's vision encoder and aggregate them through a CONTENT PROJECTOR. By encoding mid- to high-level image characteristics that are more generic across categories [66], we aim to enhance transferability across domains/classes. Unlike the literature [22, 68], we propose a learnable fusion network to aggregate these visual features with the final prompt embeddings obtained previously.

Contributions: We highlight our major contributions as:
- We introduce STYLIP, a domain-unified prompt learning strategy that leverages CLIP's frozen vision encoder to extract the domain and content information from an image and deploy them in prompt learning through light-weight learnable style and content projectors.
- We acquire prompt tokens from visual style features at various scales, facilitating the consolidation of hierarchical domain knowledge, thereby assisting in generalization across different domains. Moreover, we incorporate multi-scale visual content information into prompt embeddings, effectively mitigating overfitting and promoting generalization across different categories.
- We showcase the performance of STYLIP for multiple datasets on five major DG tasks: i) single-source and multi-source DG, ii) cross-dataset DG, iii) in-domain base to novel class DG, and iv) cross-domain base to novel class DG, a novel task we introduce in the context of prompting. Experimentally, STYLIP outperforms the competitors in all tasks at least by $0.2 - 4\%$. *To our knowledge, ours is the first attempt to extensively study the DG problem using CLIP.*

2. Related Works

Domain generalization. The DG problem has different variations. Single-source DG [49, 60] trains with one domain, while multi-source DG [11, 21, 72] considers training multiple domains simultaneously. In a closed-world setting, where the label set is shared across domains, DG approaches commonly address domain shift. Heterogeneous DG [36, 58] faces additional challenges due to different labels between the source and target domains. Previous research on DG proposed methods such as domain alignment losses [20, 32, 33, 59], self-supervised learning [5], ensemble learning [62], domain-specific networks [41], and meta-learning [47]. However, these methods often require more training domains, which can influence DG performance. To overcome this, novel pseudo-domains have been generated using domain augmentation approaches [21, 34, 72, 73]. In single-source DG models [49, 60, 65], diverse styles can be synthesized by perturbing the source domain through entropy maximization, meta-learning, and adversarial learning. Conversely, methods for heterogeneous DG [31, 37, 72] aim to improve model generalizability for novel tasks.

DPL [64] used CLIP [50] for multi-source DG by in-

ferring domain information from batch-wise visual features. However, DPL doesn't fully leverage CLIP's ability to extract domain-specific artifacts and can overfit with small batches due to challenges in obtaining an unbiased style estimate. Researchers have explored domain invariant prompts [26, 45] through text-based source domain knowledge or image patches for prompt input in ViT models, similar to VPT [19]. *Our* STYLIP *approach differs from [26, 45, 64] by considering style features at different visual encoder levels to learn individual prompt tokens and exploring multi-scale visual features in prompt learning, which have been successful in various DG tasks.*

In a recent study, Cumix [40] combines DG with the notion of zero-shot learning [61] for the recognition of new domains and classes. The following research investigated the use of structured multimodal information [8] or disentangled feature learning [43] for similar aims. *Our proposed experimental setup for a base to novel class generalization is identical; however, we are interested in analyzing the performance of the prompting techniques for VLMs in this respect, contrary to the more ad-hoc models mentioned above.* **Prompt tuning for vision-language models (VLMs).** VLMs have gained attention in language processing and computer vision [2, 3, 14, 25, 51, 54, 55]. These models utilize task-centric textual descriptions for visual data [15, 16]. Earlier prompting strategies were manual but later works focused on prompt learning. CoOp [69] optimized unified and class-specific prompts through back-propagation. Co-CoOp [68] addressed CoOp's generalization issue through input-conditioned prompt learning. CLIP-adapter [10] proposed fine-tuning feature adapters in visual and language branches. ProGrad [74] prevents knowledge forgetting from the foundation model. TPT [53] utilizes consistency among multiple image views for supervision. Probabilistic and variational models [38, 39] learn prompt distributions to match visual feature spreads. LASP [4] improves the learned prompt via text-to-text cross-entropy loss. MaPle [22] enhances compatibility between CLIP encoders at different levels. However, these approaches are not tailored to deal with multi-domain data. *In opposition, we introduce the notion of visual content-style disentanglement for prompt learning for DG tasks using CLIP.*

3. Proposed Methodology

3.1. Problem and notation

The DG problem involves $\mathcal{N}$ labelled source domains $\mathcal{S}^i = \{x_i^k, y_i^k\}_{k=1}^{n_i} \approx P_{data}^{\mathcal{S}^i}$, $1 \le i \le \mathcal{N}$, where $x_i \in \mathcal{X}^i$, $y_i \in \mathcal{Y}$, and $P_{data}^{\mathcal{S}^i}$ denote the input data, label, and the joint distribution concerning the data and the label space, respectively. Furthermore, $P_{data}^{\mathcal{S}^i} \ne P_{data}^{\mathcal{S}^j}$ $\forall i, j \in \{1, 2, \cdots, \mathcal{N}\}$, indicating that the source domains are mutually distinct. We call the setting single-source DG if $\mathcal{N} = 1$, else it is known

as multi-source DG. The goal is to train a model $f : \mathcal{X} \to \mathcal{Y}$ given $\mathcal{S} = \{\mathcal{S}^i\}_{i=1}^{\mathcal{N}}$, which is expected to generalize for a novel target domain $\mathcal{S}^{\mathcal{N}+1} = \{x_t^k, y_t^k\}_{k=1}^{n_t}$ unseen during training with $x_t \in \mathcal{X}^t$ and $y_t \in \mathcal{Y}^t$ and P_{data}^t denotes the target distribution which is different from the source distributions. Typically, we consider a closed-set setting where $\mathcal{Y} \cup \mathcal{Y}^t = \mathcal{Y} \cap \mathcal{Y}^t$. Also, for the base to new class generalization setting, we consider $\mathcal{Y} \cap \mathcal{Y}^t = \emptyset$.

3.2. The STYLIP model

Here, we introduce STYLIP, a novel approach for DG based on CLIP [50]. STYLIP leverages CLIP's frozen vision encoder (f_v) and text encoder (f_t), trained on a large volume of image-text pairs (see Fig. 2). f_v that transforms an input image into a feature embedding vector can be implemented with different architectures: in our experiments (see Section 4), we consider ResNet-50 (RN50) [13], and ViT-B/16 [12]. f_t is built upon a Transformer [56]: it is provided with an input of a sequence of word tokens and converts them into a vectorized representation.

As stated, STYLIP seeks to utilize the multi-scale visual features extracted from different levels of f_v to estimate the style and content primitives and further channel them in learning a generic prompt space regarding a concept. Typically, high-level representations of the deepest layer of a vision encoder tend to capture the abstract object semantics suitable for classification but suffer from a lack of description of local patterns like oriented edges or local shapes [66]. Therefore, the set of characteristics obtained from multiple levels is deemed more transferable between tasks than the high-level features alone. Similarly, the instance-wise feature statistics calculated from multiple layers of the encoder capture different levels of style, e.g., the texture in the top layers usually has larger granularity than those in the bottom layers [1].

To model a continuous prompt embedding space using these multilevel visual features, STYLIP (see Fig. 2) adopts a set of projector networks on top of f_v and f_t: a set of M style projectors $\{\mathcal{P}_m\}_{m=1}^M$ to encode domain characteristics into M prefix tokens $\{c_m\}_{m=1}^M$, a content projector $\mathcal{P}_C$ to encode feature responses from all the L encoder layers of f_v after reducing their dimensions using bottleneck layers $\{\mathcal{B}_l\}_{l=1}^L$, and a fusion projector $\mathcal{P}_A$. We discuss the structure of the proposed projectors in detail below.

Embedding multi-level style information into prompt tokens: For calculating the style features, let us consider the vector $\mathcal{F}_l(x) = [\vec{\mu_l}(x); \vec{\sigma_l}(x)]$ denoting the channel-wise mean and standard deviation of the feature map outputs from the lth layer ($1 \le l \le L$) of f_v, also indicated as $f_v^l(x)$. Here, $[-; -]$ denotes the concatenation operation. Specifically, if $f_v^l(x)$ is of dimensions $W \times H \times C$ (height, width, and depth dimensions), the statistics corresponding to the cth feature map $f_v^{l_c}$, (μ_l^c, σ_l^c), are calculated as:
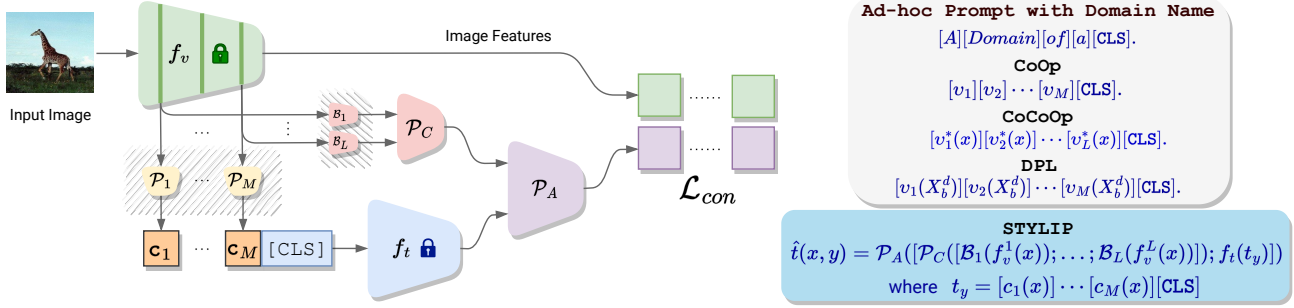
Figure 2. STYLIP generates a joint embedding space for image and prompt embeddings, leveraging style and content information extracted from the vision encoder f_v. The projectors $\{\mathcal{P}_m\}_{m=1}^M$, $\{\mathcal{B}_l\}_{l=1}^L$, and $\mathcal{P}_C$ adapt these features for the target task. For each image x, the style features from layer m of f_v are used to learn the mth token of the prompt, $c_m(x)$, through the style projector P_m. Similarly, the multi-scale content features are aggregated using $\mathcal{P}_C$ after dimensionality reduction by the bottleneck networks $\mathcal{B}_l$. The fusion projector $\mathcal{P}_A$ computes the classification weights. Unlike existing prompt tuning techniques (CoOp, CoCoOp, and DPL), STYLIP differs in its approach. v_m represents the prompt learner from a random vector, and X_b^d denotes a batch of samples for domain d.

$$\mu_l^c = \frac{1}{WH} \sum_{w,h=1}^{W,H} f_v^{l_c}(x)_{w,h} \qquad (1)$$

$$\sigma_l^c = \sqrt{\sum_{w,h=1}^{W,H} (f_v^{l_c}(x)_{w,h} - \mu_l^c)^2} \qquad (2)$$

In the simplest case, when the context length M equals the number of encoder layers L, we seek to learn the mth context vector c_m from $\mathcal{F}_m(x)$. Considering that the dimensions of $\mathcal{F}_m(x)$s are inconsistent and to appropriately input $\mathcal{F}_m$ into the text encoder f_t, we deploy the style projectors $\{\mathcal{P}_m\}_{m=1}^M$ and compute $c_m(x) = \mathcal{P}_m(\mathcal{F}_m(x))$, *i.e.* the mth context vector for the text prompt. We define

$$t_y = [c_1(x)][c_2(x)] \cdots [c_M(x)][CLS_y] \qquad (3)$$

as the prompt for (x,y) where $[CLS_y]$ is the word embedding of label y. Finally, f_t generates the embedding $f_t(t_y)$.

However, the context length in prompting is a hyper-parameter, meaning an M different from L may be preferred for a given task. To incorporate this flexibility in our prompt learning, we consider aggregation or replication of representations from $\{\mathcal{F}_l(x)\}_{l=1}^L$ depending on whether $M < L$ or $M > L$, respectively.

Supplementing the prompt embeddings with multi-scale image content features: The paradigm of t_y considers the information of the visual style of the images, but a static class embedding CLS_y for all images with the label y may limit its versatility. To further generalize the prompt embeddings, we propose to supplement t_y with content image information. As discussed, we extract multiple complementary visual characteristics associated with an image by aggregating the multilevel feature responses obtained from the L blocks of f_v.

One naive way to combine this multilevel information is by flattening the feature maps of individual blocks, fol-

lowed by concatenation. However, this leads to a very high-dimensional vector representation compared to the dimensionality of t_y, undermining the effects of t_y in the final classification weights. As a result, the contrastive task may lead to triviality. We propose reducing the feature maps' dimensions before concatenation as a remedy. This also shrinks the size of the inputs to $\mathcal{P}_C$, thus controlling its number of learnable parameters and the amount of information exchanged by the two encoders.

Precisely, given the lth-layer feature maps $f_v^l(x)$, we perform 1×1 convolution followed by flattening using $\mathcal{B}_l$ to reduce the channel depth of $f_v^l(x)$ from original C to $\hat{C} << C$, resulting in $\mathcal{B}_l(f_v^l(x)) \in \mathbb{R}^{WH\hat{C} \times 1}$. Finally, we concatenate the $\mathcal{B}(f_v^l(x))$s to obtain $\hat{f}_v(x)$:

$$\hat{f}_v(x) = [\mathcal{B}_1(f_v^1(x)); \mathcal{B}_2(f_v^2(x)); \cdots ; \mathcal{B}_L(f_v^L(x))] \qquad (4)$$

The content projector $\mathcal{P}_C$ learns the combined image embedding $\mathcal{P}_C(\hat{f}_v(x))$ through a linear transformation. To generate the classification weights for a given (x,y), we first concatenate $\mathcal{P}_C(\hat{f}_v(x))$ with $f_t(t_y)$ and transform the aggregated information through the fusion projector $\mathcal{P}_A$ to obtain $\hat{t}(x,y)$ as follows:

$$\hat{t}(x,y) = P_A([\mathcal{P}_C(\hat{f}_v(x)); f_t(t_y)]) \qquad (5)$$

3.3. Training and inference

The projectors are trained using a contrastive loss $\mathcal{L}_{con}$ between $\hat{t}(x,y)$ and the image features obtained from the final embedding layer of f_v, i.e., $f_v(x)$, as follows:

$$\mathcal{L}_{con} = \underset{\substack{\{\mathcal{P}_m\}_{m=1}^M, \\ \{\mathcal{B}_l\}_{l=1}^L, \mathcal{P}_C, \mathcal{P}_A}}{\arg\min} \underset{(x,y) \sim P_{data}^{\mathcal{S}}}{\mathbb{E}} -\log(p(\hat{t}(x,y)\|x)) \qquad (6)$$

where $P_{data}^{\mathcal{S}}$ is the joint data distribution of $\mathcal{S}$ and

$$p(\hat{t}(x^k,y^k)\|x^k) = \frac{e^{\delta(\hat{t}(x^k,y^k),f_v(x^k))/\tau}}{\sum_{n \in \mathcal{Y}} e^{\delta(\hat{t}(x^k,n),f_v(x^k))/\tau}} \qquad (7)$$

δ defines the cosine similarity and τ is the temperature hyperparameter. The contrastive loss synergistically maximizes the similarity between the image and the correct class prompt embeddings while minimizing the similarity between the image and all the opposing classes.

During inference, we calculate the compatibility between $f_v(x_t)$ and the prompt embeddings for all classes in $\mathcal{Y}^t$. The class with the highest compatibility is selected as:

$$\hat{y}_t = \arg\max_{n \in \mathcal{Y}^t} p(\hat{t}(x_t, n) \| x_t) \qquad (8)$$

4. Experimental Results

Datasets: We evaluate STYLIP over five benchmark datasets for multi-source and single-source domain generalization, namely Office-Home [57], PACS [29], VLCS [30], Digits-DG [72]], and DomainNet [48]. We further analyze the performance of STYLIP for cross-dataset generalization, where STYLIP is trained on ImageNet [24] and tested on ten other different datasets [69]. Detailed descriptions of the datasets are provided in SUPPLE.

Implementation, training, and evaluation protocols: We implement the projectors in $(\mathcal{P}_C, \mathcal{P}_A, \{\mathcal{P}_m\}_{m=1}^M)$ as single dense layers. We train the model with Adam optimizer [23] with a learning rate of $2e-2$ and betas $(0.9, 0.999)$. We consider a context length of four for all the experiments following [68, 69]. For RN50, we consider the feature map outputs from the four convolution stages to extract the style and content features, hence $L = M = 4$. For ViT-B/16, we obtain the embedding outputs from the $L = 12$ encoder layers. We further average the features for every three consecutive layers of f_v in a bottom-up manner without overlap to generate four intermediate feature representations, which are subsequently used to produce four distinct domain information vectors to be passed to $\{\mathcal{P}_m\}_{m=1}^4$. As we show in Fig. 3, the feature statistics capture the domain information in ViT-based prompt learning for STYLIP, similar to RN50. We fix the number of output channels of the bottleneck $\hat{C} = 3$ using cross-validation, where 10% images from each source domain are treated as the validation set. We further ablate $\hat{C}$ in Section 4.2 to check other architecture choices. Finally, we consider a mini-batch size of 4 for DomainNet and Office-Home, while it is 8 for the other datasets, and we train the model for 10 epochs. We report the average top-1 classification performance on $\mathcal{S}^{\mathcal{N}+1}$ over three different executions. *In terms of model complexity, STYLIP is extremely light-weight and consists of 0.18% more parameters than CoOp and CoCoOp and 0.06% more parameters than MaPLe.*

Baselines: We consider three types of methods for comparison to check the generalizability of pre-trained CLIP features and that of the prompting strategies. Our baseline is *zero-shot CLIP* with the prompt as 'A Photo of a [CLS]'. We also include *domain name* in the prompt as
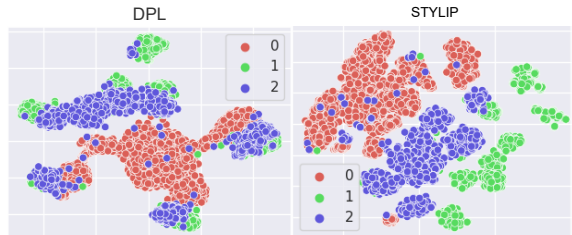

Figure 3. t-SNE visualization of the prompt embedding outputs from f_t for DPL [64] and STYLIP for multi-DG on PACS dataset. All the domains are highly clustered in STYLIP.

'A [Domain] of a [CLS]'. We use CLIP features to train a linear classifier, which we term *Linear Probing*. Furthermore, we deploy these features in conjunction with the benchmark DG technique of *CROSSGRAD* [52], where we put the learnable networks on top of frozen CLIP for back-propagation training. From the traditional DG literature, we report the performance of *SWAD* [6] for Multi-DG and SagNet [44] and DSBF [63] for Single-DG, respectively. Furthermore, we choose to compare STYLIP with existing prompt learning techniques including *CoOp* [69], *Co-CoOp* [68], *CLIP-Adapter* [10], *DPL* [64], *ProGrad* [74], *VPT* [19], *CSVPT* [26], *MaPLe* [22] and, *TPT* [53], etc.

Finally, we evaluate three variants of STYLIP intending to ablate the individual components of our approach: **i)** $\{\mathcal{P}_m\}_{m=1}^M$ are trained from random vectors (similar to CoOp), but we consider the multi-scale content feature learning of STYLIP through $(\{\mathcal{B}_l\}_{l=1}^L, \mathcal{P}_A, \mathcal{P}_C)$, respectively. (STYLIP-con). This establishes the importance of including the visual style information in the prompt tokens. **ii)** the model without content features and $(\{\mathcal{B}_l\}_{l=1}^L, \mathcal{P}_C)$ but with $\{\mathcal{P}_m\}_{m=1}^M$ (STYLIP-sty). This is to verify the importance of the multi-scale content features, and **iii)** the version of STYLIP where the features of the deepest layer of f_v are used for the content branch together with $\mathcal{B}_L$ only (STYLIP⋆). This is to assess the importance of the multi-scale content features over the single-scale high-level visual content properties as used in the literature [68].

4.1. Comparison with state-of-the-art

We discuss the experimental comparisons of STYLIP with the literature in the following order of the DG tasks: **i)** multi-source DG, **ii)** single-source DG, **iii)** cross-domain base to novel class DG, **iv)** in-domain base to novel class DG and, **v)** cross-dataset DG, respectively. We follow the leave-one-domain-out evaluation protocol for multi-source DG where all the domains except one are considered source domains while the model is to be verified on the held-out target domain. For single-source DG, we train the model on

---

[1]Methods are trained for a large number of epochs such as VPT [19] trains for 100 epochs, whereas STYLIP is trained only with 10 epochs.

Table 1. Comparison of our proposed STYLIP with the state-of-the-art methods on PACS, VLCS, Digits-DG, Office-Home, and DomainNet datasets for multi-source DG in terms of mean leave-one-out performance. † uses a different backbone than CLIP. (In %) [#1] Methods perform fine-tuning the visual backbones.

Backbone	Method	PACS	VLCS	Off.Home	Dig.DG	Dom.N.
	SWAD† [6]	88.10	79.10	70.60	-	46.50
CLIP RN50	Lin. Probing	91.65	79.48	70.17	62.22	46.10
	CROSSGRAD [52]	91.56	79.63	70.47	62.98	45.64
	ZS-CLIP [50]	90.32	76.43	66.75	56.41	44.18
	ZS-CLIP + DN	91.86	-	67.93	-	47.50
	CoOp [69]	92.28	81.87	71.65	73.11	49.71
	CoCoOp [68]	91.64	82.30	71.93	74.58	50.16
	CLIP-Adapt. [10]	92.08	82.35	72.18	73.79	50.25
	DPL [64]	91.96	82.12	72.54	74.33	50.38
	ProGrad [74]	92.01	82.23	71.85	74.45	50.27
	TPT [53]	92.16	82.39	72.07	74.68	50.30
	STYLIP-con	92.35	84.07	73.89	75.90	51.43
	STYLIP-sty	92.96	84.39	74.22	76.21	51.80
	STYLIP*	92.47	83.60	73.56	75.81	51.63
	STYLIP	93.59	84.83	74.80	76.49	52.29
CLIP ViT-B/16	Lin. Probing	96.54	82.63	80.43	70.15	57.46
	CROSSGRAD [52]	96.40	83.76	80.55	70.83	57.60
	ZS-CLIP [50]	95.81	80.57	78.57	65.79	54.08
	ZS-CLIP + DN	96.30	-	79.10	-	56.95
	CoOp [69]	97.00	82.98	81.12	76.41	59.52
	CoCoOp [68]	96.73	83.59	80.70	78.49	59.68
	CLIP-Adapt. [10]	96.41	84.32	82.23	77.86	59.90
	DPL [64]	97.07	83.93	83.00	77.32	59.86
	ProGrad [74]	96.50	83.82	82.46	78.26	59.65
	TPT [53]	96.99	83.72	82.45	78.51	59.87
	MIRO [7]	95.80	83.60	82.30	-	57.20
	VPT# [19]	97.20	84.90	85.20	-	59.80
	CSVPT# [27]	97.30	84.90	85.00	-	60.00
	DUPRG# [46]	97.10	83.90	83.60	-	59.60
	MaPLe# [22]	97.56	85.12	83.35	-	60.43
	STYLIP-con	96.82	85.61	83.90	80.63	61.51
	STYLIP-sty	97.25	86.27	84.18	80.91	61.77
	STYLIP*	97.11	85.88	83.41	80.56	61.39
	STYLIP	98.05	86.94	84.63	81.38	62.02

Table 2. Comparing STYLIP with CLIP-based SOTA methods for single-source DG on PACS, VLCS, and Office-Home datasets in terms of mean leave-all-but-one-domain-out performance. (In %)

Backbone	Method	PACS	VLCS	Office Home
	SagNet [44]	61.90	-	68.00
	DSBF [63]	85.33	-	63.91
CLIP RN50	Lin. Probing	85.67	69.42	65.99
	CoOp [69]	89.88	74.04	69.04
	CoCoOp [68]	88.69	74.80	69.48
	CLIP-Adapter [10]	88.86	75.31	69.29
	DPL [64]	89.24	74.86	69.10
	Prograd [74]	88.51	75.40	69.49
	TPT [53]	88.93	75.02	69.58
	STYLIP-con	90.56	75.92	70.41
	STYLIP-sty	91.77	76.39	70.87
	STYLIP*	89.14	75.67	69.85
	STYLIP	92.61	77.18	71.60
CLIP ViT-B/16	Lin. Probing	89.85	76.15	77.71
	CoOp [69]	95.59	80.10	80.44
	CoCoOp [68]	94.92	80.44	81.19
	CLIP-Adapter [10]	94.60	80.27	80.86
	DPL [64]	94.70	80.58	80.79
	Prograd [74]	94.82	80.38	81.37
	TPT [53]	95.14	80.57	81.43
	MaPLe [22]	95.33	80.15	81.95
	STYLIP-con	96.17	81.26	82.79
	STYLIP-sty	96.58	82.41	83.58
	STYLIP*	95.64	80.82	82.04
	STYLIP	97.03	82.90	83.89

Table 3. Analysis of the generalization from base to new classes across domains. We show results on DomainNet with *ClipArt* acting as the source domain, while others denote the target. The model is trained (backbone CLIP ViT-B/16) using 16 shots from the base classes. (In %)

Method	DomainNet						
	Base	New					
	Clip Art	Clip Art	Infograph	Painting	Quick Draw	Real	Sketch
CLIP [50]	78.00	76.55	49.80	70.84	17.56	88.11	66.54
CoOp [69]	82.79	75.60	48.60	71.38	20.90	85.19	67.39
CoCoOp [68]	82.85	77.40	52.61	72.06	20.80	88.00	68.12
CLIP-Adapter [10]	80.51	76.33	51.70	71.81	20.15	87.30	67.60
DPL [64]	82.35	76.49	52.10	71.88	20.30	87.54	67.73
ProGrad [74]	83.00	77.50	51.44	72.16	20.86	87.11	67.05
MaPLe [22]	82.84	77.61	51.65	71.95	20.51	87.35	67.82
STYLIP-con	83.71	77.86	52.04	72.53	20.97	87.20	67.70
STYLIP-sty	84.19	77.62	52.80	73.00	21.10	87.54	68.29
STYLIP*	83.34	77.30	51.92	72.36	20.93	87.39	67.97
STYLIP	84.90	78.14	53.09	73.60	21.69	87.90	68.61

for both RN50 and ViT-B/16 backbones. Our method, STYLIP, surpasses zero-shot CLIP, Linear Probing, and domain alignment approaches by at least 3% for both backbones, achieving state-of-the-art (SOTA) results. Additionally, STYLIP outperforms competitors, including DPL, CSVPT, and other prompting methods, across all datasets and vision backbones. Notably, when using ViT-B/16, STYLIP achieves outstanding performance with scores of 86.94% on VLCS, 81.38% on Digits-DG, and 62.03% on DomainNet, surpassing others by at least 3%.

Comparatively, the performance of STYLIP-con is slightly lower than STYLIP by approximately 0.5-1.3%, while STYLIP-sty performs marginally better than STYLIP-con but remains inferior to STYLIP. However, both STYLIP-sty and STYLIP-con exhibit comparable or better performance than other prompting methods. The limitations of these variants of STYLIP are that they only capture partial visual properties, leading to sub-optimal prompt learning. In contrast, STYLIP fully leverages both style and content information of images, reducing the gap between visual and semantic spaces. Moreover, STYLIP outperforms STYLIP* due to the multi-scale content features, which are more generalizable than deeper semantically oriented visual representations.

In the single-source DG setting, using PACS, VLCS, and Office-Home datasets, we report the average leave-all-but-one-domain-out in Table 2 for all domain combinations. Remarkably, STYLIP achieves a convincing improvement of approximately 1.4-2.5% over other prompting techniques, establishing a new SOTA for single-source DG. For detailed domain-wise results in both single-source and multi-source DG setups, please refer to SUPPLEMENTARY.

Generalizing across novel domains and categories: In this experiment on DomainNet, we consider ClipArt as the source domain while the others denote the target domain. We divide the classes equally, and the model is trained and tested on the disjoint class sets, following [40]. In Table 3, STYLIP outperforms the other prompting techniques in nine out of ten cases by $\approx 0.3 - 4\%$ while generalizing to novel classes from both the source and the target

one domain and test it on the remaining domains (leave-all-but-one-domain-out). We consider the standard few-shot training dataset with 16-shots, following the CLIP literature [22, 69] for all the tasks. However, we have mentioned a detailed sensitivity analysis of STYLIP against the number of available training samples in Fig. 5.

Discussions on multi-source and single-source DG: We present the mean leave-out performance of PACS, VLCS, Office-Home, Digits-DG, and DomainNet in Table 1

Table 4. Comparison with SOTA methods on base-to-new generalization. STYLIP shows better generalization performance over existing methods on 11 different recognition datasets on 16-shots and a context length of four. HM is the harmonic mean. (In %)

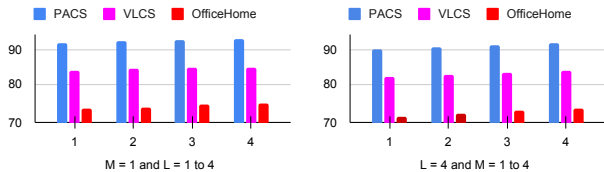| Average over 11 datasets			
Methods	Base	New	HM
CLIP [50]	69.34	74.22	71.70
CoOp [69]	82.69	63.22	71.66
CoCoOp [68]	80.47	71.69	75.83
LASP [4]	82.70	74.90	78.61
MaPLe [22]	82.28	75.14	78.55
STYLIP-con	82.64	75.39	78.85
STYLIP-sty	82.93	75.67	79.13
STYLIP*	82.30	75.24	78.61
STYLIP	**83.22**	**75.94**	**79.41**

Figure 4. Effects of using style features from an increasing number of layers from f_v for the fixed content features from the Lth layer, and using content features from an increasing number of layers for a fixed style feature ($M = 1$).

domains, respectively. For the *Real* domain of DomainNet, STYLIP lags [68] by a mere 0.21%. STYLIP is less prone to overfitting to the classes of the source domain due to the better transferability offered by our model through multi-scale feature embedding. To validate this, we repeat this experiment using the model STYLIP*, which deals with only the deepest layer visual encodings. Confirming our hypothesis, we find that the performance of STYLIP* is consistently poorer than STYLIP for all cases ($\approx 0.2 - 1.2\%$).

In-domain base to novel class generalization: In addition to the cross-domain generalization to novel categories, we show the performance of STYLIP on the 11 datasets [69] where the base and novel classes are divided for each dataset to define the source and the target domains. A context length of four and 16 samples per class are considered for training the model. We depict the average performance over all the datasets in Table 4, which shows that STYLIP beats the state-of-the-art, CoOp [69], CoCoOp [68], LASP [4], and MaPLe [22] convincingly by more than 0.8% on average H-score (H-score is the harmonic mean of the base-class and novel-class accuracies). Specifically, STYLIP is better than CoOp and CoCoOp by $\approx 8\%$ and 4%, respectively. We observe that STYLIP is able to beat the others both for the base as well as novel classes. This is important since the existing methods are mostly found to boost the performance of novel classes at the cost of decreasing base class performance. Refer to `Supplementary` for the detailed results.
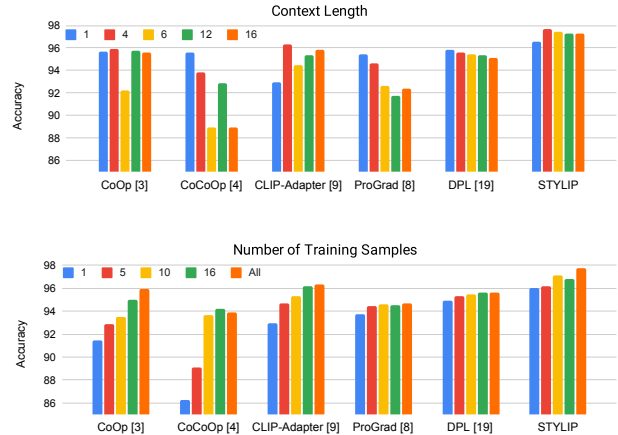

Figure 5. Sensitivity of prompting techniques (with ViT-B/16) on the context length (M) and the number of training samples per class. We show the average performance for multi-source DG on the PACS dataset.

4.2. Ablation analysis

Generalization across datasets: Following the literature [69], we perform prompt learning using 16-shots from the 1000 classes of ImageNet (source) and test on the other 10 datasets (target). On the source domain, STYLIP beats the recent [22] by almost 1.6% (Table 5). In contrast, for more specialized target datasets, such as DTD, EuroSAT and FG-VAircrafts, STYLIP beats the other competitors. For the fine-grained datasets, STYLIP shows improvements up to 2%, exhibiting much stronger transferability.

Analysis of multi-scale features: For the PACS and VLCS datasets with RN50, we conducted two experiments to investigate the impact of multi-scale features from f_v on performance. In the first experiment, we focused on the content feature corresponding to the Lth layer of f_v, varying M from 1 to 4 for the style features. In the second experiment, we fixed $M = 1$ for the deepest layer style features and varied L from 1 to 4 for the content features. As shown in Figure 4, increasing the number of layers for both style and content features positively influenced the performance.

In another experiment, we seek to show the usefulness of the multi-scale content features. In this regard, we compare STYLIP with a multi-scale version of CoCoOp [68] where we combine the multi-scale features to the input tokens instead of the deepest layer features as done in the CoCoOp paper. It can be observed from Tab. 6 that STYLIP is able to beat MS-CoCoOp on multiple datasets for the Multi-DG task. This can be attributed to the improved prompting of STYLIP using the disentangled style and content features.

Context length (M): As we mention in Fig. 5, we evaluate the effects of different context lengths for multi-source DG on PACS using the ViT backbone. We find that STYLIP outperforms the other techniques, including [10, 64, 68, 69, 74] for context lengths of 1, 4, 6, 12, and 16. To generate the style primitives for $M = 16$, we choose to replicate the feature statistics vectors for the final four en-

Table 5. Comparison of STYLIP with the prompt benchmark methods for generalization across datasets. We train the model on ImageNet using 16-shots with CLIP ViT-B/16 and test on 10 other datasets. (In %)

Method	Source	Target										
	ImgNet.	C101	Pets	Cars	Flowers	Food	Aircraft	Sun397	DTD	EuroSAT	UCF101	Average
CoOp [69]	71.51	93.70	89.14	64.51	68.71	85.30	18.47	64.15	41.92	46.39	66.55	63.88
CoCoOp [68]	71.02	94.43	90.14	65.32	71.88	86.06	22.94	67.36	45.73	45.37	68.21	65.74
MaPLe [22]	70.72	93.53	90.49	65.57	72.23	86.20	24.74	67.01	46.49	48.06	68.69	66.30
STYLIP-con	71.44	94.96	90.75	66.83	72.14	87.56	24.88	67.45	46.63	47.72	68.85	66.67
STYLIP-sty	72.05	95.13	91.44	67.02	72.29	88.31	25.17	67.92	47.64	48.09	69.12	67.25
STYLIP*	70.93	93.87	90.53	65.75	72.00	86.85	24.63	67.30	46.53	47.92	68.74	66.41
STYLIP	72.30	95.45	91.60	67.09	72.36	88.60	25.21	68.11	47.86	48.22	69.30	67.38

Table 6. Comparison of our proposed STYLIP with CoCoOp, including its extension with multi-scale features on the meta networks for PACS, VLCS, and Office-Home datasets for multi-source DG in terms of mean leave-one-out performance. (In %)

Backbone	Method	PACS	VLCS	Office Home
	CoCoOp [68]	88.69	74.80	69.48
	MS-CoCoOp	89.15	75.90	69.72
CLIP RN50	STYLIP	92.61	77.18	71.60
	CoCoOp [68]	94.92	80.44	81.19
	MS-CoCoOp	95.14	81.06	81.93
CLIP ViT-B/16	STYLIP	97.03	82.90	83.89

Table 7. Ablation analysis of STYLIP for multi-source DG on PACS and Office-Home (OH) using ViT-B/16 backbone. (In %)

Baselines	PACS	OH
Late Fusion Projector (max-pool)	95.33	82.10
Late Fusion Projector (average pool)	93.21	81.33
Depth of $\{\mathcal{P}_m\}_{m=1}^M$ and $\mathcal{P}_C$ (2 Layers)	97.28	84.47
Depth $\{\mathcal{P}_m\}_{m=1}^M$ and $\mathcal{P}_C$ (3 Layers)	97.43	84.40
$\{\mathcal{P}_m\}_{m=1}^M$ Only (μ)	96.81	83.26
$\{\mathcal{P}_m\}_{m=1}^M$ Only (σ)	97.00	83.42
$\{\mathcal{B}_l\}_{l=1}^L$ (GAP over the spatial dimensions of the feature-maps)	97.64	83.99
$\{\mathcal{B}_l\}_{l=1}^L$ (only flatten)	97.30	83.57
$\{\mathcal{B}_l\}_{l=1}^L$ (conv 1x1 with $\hat{C} = 2$)	97.57	83.79
$\{\mathcal{B}_l\}_{l=1}^L$ (conv 1x1 with $\hat{C} = 4$)	97.92	84.33
$\{\mathcal{B}_l\}_{l=1}^L$ (conv 1x1 with $\hat{C} = 16$)	96.44	83.10
STYLIP ($\hat{C} = 3$)	98.05	84.63

coder layers, i.e., $\mathcal{F}_{9-12}(x)$, in addition to that of the original 12 layers and feed them to $\{\mathcal{P}_m\}_{m=1}^{16}$. A context of 4 provides the optimal performance for STYLIP. We further find that a longer context length drastically deteriorates the performance of [68, 74], while STYLIP performs consistently across all context lengths.

Sensitivity to the number of training samples: To assess the robustness of STYLIP versus the number of training samples for the conventional DG setting, we train the single-source DG model on PACS while varying the number of training samples per class in the range $[1, 5, 10, 16, All]$. As shown in Fig. 5, the DG performance of [68, 69] degrades in the low-data regime, while [10,64,74] shows comparatively better performance. Finally, STYLIP maintains its superior performance for very few training samples and shows improvements with more shots.

Depth of style and content projectors: To check the sensitivity of STYLIP on the depth of $\{\mathcal{P}_m\}_{m=1}^M$ and $\mathcal{P}_C$, we consider cases of multi-source DG where the projectors are two-layers and three-layers deep with a consistent number of nodes per layer, respectively, in PACS and Office-Home (Tab. 7). We find that performance decreases marginally with increasing depth: $0.6-0.8\%$ for PACS and $0.2-0.4\%$

for Office-Home than STYLIP with linear projectors, suggesting STYLIP is indeed lightweight.

Learnable vs. non-learnable $\mathcal{P}_A$: Typically, $\mathcal{P}_C$ and f_t produce feature embeddings of similar dimensions; hence, one way to fuse them in $\mathcal{P}_A$ is through element-wise feature pooling. In this regard, we use the max and average feature pooling strategies and observe in Tab. 7 that such aggregations affect the performance, reducing the multi-source DG accuracies on PACS and Office-Home by $\approx 2-3\%$ in max pooling and $\approx 4-5\%$ in average pooling than STYLIP.

Analysis of style features: Typically, the mean and std. of the feature maps together are known to capture the visual style information. To validate the same, we study the model's performance with either mean or std. being used as input to the style projectors. In this regard, we see a decrease in the performance of $1-2\%$ compared to STYLIP, suggesting the importance of both statistical estimates. Interestingly, we see better accuracy when only std. is used for context learning than only mean (Tab. 7).

Sensitivity to the depth of the bottleneck layer $\hat{C}$: We consider different $\hat{C}$ values in the range $2, 3, 4, 16$ to see the effects of the bottleneck dimensions in the final accuracy (Tab. 7). While $\hat{C} = 3$ provides the best performance, we see the numbers decreasing from $\hat{C} = 4$ onwards, finally producing a dip of almost 1.5% for $\hat{C} = 16$. Besides, we consider the scenario where 1×1 convolutions are not used, and we perform global average pooling (GAP), or directly flatten the feature maps and then concatenate. Both options perform poorly compared to STYLIP by $0.5-1\%$.

5. Takeaways

In this paper, we aim to address the challenge of domain shift in DG tasks by proposing STYLIP, a domain-agnostic prompt learning strategy for CLIP. By disentangling and incorporating multi-scale visual style and content information from CLIP's frozen vision encoder into the prompt learning process, we enhance its generalizability. Extensive evaluations on various cross-domain inference tasks demonstrate the consistent state-of-the-art performance of STYLIP. Our study on task-generalizable prompt learning paves the way for new research opportunities in computer vision. Future directions could explore domain-aware prompt learning with different foundation models and extend the proposed approach to structured prediction tasks.

References

[1] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. Neural codes for image retrieval. In *European conference on computer vision*, pages 584–599. Springer, 2014. 3

[2] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021. 3

[3] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer, 2014. 3

[4] Adrian Bulat and Georgios Tzimiropoulos. Language-aware soft prompting for vision & language foundation models. *arXiv preprint arXiv:2210.01115*, 2022. 3, 7

[5] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019. 2

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

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

[8] Shivam Chandhok, Sanath Narayan, Hisham Cholakkal, Rao Muhammad Anwer, Vineeth N Balasubramanian, Fahad Shahbaz Khan, and Ling Shao. Structured latent embeddings for recognizing unseen classes in unseen domains. *arXiv preprint arXiv:2107.05622*, 2021. 3

[9] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*, 2017. 1

[10] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021. 1, 3, 5, 6, 7, 8

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

[12] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*, 2022. 3

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[14] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019. 3

[15] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International conference on machine learning*, pages 4182–4192. PMLR, 2020. 3

[16] Dat Huynh and Ehsan Elhamifar. Fine-grained generalized zero-shot learning via dense attribute-based attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4483–4493, 2020. 3

[17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015. 1

[18] 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*, pages 4904–4916. PMLR, 2021. 1

[19] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIII*, pages 709–727. Springer, 2022. 3, 5, 6

[20] Yunpei Jia, Jie Zhang, Shiguang Shan, and Xilin Chen. Single-side domain generalization for face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8484–8493, 2020. 2

[21] Juwon Kang, Sohyun Lee, Namyup Kim, and Suha Kwak. Style neophile: Constantly seeking novel styles for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7130–7140, 2022. 2

[22] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19113–19122, June 2023. 1, 2, 3, 5, 6, 7, 8

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 1, 5

[25] J Devlin M Chang K Lee and K Toutanova. Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[26] Aodi Li, Liansheng Zhuang, Shuo Fan, and Shafei Wang. Learning common and specific visual prompts for domain

generalization. In *Proceedings of the Asian Conference on Computer Vision*, pages 4260–4275, 2022. 3, 5

[27] Aodi Li, Liansheng Zhuang, Shuo Fan, and Shafei Wang. Learning common and specific visual prompts for domain generalization. In *Proceedings of the Asian Conference on Computer Vision*, pages 4260–4275, 2022. 6

[28] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018. 1

[29] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017. 5

[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*, pages 5542–5550, 2017. 5

[31] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M Hospedales. Episodic training for domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1446–1455, 2019. 2

[32] Haoliang Li, YuFei Wang, Renjie Wan, Shiqi Wang, Tie-Qiang Li, and Alex Kot. Domain generalization for medical imaging classification with linear-dependency regularization. *Advances in Neural Information Processing Systems*, 33:3118–3129, 2020. 2

[33] Jingjing Li, Erpeng Chen, Zhengming Ding, Lei Zhu, Ke Lu, and Heng Tao Shen. Maximum density divergence for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3918–3930, 2021. 2

[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/CVF International Conference on Computer Vision*, pages 8886–8895, 2021. 2

[35] Yanghao Li, Naiyan Wang, Jiaying Liu, and Xiaodi Hou. Demystifying neural style transfer. *arXiv preprint arXiv:1701.01036*, 2017. 2

[36] Yiying Li, Yongxin Yang, Wei Zhou, and Timothy Hospedales. Feature-critic networks for heterogeneous domain generalization. In *International Conference on Machine Learning*, pages 3915–3924. PMLR, 2019. 2

[37] Yiying Li, Yongxin Yang, Wei Zhou, and Timothy Hospedales. Feature-critic networks for heterogeneous domain generalization. In *International Conference on Machine Learning*, pages 3915–3924. PMLR, 2019. 2

[38] Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5206–5215, 2022. 1, 3

[39] Mohammad Mahdi Derakhshani, Enrique Sanchez, Adrian Bulat, Victor Guilherme Turrisi da Costa, Cees GM Snoek, Georgios Tzimiropoulos, and Brais Martinez. Variational prompt tuning improves generalization of vision-language models. *arXiv e-prints*, pages arXiv–2210, 2022. 3

[40] Massimiliano Mancini, Zeynep Akata, Elisa Ricci, and Barbara Caputo. Towards recognizing unseen categories in unseen domains. In *European Conference on Computer Vision*, pages 466–483. Springer, 2020. 2, 3, 6

[41] Massimiliano Mancini, Samuel Rota Bulo, Barbara Caputo, and Elisa Ricci. Best sources forward: domain generalization through source-specific nets. In *2018 25th IEEE international conference on image processing (ICIP)*, pages 1353–1357, 2018. 2

[42] Massimiliano Mancini, Lorenzo Porzi, Samuel Rota Bulo, Barbara Caputo, and Elisa Ricci. Boosting domain adaptation by discovering latent domains. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3771–3780, 2018. 2

[43] Puneet Mangla, Shivam Chandhok, Vineeth N Balasubramanian, and Fahad Shahbaz Khan. Context-conditional adaptation for recognizing unseen classes in unseen domains. *arXiv preprint arXiv:2107.07497*, 2021. 3

[44] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8690–8699, 2021. 5, 6

[45] Hongjing Niu, Hanting Li, Feng Zhao, and Bin Li. Domain-unified prompt representations for source-free domain generalization. *arXiv preprint arXiv:2209.14926*, 2022. 2, 3

[46] Hongjing Niu, Hanting Li, Feng Zhao, and Bin Li. Domain-unified prompt representations for source-free domain generalization. *arXiv preprint arXiv:2209.14926*, 2022. 6

[47] Novi Patricia and Barbara Caputo. Learning to learn, from transfer learning to domain adaptation: A unifying perspective. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1442–1449, 2014. 2

[48] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019. 1, 2, 5

[49] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12556–12565, 2020. 2

[50] 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, 2, 3, 6, 7

[51] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 3

[52] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. *arXiv preprint arXiv:1804.10745*, 2018. 5, 6

[53] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-

time prompt tuning for zero-shot generalization in vision-language models. *arXiv preprint arXiv:2209.07511*, 2022. 3, 5, 6

[54] Mainak Singha, Ankit Jha, Bhupendra Solanki, Shirsha Bose, and Biplab Banerjee. Applenet: Visual attention parameterized prompt learning for few-shot remote sensing image generalization using clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 3

[55] Mainak Singha, Harsh Pal, Ankit Jha, and Biplab Banerjee. Ad-clip: Adapting domains in prompt space using clip. *arXiv preprint arXiv:2308.05659*, 2023. 3

[56] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[57] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017. 5

[58] Yufei Wang, Haoliang Li, and Alex C Kot. Heterogeneous domain generalization via domain mixup. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3622–3626. IEEE, 2020. 2

[59] Ziqi Wang, Marco Loog, and Jan van Gemert. Respecting domain relations: Hypothesis invariance for domain generalization. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9756–9763. IEEE, 2021. 2

[60] Zijian Wang, Yadan Luo, Ruihong Qiu, Zi Huang, and Mahsa Baktashmotlagh. Learning to diversify for single domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 834–843, 2021. 2

[61] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning - the good, the bad and the ugly. *CoRR*, abs/1703.04394, 2017. 3

[62] Zheng Xu, Wen Li, Li Niu, and Dong Xu. Exploiting low-rank structure from latent domains for domain generalization. In *European Conference on Computer Vision*, pages 628–643. Springer, 2014. 2

[63] Junkun Yuan, Xu Ma, Defang Chen, Kun Kuang, Fei Wu, and Lanfen Lin. Domain-specific bias filtering for single labeled domain generalization. *International Journal of Computer Vision*, 131(2):552–571, 2023. 5, 6

[64] Xin Zhang, Yusuke Iwasawa, Yutaka Matsuo, and Shixiang Shane Gu. Amortized prompt: Lightweight fine-tuning for clip in domain generalization. *arXiv preprint arXiv:2111.12853*, 2021. 2, 3, 5, 6, 7, 8

[65] Long Zhao, Ting Liu, Xi Peng, and Dimitris Metaxas. Maximum-entropy adversarial data augmentation for improved generalization and robustness. *Advances in Neural Information Processing Systems*, 33:14435–14447, 2020. 2

[66] Liang Zheng, Yali Zhao, Shengjin Wang, Jingdong Wang, and Qi Tian. Good practice in cnn feature transfer. *arXiv preprint arXiv:1604.00133*, 2016. 2, 3

[67] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 1

[68] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022. 1, 2, 3, 5, 6, 7, 8

[69] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 1, 2, 3, 5, 6, 7, 8

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

[71] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *European conference on computer vision*, pages 561–578. Springer, 2020. 1

[72] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *European conference on computer vision*, pages 561–578. Springer, 2020. 1, 2, 5

[73] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. *arXiv preprint arXiv:2104.02008*, 2021. 1, 2

[74] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. *arXiv preprint arXiv:2205.14865*, 2022. 1, 2, 3, 5, 6, 7, 8