Single-Domain Generalization via Path Flatness-Aware Optimization of Loss Landscapes

Zizhou Wang, Yan Wang, Yangqin Feng, Jiawei Du, Joey Tianyi Zhou, *Senior Member, IEEE*, Rick Siow Mong Goh, *Senior Member, IEEE*, Yong Liu, *Senior Member, IEEE*, and Liangli Zhen, *Senior Member, IEEE*

***Abstract*—Domain generalization methods traditionally rely on multiple source domains to achieve robust performance across unseen target domains. However, single-domain generalization (SDG) presents a more practical paradigm by learning from a single source domain, addressing scenarios where access to multiple domains is limited. While existing SDG approaches primarily focus on data augmentation and style transfer techniques to enhance model robustness, these methods often incur substantial computational overhead and may inadequately capture the complexity of real-world domain shifts. In this paper, we propose Path Flatness-aware Optimization (PFO), an optimization framework that addresses the fundamental challenges of SDG. Unlike conventional approaches that rely on synthetic data generation, PFO identifies and exploits regions of flat minima within the optimization landscape of deep neural networks. The framework employs an iterative optimization strategy to construct a path through the parameter space along which an ensemble of candidate models achieves minimal empirical risk. The initialization of this optimization path is achieved through the strategic interconnection of model instances, each originating from carefully selected anchor points that are computationally determined through systematic analysis of classification decision manifolds. This optimization path serves as a mechanism for implicit distribution alignment between source and target domains within the loss landscape, consequently enhancing the model's capacity for cross-domain generalization. Empirical evaluation on multiple benchmark datasets demonstrates significant performance improvements in cross-domain generalization, validating the efficacy of our approach.**

***Index Terms*—Single domain generalization, domain generalization, path flatness-aware optimization, deep model optimization.**

I. Introduction

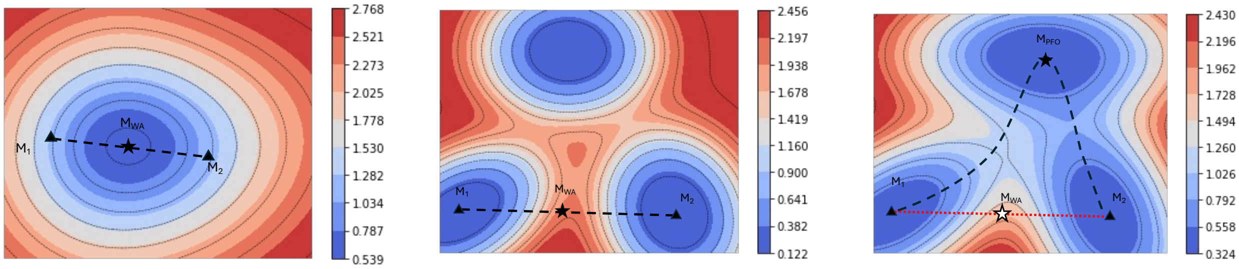
MACHINE learning systems face a fundamental challenge when deployed in real-world environments: the inherent discrepancy between training and testing data distributions, commonly known as domain shift. While conventional machine learning frameworks operate under the assumption of Independent and Identically Distributed (IID) data across training and testing sets [1], [2], this assumption frequently proves inadequate in practical applications. The performance degradation resulting from domain shift has emerged as a critical concern in contemporary machine learning research [3]–[5]. Domain Generalization (DG) has emerged as a promising framework for addressing this challenge, focusing on training models that maintain robust performance across both source and unseen target domains [6], [7]. Current state-of-the-art approaches in DG predominantly employ feature disentanglement techniques, which aim to isolate domain-invariant features while minimizing the impact of domain-specific characteristics [3]. While these methods have demonstrated considerable success [8], their reliance on multiple source domains presents a significant limitation for real-world applications where such diverse training data may be unavailable.

In this paper, we address the more challenging yet practical paradigm of Single-Domain Generalization (SDG), where models must generalize to unseen target domains using training data from only a single source domain [10]. SDG presents two fundamental challenges: 1) the limited availability of training samples for learning the underlying data distribution, and 2) the absence of knowledge about the relationships across multiple domains, hindering the training of a reliable predictive model. Current SDG approaches primarily focus on enhancing model generalization by diversifying source domain data [11]. This is achieved through various data perturbation techniques during training, including the addition of Gaussian noise or the generation of new data via operations like shearing and rotation. These techniques boost the model's robustness to disturbances and its adaptability to new domains, thereby improving generalization in real-world scenarios. These methods emphasize data-centric solutions, specifically enlarging the source domain dataset by creating more varied data. This expansion broadens the source domain's range but depends heavily on the quality of the generated data [6]. The data must be semantically coherent and recognizable, yet diverse enough in domain characteristics, such as stylistic variations, to be effective [12]. The challenge escalates with increasing image size and complexity of data composition. Moreover, generating synthetic data samples requires significant computational efforts for both model training and data generation. As the dataset complexity grows, so does the computational demand, posing a significant hurdle for practical implementations. However, the high computational cost of generating synthetic data limits their practicality.This has motivated another category of SDG strategies, which focuses on optimizing deep neural networks by explicitly identifying broader and

This work was supported by the National Research Foundation, Singapore, under its AI Singapore Programme (Award Numbers: AISG2-TC-2021-003 and AISG4-GC-2023-007-1B), and by the Singapore Digital Trust Centre through its Innovation Grant (Award Number: DTC-IGC-02). (Corresponding author: Liangli Zhen)

Z. Wang, Y. Wang, Y. Feng, Y. Liu, R. Goh and L. Zhen are with the Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore 138632, Singapore (e-mail: wang_zizhou@a-star.edu.sg, wangyan@a-star.edu.sg, fengyq@a-star.edu.sg, liuyong,@a-star.edu.sg gohsm@a-star.edu.sg, zhenll@a-star.edu.sg)

J. Du and J. Zhou are with the Centre for Frontier AI Research, Agency for Science, Technology and Research (A*STAR), Singapore 138632, Singapore (e-mail: dujw@a-star.edu.sg, zhouty@a-star.edu.sg)



(a) Optimal Case with Weight Averaging

(b) Sub-optimal Case with Weight Averaging

(c) PFO vs. Weight Averaging

Fig. 1: Comparison of our proposed method and the approach of Weight Averaging (WA) [9]. (a) Weight Averaging seeks to improve model generalization by averaging multiple models' weights along the optimization trajectory. The average of local minimum weights around the global minimum is expected to approximate the global minimum. (b) Weight Averaging may inadvertently result in a sub-optimal outcome under certain conditions. For example, such a situation can arise when the models being averaged have significantly different learned representations, leading to a divergence that challenges the viability of straightforward weight averaging between two local minima. (c) Our PFO strategy employs the concept of model connectivity to improve generalization by optimizing the entire path of model states. Our method ensures consistent high performance not just at individual points along the trajectory but throughout the entire learning process.

flatter optimal solutions within the loss landscape to enhance model generalization performance. For example, traditional model averaging approaches, represented by Stochastic Weight Averaging (SWA), improve generalization capability by averaging model weights along the optimization trajectory, as illustrated in Figure 1(a). As observed from the figure, this approach averages multiple points along the stochastic gradient descent (SGD) trajectory, achieving flatter minima compared to conventional SGD. However, as depicted in Figure 1(b), the SWA approach may inadvertently converge to suboptimal solutions when averaging substantially different local minima, thus impairing the generalization potential of the model.

To address this limitation, we propose a Path Flatness-aware Optimization (PFO) approach. Unlike existing methods, PFO explicitly constructs a continuous path connecting multiple anchor models within the weight space and ensures the stability of the entire optimization trajectory via optimization techniques. Consequently, the resulting model ensemble resides within a consistently flat region. By identifying and stabilizing such flat minima regions, our proposed strategy significantly enhances the cross-domain generalization ability of models, thereby facilitating a more robust alignment between source and target domain distributions. Our PFO method comprises two primary components: the preservation of flatness along the learning trajectory and the application of constraints on decision boundaries to generate diverse anchor models. These elements substantially improve the model's ability to generalize across various unseen domains. Specifically, PFO refines the decision boundary formulation and employs angle-space optimization to widen the decision boundaries, facilitating the generation of models with distinct and diverse boundaries. This optimization modulates the cosine of the decision boundary angle, leading to structurally coherent models with enhanced diversity. Further fine-tuning the decision boundary shift enhances the model's task-specific performance, increasing the diversity of models derived from varied initial conditions. A connection path is then established between these models to form an ensemble model. In our method, rather than optimizing individual models, we directly optimize the entire path, creating a flat landscape section to ensure robust generalization of the final model. Comprehensive experiments on four public benchmarks demonstrate the superiority of our method over various competing methods, with particularly notable improvements in complex scenarios.

The novelty and key contributions of this work are summarized as follows:

1) We propose **Path Flatness-aware Optimization (PFO)**, a novel framework for single-domain generalization that enhances cross-domain robustness by explicitly optimizing flat regions in the loss landscape, without relying on data augmentation techniques.
2) We introduce a **trajectory-aware optimization strategy** that constructs smooth parametric curves between anchor models in the weight space. This design ensures low-loss connectivity along the entire path, leading to improved generalization beyond isolated model checkpoints.
3) We develop a **dual-anchor diversification mechanism** that integrates angular space optimization with bias translation, enabling the construction of diverse yet structurally coherent decision boundaries. This enhances the model's adaptability to complex domain shifts.
4) Extensive experiments on multiple public benchmarks demonstrate that **PFO achieves state-of-the-art performance** under challenging single-domain generalization settings. Notably, it yields average accuracy gains over prior methods on CIFAR-100-C, PACS and OfficeHome, highlighting its robustness across diverse and complex scenarios.

II. Related work

A. Domain Generalization and Single Domain Generalization

Domain Generalization (DG) tackles the problem of domain shift by training models on multiple source domains to generalize effectively to unseen target domains. Existing DG methods are typically categorized into five paradigms: alignment, regularization, data augmentation, meta-learning, and feature disentanglement [13]–[15].

In contrast, Single Domain Generalization (SDG) presents a more challenging scenario, where models must generalize from only a single source domain. The absence of domain diversity makes learning robust invariant features substantially more difficult. To address this, SDG methods predominantly rely on data augmentation to simulate domain shifts and increase generalization. SDG approaches are generally grouped into adversarial and style-based augmentation. Adversarial methods perturb source images to generate hard examples, often by maximizing classification loss or entropy [10], [11]. Representative works include ADA [16], which injects adversarial examples during training, and ME-ADA [10], which expands domain coverage by maximizing prediction entropy. Style-based approaches, by contrast, synthesize out-of-distribution samples using generative models [17], [18]. M-ADA [6] leverages Wasserstein Auto-Encoders to increase style diversity, and CADA [19] employs angular center loss to generate samples that deviate from class centers. Additional techniques such as random convolution [20] introduce stylistic variation through texture alteration, while Crafting-Shifts [21] provides a principled framework for modeling domain shifts via synthetic distribution generation with theoretical guarantees. Moreover, MetaCNN [12] introduces meta-convolutional layers to decompose features into reusable and task-relevant components, effectively filtering out domain-specific noise. SRCD [22] introduces semantic reasoning with compound domain representations to enhance single-domain generalized object detection by bridging visual and semantic spaces, achieving superior performance on unseen categories. While these SDG methods have shown promising results, they often depend on computationally intensive, data-centric strategies, which may still fall short in capturing the full complexity of real-world domain shifts. This underscores the need for more efficient and theoretically grounded solutions for practical deployment.

B. Flatness-Aware Optimization

Enhancing model generalization under distributional shifts remains a fundamental challenge in domain generalization (DG). Recent advances have highlighted flatness-aware optimization as a promising strategy for improving model robustness, often complementing traditional model averaging approaches that implicitly promote flatness in the loss landscape. Model averaging techniques, which consolidate checkpoints along training trajectories to converge toward flatter loss regions, have demonstrated considerable success. Stochastic weight averaging (SWA) and its domain-adaptive variant SWAD [9], [23] exemplify this approach. The ”Model soups” methodology [24] further extends this paradigm by employing weight averaging algorithms to achieve superior performance across multiple tasks. Additionally, DiWA and EoA emphasize the importance of model diversity in the averaging process to enhance generalization capabilities [25], [26]. Despite their efficacy, these methods may underperform when averaged solutions fall between poorly aligned optimization modes.

From another perspective, explicit flatness-aware optimization directly pursues solutions residing in broad, low-loss valleys. Sharpness-Aware Minimization (SAM) [27] pioneers this approach by introducing a surrogate objective that minimizes neighborhood sharpness, subsequently inspiring several DG-specific extensions. Gradient-Aligned Minimization (GAM) [28] integrates gradient alignment with flatness objectives, while Sharpness-Aware Group Minimization (SAGM) [29] enforces sharpness minimization across heterogeneous domains. Inconsistency-Aware Domain Adaptation (IADA) [30] mitigates domain inconsistencies through adversarial perturbations, indirectly promoting flatter minima. Zhang *et al.* [31] proposed a unified Flatness-Aware Minimization (FAM) framework to penalize sharp loss directions, while Li *et al.* [32] refined the loss landscape to identify consistent flat minima shared across domains, thereby improving generalization performance and training stability. Advancing beyond pointwise solutions, trajectory-based approaches optimize continuous low-loss paths connecting multiple models, ensuring smoothness throughout the parameter space while directly regulating the solution geometry [33], [34]. These methods often incorporate diversity constraints to enhance functional complementarity between models along the trajectory. Building upon this perspective, our proposed Path Flatness-aware Optimization (PFO) framework learns a parameterized path across diverse models while enforcing uniform flatness along the entire trajectory. Our approach yields more robust and transferable representations for unseen target domains, addressing a critical challenge in SDG.

III. Our Proposed Method

We present a novel method to improve the model generalization by constraining the training error based on the flat path. Different from the common multi-source domain generalization, the number of source domain is one and there may exist more than one target domains. The model cannot rely on relationships between different domains to obtain the domain invariant representation to improve its generalization. By channeling attention towards the model's intrinsic improvement, significant advancements in performance can be achieved. The prevailing consensus in the domain of optimization affirms that using multiple models in a collection process helps to combine the advantages of multiple models, making the loss function of the collection model superior to the local minimum of each model. This set-based approach can be regarded as a means to achieve model flatness, which prompts us to explore a new method to achieve single-domain generalization by expanding the minimum value region in the landscape of the deep neural networks.

A. Preliminaries

We denote X as the input space of images, Y the label space and the given one source domain $D_S = \{x_i, y_i\}_{i=1}^{N_S}$ where S is the training (source) domain with distribution $\mathcal{P}_S$. And the $D_T = \{x_i, y_i\}_{i=1}^{N_T}$ is one test (target) domain with distribution $\mathcal{P}_T$. We define the model $\mathcal{F}(;\theta)$ trained on the source domain as where model parameter $\theta \in \Theta$ and $\ell(\cdot,\cdot)$ can be generalized for any bounded loss function. Then, we consider $\mathcal{E}_{\mathcal{S}}(\theta) \triangleq \mathbb{E}_{(x,y)\sim\mathcal{P}_S}[\ell(f_\theta(x^i), y^i)]$ as the optimization objective, where $f(\cdot;\theta)$ is a model with θ as a parameter. Let θ^* denote the optimal parameter minimizing $\mathcal{E}_{\mathcal{S}}(\theta)$. During one-time training, we may obtain one model with parameter θ_i locating nearby the θ^*. In practice, ERM, *i.e.*, by $\arg\min_\theta \mathcal{E}_{\mathcal{S}}(\theta)$ constrains the training process to make an approximate θ^* by optimization. For domain generalization task, we then expect that this optimal solution can also perform well on the target domain, *i.e.* it can be the lowest risk $\mathcal{E}_{\mathcal{T}}(\theta^*) \triangleq \mathbb{E}_{(x,y)\sim\mathcal{P}_T}[\ell(f_{\theta^*}(x^i), y^i)]$ on the target domain. However, because of the differences in distribution between domains, the θ^* that minimises risk on the source domain is often not guaranteed to minimise risk on the target domain. The differences can be shown in Eq. (1). The divergence metric, denoted as $\mathrm{Div}(\mathcal{P}_S, \mathcal{P}_T)$, is defined as $\sup_A |\mathcal{P}_S(A) - \mathcal{P}_T(A)|$, where $\mathcal{P}_S$ and $\mathcal{P}_T$ represent the marginal distributions of the source and target domains and A represents the set of can-be-identified samples, respectively. This metric serves as a quantification of diversity shift between the two distributions [23]:

$$\mathcal{E}_{\mathcal{T}}(\theta) \leq \frac{1}{2}\,\mathrm{Div}(\mathcal{P}_S, \mathcal{P}_T) + \mathcal{E}_{S}(\theta), \tag{1}$$

$$\mathcal{E}_{\mathcal{T}}(\theta^*) \leq \frac{1}{2}\,\mathrm{Div}(\mathcal{P}_S, \mathcal{P}_T) + \mathcal{E}_{S}^{\gamma}(\theta^*) + \mathcal{L}_{cb}, \tag{2}$$

$$\mathcal{L}_{cb} = \max\sqrt{\frac{(v_k[\ln(N_S/v_k) + 1] + \ln(\eta/\delta))}{2N_S}}, \tag{3}$$

where $\mathcal{E}_{S}^{\gamma}(\theta^*)$ is the robust empirical loss on the source training dataset D_S from S of size N_S. It is defined as the maximum value attained over parameter perturbations Δ within a specified γ-radius. Mathematically, this can be expressed as:

$$\mathcal{E}_{S}^{\gamma}(\theta^*) \triangleq \max_{\|\Delta\|\leq\gamma} \mathcal{E}_{S}(\theta + \Delta), \tag{4}$$

where $\mathcal{E}_{S}(\theta + \Delta)$ represents the empirical loss on D_S for parameters $\theta + \Delta$, where $\mathbb{E}_{(x,y)\in D_S}[\ell\left(f_{\theta+\Delta}(x);y\right)]$ quantifies the expectation of the loss ℓ over the source domain S. The maximization is performed over parameter perturbations within the γ-radius, capturing the range of variations that maintain robustness in the face of potential adversarial perturbations.

Model averaging has emerged as an effective strategy to enhance generalization by exploiting the geometry of the loss landscape. Wang et al. [13] demonstrated that averaging the weights of η models, $\theta^* = \mathcal{H}_{wa}(\theta_i)$ for $i = 1,\ldots,\eta$, can approximate an optimal solution that generalizes well to target domains. Our method extends this idea by constructing a more powerful ensemble, $\mathcal{H}_{wa}^{C}$, which improves upon conventional direct averaging to further enhance generalization efficiency.

The generalization behavior of the averaged model θ^* is governed by three key factors, as described in Eq. (2): (a) the robust empirical loss $\mathcal{E}_{S}^{\gamma}(\theta^*)$; (b) the divergence between the source and target distributions; and (c) the confidence bound $\mathcal{L}_{cb}$, which depends on the neighborhood radius γ and the source sample size N_S as defined in Eq. (3). This decomposition can be theoretically grounded in the PAC-Bayesian framework [35], which links flat minima to robust generalization. The generalization bound can be expressed as:

$$\mathbb{E}_T(\theta^*) \leq \frac{1}{2}\mathrm{Div}(P_S, P_T) + \mathbb{E}_S^{\gamma}(\theta^*) + L_{cb}, \tag{5}$$

where $\mathrm{Div}(P_S, P_T)$ denotes the domain divergence, and $\mathbb{E}_S^{\gamma}(\theta^*)$ is the robust empirical risk defined by:

$$\mathbb{E}_S^{\gamma}(\theta^*) \triangleq \max_{\|\Delta\|\leq\gamma} \mathbb{E}_S(\theta^* + \Delta). \tag{6}$$

In the single-domain generalization (SDG) setting, we lack access to the target distribution P_T, making it infeasible to directly minimize $\mathrm{Div}(P_S, P_T)$. Therefore, we focus on reducing $\mathbb{E}_S^{\gamma}(\theta^*)$, which captures the worst-case loss under parameter perturbations within a radius γ. Models located in flatter regions of the loss landscape are inherently more robust to such perturbations, yielding lower values of $\mathbb{E}_S^{\gamma}(\theta^*)$ and thus better generalization. To explicitly pursue such flat solutions, we propose the Path Flatness-aware Optimization (PFO) method. PFO constructs a curve $\phi_\theta(t)$ connecting two anchor points w_1 and w_2 via a learnable control point θ_c. By minimizing the training loss along this continuous path, we ensure:

$$\max_{\|\Delta\|\leq\gamma} \mathbb{E}_S(\theta_{PFO} + \Delta) \leq \max_{\|\Delta\|\leq\gamma} \mathbb{E}_S(\theta_{SGD} + \Delta), \tag{7}$$

which lowers the worst-case perturbed risk relative to standard optimization approaches.

This path-based approach is inspired by recent insights into mode connectivity, which show that flat minima are often connected by low-loss trajectories in parameter space. By leveraging this connectivity [33], the resulting ensemble $\mathcal{H}_{wa}^{C}$ benefits from enhanced robustness and generalization compared to traditional weight-averaged models.

B. Flatness of Weight Paths Between Models

Inspired by practical ensemble learning techniques particularly Fast Geometric Ensembling (FGE) and grounded in the concept of mode connectivity [33], our work introduces a novel and efficient weight-averaging method specifically tailored for single-domain generalization tasks. The theoretical foundation for connecting anchor model weights via continuously piecewise smooth parametric curves stems from recent studies on the mode connectivity in deep learning. The mode connectivity hypothesis posits that independently trained models, even from different initializations, often lie within a shared low-loss region of the parameter space [34]. Therefore, by leveraging non-linear, continuously piecewise smooth parameterized curves, we can effectively connect these models in weight space, ensuring the path remains within a low-loss region. This design facilitates improved generalization by allowing explicit control over the flatness of the trajectory during optimization, thereby optimizing model performance across the entire path rather than isolated checkpoints. Such a path-centric optimization strategy is particularly critical for single-domain generalization, enabling the model to remain stable under unseen domain shifts.

By incorporating this trajectory-based training framework, we systematically enforce flatness between the regions surrounding the anchor models. This flatness serves as a robustness guarantee, enhancing the effectiveness of the averaging process. The dual constraints applied during path training significantly improve the efficiency and resource economy of single-domain generalization, without relying on explorative ensemble construction. In our study, we define a trainable model $\mathcal{F}$ parameterized by a set of weights θ. After a single training process, we obtain an optimized weight set w. Specifically, we consider two distinct weight sets, denoted as w_1 and w_2, both of which belong to the Euclidean space $\mathbb{R}^{|\mathcal{F}|}$, where $|\mathcal{F}|$ is the total number of parameters in the DNN. These two sets represent independently trained neural networks, each optimized with respect to the cross-entropy loss objective.

Furthermore, we introduce continuously piece wise smooth parametric curves to simulate paths connected between anchors, denoted by $\phi_\theta : [0,1] \rightarrow \mathbb{R}^{|\mathcal{F}|}$. The curve is parameterized by the weights θ. A notable feature of this curve is its boundary conditions: it is defined such that $\phi_\theta(0) = w_1$ and $\phi_\theta(1) = w_2$, where w_1 and w_2 represent two trained models. To instantiate ϕ_θ, we adopt a quadratic Bézier curve defined as:

$$\phi_\theta(t) = (1-t)^2 w_1 + 2(1-t)t\theta_c + t^2 w_2, \tag{8}$$

where θ_c is a learnable control point that determines the shape of the curve between the endpoints. This construction enables us to control the geometry of the interpolation path and to search for flatter solutions in the weight space. These boundary conditions ensure that the curve originates at the point in the weight space corresponding to w_1, and terminates at the point corresponding to w_2. This delineation of boundary conditions is not merely a mathematical construct; it serves as a critical pathway in tracing the evolution of weights in the neural network's configuration space. By rigorously controlling ϕ_θ, we gain the capability to manipulate and monitor the transition of neural network weights along a specified path. This path, extending from one trained state (w_1) to another (w_2) via a set of trainable parameters θ_c, is more than a simple path in weight space. It ensures a period of flat areas $\|\Delta\| \le \gamma$ in the network's learning process. By optimizing the path parameters θ, we manipulate and monitor the transition of network weights along the curve. This path, extending from w_1 to w_2, aims to remain within flat regions of the loss landscape (i.e., where $\|\Delta\| \le \gamma$), improving generalization across domains.

For simplicity, we indistinctly introduce a loss function across a parametric curve in the weight space of a neural network. This evaluation is articulated in Eq. (9), which is defined as follows:

$$\ell(\theta) = \int_0^1 \mathcal{L}(\phi_\theta(t))dt = \mathbb{E}_{t\sim U(0,1)}\mathcal{L}(\phi_\theta(t)), \tag{9}$$

where $\mathcal{L}(\phi_\theta(t))$ represents the loss function evaluated along the continuous piecewise smooth curve $\phi_\theta(t)$ in the weight space. The variable t ranges from 0 to 1 and follows a uniform distribution $U(0,1)$. This distribution is pivotal as it ensures an even sampling across the entire trajectory of the weight curve, guaranteeing a full and fair estimation of the subsequent loss landscape.

To optimize the model to achieve a period of flat region, we first construct the loss function $\ell(\theta)$ and then train it using an iterative manner. Since this section of the region is continuous cannot be optimized directly, we achieve this objective through iterative optimization. In each iteration, a sample $\tilde{t}$ is drawn from a uniform distribution $U(0,1)$. Subsequently, a gradient descent step is performed on the weights θ relative to the loss function evaluated at $\phi_\theta(\tilde{t})$, as shown in the following approximation:

$$\begin{aligned}\nabla_\theta \mathcal{L}(\phi_\theta(\tilde{t})) &\simeq \mathbb{E}_{t\sim U(0,1)}\nabla_\theta \mathcal{L}(\phi_\theta(t)) \\ &= \nabla_\theta \mathbb{E}_{t\sim U(0,1)}\mathcal{L}(\phi_\theta(t)) \\ &= \nabla_\theta \ell(\theta).\end{aligned} \tag{10}$$

This method of random sampling on the path, with the increase of sampling times, can be approximately regarded as the operation of the whole path, so as to achieve the purpose of optimizing the whole path. The iterative process is repeated until convergence is achieved. This technique navigates the complex landscape of the loss function in training. By integrating a random sampling strategy and iterative gradient updates, this method facilitates a more flat landscape optimization process, which is crucial for generalization.

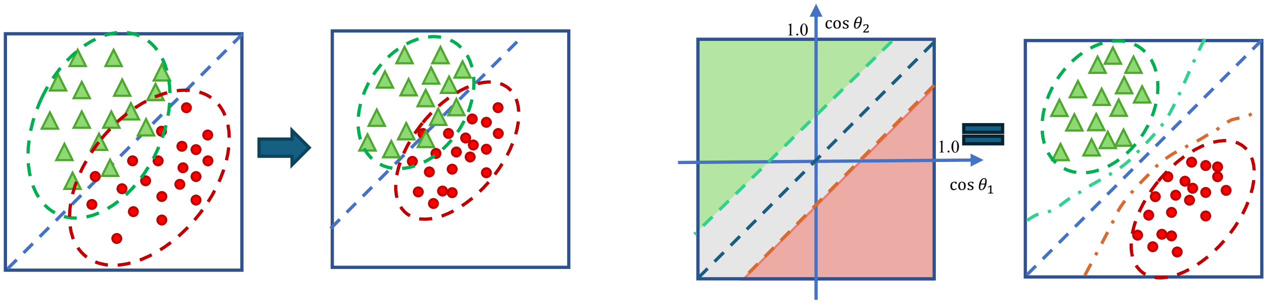
C. Enhancing Diversity of Anchor models by Constraints on the Decision Boundaries

In the progression of this study, following the construction of a pathway interlinking various neural network models, our research focus shifted towards the methodology for selecting two distinct "anchor" models situated at the boundaries of this pathway. This selection process is rooted in both observational insights and a thorough analysis of prior research. During the process of model weight averaging, a greater disparity in the weights being averaged from the respective models tends to yield more effective results post-averaging. We emphasize the importance of model diversity in this process, particularly noting that structural variations in model decision boundaries are crucial for achieving enhanced post-averaging performance.

Leveraging insights from prior research [36], [37], we develop a novel approach to enhance the diversity between models. In this work, we achieve this by modulating the decision boundaries of two distinct anchor models, according to the critical role that decision boundaries play in shaping model behavior. We observe that manipulating the boundaries through angular space optimization and bias translation results in greater diversity compared to conventional approaches that rely solely on varying initialization.

Therefore, our method introduces a technique for optimizing decision boundary operations within angular space, inspired by [37]. We tailor the angular space to implement multiple anchor model across a range of parameter configurations. The core of this optimization is realized through a novel redefinition of the decision boundary equation, expressed as:

$$f_j = \|W_j\|\|x\|\cos(\alpha_j), \tag{11}$$



(a) Adjusting the network's bias parameter

(b) Angle space optimization [37]

Fig. 2: Comparison of the results of decision boundary movements with two different strategies. In (a), the boundary movement is achieved by adjusting the bias parameter of the network. The bias adjustment typically translates the decision boundary without altering the relationship between samples, demonstrating how different bias values can lead to similar classification outcomes for the same data points. In (b), the boundary is manipulated in angular space [37], achieving not just positional translation but also rotational transformations, reflecting a more adaptive response to task-specific data distributions.

where f_j denotes the activation of the fully connected layer, W_j is the weight vector, and α_j is the angle between W_j and the input feature vector x for class j.

Angular space transformations enable a more controlled and meaningful manipulation of the classification strategy by maintaining a constant weight norm while diversifying decision behaviors. Meanwhile, this highlights the pivotal role of α_j in governing the decision boundary. Unlike prior methods that rely solely on bias shifting—as illustrated in Figure 2(a)—angular adjustment offers more refined control over boundary transformations, enabling a wider spectrum of class-specific clustering behavior. Such transformations allow the decision boundary of models like those shown in Figure 2(b). Unlike bias-induced movements, angular adjustments alter not only the position but also the orientation of decision boundaries, enabling compression of intra-class regions and more complex adaptation to feature distributions. The resulting expansion of the inter-anchor region significantly increases the coverage of the model ensemble in high-dimensional feature space. Ultimately, angular space optimization improves the effectiveness of connectivity pathways. It strikes an effective balance between maintaining structural consistency and promoting diversity among models.

To maintain the norm of W, we apply L_2 normalization, setting $\|W_j\|\|x\|$ to a constant n. This focuses the adjustment of the decision boundary on the cosine of the angle, improving category compactness. The cross-entropy loss is thus modified as:

$$\begin{aligned} L &= \frac{1}{N}\sum_{i=1}^{N} -\log\frac{e^{f_{y_i}}}{\sum_{j=1}^{C} e^{f_j}} \\ &= \frac{1}{N}\sum_{i=1}^{N} -\log\frac{e^{n\cos(\alpha_{y_i,i})}}{\sum_{j=1}^{C} e^{s\cos(\alpha_{j,i})}} \end{aligned} \tag{12}$$

and

$$\cos(\alpha_j, i) = \frac{{W_j}^T x_i}{\|W_j\|\|x_i\|}. \tag{13}$$

Moreover, by designing the decision function to depend only on shared, normalized weights and angular cosine values, the inference remains consistent across both anchor and interpolated models. This structural consistency ensures efficient model connectivity without extra modifications, while preserving diversity through meaningful angular variation along a smooth, differentiable path in the weight space.

Then introducing a shift parameter "s" into the cosine term refines the decision boundary:

$$J = \frac{1}{N}\sum_{i=1}^{N} -\log\frac{e^{n(\cos(\alpha_{y_i,i})-s)}}{e^{n(\cos(\alpha_{y_i,i})-s)} + \sum_{j\neq y_i} e^{n\cos(\alpha_{j,i})}}, \tag{14}$$

which enables precise control over the decision boundary, enhancing the model's ability to differentiate categories more effectively. The shift parameter "s" introduces flexibility, allowing for the fine-tuning of the boundary to achieve diverse model performance. Its control effect is shown in Figure 2 (b). By optimizing this shift parameter strategically, we ensure greater model variability and alignment with target clustering metrics, paving the way for efficient pathway optimizations. This methodology aims to maximize the variability in models originating from diverse initial conditions. The adjustment of the term $cos(\alpha) - s$, with "s" set greater than 0, effectively imposes more stringent criteria for the task at hand. Such a configuration predisposes the models under these conditions to develop a more distant landscape during their training phase.

This landscape is an anticipated phenomenon, aligning with our preliminary objectives for the model's path connections. We employ a lower-loss anchor model, which serves as a pivotal component in establishing pathways for model connectivity. This strategy provides an advantageous initial state, facilitating the expedited optimization of the entire path. Moreover, as our focus extends to optimizing the whole path, these models contribute to the formation of a more robust model, one that transcends the individual limitations of each anchor. This ensemble model effectively disregards the shortcomings inherent in its constituent parts, embodying a more efficient learning mechanism.

D. Convergence Analysis

We provide theoretical guarantees for the convergence of our Path Flatness-aware Optimization (PFO) method to flat minima. Let $L(\cdot)$ be a β-smooth and L-Lipschitz continuous loss function. We analyze the convergence behavior of the iterative optimization process described in Algorithm 1, which aims to minimize a path-integrated loss by optimizing a parameterized curve $\phi_\theta(t)$ between two pretrained anchor models w_1 and w_2.

Theorem 1: Assuming β-smoothness and L-Lipschitz continuity of the loss function L, and setting the learning rate $\eta = 1/\beta$, the PFO algorithm converges to a stationary point of the path loss $\ell(\theta) = \mathbb{E}_{t\sim U(0,1)}[L(\phi_\theta(t))]$ at a rate of $O(1/\sqrt{T})$, where T is the number of iterations.
At each iteration, we sample $t \sim U(0,1)$ and update θ via:

$$\theta_{k+1} = \theta_k - \eta \nabla_\theta L(\phi_{\theta_k}(t)). \tag{15}$$

By the β-smoothness of $\ell(\theta)$, we have:

$$\ell(\theta_{k+1}) \leq \ell(\theta_k) - \eta \langle \nabla \ell(\theta_k), \nabla_\theta L(\phi_{\theta_k}(t))\rangle + \frac{\eta^2 \beta}{2} \|\nabla_\theta L(\phi_{\theta_k}(t))\|^2. \tag{16}$$

Taking the expectation over $t \sim U(0,1)$ and using the identity $\nabla \ell(\theta_k) = \mathbb{E}[\nabla_\theta L(\phi_{\theta_k}(t))]$, we obtain:

$$\mathbb{E}[\ell(\theta_{k+1})] \leq \ell(\theta_k) - \eta \|\nabla \ell(\theta_k)\|^2 + \frac{\eta^2 \beta}{2} \mathbb{E}[\|\nabla_\theta L(\phi_{\theta_k}(t))\|^2]. \tag{17}$$

By Jensen's inequality:

$$\|\nabla \ell(\theta_k)\|^2 \leq \mathbb{E}[\|\nabla_\theta L(\phi_{\theta_k}(t))\|^2], \tag{18}$$

which allows us to simplify:

$$\mathbb{E}[\ell(\theta_{k+1})] \leq \ell(\theta_k) - \eta \left(1 - \frac{\eta\beta}{2}\right) \mathbb{E}[\|\nabla_\theta L(\phi_{\theta_k}(t))\|^2]. \tag{19}$$

With $\eta = 1/\beta$, this becomes:

$$\mathbb{E}[\ell(\theta_{k+1})] \leq \ell(\theta_k) - \frac{1}{2\beta} \mathbb{E}[\|\nabla_\theta L(\phi_{\theta_k}(t))\|^2]. \tag{20}$$

Summing over T iterations yields:

$$\frac{1}{T} \sum_{k=0}^{T-1} \mathbb{E}[\|\nabla_\theta L(\phi_{\theta_k}(t))\|^2] \leq \frac{2\beta(\ell(\theta_0) - \ell^*)}{T}, \tag{21}$$

where ℓ^* is the global minimum of $\ell(\theta)$. This implies a convergence rate of $O(1/\sqrt{T})$ toward stationary points of the path loss. In addition, the optimized parameter path θ_{PFO} satisfies a flatness condition. For any $t_1, t_2 \in [0,1]$ with $|t_1 - t_2| \leq \delta$, we have:

$$|L(\phi_\theta(t_1)) - L(\phi_\theta(t_2))| \leq \varepsilon, \tag{22}$$

indicating that the loss remains nearly constant along the path, and thus, the optimization converges to flat minima in the parameter space.

Algorithm 1 Path Flatness-aware Optimization (PFO) for Single-Domain Generalization

```
Input: Source dataset $\mathcal{D}_S = \{x_i, y_i\}_{i=1}^{N_S}$, pretrained model weights $w_1$ and $w_2$, learning rate $\eta$, maximal number of iterations $T$.
Output: Generalized model weights $\theta^*$
Anchor Training: Train two anchor models with distinct decision boundaries by angular space optimization and shift translation $s$:
    $f_1 = W_1^T x = \|W_1\|\|x\|(\cos\alpha - s_1),$
    $f_2 = W_2^T x = \|W_2\|\|x\|(\cos\alpha - s_2)$
Path Initialization: Construct a parametric Bézier curve $\phi_\theta(t)$ in weight space with endpoints:
    $\phi_\theta(0) = w_1, \quad \phi_\theta(1) = w_2$
  where $\theta$ parameterizes intermediate control points of the Bézier curve.
Define Path Loss:
    $\ell(\theta) = \mathbb{E}_{t\sim U(0,1)} \mathcal{L}(\phi_\theta(t)),$
  where $\mathcal{L}$ is cross-entropy loss on $\mathcal{D}_S$
for $k = 1$ to $T$ do
    Sample interpolation factor $t \sim U(0,1)$
    Instantiate model weights along path: $w_t = \phi_\theta(t)$
    Evaluate path loss: $\mathcal{L}(w_t) = \text{CrossEntropy}(f(w_t(x)), y)$
    Compute path gradient via backpropagation:
        $\nabla_\theta \mathcal{L}(\phi_\theta(t)) = \text{Backprop}(\mathcal{L}(w_t))$
    Update path parameters:
        $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}(\phi_\theta(t))$
end for
Model Fusion: Average the weights from both anchors and optimized path center to obtain the final model:
    $\theta^* = \frac{1}{3}(w_1 + w_2 + \phi_\theta(0.5))$
Return: Final generalized model weights $\theta^*$
```

IV. Experimental Study

A. Datasets

We verify the effectiveness of our proposed method on four widely-used benchmark datasets that span a broad spectrum of object recognition scenarios for SDG :

5-Digits. The 5-Digits dataset consists of five datasets: MNIST [38], MNIST-M [39], SVHN [40], USPS [41], and SYN [39], with $0-9$ the 10 categories. Then following the previous work [6], MNIST is utilized as the source domain, and the remaining four datasets serve as the target domains. Following prior work [19], the training comprises the initial $10,000$ images from the MNIST training set. Then the trained model is evaluated on the four target domains with various differences in the background, style, color and image quality. To ensure that the model is generic, all the images are converted to the 32×32 and 3 channel-wise like RGB.

CIFAR10-C. The CIFAR10-C dataset [42] denotes the corrupted CIFAR-10 which is the test set of CIFAR-10 [43] consisting of 19 types of corruptions to evaluate the robustness of classification model. Each corruption has with five levels of severity from 1 to 5 and they can be divided into 4 main categories, weather, blur, noise, and digital. The basic CIFAR10 is employed as the source domain, while CIFAR10-C is used as the target domains. Following [44], only 15 types corruptions of level "5" are used for evaluation, because the most serious corruption can better demonstrate the model generalization of performance.

CIFAR100-C. The CIFAR100-C dataset [42] is similar with the CIFAR10-C dataset but with 100 categories. It also includes 19 types of corruptions with five levels of severity. In accordance with the configuration presented in [44], CIFAR100 [43] is employed as the source domain and the selection of 15 corruption types across highest severity levels are used for evaluation.

PACS. The dataset serves as a domain generalization benchmark, encompassing four diverse domains: art painting, cartoon, photo, and sketch. Across these domains, there are a total of $9,991$ 224×224 images spanning seven categories. The dataset's complexity is rooted in the significant stylistic disparities between domains. For training, there is classical split that each of the four domains serves as the source domain and the other three as the target domain. This structure generates four distinct training and test domain pairs, highlighting the dataset's challenging nature.

OfficeHome. The dataset is a widely used benchmark for domain generalization and domain adaptation, comprising four visually distinct domains: Art, Clipart, Product, and Real World. It consists of a total of 15,588 images, each belonging to one of 65 object categories. The dataset's inherent complexity make it particularly valuable for assessing model robustness across diverse domains in real-world applications. For training and evaluation, the split involves using one domain as the source domain while the remaining three serve as target domains as PACS.

B. Implementation Details

In our study, the selection of neural network architectures for different datasets follows established practices in prior research [45]. Specifically, for the 5-Digits datasets, we adopt the ConvNet architecture [46]; for the CIFAR10-C and CIFAR100-C datasets, we use a 16-layer WideResNet [47] with a widening factor of 4; and for the PACS and OfficeHome datasets, we follow standard domain generalization benchmarks by employing a pretrained ResNet50 backbone.

Data augmentations are widely adopted in existing studies, such as Crafting-Shifts [21], and our method is fully compatible with various augmentation techniques. By following [21], we apply the augmentation strategies to our method for small-scale image datasets (*i.e.*, 5-Digits and CIFAR10-C) in our experiments. However, because these augmentations are computationally expensive for large-scale datasets, we omit them when evaluating our method on CIFAR100-C, PACS, and OfficeHome.

In all experiments, we train our model on CIFAR with SGD [48] and on all other datasets with AdamW [49]. For the 5-Digits datasets, the learning rate is set to 1×10^{-3}, weight decay to 1×10^{-3}, and batch size to 128. For CIFAR10-C and CIFAR100-C, the learning rate is 1×10^{-1}, weight decay is 1×10^{-4}, and batch size is 256, with the learning rate decayed by a factor of 0.2 at the 200th, 300th, and 400th epochs. For PACS and OfficeHome, the learning rate is 5×10^{-5} and weight decay is 1×10^{-4}.

In the dual-anchor setup, two single models are trained independently with different random seeds and different shift parameters ($s = 0.2$ and $s = 0.25$) to ensure sufficient diversity in decision boundaries. The path $\phi_\theta(t)$ between these models is parameterized using a cubic Bézier curve, where θ controls the intermediate curve points. Training of the path model involves sampling $t \sim U(0, 1)$, evaluating the cross-entropy loss $\mathcal{L}(\phi_\theta(t))$ on the source domain, and performing backpropagation to update θ using the learning rate η. We set $\eta = 10^{-4}$ and use a cosine annealing learning rate schedule throughout the path optimization phase to ensure smooth convergence to flat minima, consistent with our theoretical analysis in Section III.D.

The final generalized model weights θ^* are obtained by averaging the weights of the two anchor models and the midpoint model $\phi_\theta(0.5)$ on the learned curve. This practical procedure exactly mirrors the update and averaging steps described in Algorithm 1 and the convergence proof.

C. Comparison with Peer Methods

We compare our method with a wide range of SOTA SDG methods in different settings, following the experimental setups of previous works to ensure fair comparison. The compared methods include foundational approaches (e.g., ERM [50]), domain-invariant feature learning methods (e.g., CCSA [51], d-SNE [52], JiGen [53]), and data augmentation-based strategies (e.g., GUD [16], M-ADA [18], ME-ADA [10], PDEN [17], L2D [6], RC [20], RSDA [54], RSC [55], ASR [56], FFM [44], CADA [19], MCL [57], Crafting-Shifts [21], and PhysAug [58]). The results of these methods are reported based on previous publications [19], [44], [57].Building upon this, we further benchmark our method against a unified set of strong and representative baselines to ensure consistent evaluation across all datasets. These include classical domain generalization methods (e.g., IRM [59], Mixup [60], VREx [61], and RIDG [62]), as well as flatness-aware optimization approaches that aim to improve generalization through gradient-based methods (e.g., SAM [27], GAM [28], SAGM [29], IADA [30], FSAM [63], SSESAM [64], GCSAM [65], and SAML [66]). This unified set of benchmarks is evaluated across all datasets, enhancing the fairness, transparency, and comprehensiveness of our experimental analysis, and offering a rigorous assessment of our method's effectiveness in diverse SDG scenarios.

Evaluation on 5-Digits. Table I summarizes results when training on MNIST and evaluating on SVHN, MNIST-M,

TABLE I: Generalization Accuracy (%) for Single Domain on Digits: Models trained on MNIST and tested on other digit datasets.

Method	Venue Year	SVHN	MNIST-M	SYN	USPS	Avg.
ERM [50]	Springer, 2011	27.83	52.72	39.65	76.94	49.29
CCSA [51]	ICCV, 2017	25.89	49.29	37.31	83.72	49.05
d-SNE [52]	CVPR, 2019	26.22	50.98	37.83	93.16	52.05
JiGen [53]	CVPR, 2019	33.80	57.80	43.79	77.15	53.14
ADA [16]	NeurIPS, 2018	35.51	60.41	45.32	77.26	54.62
M-ADA [18]	CVPR, 2020	42.55	67.94	48.95	78.53	59.49
ME-ADA [10]	NeurIPS, 2020	42.56	63.27	50.39	81.04	59.32
RSDA [54]	ICCV, 2019	47.40	81.50	62.00	83.10	68.50
RSDA+ASR [56]	CVPR, 2021	52.80	80.80	64.50	82.40	70.10
L2D [6]	ICCV, 2021	62.86	87.30	63.72	83.97	74.46
RC [20]	CoRR, 2020	62.07	87.89	63.90	84.39	74.56
FFM [44]	WACV, 2023	64.11	82.25	63.91	83.56	73.45
CADA [19]	WACV, 2023	67.27	78.66	79.34	96.96	80.56
MCL [57]	CVPR, 2023	69.94	78.47	78.34	88.54	78.82
Crafting-Shifts [21]	WACV, 2024	67.82	84.28	79.64	98.68	**82.61**
PhysAug [58]	AAAI, 2025	62.24	80.46	59.98	95.37	74.51
IRM [59]	CoRR, 2019	56.85	63.68	45.39	90.73	64.16
Mixup [60]	CoRR, 2020	55.72	79.25	53.28	94.22	70.62
VREx [61]	ICML, 2021	51.36	78.49	56.04	94.02	69.98
RIDG [62]	ICCV, 2023	54.96	77.64	55.32	93.32	70.31
SAM [27]	ICLR, 2021	55.37	80.26	55.31	93.97	71.23
GAM [28]	CVPR, 2023	54.38	77.93	54.65	94.02	70.25
SAGM [29]	CVPR, 2023	56.42	79.78	54.99	94.17	71.34
IADA [30]	ICLR, 2024	65.13	81.45	62.29	95.17	76.01
FSAM [63]	CVPR, 2024	55.53	80.24	58.59	94.52	72.22
GCSAM [65]	TMM, 2025	52.41	80.84	54.84	94.12	70.55
SSESAM [64]	AAAI, 2025	56.65	78.08	53.04	93.42	70.30
SAML [66]	ICLR, 2025	54.47	74.67	58.69	92.18	70.00
PFO (Ours)	-	63.84	82.32	70.60	88.39	76.29
PFO* (Ours)	-	68.25	79.88	84.48	95.76	82.09

SYN, and USPS. Consistent with prior observations, USPS is the easiest task due to its high visual similarity to MNIST (several methods exceed 95%), whereas SVHN remains the most challenging because of its different color statistics and cluttered, real-world backgrounds. Augmentation-oriented methods such as Crafting-Shifts achieve the best overall average (82.61%) and the strongest USPS score (98.68%). PhysAug also performs well on USPS (95.37%) but is less stable on synthetic domains (59.98% on SYN). SAM-family approaches (e.g., FSAM, GCSAM) are competitive on average but generally lag on SVHN and SYN. Our PFO, even without data augmentations, attains an average accuracy of 76.29%, outperforming several augmentation-dependent methods and yielding balanced performance across SVHN (63.84%), MNIST-M (82.32%), SYN (70.60%), and USPS (88.39%). When augmented with the data augmentation strategy employed by Crafting-Shifts (denoted as PFO*), our method achieves an average accuracy of 82.09%, matching the best performance reported by Crafting-Shifts. This improvement stems not only from substantial gains on SVHN (+4.41 points) and SYN (+13.88 points) but also from achieving competitive peak results on USPS (95.76%) and MNIST-M (79.88%).

Evaluation on CIFAR10-C. Table II reports the performance of our method on the CIFAR-10-C benchmark under 15 corruption types at severity level 5. Without data augmentations, PFO achieves a strong average accuracy of 75.29%, outperforming most classical DG and SAM-based approaches, and showing strong performance across weather, blur, noise, and digital-type corruptions. When equipped with the same augmentation protocol as augmentation-driven baselines (denoted as PFO*), the average accuracy further improves to 78.64%, outperforming all peer methods. This gain is largely attributed to substantial improvements under noise-related corruptions (Shot: +11.53,) and glass blur (+11.26), while maintaining competitive performance on weather and digital distortions. Compared with the best augmentation-based baseline FFM (77.77%), PFO* not only achieves a higher overall score but also delivers a more stable and consistent performance across diverse corruption types, avoiding large drops in challenging settings. Notably, PFO demonstrates robust resilience to weather-related distortions and excels under blur-related corruptions, reflecting its ability to preserve semantic representations under severe visual degradations. Among other recent methods, FSAM (66.91%) and GCSAM (64.30%) show promising results, particularly on digit-related corruptions. Compared with these approaches, as well as representative DG and SAM-based baselines such as SAML (64.48%), and IADA (73.60%), PFO consistently ranks among the top performers.

Evaluation on CIFAR100-C. For CIFAR-100-C, similar experimental conditions of CIFAR10-C are applied, with summarized results presented in Table III. Our method (PFO) registers an average accuracy of 62.88%, surpassing classical DG approaches such as Mixup (49.36%), VREx (49.74%), and RIDG (41.44%). PFO also exceeds recent SAM-based methods like IADA (56.77%), FSAM (57.55%), GCSAM (58.40%), SSESAM (53.60%), and SAML (54.11%). Compared to CADA (59.96%), PhysAug (50.82%) and RC (57.38%), which are among the strongest SDG baselines, PFO demonstrates improved overall consistency and higher peak performance across all corruption categories. Notably, PFO achieves substantial improvements in challenging Weather (65.34%) and Digital (65.95%) corruption categories, highlighting its exceptional robustness and consistent generalization capability across various corruption types.

Comparing CIFAR-100-C with CIFAR-10-C, despite their similarities, the complexity of tasks in CIFAR-100-C is notably higher, raising the bar for methodological requirements, particularly for those reliant on data augmentation. The increased dataset complexity amplifies the challenges associated with data enhancement, diminishing the relative impact of such techniques compared to our method's performance on CIFAR-10-C. This underscores the nuanced demands placed on augmentation-based algorithms as task complexity escalates, highlighting areas where our proposed method retains a comparative advantage. Its consistent effectiveness in both CIFAR-10-C and CIFAR-100-C highlights its scalability and adaptability to increasing task complexity, making it a stable solution for robust image classification in real-world, corruption-rich environments.

Evaluation on PACS. The results on the PACS dataset are reported in Table IV, where models are trained on one domain and tested on the remaining three. PFO achieves the highest average accuracy of 73.84%, surpassing previous state-of-the-art approaches such as Crafting-Shifts (72.05%), PhysAug (69.98%), IADA (71.83%), and FSAM (67.75%). While the competitive performance of Crafting-Shifts benefits from its extensive data augmentation strategies, PFO attains superior results without relying on such techniques, demonstrating the strength of its optimization-based design. Notably, PFO sets a new SOTA on the challenging Sketch domain with an

TABLE II: Generalization Accuracy (%) for Single Domain on CIFAR-10: Models trained on CIFAR-10 and tested on 15 different types of corruption at the severity level 5 in CIFAR-10-C.

Method	Venue, Year	Weather			Blur				Noise				Digit				Avg.
		Fog	Snow	Frost	Zoom	Defocus	Glass	Motion	Shot	Impulse	Gaussian	Speckle	Pixelate	Elastic	Brightness	Contrast	
ERM [50]	Springer, 2011	65.92	74.36	61.57	59.97	53.71	49.44	63.81	35.41	25.65	29.01	69.90	41.07	72.40	91.25	36.87	56.15
CCSA [51]	ICCV, 2017	66.94	74.55	61.49	61.96	56.11	48.46	64.73	33.79	24.56	27.85	69.68	40.94	72.36	91.00	35.83	56.31
M-ADA [18]	CVPR, 2020	69.36	80.59	76.66	68.04	61.18	61.59	64.23	60.58	45.18	56.88	77.14	52.25	75.61	90.78	29.71	65.59
MEADA [10]	NeurIPS, 2020	60.07	81.72	82.10	75.45	67.71	72.55	70.86	59.73	46.78	58.65	85.52	77.48	79.80	88.16	23.92	69.15
L2D [6]	ICCV, 2021	69.21	78.70	81.35	72.86	64.58	61.53	68.52	78.32	13.61	74.81	82.31	53.19	76.50	91.33	48.16	69.08
FFM [44]	WACV, 2023	80.23	84.62	84.86	81.01	79.94	67.50	83.71	82.67	23.16	80.90	81.80	70.17	77.40	90.82	78.54	77.77
PhysAug [58]	AAAI, 2025	62.62	59.98	60.68	63.52	64.66	48.76	63.78	47.38	41.20	43.73	64.89	58.65	59.88	72.84	28.34	56.06
IRM [59]	Arxiv, 2019	39.73	41.68	38.66	44.97	41.97	40.92	36.62	44.38	35.56	43.50	52.54	41.54	51.57	56.43	21.57	42.11
Mixup [60]	Arxiv, 2020	71.79	65.44	55.46	63.09	62.14	42.10	62.27	43.92	37.64	44.00	67.17	50.16	64.14	82.29	64.15	58.38
VREx [61]	ICML, 2021	71.98	61.69	54.97	58.51	56.32	45.59	62.87	47.74	48.76	45.18	67.00	51.36	61.74	81.70	64.56	58.66
RIDG [62]	ICCV, 2023	68.64	60.06	51.38	61.81	60.11	39.07	56.87	36.48	36.44	33.36	65.31	48.34	63.40	78.53	53.86	54.24
SAM [27]	ICLR, 2021	75.61	63.41	56.53	60.90	60.10	40.83	61.40	41.39	42.28	39.16	67.34	47.23	65.48	82.59	66.74	58.07
GAM [28]	CVPR, 2023	72.40	62.83	53.10	63.62	59.55	40.37	62.32	40.23	39.54	37.50	65.34	49.10	64.00	80.59	60.68	56.74
SAGM [29]	CVPR, 2023	74.21	65.10	56.99	63.12	62.76	43.81	64.24	44.93	44.21	40.76	64.75	44.93	64.07	82.75	67.03	58.91
IADA [30]	ICLR, 2024	76.34	75.86	74.52	79.14	79.52	57.27	76.07	71.61	64.18	69.22	71.97	72.05	69.35	86.73	80.15	73.60
FSAM [63]	CVPR, 2024	79.25	73.67	64.54	75.89	74.63	48.68	73.64	48.80	48.32	46.33	73.94	54.71	69.42	89.39	82.38	66.91
GCSAM [65]	TMM, 2025	77.33	70.99	59.76	72.70	70.83	46.33	71.51	44.07	47.88	39.98	72.80	51.82	69.18	88.52	80.77	64.30
SSESAM [64]	AAAI, 2025	64.35	76.12	69.44	69.73	78.30	59.60	66.37	48.75	51.08	36.82	73.03	52.18	60.24	74.10	79.49	63.97
SAML [66]	ICLR, 2025	78.75	73.35	66.97	78.66	65.67	41.86	70.72	52.45	56.90	41.77	70.48	51.24	68.20	76.43	73.74	64.48
PFO (Ours)	-	78.06	77.86	77.76	81.96	81.32	55.94	78.54	69.94	64.99	78.55	71.64	75.05	69.55	90.73	77.41	75.29
PFO* (Ours)	-	71.83	79.68	81.11	82.67	82.87	67.20	78.52	81.47	74.03	80.22	80.95	77.66	73.88	86.16	81.34	**78.64**

TABLE III: Generalization Accuracy (%) for Single Domain on CIFAR-100: Models trained on CIFAR-100 and tested on the 4 main categories of corruption at the severity level 5 in CIFAR-100-C.

Method	Venue, Year	Weather	Blur	Noise	Digital	Avg.
ERM [50]	Springer, 2011	24.67	45.75	55.00	56.00	46.78
ADA [16]	NeurIPS, 2018	33.00	51.75	55.33	58.20	50.97
ME-ADA [10]	NeurIPS, 2020	52.67	53.00	52.33	56.20	53.93
L2D [6]	ICCV, 2021	25.40	37.91	43.34	46.07	38.18
FFM [44]	WACV, 2023	33.06	49.51	51.74	51.98	46.57
RC [20]	Arxiv, 2020	56.99	57.17	57.88	57.48	57.38
CADA [19]	WACV, 2023	59.81	59.28	60.19	59.66	59.96
PhysAug [58]	AAAI, 2025	59.37	49.07	45.42	49.41	50.82
IRM [59]	Arxiv, 2019	40.18	37.35	40.18	40.18	39.47
Mixup [60]	Arxiv, 2020	51.22	46.59	46.47	53.14	49.36
VREx [61]	ICML, 2021	51.89	48.10	45.03	53.93	49.74
RIDG [62]	ICCV, 2023	37.54	39.90	45.08	43.26	41.44
SAM [27]	ICLR, 2021	53.78	47.63	45.19	55.15	50.44
GAM [28]	CVPR, 2023	51.04	45.14	43.46	52.41	48.01
SAGM [29]	CVPR, 2023	52.69	47.87	44.73	53.97	49.81
IADA [30]	ICLR, 2024	58.18	56.07	53.94	58.91	56.77
FSAM [63]	CVPR, 2024	59.27	59.33	51.10	60.52	57.55
GCSAM [65]	TMM, 2025	60.10	60.13	51.89	61.48	58.40
SSESAM [64]	AAAI, 2025	55.39	55.40	46.82	56.77	53.60
SAML [66]	ICLR, 2025	58.29	52.14	52.32	53.69	54.11
PFO (Ours)	-	65.34	60.35	59.86	65.95	**62.88**

TABLE IV: Generalization Accuracy (%) for Single Domain on PACS: Models trained on one domain (Source domain) and tested on the other three domains (Target domains).

Method	Venue, Year	Artpaint	Cartoon	Sketch	Photo	Avg.
ERM [50]	Springer, 2011	70.90	76.50	53.10	42.20	60.70
RSC [55]	ECCV, 2020	73.40	75.90	56.20	41.60	61.80
ADA [16]	NeurIPS, 2018	71.60	76.80	52.40	43.70	61.10
ME-ADA [10]	NeurIPS, 2020	71.50	76.80	46.20	46.30	60.20
RC [20]	Arxiv, 2020	73.70	74.90	55.40	46.80	62.70
L2D [6]	ICCV, 2021	76.90	77.90	53.70	52.30	65.20
RSC+ASR [56]	CVPR, 2021	76.70	79.30	61.60	54.60	68.10
CADA [19]	WACV, 2023	76.30	79.10	61.60	56.70	68.40
FFM [44]	WACV, 2023	80.50	77.70	62.10	61.40	70.40
MCL [57]	CVPR, 2023	77.13	80.14	62.55	59.60	69.86
Crafting-Shifts [21]	Arxiv, 2024	81.14	78.34	68.13	60.59	72.05
PhysAug [58]	AAAI, 2025	79.04	81.28	62.16	57.45	69.98
IRM [59]	Arxiv, 2019	74.80	76.52	52.04	49.69	63.26
Mixup [60]	Arxiv, 2020	80.51	83.01	63.55	49.70	69.19
VREx [61]	ICML, 2021	79.03	81.25	62.74	56.25	69.82
RIDG [62]	ICCV, 2023	80.44	81.05	63.24	48.73	68.36
SAM [27]	ICLR, 2021	80.75	83.02	66.98	57.03	71.95
GAM [28]	CVPR, 2023	81.20	81.99	60.89	53.10	69.29
SAGM [29]	CVPR, 2023	78.53	83.13	59.99	48.35	67.50
IADA [30]	ICLR, 2024	80.49	80.28	65.64	60.94	71.83
FSAM [63]	CVPR, 2024	79.23	82.55	60.40	48.83	67.75
GCSAM [65]	TMM, 2025	78.00	82.84	57.50	50.35	67.17
SSESAM [64]	AAAI, 2025	77.14	82.08	65.78	46.72	67.93
SAML [66]	ICLR, 2025	68.18	81.45	63.89	49.53	65.77
PFO (Ours)	-	81.38	82.67	67.95	63.34	**73.84**

accuracy of 67.95%, indicating strong adaptability to abstract and texture-deficient representations. Although slightly trailing the very best results in domains like Artpaint and Cartoon, PFO maintains consistently high performance across all target domains. These results highlight its robustness in handling substantial domain shifts with limited source supervision and reinforce its effectiveness as a general-purpose solution for cross-domain image classification.

Evaluation on OfficeHome. We evaluate our method (PFO) on the OfficeHome dataset under the single-domain generalization setting, where the model is trained on one domain and tested on the remaining three. As shown in Table V, PFO achieves the highest average accuracy of 58.42%, outperforming all listed baselines. For example, the next-best method, GAM, reaches 57.72%, while Crafting-Shifts and SAM obtain 57.31% and 55.24%, respectively. Although FSAM (57.34%) and GCSAM (57.15%) achieve competitive results, they still

TABLE V: Generalization Accuracy (%) for Single Domain on OfficeHome: Models trained on one domain (Source domain) and tested on the other three domains (Target domains).

Method	Venue, Year	Art	Clipart	Product	Real World	Avg.
ERM [50]	Springer, 2011	54.28	51.68	49.22	60.06	53.81
Crafting-Shifts [21]	WACV, 2024	59.77	55.31	51.46	63.10	57.31
PhysAug [58]	AAAI, 2025	54.30	52.43	49.57	60.10	54.10
IRM [59]	Arxiv, 2019	54.84	53.17	48.45	58.38	53.71
Mixup [60]	Arxiv, 2020	55.26	52.56	49.62	59.64	54.27
VREx [61]	ICML, 2021	57.06	54.42	49.12	60.25	55.21
RIDG [62]	ICCV, 2023	56.77	54.75	50.34	63.07	56.23
SAM [27]	ICLR, 2021	54.73	52.71	51.10	62.41	55.24
GAM [28]	CVPR, 2023	59.53	55.47	52.65	63.22	57.72
SAGM [29]	CVPR, 2023	57.19	56.54	52.21	61.69	56.90
IADA [30]	ICLR, 2024	51.56	48.06	45.75	58.19	50.89
FSAM [63]	CVPR, 2024	58.15	56.41	52.59	62.20	57.34
GCSAM [65]	TMM, 2025	56.86	56.45	53.06	62.24	57.15
SSESAM [64]	AAAI, 2025	55.16	54.82	50.65	60.08	55.18
SAML [66]	ICLR, 2025	44.56	56.48	53.04	62.18	54.07
PFO (Ours)	-	58.23	55.96	55.56	63.93	**58.42**

fall short of PFO due to domain-specific weaknesses, whereas PFO delivers consistently higher and more stable accuracy across all target domains. This stability is particularly evident in challenging domains like Art, which exhibit significant appearance shifts and high intra-domain variability. Furthermore, while the strong performance of Crafting-Shifts can be attributed to extensive augmentation strategies, PFO achieves superior results without relying on such techniques, highlighting the advantages of an optimization-based approach on complex multi-domain datasets. These comparisons reinforce the strength of PFO's optimization strategy in navigating cross-domain discrepancies without specialized augmentations or domain-specific modules, underscoring its effectiveness in learning transferable semantic representations and maintaining robust generalization under distribution shifts with limited source supervision.

In summary, although PFO does not achieve the highest average accuracy across the five benchmarks, it demonstrates strong cross-domain consistency across all target domains. While augmentation-based methods often exhibit superior performance in certain domains, they tend to suffer from high cross-domain variance, which limits their robustness. In contrast, PFO maintains stable performance across all domains, indicating better generalization stability under the single-domain generalization (SDG) setting. Moreover, although augmentation-based methods perform well on visually simple datasets such as digits, they struggle to scale effectively to more complex benchmarks like PACS. In these more challenging domain shift scenarios, PFO and SAM-based methods consistently outperform augmentation-based approaches, highlighting the advantages of model-centric optimization strategies in achieving better scalability and broader applicability. A recent example is *Crafting Shifts*, which surpasses prior SOTA results by leveraging extensive data augmentation and object-centric features. Nevertheless, on more complex datasets such as PACS and OfficeHome, PFO and SAM-based methods are able to match or even exceed its performance. This finding further validates our motivation to reduce reliance on handcrafted augmentations and instead prioritize model-centric optimization as a more effective and generalizable solution.

TABLE VI: Compare the ACC(%) for different Settings on 5-Digits.

Method	SVHN	MNIST-M	SYN	USPS	Avg.
ERM	27.83	52.72	39.65	76.94	49.29
ERM w/ s	27.35	52.38	39.92	75.21	48.72
PFO w/o s	48.75	73.46	49.31	81.23	63.19
WA	50.16	77.31	55.33	82.39	66.30
PFO (Ours)	63.84	82.32	70.60	88.39	**76.29**

TABLE VII: Compare the ACC(%) for different Settings on CIFAR-10-C.

Method	Weather	Blur	Noise	Digital	Avg.
ERM	67.28	56.73	30.02	62.3	54.08
ERM w/ s	61.35	54.13	31.89	61.81	52.30
PFO w/o s	73.28	65.13	59.31	72.93	67.66
WA	75.56	69.10	66.16	70.91	70.43
PFO (Ours)	77.89	73.07	73.01	76.88	**75.21**

TABLE VIII: Compare the ACC(%) for the relationship between source-domain validation set and target-domain test set in different Settings on CIFAR-10-C.

Method	CIFAR-10 validation	CIFAR-10-C test Avg.
ERM	92.31	54.08
ERM w/ s	93.95	52.30
PFO w/o s	93.81	67.66
WA	94.10	70.43
PFO (Ours)	93.02	**75.21**

TABLE IX: Comparison of Training Resource Consumption in different Settings on CIFAR-10-C.

Method	#Training params (M)	Time cost (ms) per batch
ERM	1.55	9.1
ERM w/ s	1.55	9.6
PFO w/o s	4.51	10.3
WA	1.55	9.2
PFO (Ours)	4.51	11.1

D. Ablation Study and Analysis

In this subsection, we delve into the individual contributions of each component within our methodology and scrutinize the influence of hyperparameters on performance. Our comparative analysis encompasses a variety of methods including Empirical Risk Minimization (ERM), ERM augmented with boundary shift (ERM w/ s), Path-Aware Flattening without boundary shift (PFO w/o s), Weighted Averaging (WA), and the full version of PFO.

The domain generalization performance, as detailed in Table VI for digits recognition, reveals our method's superior capabilities across four image recognition datasets, achieving a notable average accuracy of 74.71%. This outstrips the performances of other techniques, with WA and PFO w/o s also showing strong results with average accuracies of 66.30% and 63.19%, respectively. These findings underscore the efficacy of model ensembles in single domain generalization tasks.

Table VII shows the assessment to robustness against image corruptions on the CIFAR-10-C dataset, both WA and PFO w/o s methods bolster robustness, reaching average accuracies of 70.43% and 67.66%. The variability in performance enhancements across different domains underscores the distinct optimization focuses of each integration algorithm. Our proposed method significantly boosts domain generalization capabilities and resilience by amalgamating the strengths of preceding strategies.

Further analysis reveals a correlation between source domain validation set outcomes and performance on unknown target domains in Table VIII. While ERM w/ s may offer marginal improvements over ERM in CIFAR-10, its efficacy

TABLE X: Compare the ACC(%) about impact of boundary shift "s" on CIFAR-10-C.

Method	Weather	Blur	Noise	Digital	Avg.
s(0.0, 0.0)	67.28	56.73	30.02	62.3	54.08
s(0.0, 0.1)	69.91	65.88	64.69	68.64	67.28
s(0.0, 0.2)	76.81	72.35	69.17	70.56	72.22
s(0.1, 0.1)	75.56	73.10	72.66	75.35	74.16
s(0.2, 0.2)	77.89	73.07	73.01	76.88	**75.21**
s(0.3, 0.3)	76.16	71.33	71.62	72.48	72.89

diminishes in CIFAR-10-C, potentially due to boundary shifts compromising the model's adaptability to corrupted data. Our method's exceptional performance on CIFAR-10-C suggests a design emphasis on model flatness across pairs rather than solely optimizing for clean data, highlighting the necessity for machine learning models and domain adaptation techniques that accommodate both pristine and corrupted data scenarios.

Table IX shows minimal differences in training resource consumption across methods, with throughput ranging from 9.1 to 11.1 ms/batch. While PFO introduces slightly more parameters due to the curve representation, the per-batch training time increases only marginally, which approximately increase +2ms compared to ERM). Despite these differences, the computational overhead remains manageable. In contrast, WA typically involves training a pool of candidate models followed by an ensemble selection process, which incurs substantial training and validation cost. PFO avoids this by training only two anchor models and combining them through a single-stage, differentiable curve optimization process.

As shown in Figure 3, our method also demonstrates faster convergence during training. On the Real World domain of the OfficeHome dataset, PFO achieves a significantly lower and more stable training loss within fewer iterations compared to ERM, SAM, and SAGM. This suggests that the flatness-aware path optimization accelerates the model's ability to capture effective patterns, leading to both computational efficiency and robust generalization.

Table X reports the results on the impact of different boundary shift configurations, with s(0.1, 0.2) denoting the boundary shift values "s", which are 0.1 and 0.2 for two anchor models. The performance of method s fluctuates considerably across configurations and domains, with s(0.2, 0.2) achieving the highest overall average of 75.21%. This indicates that a balanced approach to diverse image domains is attainable with specific settings, emphasizing the critical role of parameter tuning in domain adaptation and image classification. The variation in sensitivity to parameter adjustments among domains, particularly in Noise domains, highlights the nuanced gains achievable with different configurations. The model with s(0.3, 0.3) is slightly less effective than s(0.2, 0.2) at 72.89%. The potential reason is the over-fitting to source domain characteristics. These results indicate that the setting of the boundary shift has a high impact to the performance of the model.

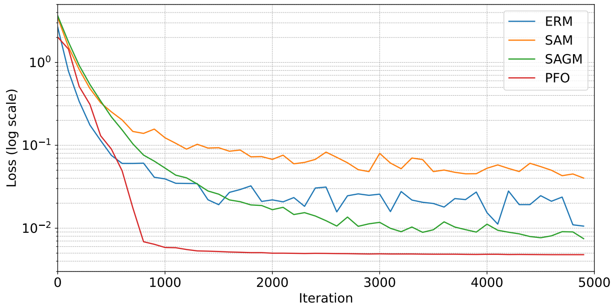


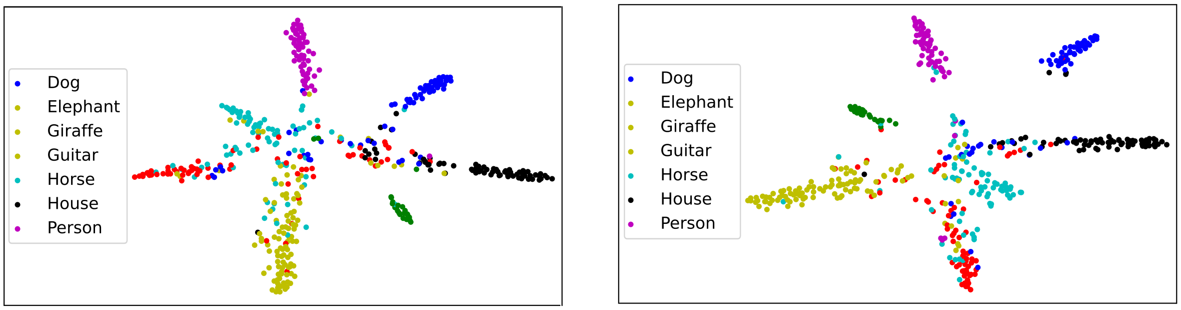
Fig. 3: The training loss curve for the results obtained by ERM, SAM, SAGM, and PFO methods on the OfficeHome dataset.

Visualization Analysis. We employed t-SNE [67] to project the learned feature representations onto a 2D plane, enhancing our understanding of our method's effectiveness. Fig. 4 illustrates the results yielded by our PFO method and the ERM when trained with sketch images (source domain) and applied to images from the cartoon domain (target domain). Specifically, we take the outputs from the penultimate Layer of a ResNet-18 Backbone as the feature representations of the input images. The results reveal that the ERM's representations for certain categories may be more scattered or less defined, showing possible overlaps between categories such as "Dog" and "Horse". Conversely, the feature representations derived from our PFO method tend to be more compact for some categories, suggesting improved separation of certain classes compared to the ERM. The comparative analysis demonstrates that the PFO method achieves better clustering performance, forming clusters that are internally cohesive and distinctly separated, when contrasted with the ERM.

V. Conclusion

In this study, we presented a novel approach to enhancing model generalization in single-domain scenarios by leveraging the path-aware flat minima, which form a more robust ensemble. Our method emphasizes the optimization of model connections to achieve the flatness of the learning path, thereby significantly improving generalization across various unseen domains. The introduction of diverse anchor models through decision boundary constraints further contributes to our method's effectiveness, enabling the creation of models with enhanced differentiation and generalization capabilities. By focusing on the intrinsic characteristics of the model rather than relying on extensive data augmentation, our approach offers a robust and efficient solution to the challenges of domain generalization. Empirical results on multiple public benchmarks validated the superiority of our proposed approach to current state-of-the-art methods, especially in complex scenarios. In future, we will investigate how to identify the elite model candidates and save them as an archive to guide and seed up the optimal model searching process.

While our method substantially reduces the additional pretraining time compared to Weight Averaging, it incurs increased GPU memory consumption due to the requirement of training multiple anchor models during curve learning. Moreover, effective path optimization depends on the quality of these anchor models, whose initialization currently relies on diverse angle settings and sufficient training. As widely acknowledged, there exists a trade-off between a model's generalization ability and task-specific performance. If the anchor models fail to reach a satisfactory initial state, optimizing the entire path toward desirable performance becomes challenging. To address these limitations, our future work will focus on developing more rigorous theoretical guarantees to better understand the relationship between path flatness and generalization capability. Additionally, we plan to explore integrating our path-flatness method with complementary techniques such as adversarial training or meta-learning. Building upon this foundation, we aim to investigate more efficient parameterizations



(a) ERM

(b) PFO

Fig. 4: The visualization of the t-SNE embeddings for the results obtained by our PFO method and the ERM when trained with sketch images (source domain) and applied to images from the cartoon domain (target domain).

of the connecting paths to reduce memory usage and improve computational efficiency. Furthermore, we intend to extend our approach to a broader range of scenarios by exploring how path construction can be dynamically adapted to different tasks and data modalities. Tailoring the path-building process based on specific task and data characteristics represents a promising direction for future research and may further enhance the versatility of our method.

References

[1] Z. Chen, Y. Luo, R. Qiu, S. Wang, Z. Huang, J. Li, and Z. Zhang, "Semantics Disentangling for Generalized Zero-Shot Learning," in *Proceedings of the International Conference on Computer Vsion*. IEEE, 2021, pp. 8692–8700.

[2] R. Dai, Y. Zhang, Z. Fang, B. Han, and X. Tian, "Moderately Distributional Exploration for Domain Generalization," in *Proceedings of Machine Learning Research*, vol. 202, 2023, pp. 6786–6817.

[3] J. Kang, S. Lee, N. Kim, and S. Kwak, "Style Neophile: Constantly Seeking Novel Styles for Domain Generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2022, pp. 7120–7130.

[4] Z. Wang, X. Shu, Y. Wang, Y. Feng, L. Zhang, and Z. Yi, "A Feature Space-Restricted Attention Attack on Medical Deep Learning Systems," *IEEE Transactions on Cybernetics*, vol. 53, no. 8, pp. 5323–5335, 2023.

[5] L. Wang, L. Zhang, X. Qi, and Z. Yi, "Deep Attention-Based Imbalanced Image Classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 8, pp. 3320–3330, 2022.

[6] Z. Wang, Y. Luo, R. Qiu, Z. Huang, and M. Baktashmotlagh, "Learning to diversify for single domain generalization," in *Proceedings of the International Conference on Computer Vsion*, 2021, pp. 834–843.

[7] W. Ying, D. Wang, X. Hu, Y. Zhou, C. C. Aggarwal, and Y. Fu, "Unsupervised Generative Feature Transformation via Graph Contrastive Pre-training and Multi-objective Fine-tuning," in *Proceedings of the Conference on Knowledge Discovery and Data Mining*. ACM, 2024, pp. 3966–3976.

[8] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4396–4415, 2022.

[9] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, and A. G. Wilson, "Averaging weights leads to wider optima and better generalization," in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2018, pp. 876–885.

[10] L. Zhao, T. Liu, X. Peng, and D. Metaxas, "Maximum-entropy adversarial data augmentation for improved generalization and robustness," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 33, pp. 14 435–14 447, 2020.

[11] Y. Tsuzuku and I. Sato, "On the structural sensitivity of deep convolutional networks to the directions of fourier basis functions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 51–60.

[12] C. Wan, X. Shen, Y. Zhang, Z. Yin, X. Tian, F. Gao, J. Huang, and X.-S. Hua, "Meta convolutional neural networks for single domain generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4682–4691.

[13] J. Wang, C. Lan, C. Liu, Y. Ouyang, and T. Qin, "Generalizing to Unseen Domains: A Survey on Domain Generalization," pp. 4627–4635, 2021.

[14] M. Wang, J. Liu, G. Luo, S. Wang, W. Wang, L. Lan, Y. Wang, and F. Nie, "Smooth-Guided Implicit Data Augmentation for Domain Generalization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 3, pp. 4984–4995, 2025.

[15] J. Lin, Y. Tang, J. Wang, and W. Zhang, "Constrained Maximum Cross-Domain Likelihood for Domain Generalization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 2, pp. 2013–2027, 2025.

[16] R. Volpi, H. Namkoong, O. Sener, J. C. Duchi, V. Murino, and S. Savarese, "Generalizing to unseen domains via adversarial data augmentation," *Proceedings of the Advances in neural information processing systems*, vol. 31, 2018.

[17] L. Li, K. Gao, J. Cao, Z. Huang, Y. Weng, X. Mi, Z. Yu, X. Li, and B. Xia, "Progressive domain expansion network for single domain generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 224–233.

[18] F. Qiao, L. Zhao, and X. Peng, "Learning to learn single domain generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 556–12 565.

[19] T. Chen, M. Baktashmotlagh, Z. Wang, and M. Salzmann, "Center-aware Adversarial Augmentation for Single Domain Generalization," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2023, pp. 4157–4165.

[20] Z. Xu, D. Liu, J. Yang, C. Raffel, and M. Niethammer, "Robust and generalizable visual representation learning via random convolutions," *CoRR*, 2020. [Online]. Available: https://arxiv.org/abs/2007.13003

[21] N. Efthymiadis, G. Tolias, and O. Chum, "Crafting Distribution Shifts for Validation and Training in Single Source Domain Generalization," *CoRR*, vol. abs/2409.19774, 2024.

[22] Z. Rao, J. Guo, L. Tang, Y. Huang, X. Ding, and S. Guo, "SRCD: Semantic Reasoning With Compound Domains for Single-Domain Generalized Object Detection," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–10, 2024.

[23] J. Cha, S. Chun, K. Lee, H.-C. Cho, S. Park, Y. Lee, and S. Park, "Swad: Domain generalization by seeking flat minima," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 34, pp. 22 405–22 418, 2021.

[24] M. Wortsman, G. Ilharco, S. Y. Gadre, R. Roelofs, R. Gontijo-Lopes, A. S. Morcos, H. Namkoong, A. Farhadi, Y. Carmon, S. Kornblith *et al.*, "Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time," in *Proceedings of the International Conference on Machine Learning*, 2022, pp. 23 965–23 998.

[25] D. Arpit, H. Wang, Y. Zhou, and C. Xiong, "Ensemble of averages: Improving model selection and boosting performance in domain generalization," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 35, pp. 8265–8277, 2022.

[26] A. Rame, M. Kirchmeyer, T. Rahier, A. Rakotomamonjy, P. Gallinari, and M. Cord, "Diverse weight averaging for out-of-distribution general-

ization," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 35, pp. 10 821–10 836, 2022.

[27] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, "Sharpness-aware Minimization for Efficiently Improving Generalization," in *Proceedings of the International Conference on Learning Representations*, 2021, pp. 1–19.

[28] X. Zhang, R. Xu, H. Yu, H. Zou, and P. Cui, "Gradient Norm Aware Minimization Seeks First-Order Flatness and Improves Generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2023, pp. 20 247–20 257.

[29] P. Wang, Z. Zhang, Z. Lei, and L. Zhang, "Sharpness-Aware Gradient Matching for Domain Generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2023, pp. 3769–3778.

[30] S. Shin, H. Bae, B. Na, Y. Kim, and I. Moon, "Unknown Domain Inconsistency Minimization for Domain Generalization," in *Proceedings of the International Conference on Learning Representations*, 2024, pp. 1–25.

[31] X. Zhang, R. Xu, H. Yu, Y. Dong, P. Tian, and P. Cui, "Flatness-Aware Minimization for Domain Generalization," in *IEEE/CVF International Conference on Computer Vision*. IEEE, 2023, pp. 5166–5179.

[32] A. Li, L. Zhuang, X. Long, M. Yao, and S. Wang, "Seeking Consistent Flat Minima for Better Domain Generalization via Refining Loss Landscapes," *CoRR*, vol. abs/2412.13573, 2024.

[33] T. Garipov, P. Izmailov, D. Podoprikhin, D. P. Vetrov, and A. G. Wilson, "Loss Surfaces, Mode Connectivity, and Fast Ensembling of DNNs," in *Proceedings of the Advances in Neural Information Processing Systems*, 2018, pp. 8803–8812.

[34] F. Draxler, K. Veschgini, M. Salmhofer, and F. Hamprecht, "Essentially no barriers in neural network energy landscape," in *Proceedings of the International Conference on Machine Learning*, 2018, pp. 1309–1318.

[35] D. A. McAllester, "PAC-Bayesian Model Averaging," in *Proceedings of the Twelfth Annual Conference on Computational Learning Theory*, S. Ben-David and P. M. Long, Eds. ACM, 1999, pp. 164–170.

[36] N. Ubayashi, J. Nomura, and T. Tamai, "Archface: a contract place where architectural design and code meet together," in *Proceedings of ACM/IEEE International Conference on Software Engineering*. ACM, 2010, pp. 75–84.

[37] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "CosFace: Large Margin Cosine Loss for Deep Face Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5265–5274.

[38] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[39] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proceedings of the International Conference on Machine Learning*, 2015, pp. 1180–1189.

[40] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A. Y. Ng *et al.*, "Reading digits in natural images with unsupervised feature learning," in *Proceedings of the Advances in Neural Information Processing Systems workshop on deep learning and unsupervised feature learning*, vol. 2011, no. 5. Granada, Spain, 2011, p. 7.

[41] J. J. Hull, "A database for handwritten text recognition research," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 550–554, 1994.

[42] D. Hendrycks and T. G. Dietterich, "Benchmarking Neural Network Robustness to Common Corruptions and Perturbations," in *Proceedings of the International Conference on Learning Representations*, 2019, pp. 1–16.

[43] A. Krizhevsky and G. Hinton, "Learning Multiple Layers of Features from Tiny Images," Master's thesis, Department of Computer Science, University of Toronto, Toronto, Ontario, Canada, 2009.

[44] Z. Wang, Y. Luo, Z. Huang, and M. Baktashmotlagh, "FFM: Injecting Out-of-Domain Knowledge via Factorized Frequency Modification," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2023, pp. 4135–4144.

[45] I. Gulrajani and D. Lopez-Paz, "In Search of Lost Domain Generalization," in *Proceedings of the International Conference on Learning Representations*, 2021, pp. 1–29.

[46] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[47] S. Zagoruyko and N. Komodakis, "Wide residual networks," *CoRR*, 2016. [Online]. Available: https://arxiv.org/abs/1605.07146

[48] H. Robbins and S. Monro, "A Stochastic Approximation Method," *The Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400–407, 1951.

[49] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in *Proceedings of the International Conference on Learning Representations*, 2019, pp. 1–18.

[50] V. 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, vol. 2033.

[51] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto, "Unified deep supervised domain adaptation and generalization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5715–5725.

[52] X. Xu, X. Zhou, R. Venkatesan, G. Swaminathan, and O. Majumder, "d-sne: Domain adaptation using stochastic neighborhood embedding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2497–2506.

[53] F. M. Carlucci, A. D'Innocente, S. Bucci, B. Caputo, and T. Tommasi, "Domain generalization by solving jigsaw puzzles," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2229–2238.

[54] R. Volpi and V. Murino, "Addressing model vulnerability to distributional shifts over image transformation sets," in *Proceedings of the International Conference on Computer Vsion*, 2019, pp. 7980–7989.

[55] Z. Huang, H. Wang, E. P. Xing, and D. Huang, "Self-challenging improves cross-domain generalization," in *Proceedings of the European Conference on Computer Vision*. Springer, 2020, pp. 124–140.

[56] X. Fan, Q. Wang, J. Ke, F. Yang, B. Gong, and M. Zhou, "Adversarially adaptive normalization for single domain generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8208–8217.

[57] J. Chen, Z. Gao, X. Wu, and J. Luo, "Meta-causal learning for single domain generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7683–7692.

[58] X. Xu, J. Yang, W. Shi, S. Ding, L. Luo, and J. Liu, "PhysAug: A Physical-guided and Frequency-based Data Augmentation for Single-Domain Generalized Object Detection," in *Proceedings of the the Association for the Advancement of Artificial Intelligence*. AAAI Press, 2025, pp. 21 815–21 823.

[59] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz, "Invariant Risk Minimization," *CoRR*, vol. abs/1907.02893, 2019.

[60] S. Yan, H. Song, N. Li, L. Zou, and L. Ren, "Improve Unsupervised Domain Adaptation with Mixup Training," *CoRR*, vol. abs/2001.00677, 2020.

[61] D. Krueger, E. Caballero, J. Jacobsen, A. Zhang, J. Binas, D. Zhang, R. L. Priol, and A. C. Courville, "Out-of-Distribution Generalization via Risk Extrapolation (REx)," in *Proceedings of International Conference on Machine Learning*, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 2021, pp. 5815–5826.

[62] L. Chen, Y. Zhang, Y. Song, A. van den Hengel, and L. Liu, "Domain Generalization via Rationale Invariance," in *Proceedings of the International Conference on Computer Vsion*. IEEE, 2023, pp. 1751–1760.

[63] T. Li, P. Zhou, Z. He, X. Cheng, and X. Huang, "Friendly Sharpness-Aware Minimization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2024, pp. 5631–5640.

[64] X. Lyu, Q. Xu, Z. Yang, S. Lyu, and Q. Huang, "SSE-SAM: Balancing Head and Tail Classes Gradually Through Stage-Wise SAM," in *Proceedings of the the Association for the Advancement of Artificial Intelligence*, T. Walsh, J. Shah, and Z. Kolter, Eds. AAAI Press, 2025, pp. 19 278–19 286.

[65] L. Liu, N. Wang, D. Zhou, D. Liu, X. Yang, X. Gao, and T. Liu, "Generalizable Prompt Learning via Gradient Constrained Sharpness-Aware Minimization," *IEEE Transactions on Multimedia*, vol. 27, pp. 1100–1113, 2025.

[66] Z. Zhou, M. Wang, Y. Mao, B. Li, and J. Yan, "Sharpness-Aware Minimization Efficiently Selects Flatter Minima Late In Training," in *Proceedings of the International Conference on Learning Representations*, 2025, pp. 1–32.

[67] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE." *Journal of machine learning research*, vol. 9, no. 11, 2008.