Bi-Level Optimization for Single Domain Generalization

Marzi Heidari*, Hanping Zhang*, Hao Yan*, Yuhong Guo*†
*School of Computer Science, Carleton University, Ottawa, Canada
†Canada CIFAR AI Chair, Amii, Canada
{marziheidari@cmail., jagzhang@cmail., haoyan6@cmail., yuhong.guo@}carleton.ca

Abstract

Generalizing from a single labeled source domain to unseen target domains, without access to any target data during training, remains a fundamental challenge in robust machine learning. We address this underexplored setting, known as Single Domain Generalization (SDG), by proposing BiSDG, a bi-level optimization framework that explicitly decouples task learning from domain modeling. BiSDG simulates distribution shifts through surrogate domains constructed via label-preserving transformations of the source data. To capture domain-specific context, we propose a domain prompt encoder that generates lightweight modulation signals to produce augmenting features via feature-wise linear modulation. The learning process is formulated as a bi-level optimization problem: the inner objective optimizes task performance under fixed prompts, while the outer objective maximizes generalization across the surrogate domains by updating the domain prompt encoder. We further develop a practical gradient approximation scheme that enables efficient bi-level training without second-order derivatives. Extensive experiments on various SGD benchmarks demonstrate that BiSDG consistently outperforms prior methods, setting new state-of-the-art performance in the SDG setting.

1. Introduction

Traditional deep learning methods typically assume that training and testing data share the same distribution, resulting in limited generalization to out-of-domain scenarios. Domain Generalization (DG) addresses this limitation by training models that can generalize to unseen target domains. Recent DG approaches either aim to learn domain-invariant representations via feature alignment [20, 22], simulate train-test splits through meta-learning [10, 18], or seek flat minima of empirical risk to improve robustness [3, 37]. However, conventional DG assumes access to data from multiple source domains, which restricts its applicability in data-sparse scenarios. Additionally, the typical evaluation setting involves testing on a single target domain, limiting assessment of generalization to diverse out-of-domain scenarios.

Single Domain Generalization (SDG) addresses these limitations by training on a single source domain while evaluating on multiple unseen target domains, thus posing a more challenging and realistic setting. Since standard DG approaches often rely on multi-domain data, they are not directly applicable to SDG. Early SDG methods [5, 6, 9] introduce various data augmentation techniques to enhance in-domain generalization and robustness to corruptions. Other methods [32, 39, 40] adopt adversarial data augmentation to improve out-of-domain generalization performance. Additionally, generative approaches [4, 19, 29] synthesize auxiliary data to aid generalization from a single domain. To date, most existing SDG methods adopt target-agnostic strategies, due to the constraint of not accessing target domains. Such strategies face a "lottery ticket" dilemma: models tend to perform well on target domains similar to the augmented source domain, but poorly on dissimilar ones.

This motivates us to propose a domain-aware method for SDG that respects the constraint of non-access to target domains, while achieving improved self-adaptability. We explicitly extract domain knowledge using a domain prompt encoder and inject this knowledge into the feature representations. The prediction model is trained to process these fused features effectively. Since a single domain cannot support training a generalizable domain prompt encoder alone, we simulate a diverse set of surrogate domains by applying groups of semantically coherent data transformations to the source domain data. Inspired by FiLM [28], we modulate features using domain-specific prompts via feature-wise linear modulation.

A key challenge is that joint optimization of the task model and the domain prompt encoder does not guarantee meaningful domain knowledge extraction or effective feature fusion to support generalization. This often leads to a trivial solution where both components collapse into a naive data-augmentation-based model. In practice, the

task model benefits from a well-trained domain prompt encoder, while the encoder's effectiveness also depends on a well-optimized task model. To address this interdependence, we formulate SDG as a bi-level optimization problem. The inner objective optimizes the task model using the original data from the training domain, while the outer objective optimizes the domain prompt encoder using data from the synthetic surrogate domains. We solve this bi-level objective using gradient estimation via implicit differentiation and gradient approximation. We evaluate our proposed method, Bi-level Optimization for Single Domain Generalization (BiSDG), under the standard SDG setting and compare it with existing baselines. Experimental results demonstrate that BiSDG achieves state-of-the-art performance on widely used SDG benchmarks.

2. Related Works

2.1. Single-Domain Generalization

Single Domain Generalization (SDG) aims to train models that can generalize to unseen test domains using data from only a single training domain. This setting is more challenging than traditional multi-source domain generalization, which assumes access to data from multiple training domains. Existing SDG approaches generally fall into three main groups.

The first group focuses on traditional data augmentation techniques to enhance in-domain generalization and robustness to corruptions. For instance, CutOut [9] improves regularization by randomly masking square regions in input images, encouraging robust feature learning. MixUp [36] improves generalization by training on convex combinations of input pairs and their labels. AugMix [12] enhances robustness and uncertainty estimation by stochastically mixing multiple augmentations of an image and applying a consistency loss. AutoAugment [5] learns optimal augmentation policies by exploring combinations of transformations with different probabilities and magnitudes, while RandAugment [6] simplifies the process by removing the search phase and using a reduced, interpretable space of augmentations. Although these techniques improve robustness and in-domain performance, they do not explicitly address out-of-domain generalization. A more recent approach, ACVC [7], applies traditional visual corruptions and enforces attention consistency between original and corrupted versions to improve domain generalization.

A second line of research introduces adversarial data augmentation methods aimed at out-of-domain generalization. ADA [32] proposes an iterative strategy that generates hard adversarial examples from fictitious target domains, using only single-source training data. ME-ADA [39] incorporates a regularization term derived from the information bottleneck principle to encourage high-entropy perturbations, improving robustness to domain shifts. Zhang et al. [38] propose adversarial perturbations applied to feature statistics, enabling models to learn representations that are robust to style variations. AdvST [40] treats standard augmentations as learnable semantic transformations and uses adversarial training to create diverse semantic variants, optimizing a distributionally robust objective to enhance generalization to unseen domains.

A third group of methods leverages generative modeling to synthesize auxiliary training data that simulate domain shifts. M-ADA [29] generates challenging fictitious domains through adversarial training in a meta-learning framework, guided by a Wasserstein Auto-Encoder. L2D [33] introduces a style-complement module that diversifies training data by synthesizing style-varied but semantically consistent images using mutual information. PDEN [19] expands the training domain through photometric and geometric transformations, guided by contrastive learning to enforce class separation and improve generalization. MCL [4] applies meta-causal learning to infer and align causal factors underlying domain shifts by simulating auxiliary domains. Xu et al. [34] generate low-confidence samples by maximizing entropy and minimizing cross-entropy, using dual-view generators guided by a classifier.

Despite their differences, most existing SDG methods adopt a target-agnostic design philosophy due to the constraint of non-access to target domains. This results in what we call the "lottery dilemma": models tend to perform better on target domains that are similar to the augmented training data, and worse on dissimilar ones. To address this limitation, we propose a domain-aware approach to SDG that explicitly extracts domain knowledge and injects it into the learning process to enhance adaptation and generalization. Our method uses a bi-level optimization framework, where a domain prompt encoder captures domain-specific knowledge, and the task model learns to effectively process the fused representation. This approach allows us to simulate domain shifts and enforce knowledge transfer, while respecting the constraint of not accessing target domain data.

2.2. Bi-Level Optimization

Bi-level optimization is a powerful framework that enables the joint optimization of a nested pair of objectives, where the outer (upper-level) objective depends implicitly on the solution of the inner (lower-level) problem. This approach has been widely adopted across a range of tasks, such as hyperparameter tuning [25], neural architecture search [21], and semi-supervised learning [**?**], due to its ability to model hierarchical dependencies in learning systems. Recent works leverage bi-level optimization to tackle domain generalization across practical scenarios. Jia and Zhang [13] introduce a meta-learning algorithm with inner-loop and outer-loop objectives on a discrepancy measure to learn

invariant representations. By adversarially minimizing both covariate and conditional shifts between source and simulated target domains in a bi-level setup, their method improves out-of-domain feature alignment and model robustness. To address data scarcity, Qin et al. [30] propose a two-tier meta-learning approach for few-shot domain generalization: an inner loop learns domain-specific embedding subspaces while an outer loop learns a shared base space, jointly optimized in a bi-level manner. Beyond closed-set shifts, Peng et al. [26] tackle open-set domain generalization with an evidential bi-level domain scheduler. Collectively, these studies demonstrate that integrating bi-level optimization into domain generalization pipelines can address a spectrum of practical challenges, leading to improved performance on unseen domains.

3. Method

3.1. Problem Setup

We study the task of Single Domain Generalization (SDG), where the objective is to train a model using labeled data from a single source domain and achieve robust performance on unseen target domains. Formally, let the source domain dataset be denoted by $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$, where each input-label pair $(\mathbf{x}_i, \mathbf{y}_i)$ is drawn i.i.d. from a distribution $\mathcal{P}_\text{s}$ over $\mathcal{X} \times \mathcal{Y}$. Our goal is to learn a prediction function $h_\phi \circ f_\theta : \mathcal{X} \to \mathcal{Y}$ that generalizes well to samples from an unknown target distribution $\mathcal{P}_\text{t} \neq \mathcal{P}_\text{s}$, which remains inaccessible during training. The prediction model is composed of a feature extractor $f_\theta : \mathcal{X} \to \mathcal{Z}$, parameterized by θ, and a classifier sub-network $h_\phi : \mathcal{Z} \to \mathcal{Y}$, parameterized by ϕ. The primary challenge of SDG is that no domain alignment, calibration, or adaptation is possible using target data; instead, the model must learn domain-invariant representations solely from the available source domain data.

3.2. Approach

We propose Bi-Level Optimization for Single-Domain Generalisation (BiSDG), a principled framework designed to tackle the challenge of generalizing from a single labeled domain to unseen target distributions. The core idea is to simulate plausible domain shifts during training and explicitly disentangle the learning of semantic representations from the modeling of domain-specific variations. To simulate distribution shifts, BiSDG constructs a diverse set of surrogate domains via stochastic, label-preserving transformations applied to source data. These synthetic domains expose the model to a controlled spectrum of low-level perturbations without requiring access to target samples. To capture domain-specific context, BiSDG introduces a domain prompt module that extracts a compact latent embedding from each domain. This embedding is used to condition the feature extractor via FiLM-based modulation, enabling adaptive feature normalization tailored to each domain's characteristics. Crucially, BiSDG formulates the learning process as a bi-level optimization problem. The lower-level objective updates the feature extractor and classifier to minimize task loss under fixed domain prompts, while the upper-level objective updates the prompt encoder to maximize generalization across the surrogate domains. This separation allows the backbone to focus on learning robust task-relevant features, while the prompt encoder is guided to emphasize variations that challenge generalization. The overall framework in shown in Figure 1.

3.2.1. Surrogate-Domain Synthesis

In the single-domain generalization setting, the absence of target-domain samples prohibits direct exposure to distribution shift. To mitigate this, we construct a diverse collection of surrogate domains that serve as plausible stand-ins for unknown target environments. These surrogate domains are derived by applying label-preserving image transformations to source data, thereby inducing systematic low-level variations while preserving semantic content.

Specifically, we define a transformation pool $\{\mathcal{A}_i\}_{i=1}^{N^\text{aug}}$ containing a diverse set of photometric and geometric augmentations, such as Gaussian blur, color jitter, and histogram equalization. Rather than selecting augmentations at random, we group them into semantically coherent transformation pipelines, where each surrogate domain $\mathcal{D}^{(k)}$ is associated with a specific type of shift, e.g., texture, photometric, or geometric. Each pipeline $\mathcal{T}^{(k)}$, composed of m augmentations, is applied consistently to all source samples to construct the k-th surrogate domain. Formally, we define:

$$\mathcal{D}^{(k)} = \left\{(\mathbf{x}_i^{(k)}, \mathbf{y}_i) : \mathbf{x}_i^{(k)} = \mathcal{T}^{(k)}(\mathbf{x}_i), (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}\right\}. \quad (1)$$

By deploying K such transformation pipelines, we obtain a family of surrogate domains $\{\mathcal{D}^{(k)}\}_{k=1}^{K}$ that span a broad spectrum of visual attributes. This augmentation-driven diversification compels the model to learn features that are stable under a variety of appearance shifts, thereby simulating the challenge of generalizing to unseen domains.

3.2.2. Domain Prompt Encoder

A major challenge in training a shared feature extractor across multiple domains lies in the risk of representation collapse, wherein the network converges to features that reflect an average over all domains, thereby suppressing meaningful domain-specific variations. To retain sensitivity to these variations while preserving parameter efficiency, we introduce a domain prompt encoder $\mathcal{M}_\omega$ that learns to generate domain-aware modulation signals.

For each surrogate domain $\mathcal{D}^{(k)}$, we associate a latent prompt instantiated through a pair of modulation vectors: a scaling vector $\boldsymbol{\gamma}^{(k)}$ and a shifting vector $\boldsymbol{\beta}^{(k)}$. These are inferred from a mini-batch $X_b^{(k)} = \{\mathbf{x}_j^{(k)}\}_{j=1}^{N^\mathcal{B}}$ sampled from

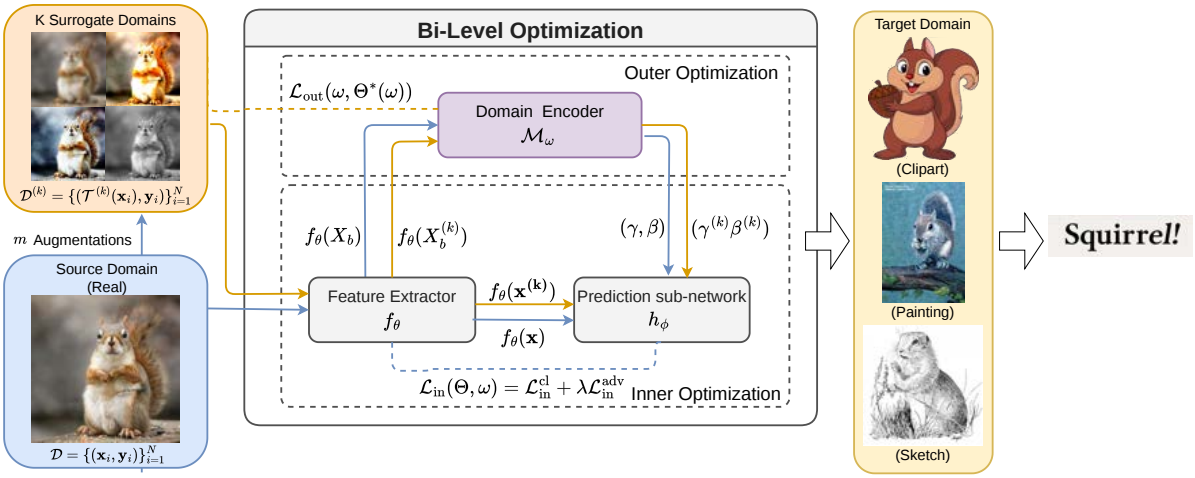


Figure 1. Overview of the BiSDG framework for Single Domain Generalization (SDG). Starting from a labeled source dataset, BiSDG constructs multiple surrogate domains via semantically coherent, label-preserving transformation pipelines to simulate unseen distribution shifts. A domain prompt encoder is introduced to capture domain-specific knowledge, which is deployed to perform FiLM-based feature modulation, supporting generalizable prediction. BiSDG formulates SGD as a bi-level optimization problem: the inner objective updates the model parameters $\Theta = (\theta, \phi)$ using the source domain, combining supervised cross-entropy loss $\mathcal{L}_{\text{in}}^{\text{cl}}$ and adversarial consistency loss $\mathcal{L}_{\text{in}}^{\text{adv}}$. The outer objective updates the prompt encoder parameters ω to maximize generalization across surrogate domains.

$\mathcal{D}^{(k)}$. To encode the domain-level distributional context in a permutation-invariant manner, we apply a domain encoder $\mathcal{M}_\omega$ over frozen features extracted by a stop-gradient copy of the backbone, denoted $f_{\bar{\theta}}$:

$$(\boldsymbol{\gamma}^{(k)}, \boldsymbol{\beta}^{(k)}) = \mathcal{M}_\omega\big(f_{\bar{\theta}}(X_b^{(k)})\big). \tag{2}$$

Here, $\mathcal{M}_\omega$ is implemented as a Set Transformer [16], which naturally handles unordered inputs and enables the prompt to summarize the aggregate style of the domain batch.

Given an input $\mathbf{x}_i^{(k)}$ from the surrogate domain k, we extract its features $\mathbf{z}_i^{(k)} = f_\theta(\mathbf{x}_i^{(k)})$ and apply feature-wise linear modulation (FiLM) [28] using the prompt-specific parameters, while the features are standardized using batch-wise statistics $(\boldsymbol{\mu}^{(k)}, \boldsymbol{\sigma}^{(k)})$ before modulation:

$$\tilde{\mathbf{z}}_i^{(k)} = \boldsymbol{\gamma}^{(k)} \odot \left(\frac{\mathbf{z}_i^{(k)} - \boldsymbol{\mu}^{(k)}}{\boldsymbol{\sigma}^{(k)}} \right) + \boldsymbol{\beta}^{(k)}, \tag{3}$$

where $\odot$ denotes element-wise multiplication. The resulting modulated features $\tilde{\mathbf{z}}_i^{(k)}$ are then fed into a domain-aware prediction head that conditions jointly on the original instance features and the domain prompt modulated features. We denote this modulated prediction sub-network as h_ϕ that includes both the modulation process in Eq. (3) and the prediction head, yielding predictions of the form $\hat{\mathbf{y}}_i^{(k)} = h_\phi(f_\theta(\mathbf{x}_i^{(k)}), \mathcal{M}_\omega(f_{\bar{\theta}}(X_b^{(k)})))$.

This FiLM-based strategy equips the model with lightweight, domain-adaptive behavior while keeping the shared backbone fixed across domains. Importantly, the modulation is global (feature-wise) rather than spatial, ensuring parameter efficiency and compatibility with standard architectures. By conditioning the model on learned domain prompts, we enable domain-informative inference to enhance adaptation and generalization.

3.2.3. Bi-Level Optimization for SDG

We formulate *Single-Domain Generalization* (SDG) as a bi-level optimization problem, in which the goal is to simultaneously learn a robust prediction model that generalizes well across domain shifts, and a domain prompt encoder that tackles domain variability via feature modulation. To decouple these two objectives, we introduce a bi-level formulation where the prediction model parameters are optimized at the inner level for task performance, while the prompt encoder is optimized at the outer level to maximize cross-domain generalization.

In this setting, the prediction model is parameterized by $\Theta = \{\theta, \phi\}$, where f_θ is the shared feature extractor and h_ϕ is the domain-aware prediction sub-network. At each iteration, we draw a batch with $N^{\mathcal{B}}$ labeled instances $(X_b, Y_b) = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N^{\mathcal{B}}}$ from the source domain and generate K surrogate domains $\{X_b^{(k)}\}_{k=1}^{K}$ via semantically coherent transformation pipelines.

Inner objective. Given fixed prompt parameters ω, we update the model parameters Θ by minimizing a loss that combines standard cross-entropy with an adversarial regularizer that promotes local smoothness under input perturbations. The inner supervised loss on a training batch

$(X_b, Y_b) = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N^{\mathcal{B}}}$ is defined as:

$$\mathcal{L}_{\text{in}}^{\text{cl}}(\Theta) = \frac{1}{N^{\mathcal{B}}} \sum_{i=1}^{N^{\mathcal{B}}} \ell_{\text{CE}}\big(h_\phi(f_\theta(\mathbf{x}_i), \mathcal{M}_\omega(f_{\bar{\theta}}(X_b))), \mathbf{y}_i\big), \tag{4}$$

where ℓ_{CE} is the cross-entropy loss and $\mathcal{M}_\omega(X_b)$ denotes the domain prompts derived from the batch. To improve robustness, we further include an adversarial KL term that encourages consistency under worst-case perturbations:

$$\mathcal{L}_{\text{in}}^{\text{adv}}(\Theta) = \frac{1}{N^{\mathcal{B}}} \sum_{i=1}^{N^{\mathcal{B}}} \max_{\|\epsilon\|_2 \le \rho} \text{KL}\Big(\begin{matrix} h_\phi(f_\theta(\mathbf{x}_i), \mathcal{M}_\omega(f_{\bar{\theta}}(X_b))), \\ h_\phi(f_\theta(\mathbf{x}_i + \epsilon), \mathcal{M}_\omega(f_{\bar{\theta}}(X_b))) \end{matrix}\Big), \tag{5}$$

where ρ controls the strength of the perturbation ϵ added to input $\mathbf{x}_i$. The total inner objective is:

$$\mathcal{L}_{\text{in}}(\Theta, \omega) = \mathcal{L}_{\text{in}}^{\text{cl}} + \lambda \mathcal{L}_{\text{in}}^{\text{adv}}, \tag{6}$$

where λ balances classification accuracy and adversarial robustness. Minimizing the inner loss in Eq. (6) over Θ yields an updated model $\Theta^*(\omega)$ given the current prompt encoder.

Outer objective. Given model parameters $\Theta^*(\omega)$, the outer objective updates ω by optimizing generalization performance across all the surrogate domains. Specifically, for each surrogate domain k, we sample a batch $(X_b^{(k)}, Y_b) = \{(\mathbf{x}_i^{(k)}, \mathbf{y}_i)\}_{i=1}^{N^{\mathcal{B}}}$, and compute the outer objective as the cross-entropy loss on all the K surrogate domains:

$$\mathcal{L}_{\text{out}}(\omega, \Theta^*(\omega)) = \frac{1}{K \cdot N^{\mathcal{B}}} \sum_{k=1}^{K} \sum_{i=1}^{N^{\mathcal{B}}} \ell_{\text{CE}}\left(\hat{\mathbf{y}}_i^{(k)}, \mathbf{y}_i\right), \tag{7}$$

$$\text{where} \quad \hat{\mathbf{y}}_i^{(k)} = h_{\phi^*}(f_{\theta^*}(\mathbf{x}_i^{(k)}), \mathcal{M}_\omega(f_{\bar{\theta}^*}(X_b^{(k)}))).$$

Here, $\Theta^*(\omega) = \{\theta^*, \phi^*\}$ denote the inner-optimal parameters that can be treated as functions of the outer parameters ω. By minimizing this loss, the prompt encoder learns to automatically adapt the prediction model to enhance its generalization performance.

Bi-level formulation. Combining the inner and outer objectives, BiSDG is formulated as the following bi-level optimization problem:

$$\begin{aligned} \min_{\omega} \quad & \mathcal{L}_{\text{out}}(\omega, \Theta^*(\omega)) \\ \text{s.t.} \quad & \Theta^*(\omega) = \arg\min_{\Theta} \mathcal{L}_{\text{in}}(\Theta, \omega). \end{aligned} \tag{8}$$

This training scheme enables the model to learn generalizable representations, while encouraging the prompt encoder to expose informative cross-domain variability to enhance adaptation. The optimization procedure will be illustrated in the following section and the bi-level training algorithm in presented in Algorithm 1.

Algorithm 1 Training Algorithm for BiSDG

Input: training dataset $\mathcal{D}$; initialized parameters $\Theta = (\theta, \phi)$ and ω; K transformation pipelines $\{\mathcal{T}_k\}_{k=1}^K$; and hyperparameters.
Output: learned model parameters Θ^* and ω^*

```
Generate K surrogate domains {D^(k)}_{k=1}^K via Eq. (1).
Set t = 1, Θ^1 = Θ.
for iter = 1 to maxiters do
  for minibatch X_b ∈ D do
    Sample K surrogate variants {X_b^(k)}_{k=1}^K of X_b from surrogate domains {D^(k)}_{k=1}^K.
    Compute inner loss L_in on X_b via Eqs. (4), (5), (6)
    Calculate Θ^{t+1} using Eq. (10)
    Compute L_out(ω, Θ^{t+1}) on surrogate batches {X_b^(k)}_{k=1}^K via Eq. (7)
    Calculate δ(Θ^{t+1}) using Eq. (11)
    Calculate gradient ∇_ω L_out on X_b and {X_b^(k)}_{k=1}^K via Eq. (14)
    Update ω ← ω − α_ω ∇_ω L_out
    Update Θ^{t+1} ← Θ^t − α_Θ ∇_Θ L_in(Θ^t, ω)
    t ← t + 1
  end for
end for
Θ* = Θ^t,   ω* = ω
```

3.3. Optimization Procedure

The bi-level optimization problem in Eq. (8) can be solved by minimizing the outer objective with respect to the outer parameters ω, while the inner-level parameters Θ^* are implicit functions of ω determined by the inner minimization.

In each iteration of the minimization, we compute the gradient of the outer loss with respect to ω and account for the bi-level structure using the chain rule as follows:

$$\nabla_\omega \mathcal{L}_{\text{out}} = \nabla_{\Theta^*} \mathcal{L}_{\text{out}} \cdot \nabla_\omega \Theta^* + \nabla_\omega \mathcal{L}_{\text{out}}(\omega, \bar{\Theta}^*), \tag{9}$$

where $\bar{\Theta}^*$ denotes the stop gradient version of Θ^*. Here, the first term represents the indirect gradient path through Θ^* using the chain rule, while the second term captures the direct dependence of the outer loss on ω.

For simplicity, at the t-th iteration, the inner-optimal parameters Θ^* can be approximated using a single-step gradient descent update over the inner loss:

$$\Theta^* = \Theta^{t+1} = \Theta^t - \alpha_\Theta \nabla_\Theta \mathcal{L}_{\text{in}}(\Theta^t, \omega), \tag{10}$$

where α_Θ is the learning rate for the respective update. Moreover, we denote the outer gradient over Θ as $\delta(\Theta)$, such that

$$\delta(\Theta^{t+1}) = \nabla_\Theta \mathcal{L}_{\text{out}}(\omega, \Theta^{t+1}). \tag{11}$$

The full gradient computation can then be expressed through the following propositions.

Proposition 1. *Using the chain rule, at the t-th iteration, the total gradient of $\mathcal{L}_{out}$ with respect to ω is given by:*

$$\begin{aligned}\nabla_\omega \mathcal{L}_{out} = & -\alpha_\Theta \cdot \delta(\Theta^{t+1}) \cdot \nabla_\omega \nabla_\Theta \mathcal{L}_{in}(\Theta^t, \omega) \\ & + \nabla_\omega \mathcal{L}_{out}(\omega, \bar{\Theta}^{t+1})\end{aligned} \tag{12}$$

Proof. From Eq. (10), the gradient of Θ^* with respect to ω can be computed as:

$$\begin{aligned}\nabla_\omega \Theta^* = \nabla_\omega \Theta^{t+1} &= \nabla_\omega \left(\Theta^t - \alpha_\Theta \nabla_\Theta \mathcal{L}_{\text{in}}(\Theta^t, \omega)\right) \\ &= -\alpha_\Theta \cdot \nabla_\omega \nabla_\Theta \mathcal{L}_{\text{in}}(\Theta^t, \omega).\end{aligned} \tag{13}$$

Substituting this expression into Eq. (9), we obtain Eq.(12). □

Considering the difficulty of computing the second-order derivative $\nabla_\omega \nabla_\Theta \mathcal{L}_{\text{in}}(\Theta^t, \omega)$, we propose using a finite difference approximation to avoid the expensive computation while retaining the influence of second-order terms. This is summarized in the following proposition.

Proposition 2. *Let ϵ_Θ be a very small positive constant. The total gradient $\nabla_\omega \mathcal{L}_{out}$ can be approximated as follows without directly computing second-order derivatives:*

$$\begin{aligned}\nabla_\omega \mathcal{L}_{out} \approx & \nabla_\omega \mathcal{L}_{out}(\omega, \bar{\Theta}^*) - \\ & \frac{\alpha_\Theta}{2\epsilon_\Theta} \left(\nabla_\omega \mathcal{L}_{in}(\Theta^+, \omega) - \nabla_\omega \mathcal{L}_{in}(\Theta^-, \omega)\right),\end{aligned} \tag{14}$$

where the perturbed variables Θ^+ and Θ^- are defined as:

$$\begin{aligned}\Theta^+ &= \Theta^t + \epsilon_\Theta \cdot \delta(\Theta^{t+1}), \\ \Theta^- &= \Theta^t - \epsilon_\Theta \cdot \delta(\Theta^{t+1}).\end{aligned} \tag{15}$$

Proof. We approximate the second-order term $\nabla_\omega \nabla_\theta \mathcal{L}_{\text{in}}$ using symmetric finite differences [1]. Specifically, using the small perturbation $\epsilon_\Theta \cdot \delta(\Theta^{t+1})$, we write:

$$\nabla_\Theta \mathcal{L}_{\text{in}}(\Theta^t, \omega) \approx \frac{\mathcal{L}_{\text{in}}(\Theta^+, \omega) - \mathcal{L}_{\text{in}}(\Theta^-, \omega)}{2\epsilon_\Theta \cdot \delta(\Theta^{t+1})} \tag{16}$$

Therefore, the chain rule term in the total gradient becomes:

$$\begin{aligned}&\nabla_{\Theta^*} \mathcal{L}_{\text{out}} \cdot \nabla_\omega \Theta^* \\ &= -\alpha_\Theta \cdot \delta(\Theta^{t+1}) \cdot \nabla_\omega \nabla_\Theta \mathcal{L}_{\text{in}}(\Theta^t, , \omega) \\ &\approx -\alpha_\Theta \cdot \delta(\Theta^{t+1}) \cdot \frac{\nabla_\omega \mathcal{L}_{\text{in}}(\Theta^+; \omega) - \nabla_\omega \mathcal{L}_{\text{in}}(\Theta^-, \omega)}{2\epsilon_\Theta \cdot \delta(\Theta^{t+1})} \\ &= -\frac{\alpha_\Theta}{2\epsilon_\Theta} \left(\nabla_\omega \mathcal{L}_{\text{in}}(\Theta^+, \omega) - \nabla_\omega \mathcal{L}_{\text{in}}(\Theta^-, \omega)\right)\end{aligned} \tag{17}$$

Substituting this approximation into Eq. (9), we obtain the approximated total gradient in Eq.(14). □

4. Experiments

4.1. Experimental Setup

Datasets. We evaluate our approach on three widely-used domain generalization benchmarks: Digits, PACS, and DomainNet. The **Digits** benchmark consists of five digit classification datasets, MNIST [15], MNIST-M [11], SVHN [24], SYN [11], and USPS [8], all sharing the same 10 digit classes (0–9). Following standard practice, we use MNIST as the source domain and evaluate generalization on the remaining four datasets, each exhibiting distinct visual styles and acquisition conditions. The **PACS** dataset [17] contains images from four distinct domains, Art, Cartoon, Photo, and Sketch, featuring seven shared object categories. We designate the Photo domain as the source and test generalization to the remaining three domains, which differ substantially in terms of texture, abstraction, and visual style. The **DomainNet** dataset [27] is the most challenging benchmark in our study, comprising six highly diverse domains: Real, Infograph, Clipart, Painting, Quickdraw, and Sketch, spanning 345 object categories. We use the Real domain for training and evaluate performance on the remaining five, which exhibit large domain shifts and high intra-class variability.

Implementation Details. We follow established protocols for architecture and training across all datasets, consistent with prior work [33]. For the Digits benchmark, we employ LeNet as the backbone and train on the first 10,000 images from MNIST, with all samples resized to 32×32 and converted to RGB. The model is trained for 50 epochs using a batch size of 32 and initial learning rates of $\alpha_\Theta = 10^{-4}$ and $\alpha_\omega = 10^{-5}$, which are reduced by a factor of 10 after 25 epochs. For PACS, we fine-tune a ResNet-18 model pretrained on ImageNet, using images resized to 224×224. Training is conducted for 50 epochs with a batch size of 32 and learning rates of $\alpha_\Theta = 10^{-3}$ and $\alpha_\omega = 10^{-4}$, decayed using a cosine annealing schedule. For the DomainNet benchmark, we again use ResNet-18 and train for 200 epochs with a batch size of 128, also using cosine learning rate scheduling. We use the following hyperparameters for BiSDG: adversarial loss weight $\lambda = 0.5$, number of surrogate domains $K = 5$, number of augmentations per surrogate domain $m = 3$, gradient approximation scale $\epsilon_\Theta = 0.01$, and perturbation strength $\rho = 1$. To simulate diverse domain shifts during training, we construct five surrogate domains by composing distinct triplets of augmentations from a shared transformation pool. The Color-Shifted Domain applies HSV shift, contrast adjustment, and solarization to simulate lighting and sensor variation. The Geometric Distortion Domain introduces rotation, translation, and shear to capture spatial deformations. The Photometric Degradation Domain uses inversion, posterization, and histogram equalization to mimic degraded imaging condi-

Table 1. Classification accuracy and standard deviation results (%) on the PACS dataset. Best results are in bold font.

Target	MixUp	CutOut	ADA	ME-ADA	AugMix	RandAug	ACVC	L2D	AdvST	BiSDG (Ours)
Art	52.8	59.8	58.0	60.7	63.9	67.8	67.8	67.6	$69.2_{(1.4)}$	$\mathbf{70.5}_{(1.4)}$
Cartoon	17.0	21.6	25.3	28.5	27.7	28.9	30.3	42.6	$55.3_{(2.0)}$	$\mathbf{55.9}_{(1.7)}$
Sketch	23.2	28.8	30.1	29.6	30.9	37.0	46.4	47.1	$67.7_{(1.5)}$	$\mathbf{69.5}_{(1.9)}$
Avg.	31.0	36.7	37.8	39.6	40.8	44.6	48.2	52.5	64.1	**65.3**

Table 2. Classification accuracy and standard deviation results (%) on the four target domains (SVHN, MNIST-M, SYN, and USPS) of the Digits benchmark, with MNIST as the source domain. Best results are in bold font.

Method	SVHN	MNIST-M	SYN	USPS	Avg.
ERM [14]	27.8	52.7	39.7	76.9	49.3
CCSA [23]	25.9	49.3	37.3	83.7	49.1
JiGen [2]	33.8	57.8	43.8	77.2	53.1
ADA [32]	35.5	60.4	45.3	77.3	54.6
ME-ADA [39]	42.6	63.3	50.4	81.0	59.3
M-ADA [29]	42.6	67.9	49.0	78.5	59.5
AutoAug [5]	45.2	60.5	64.5	80.6	62.7
RandAug [6]	54.8	74.0	59.6	77.3	66.4
PDEN [19]	62.2	82.2	69.4	85.3	74.8
MCL[4]	69.9	78.3	78.4	88.5	78.8
RSDA [31]	$47.7_{(4.8)}$	$81.5_{(1.6)}$	$62.0_{(1.2)}$	$83.1_{(1.2)}$	68.5
AdvST [40]	$67.5_{(0.7)}$	$79.8_{(0.7)}$	$78.1_{(0.9)}$	$94.8_{(0.4)}$	80.1
BiSDG (Ours)	$\mathbf{70.1}_{(0.6)}$	$\mathbf{85.2}_{(0.3)}$	$\mathbf{79.8}_{(0.5)}$	$\mathbf{96.0}_{(0.2)}$	**82.7**

tions. The Texture Alteration Domain perturbs fine textures via sharpening, cutout, and contrast changes. Lastly, the Scale and Shape Variation Domain applies scaling, rotation, and cutout to model geometric variation and partial occlusion. These curated domains introduce controlled but substantial variability, promoting generalization to unseen distributions. All experiments are repeated five times with different random seeds, and we report the mean accuracy and standard deviation to ensure statistical robustness.

4.2. Comparison Results

We evaluate our method against a broad spectrum of baselines, including MixUp [36], CutOut [9], CutMix [35], AutoAugment [5], RandAugment [6], and AugMix [12], ACVC [7], ERM [14], CCSA [23], JiGen [2], ADA [32], ME-ADA [39], RSDA [31], L2D [33], PDEN [19], MCL [4], and AdvST [40].

Table 1 reports the test classification accuracy on the three target domains of the PACS dataset using ResNet-18 as the backbone. This benchmark features significant domain shifts between the Photo source domain and the target domains of Art, Cartoon, and Sketch. BiSDG achieves the highest accuracy on all three target domains and the best overall average accuracy of 65.3%, outperforming the strongest baseline, AdvST (64.1%), by 1.2%. Notably, BiSDG achieves an accuracy of 55.9% on the most challenging domain, Cartoon, compared to 55.3% by AdvST. These results highlight the effectiveness of our method in learning semantically meaningful features that generalize across domains with diverse visuals.

In Table 2, we report generalization results on a digit recognition benchmark, Digits, using MNIST as the source domain and SVHN, MNIST-M, SYN, and USPS as target domains. BiSDG significantly outperforms all prior methods, achieving an average accuracy of 82.7%, surpassing the second-best method (AdvST at 80.1%) by a substantial margin. BiSDG obtains the best results on all four target domains, including 70.1% on SVHN, 85.2% on MNIST-M, 79.8% on SYN, and 96.0% on USPS.

Table 3 presents results on the challenging DomainNet benchmark, which involves large-scale classification across multiple domains with significant visual diversity. BiSDG achieves the best performance on all five target domains, with an average accuracy of 28.3%, outperforming AdvST (27.1%) by 1.2%. The improvements are especially notable on Infograph (15.9% vs. 14.8%) and Quickdraw (7.1% vs. 5.9%). These results confirm that BiSDG effectively captures core features that generalize across unseen domains despite substantial distribution shifts.

4.3. Ablation Studies

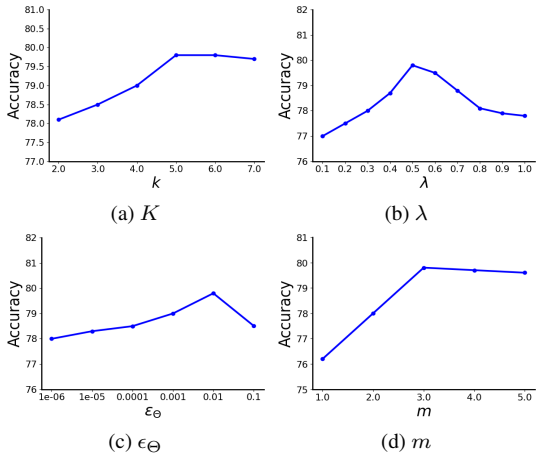
To evaluate the effectiveness of key components in our framework, we design two ablation variants of BiSDG: (1) $-$w/o $\mathcal{L}_{\text{in}}^{\text{adv}}$, which removes the adversarial regularizer from the inner objective to assess its contribution to robust feature learning; and (2) $-$w/o standardization, which disables the feature standardization step applied before domain-aware modulation, in order to examine the role of normalization

Table 3. Classification accuracy and standard deviation results (%) on the DomainNet dataset.

Target	MixUp	CutOut	CutMix	ADA	ME-ADA	RandAug	AugMix	ACVC	AdvST	BiSDG (Ours)
Painting	38.6	38.3	38.3	38.2	39.0	41.3	40.8	41.3	$42.3_{(0.1)}$	$\mathbf{43.1}_{(0.3)}$
Infograph	13.9	13.7	13.5	13.8	14.0	13.6	13.9	12.9	$14.8_{(0.1)}$	$\mathbf{15.9}_{(0.3)}$
Clipart	38.0	38.4	38.7	40.2	41.0	41.1	41.7	42.8	$41.5_{(0.4)}$	$\mathbf{43.5}_{(0.4)}$
Sketch	26.0	26.2	26.9	24.8	25.3	30.4	29.8	30.9	$30.8_{(0.3)}$	$\mathbf{31.9}_{(0.4)}$
Quickdraw	3.7	3.7	3.6	4.3	4.3	5.3	6.3	6.6	$5.9_{(0.2)}$	$\mathbf{7.1}_{(0.2)}$
Avg.	24.0	24.1	24.2	24.3	24.7	26.3	26.5	26.9	$27.1_{(0.2)}$	**28.3**

Table 4. Ablation study results (%) on the four target domains of the Digits benchmark, with MNIST as the source domain.

Method	SVHN	MNIST-M	SYN	USPS	Avg.
BiSDG (Ours)	$\mathbf{70.1}_{(0.6)}$	$\mathbf{85.2}_{(0.3)}$	$\mathbf{79.8}_{(0.5)}$	$\mathbf{96.0}_{(0.2)}$	**82.7**
−w/o $\mathcal{L}_{\text{in}}^{\text{adv}}$	$68.1_{(1.4)}$	$80.1_{(0.6)}$	$75.3_{(0.3)}$	$93.3_{(0.5)}$	79.2
−w/o standardization	$69.8_{(0.9)}$	$83.5_{(0.7)}$	$77.0_{(0.4)}$	$94.7_{(0.2)}$	81.2



(a) K (b) λ (c) ϵ_Θ (d) m

Figure 2. Sensitivity analysis for four hyper-parameters K, λ, ϵ_Θ and m on test domain SYN of the Digits benchmark.

in stabilizing prompt-guided adaptation. As shown in Table 4, both components play a crucial role in the final performance. Removing the adversarial loss $\mathcal{L}_{\text{in}}^{\text{adv}}$ results in a significant drop in average accuracy from 82.7% to 79.2%, confirming that enforcing consistency under perturbations is essential for improving robustness to domain shifts. Similarly, omitting the standardization step reduces performance to 81.2%, with particularly noticeable declines on SVHN and SYN, two domains that exhibit large visual disparities from MNIST. These findings demonstrate that both adversarial consistency and standardized feature modulation are critical to BiSDG's strong generalization performance.

4.4. Hyper-parameter Sensitivity Analysis

We perform a comprehensive sensitivity analysis to assess the robustness of BiSDG with respect to key hyperparameters. Specifically, we investigate the impact of the number of surrogate domains K, the trade-off weight λ for the adversarial loss in the inner objective, the perturbation magnitude ϵ_Θ used in the finite difference gradient approximation, and the number of augmentations m used to construct each surrogate domain. All experiments are conducted on the Digits benchmark, using classification accuracy on the SYN domain as the evaluation metric.

As shown in Figure 2, BiSDG exhibits relatively stable performance across a wide range of hyperparameter settings. Increasing the number of surrogate domains K (Figure 2a) and the number of augmentations m (Figure 2d) initially improves performance, reaching optimal values at $K = 5$ and $m = 3$, respectively. Beyond these points, performance shows a slight decline as the hyperparameter values increase, but the overall drop is marginal, indicating that BiSDG maintains strong generalization capability without requiring precise tuning of these parameters. The trade-off weight λ and perturbation magnitude ϵ_Θ exhibit a similar trend (Figures 2b and 2c), though performance degrades more rapidly after reaching their optimal values at $\lambda = 0.5$ and $\epsilon_\Theta = 0.01$. This suggests that these two hyperparameters are more sensitive and play a more critical role in determining model performance.

5. Conclusion

We have introduced BiSDG, a bi-level optimization framework designed for the challenging setting of Single Domain Generalization (SDG). By simulating domain shifts through curated surrogate domains and decoupling domain-specific modulation from task learning, BiSDG effectively exposes the model to distributional variability while maintaining semantic consistency. The proposed domain prompt encoder enables lightweight yet expressive domain-aware adaptation. Our bi-level formulation ensures that the backbone focuses on robust representation learning while the prompt encoder is optimized to induce generalization over domain shifts. Extensive experiments on multiple benchmarks demonstrate that BiSDG consistently outperforms existing state-of-the-art approaches.

References

[1] Léon Bottou. Stochastic gradient descent tricks. *Neural Networks: Tricks of the Trade: Second Edition*, 2012. 6

[2] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 7

[3] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. In *Neural Information Processing Systems (NeurIPS)*, 2021. 1

[4] Jin Chen, Zhi Gao, Xinxiao Wu, and Jiebo Luo. Meta-causal learning for single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2, 7

[5] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 7

[6] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2, 7

[7] Ilke Cugu, Massimiliano Mancini, Yanbei Chen, and Zeynep Akata. Attention consistency on visual corruptions for single-source domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2022. 2, 7

[8] John S Denker, WR Gardner, Hans Peter Graf, Donnie Henderson, Richard E Howard, W Hubbard, Lawrence D Jackel, Henry S Baird, and Isabelle Guyon. Neural network recognizer for hand-written zip code digits. In *Advances in Neural Information Processing Systems (NeurIPS)*, 1989. 6

[9] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 1, 2, 7

[10] Yingjun Du, Jun Xu, Huan Xiong, Qiang Qiu, Xiantong Zhen, Cees GM Snoek, and Ling Shao. Learning to learn with variational information bottleneck for domain generalization. In *European Conference on Computer Vision (ECCV)*. Springer, 2020. 1

[11] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning (ICML)*, 2015. 6

[12] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations (ICLR)*, 2019. 2, 7

[13] Chen Jia and Yue Zhang. Meta-learning the invariant representation for domain generalization. *Machine Learning*, 2024. 2

[14] Vladimir Koltchinskii. *Oracle inequalities in empirical risk minimization and sparse recovery problems: École D'Été de Probabilités de Saint-Flour XXXVIII-2008*. Springer Science & Business Media, 2011. 7

[15] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. 6

[16] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning (ICML)*. PMLR, 2019. 4

[17] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017. 6

[18] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence (AAAI)*, 2018. 1

[19] Lei Li, Ke Gao, Juan Cao, Ziyao Huang, Yepeng Weng, Xiaoyue Mi, Zhengze Yu, Xiaoya Li, and Boyang Xia. Progressive domain expansion network for single domain generalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 2, 7

[20] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *European Conference on Computer Vision (ECCV)*, 2018. 1

[21] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. In *International Conference on Learning Representations (ICLR)*, 2019. 2

[22] Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. In *International Conference on Machine Learning (ICML)*, 2021. 1

[23] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017. 7

[24] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. *NeurIPS workshop on deep learning and unsupervised feature learning (NeurIPSW)*, 2011. 6

[25] Fabian Pedregosa. Hyperparameter optimization with approximate gradient. In *International conference on machine learning (ICML)*, 2016. 2

[26] Kunyu Peng, Di Wen, Kailun Yang, Ao Luo, Yufan Chen, Jia Fu, M Saquib Sarfraz, Alina Roitberg, and Rainer Stiefelhagen. Advancing open-set domain generalization using evidential bi-level hardest domain scheduler. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 3

[27] 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 (ICCV)*, 2019. 6

[28] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence (AAAI)*, 2018. 1, 4

[29] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 7

[30] Xiaorong Qin, Xinhang Song, and Shuqiang Jiang. Bi-level meta-learning for few-shot domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2023. 3

[31] Riccardo Volpi and Vittorio Murino. Addressing model vulnerability to distributional shifts over image transformation sets. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 7

[32] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 1, 2, 7

[33] 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 (ICCV)*, 2021. 2, 6, 7

[34] Qinwei Xu, Ruipeng Zhang, Yi-Yan Wu, Ya Zhang, Ning Liu, and Yanfeng Wang. Simde: A simple domain expansion approach for single-source domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[35] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, 2019. 7

[36] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018. 2, 7

[37] Xingxuan Zhang, Renzhe Xu, Han Yu, Yancheng Dong, Pengfei Tian, and Peng Cui. Flatness-aware minimization for domain generalization. In *International Conference on Computer Vision (ICCV)*, 2023. 1

[38] Yabin Zhang, Bin Deng, Ruihuang Li, Kui Jia, and Lei Zhang. Adversarial style augmentation for domain generalization. In *arXiv preprint arXiv:2301.12643*, 2023. 2

[39] Long Zhao, Ting Liu, Xi Peng, and Dimitris Metaxas. Maximum-entropy adversarial data augmentation for improved generalization and robustness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2, 7

[40] Guangtao Zheng, Mengdi Huai, and Aidong Zhang. Advst: Revisiting data augmentations for single domain generalization. In *Proceedings of the AAAI conference on artificial intelligence (AAAI)*, 2024. 1, 2, 7