A2XP: Towards Private Domain Generalization

Geunhyeok Yu Hyoseok Hwang
Department of Software Convergence, Kyung Hee University, Republic of Korea
{geunhyeok, hyoseok}@khu.ac.kr

Abstract

Deep Neural Networks (DNNs) have become pivotal in various fields, especially in computer vision, outperforming previous methodologies. A critical challenge in their deployment is the bias inherent in data across different domains, such as image style and environmental conditions, leading to domain gaps. This necessitates techniques for learning general representations from biased training data, known as domain generalization. This paper presents Attend to eXpert Prompts (A2XP), a novel approach for domain generalization that preserves the privacy and integrity of the network architecture. A2XP consists of two phases: Expert Adaptation and Domain Generalization. In the first phase, prompts for each source domain are optimized to guide the model towards the optimal direction. In the second phase, two embedder networks are trained to effectively amalgamate these expert prompts, aiming for an optimal output. Our extensive experiments demonstrate that A2XP achieves state-of-the-art results over existing non-private domain generalization methods. The experimental results validate that the proposed approach not only tackles the domain generalization challenge in DNNs but also offers a privacy-preserving, efficient solution to the broader field of computer vision. Code is available at https://github.com/AIRLABkhu/A2XP.

1. Introduction

Deep Neural Networks (DNNs) are recognized as the most potent models in machine learning. They have achieved remarkable success in various fields, particularly in computer vision, where they have surpassed previous methodologies. Although DNNs are versatile and universal function approximators, the data they process often carry biases related to factors such as image style [42], sensor parameters [46], as well as painting styles [23]. These biases create distinct distributions, known as domains, with inherent gaps between them. The inability of DNNs to generalize across these domains necessitates an impractically large amount of unbiased training data to mitigate the model's bias. Consequently, this limitation underscores the importance of developing techniques that can learn general representations from biased training data. This challenge, extensively studied in various researches [48, 52], is referred to as domain generalization.

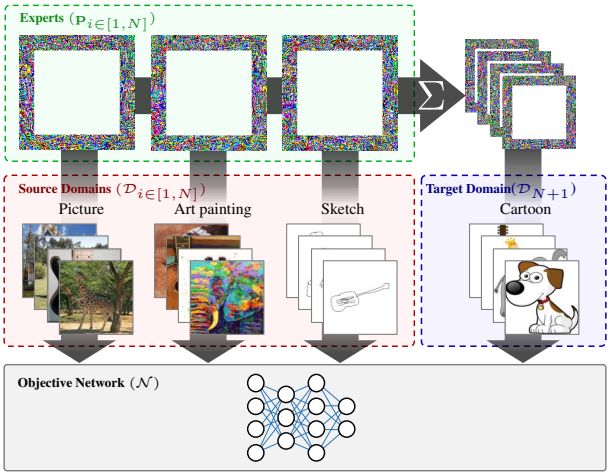


Figure 1. Flow diagram of the proposed method A2XP.

To address the generalization issue, various research topics such as domain adaptation [4, 17, 27, 34], meta-learning [7, 35, 47], and transfer learning [21, 37, 41] have been explored. Domain adaptation, which shares similarities with domain generalization, specifically aims to mitigate domain gaps. The primary difference between the two lies in the visibility of the target domain [52]. In domain adaptation, the target domain is known and the goal is to adapt a pre-trained network to this specific domain. This involves learning new knowledge from the target domain while utilizing existing knowledge from source domains [49], a task that is generally more straightforward than domain generalization. In contrast, domain generalization operates without the need for target domain data, focusing on making the network robust to shift from a source domain to an unknown target domain. While these two approaches are distinct, the ability of domain adaptation to understand domain shifts can be beneficial for domain generalization. Our approach is based on this concept. We propose that if a network can effectively map input from any ar-

bitrary domain into a generalized manifold space, the challenge of domain generalization could be transformed into a regression problem. In this scenario, adaptation strategies could provide crucial insights for determining the direction of this regression.

Most of the methods that mitigate domain gaps necessitate access to the architecture and parameters of the target network [2, 14, 19, 22, 33]. For instance, Domain Adversarial Neural Network (DANN) [14] and Style-Agnostic Network (SagNet) [33] aims to fine-tune the backbone network to extract domain-agnostic features. Similarly, Common and Specific Visual Prompt Tuning (CSVPT) [22] employs prompt tokens in conjunction with a Vision Transformer (ViT) [6] to address these challenges. However, these approaches often require modifications to the network's architecture or parameters, which can pose significant privacy concerns.

Visual Prompting (VP) [1] provides a solution to privacy concerns by fine-tuning an objective network through adversarial reprogramming without altering the network's architecture or parameters. It only tunes additional parameters known as prompts, which are added to the input image rather than being embedded within the network. Inspired by this, we added a prompt to the input to address the privacy issue [26]. However, VP faces a limitation: an excessive number of pixels in a prompt can disrupt training. To overcome this, we train multiple prompts, referred to as "experts," and integrate them using an attention mechanism. This strategy aligns with the concept of addressing domain generalization as a direction regression problem, where these experts serve as guides to identify the optimal direction for generalization.

In this study, we aim to disentangle the domain generalization problem into two steps: expert adaptation and domain generalization, while keeping the privacy of the objective network. We propose ***A**ttend **to** e**X**pert **P**rompts (A2XP)* which is a novel domain generalization method that solves this issue. In the expert adaptation step, we optimize prompts for each source domain to prepare the hints to find the optimal direction. In the domain generalization step, two embedder networks are trained to properly mix the expert prompts so that the output is in the optimal direction. The main contributions of this study can be summarized as follows:

- Inspired by VP, we introduce A2XP, which is a novel and simple domain generalization method that protects privacy.
- We mathematically analyze the generalization issue as an optimization of a linear combination problem.
- We further demonstrate the effectiveness and characteristics of A2XP and its component factors through extensive experiments.
- A2XP achieves SOTA over existing non-private domain generalization methods with significantly lower computational resource requirements.

2. Related Works

2.1. Domain Generalization

The objective of domain generalization is to reduce the gaps between visible source domains and unseen target domains. There are several approaches such as domain alignment [2, 14, 24, 25, 31–33, 40], meta learning [7, 8, 35, 47], ensemble learning [9, 19, 30, 50] and, representation disentanglement [3, 35] as categorized by Zhou *et al*. [52].

Ganin *et al*. [14] introduced DANN that discriminates the domains so that the network can find domain-agnostic features. SagNet [33] also discriminates the domains by adversarially learning content bias and style bias. Cha *et al*. [2] aligned domains by employing a regularization term to the loss function based on mutual information among domains. Diversify-Aggregate-Repeat Training (DART) [19] is an ensemble learning method that diversifies the source domain by applying data augmentation to independently capture diverse features using multiple networks, then aggregates networks and repeats these procedures. DART can enhance the generalization performance, but it also takes a massive amount of memory.

Our approach basically follows the idea of domain alignment and ensemble learning. We train multiple expert prompts that align source domains each. Then, it aggregates the experts to align a novel target domain. The experts give a hint to find the direction to the optima of a target domain on the fly, and we take different simple generalization steps to each sample of the target domain.

2.2. Prompt Tuning in Computer Vision

Prompt tuning is a transfer learning technique that requires a tiny amount of additional parameters. Prompt tuning in computer vision was first introduced by Visual Prompt Tuning (VPT) [20] for transfer learning with a small number of parameters. VPT proved that prompt tuning is a stronger transfer learning technique than full fine-tuning and linear probing. However, access to change the architecture of the network is required to apply VPT. Bahng *et al*. [1] introduced adversarial reprogramming [10]-based prompting for general pre-training using vision-language relationships. They successfully incorporated visual and lingual representations only using an optimized perturbation to the inputs. We will call this prompting "input prompting". Huang *et al*. [18] proposed Diversity-Aware Meta Visual Prompting (DAM-VP) that transfers a network to another target dataset that contains diverse representation distribution. DAM-VP separates a set of data into clusters and updates the prompt using each of the clusters. Then, it gathers all prompts from clusters to capture the diversity and

provides a detailed representation of the whole data distribution. Inspired by DAM-VP, we captured the diversity of data distribution from the source domains and generalized the target domain.

2.3. Attention Mechanism

The key idea of the attention mechanism is activating important features and silencing less important features. Many of the modern deep learning architectures have employed attention mechanism [16, 45]. Squeeze-and-Excitation Networks [16] focused on weighting each channel of a large feature map before aggregating them. Transformer [6, 45], one of the most effective architectures, lies its core on the attention mechanism. Transformers have two different types of attention mechanisms with different origins of the "query". Cross-attention builds "query" from the same source of "key" and "value" while self-attention builds from a different source. Cross-attention is used to capture the importance of "values" depending on the relationship with other data. We used the cross-attention mechanism to properly combine multiple experts.

3. Methods

Domain generalization is a task that generally fits a model to unseen target domains using known source domains. In this section, we describe A2XP, our novel domain generalization method, using input prompting.

3.1. Algorithm Overview

A2XP operates through a two-phase approach. Initially, it performs source-wise adaptation by crafting 'experts' - specific adaptation prompts for each source domain. This step is conducted end-to-end, predominantly via error back-propagation. The subsequent phase is dedicated to domain generalization, where image-specific prompts for the target domain are created for each input image by averaging the weights of all experts, determined through an attention-based algorithm. In this phase, the system utilizes two separate trainable encoders: one for the input images and another for the pre-trained experts. An expert's weight is derived from the similarity between the encoded input image and the expert's embedding. These phases are termed *Expert Adaptation* and *Attention-based Generalization*, respectively. The A2XP algorithm is detailed in Algorithm 1, with the validation process illustrated in Figure 2.

3.2. Idea Formulation

We first formulate our idea as a concrete guideline for detailed understanding. Domain generalization using input prompting can be formulated as follows. For $N+1$ domains $X_{i\in[1,N+1]}$, we can select X_{N+1} as a target domain and others as source domains. The network named $\mathcal{N}$ is

Algorithm 1 Training and Inference Scenario of A2XP

Input: $X_1, X_2, \cdots, X_{N+1}$
Parameter: Objective network $\mathcal{N}$
Parameter: Meta prompt $\mathbf{p}_{\text{meta}}$
Parameter: Learning rates α_A, α_G
Output: Experts $\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_N$
Output: Encoder head parameters $\theta_{\mathcal{E}_\text{T}}, \theta_{\mathcal{E}_\text{E}}$

▷ Training $\mathbf{p}_{i\in[1,N]}$
for $X_{i\in[1,N]}$ **do**
  $\mathbf{p}_i \leftarrow \mathbf{p}_{\text{meta}}$
  for $(\mathbf{x}_{i,j}, \mathbf{y}_{i,j}) \in X_i$ **do**
    $\mathbf{p}_i \leftarrow \mathbf{p}_i - \alpha_A \partial \mathcal{L}_{\text{KL}}(\mathcal{N}(\mathbf{x}_{i,j} + \mathbf{p}_i), \mathbf{y}_{i,j}) / \partial \mathbf{p}_i$
  end for
end for
$\mathbf{p}_i \leftarrow \mathbf{p}_i / \|\mathbf{p}_i\|_2$ ▷ Normalizing expert prompts
▷ Training $\theta_{\mathcal{E}_\text{T}}, \theta_{\mathcal{E}_\text{E}}$
for $X_{i\in[1,N]}$ **do**
  for $(\mathbf{x}_{i,j}, \mathbf{y}_{i,j}) \in X_i$ **do**
    $Q, K \leftarrow \mathcal{E}_\text{T}(\mathbf{x}_{i,j}), \mathcal{E}_\text{E}(\mathbf{p}_{k\in[1,N]})$
    $\mathbf{p}_{i,j} \leftarrow \sum_{k=1}^{N} \mathbf{p}_k \tanh(QK_k^\top)$
    $l \leftarrow \nabla \mathcal{L}_{\text{KL}}(\mathcal{N}(\mathbf{x}_{i,j} + \mathbf{p}_{i,j}), \mathbf{y}_{i,j})$
    $\theta_{\mathcal{E}_\text{T}} \leftarrow \theta_{\mathcal{E}_\text{T}} - \alpha_G \partial l / \partial \theta_{\mathcal{E}_\text{T}}$ ▷ Update θ, not $\mathbf{p}$
    $\theta_{\mathcal{E}_\text{E}} \leftarrow \theta_{\mathcal{E}_\text{E}} - \alpha_G \partial l / \partial \theta_{\mathcal{E}_\text{E}}$
  end for
end for
▷ Inference on unseen X_{N+1}
for $\mathbf{x}_{N+1,j} \in X_{N+1}$ **do**
  $Q, K \leftarrow \mathcal{E}_\text{T}(\mathbf{x}_{N+1,j}), \mathcal{E}_\text{E}(\mathbf{p}_{k\in[1,N]})$
  $\mathbf{p}_{N+1,j} \leftarrow \sum_{k=1}^{N} \mathbf{p}_k \tanh(QK_k^\top)$
  $\hat{\mathbf{y}}_{N+1,j} \leftarrow \mathcal{N}(\mathbf{x}_{N+1,j} + \mathbf{p}_{N+1,j})$ ▷ Prediction
end for

given with fixed pre-trained parameters; there exist decision boundaries of the network. Let an expert for the i-th domain be $\mathbf{p}_i \in \mathbb{R}^{d_\text{prompt}}$ where d_prompt is the dimension of a prompt. Then, $\mathbf{p}_{i\in[1,N]}$ represents the optimal direction that shifts the inputs in source domains and we can optimize those with the known source data. Prompt for the target domain $\mathbf{p}_{N+1}$ cannot be directly optimized because the target domain is invisible.

We approximate $\mathbf{p}_{N+1}$ as a linear combination of $\mathbf{p}_{i\in[1,N]}$ as following equation:

$$\mathbf{p}_{N+1} = \sum_{i=1}^{N} \lambda_i \mathbf{p}_i, \quad \lambda_i = \Lambda(\mathbf{p}_i | \mathbf{x} \in X_i) \tag{1}$$

where Λ is a conditional function that represents the optimal weights for $\mathbf{p}_i$ when $\mathbf{x} \in X_i$ is given. Let say

$$J(\lambda_i) = \text{KL}(\mathcal{N}(\mathbf{x}_{N+1} + \mathbf{p}_{N+1}) \| \mathcal{D}_{N+1}) \tag{2}$$

be the objective function where $\mathbf{x}_{N+1} \in X_{N+1}$, $\mathcal{D}_{N+1}$ is the target distribution for $\mathcal{N}$ of $\mathbf{x}_{N+1} + \mathbf{p}_{N+1}$ and KL refers to the KL-Divergence function. Then the likelihood function L has a relationship as following

$$L(\mathcal{D}_{N+1} | \mathcal{N}(\mathbf{x}_{N+1} + \mathbf{p}_{N+1})) \propto e^{-J(\lambda_i)}. \tag{3}$$

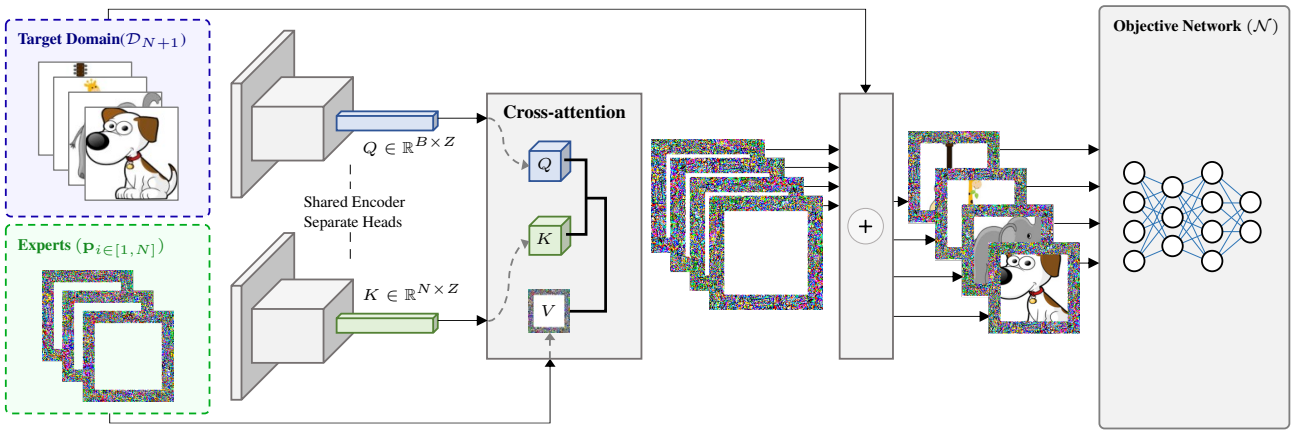


Figure 2. Inference procedure of A2XP. There are experts from source domains and target images of an unseen target domain. The experts are image-dependently mixed through an attention-based algorithm and added to the specific image.

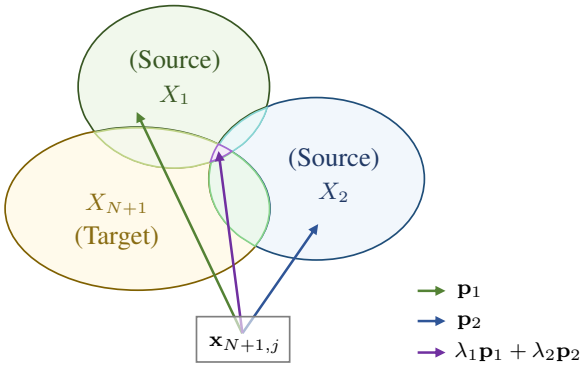


Figure 3. Geometric concept of A2XP as a linear combination in 2D manifold space with two source domains.

This formulation shows that

$$J(\lambda_i) \propto -\log L(\mathcal{D}_{N+1}|\mathcal{N}(\mathbf{x}_{N+1} + \mathbf{p}_{N+1})), \quad (4)$$

minimizing J by training Λ is equivalent to maximizing L. This idea can be explained as follows. If there are ranges of optimal prompts for each domain, an expert must be a point inside the range. And because the target prompts are formulated as Equation 1, the geometry of the prompt space can be conceptually visualized like Figure 3.

3.3. Expert Adaptation

Our objective is to mix multiple expert prompts into a single prompt. For this to be effective, each expert must be proficiently trained in their primary field, which in our case is the domain. We utilize adversarial reprogramming [10], a straightforward gradient-based method, for adapting these experts. While this approach suffices in specific scenarios, it falls short in domains vastly different from the pre-training domain. To address this, we employed meta prompts [18] to initialize the expert prompts. Meta prompt refers to pre-trained prompts that can be used to initialize a visual prompt.

3.4. Attention-based Generalization

Our key idea is to combine the experts in a way that makes images from unseen domains to be correctly classified. We combined experts by weight-averaging them. A weight must indicate how much an expert is needed for a given specific image. This requirement can be implemented using the cross-attention mechanism. In this case, the experts become "keys" (K) and "values" (V), a target image becomes "query" (Q) of attention. The attention weight is calculated as the similarity between Q and K. Instead of directly comparing Q and K, we used embedding vectors. We have a pre-trained network as a shared embedder network and two different trainable head linear layers for Q and K each. Q and K are embedded through a shared encoder and their respective linear heads. Then scalar attention weights are obtained as much as the number of the experts, and $VQK^\top$ becomes the prompt for the target image.

However, there are two problems. First, the experts are independently optimized in different domains, which makes a significant difference in scales. We solved this by dividing the experts with the L_2-norm of each of themselves for normalization. The second problem is that the weights can be saturated too much because the weights are independently calculated without scaling such as $\mathrm{softmax}$ function. Mapping the weights into $[-1, 1]$ using $\tanh$ function mitigates this problem. As a result, the prompt ($\mathbf{p}_{N+1,k}$) for a k-th target image ($\mathbf{x}_{N+1,k} \in X_{N+1}$) can be formulated as:

$$\mathbf{p}_{N+1,k} = \sum_{i=1}^{N} \frac{\mathbf{p}_i}{\|\mathbf{p}_i\|_2} \mathcal{E}_\mathrm{T}(\mathbf{x}_{N+1,k}) \mathcal{E}_\mathrm{E}(\frac{\mathbf{p}_i}{\|\mathbf{p}_i\|_2})^\top, \quad (5)$$

where $\mathcal{E}_\mathrm{T}$ and $\mathcal{E}_\mathrm{E}$ denote the embedders for target images and experts respectively. Once the generalization is trained, the embedding vectors of the experts are fixed because the experts will not be changed. Thus, the expert embedding procedure is no longer needed in evaluation.

Method	DART [19] Supported	PACS [23] Picture	Art	Cartoon	Sketch	Avg.	VLCS [42] VOC 2007	LabelMe	Caltech101	SUN09	Avg.
SAM [13]	✓	18.41	15.13	21.38	19.12	18.51	44.72	46.02	61.13	41.62	48.38
ERM [44]	✓	97.08	87.19	86.25	82.38	88.22	75.60	64.47	97.08	77.49	78.66
SagNet [33]	✓	91.99	84.56	69.19	20.07	66.45	51.02	62.63	61.13	61.16	58.98
DANN [14]	✓	97.68	89.93	86.41	81.11	88.78	77.86	66.97	98.59	73.53	79.24
MIRO [2]	✓	96.48	90.79	90.46	83.59	90.33	78.05	66.68	97.53	71.97	78.56
A2XP (ours)	✗	**99.07**	**95.27**	**98.07**	**87.85**	**95.07**	**84.07**	**68.72**	**99.62**	**80.19**	**83.15**

(a) Comparison with other methods in the target domain. DART [19] was applied to the baselines for their best performance.

Source	Target Picture	Art	Cartoon	Sketch	**Avg.**	Source	Target VOC 2007	LabelMe	Caltech101	SUN09	**Avg.**
P	-	99.88	99.76	99.52	**99.72**	V	-	78.20	99.79	87.84	**88.61**
A	96.53	-	96.39	94.87	**95.93**	L	89.28	-	99.36	84.19	**90.94**
C	98.63	98.76	-	98.17	**98.52**	C	88.48	78.58	-	84.16	**83.74**
S	91.45	91.12	91.98	-	**91.52**	S	90.23	76.84	100.00	-	**89.02**
Avg.	**95.54**	**96.59**	**96.04**	**97.52**	**96.42**	**Avg.**	**89.33**	**77.87**	**99.72**	**85.40**	**88.08**

(b) Source domain evaluation on PACS [23] (left) and VLCS [42] (right) datasets.

Table 1. Target domain and source domain evaluations. Target domain evaluation was conducted to compare A2XP with other state-of-the-art methods. Source domain evaluation was conducted to see if it is still effective in the source domains.

4. Experiments and Analysis

In this section, we perform leave-one-domain-out evaluation and more extensive experiments mainly on PACS [23] and VLCS [42] datasets and partially on Office-Home [46] dataset to demonstrate the effectiveness and characteristics of A2XP. PACS dataset consists four domains: **P**icture, **A**rt painting (**A**rt), **C**artoon, and **S**ketch. VLCS dataset is composed of four subdatasets, each representing a different domain: **V**OC 2007 [11], **L**abel Me [38], **C**altech101 [12], and **S**UN09 [51]. The Office-Home [46] dataset consists of four domains: Art painting, Clipart, Product, and Real image. The experiments were conducted on Ubuntu Server 18.04 with an Intel Xeon Gold 6226R 2.90GHz and NVIDIA RTX 3090.

4.1. Implementation Details

For our study, we selected a CLIP [37]-pre-trained ViT [6] as the objective network. The experts within this framework were optimized through end-to-end backpropagation. The prompt size was chosen based on the specifications of VP [1], which employs a padding size of 30. We used a learning rate of 1.0E-4 and stochastic gradient descent with momentum [36] for optimization. Given that a tiny network suffices for the shared embedder networks of A2XP, we opted for an ImageNet [5]-pre-trained ResNet18 [15] as the backbone. Following the shared encoder, two distinct trainable linear heads are attached to specialize the features into Q and K. To demonstrate A2XP's efficiency in simplifying problems, we limited the number of updates to 1,000, unless otherwise specified. For optimization during generalization, we used AdamW [29]. We implemented a learning rate decay to 10% of its initial value, utilizing the Cosine Annealing with Warm Restarts [28] algorithm, across the entire generalization procedure.

4.2. Leave-One-Domain-Out Evaluation

We conducted a leave-one-domain-out evaluation to assess the domain generalization performance, the results of which are detailed in Table 1a. In this experiment, we evaluated several methods, including domain generalization methods such as SagNet [33], DANN [14], and Mutual Information Regularization with Oracle (MIRO) [2], as well as non-domain generalization methods like Sharpness-Aware Minimization (SAM) [13] and Empirical Risk Minimization (ERM) [44], following the approach used by DART [19]. These five baselines were augmented using DART, which is an ensemble learning-based method for domain generalization. A2XP outperformed all other methods in each target domain on both PACS and VLCS datasets. Notably, it achieved a 4.74% increase in average accuracy on PACS dataset and a 4.99% increase on VLCS dataset. It is important to mention that DART does not ensure the privacy of the objective network.

4.3. Evaluation on Source Domains

Domain generalization focuses on adapting models to both unseen and known source domains. We evaluated the generalizability of A2XP in source domains, utilizing the expertise of these domains for the evaluation. Evaluation on all source domains well performed as much as on the target domain as shown in Table 1b. Notably, in PACS, A2XP achieved an average accuracy that was 2.9% higher than the domain adaptation performance, as detailed in Table 2.

	Expert Adaptation					Attention-based Generalization				
	$P \rightarrow P$	$A \rightarrow A$	$C \rightarrow C$	$S \rightarrow S$	Avg.	$ACS \rightarrow P$	$PCS \rightarrow A$	$PAS \rightarrow C$	$PAC \rightarrow S$	Avg.
Zero	**97.54**	73.88	**95.52**	94.55	90.37	99.07	95.07	98.12	**88.22**	**95.12**
Uniform	78.62	60.25	87.63	87.76	78.57	**99.15**	94.97	98.17	88.02	95.08
Normal	87.72	73.00	84.90	**97.89**	85.88	98.99	95.15	**98.39**	87.81	95.08
Meta [18]	94.07	**93.12**	93.60	93.28	**93.52**	99.07	**95.27**	98.07	87.85	95.07
	$A \rightarrow A$	$C \rightarrow C$	$P \rightarrow P$	$R \rightarrow R$	Avg.	$CPR \rightarrow A$	$APR \rightarrow C$	$ACR \rightarrow P$	$ACP \rightarrow R$	Avg.
Zero	21.18	38.95	61.93	43.10	41.29	67.57	57.98	66.55	71.29	65.85
Uniform	21.63	32.94	44.92	46.71	36.55	67.41	58.33	67.27	71.77	66.20
Normal	28.92	32.51	40.17	23.36	31.24	67.74	58.35	67.83	71.22	66.29
Meta [18]	**47.05**	**54.39**	**69.66**	**52.03**	**56.35**	**77.42**	**65.73**	**81.93**	**83.15**	**77.06**

Table 2. Generalization and adaptation performance in PACS [23] (top) and Office-Home [46] (bottom) datasets using different prompt initialization before adaptation. Zero initializes as zero tensor, Uniform initializes using uniform distribution $\mathcal{U}(-0.03, 0.03)$, and Normal initializes using Gaussian distribution $\mathcal{N}(0, 0.03^2)$.

4.4. Importance of Expert Processing

Our study demonstrates that normalizing and scaling experts are crucial for the effective functioning of the A2XP module in mixing experts. We conducted an ablation study focusing on three aspects: expert normalization, softmax, and the hyperbolic tangent function, with results detailed in Table 3. We calculated the performance gain of each factor by averaging the gain of every combination of the other two factors. Expert normalization makes experts initially have the same scales by following the normalization in Equation 5. This normalization contributed to a significant accuracy gain of 39.09% in the leave-one-domain-out evaluation. The Softmax function takes a role as an amplifier of attention weights. It was observed to decrease the average accuracy by 4.35%. This decrease is attributed to its tendency to significantly reduce the effect of experts with lower attention weights, even if the differences are insignificant. The attention weights can be saturated during training since the calculation for each weight is independent of other experts. The Hyperbolic tangent function was applied to prevent such saturation problems and it led to 4.39% accuracy gain. Consequently, the combination of expert normalization and hyperbolic tangent, without the softmax function, proved to be the most effective among the tested factor combinations.

4.5. Impact of Prompt Initialization

In this experiment, we compare several initialization strategies including zero, uniform distribution, Gaussian distribution, and meta prompting to justify the effectiveness of meta prompt initialization. While good initialization might be optional for simpler tasks, its importance escalates with increasing task complexity. For instance, as indicated in Table 2, meta prompt initialization did not show a marked advantage in scenarios where adaptation performance was not outstanding (see the result of PACS). However, in more challenging situations, such as with the Office-Home dataset, meta prompt initialization significantly enhanced performance, from expert training through to generalization training. For example, the adaptation performance of zero initialization was the best among others which is 15.06% lower accuracy. Consequently, the generalization performance of zero initialization is 11.21% lower than meta prompt initialization. we set the number of updates to 10K for the evaluation of the Office-Home dataset. It is noteworthy that there was no significant difference among other initialization strategies, and the correlation between adaptation and generalization was not linear. This suggests that effective expert adaptation is a critical foundation for A2XP, and good initialization is a key factor in achieving good adaptation.

Expert Normalization	Softmax	tanh	Avg. Accuracy
			49.35
✓			88.01
	✓		46.96
		✓	57.55
	✓	✓	49.25
✓		✓	**95.07**
✓	✓		88.19
✓	✓	✓	88.19

Table 3. Ablation study about the A2XP module on PACS dataset.

	Picture	Art	Cartoon	Sketch	Avg.
FT	23.71	42.72	56.61	29.12	38.04
LP	83.11	94.04	86.95	86.79	87.72
A2XP + FT	68.62	26.61	17.28	18.83	32.84
A2XP + LP	**99.07**	**95.27**	**98.07**	**87.85**	**95.07**

Table 4. Comparison of tuning range on the objective network with and without A2XP. FT and LP refer to **F**ull **T**uning and **L**inear **P**robing, respectively.

4.6. Effectiveness of A2XP Module

We utilized a CLIP-pre-trained ViT as the objective network, which is also recognized for its well-generalized pre-

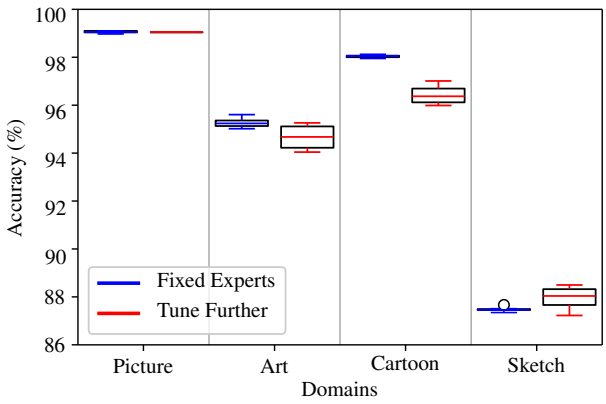


Figure 4. Comparison of two generalization strategies about fixing or tuning the experts in the generalization step.

trained model. We performed an ablation study to demonstrate the efficacy of the A2XP module by quantifying its impact on accuracy enhancement with commonly used fine-tuning approaches such as linear probing and full tuning. Initially, without A2XP, linear probing outperformed full tuning in the domain of generalization. Specifically, linear probing achieved an average accuracy of 38.04%, compared to 32.84% for full tuning. As shown in Table 4, tuning the hidden layers appeared to impact the tuning of the output layer negatively. With the integration of A2XP in linear probing, accuracy was significantly increased across all tested domains. However, in the case of full tuning, the inclusion of A2XP was counterproductive. We analyzed that full tuning is inherently unstable; thus, the A2XP module, positioned before the hidden layers, was adversely affected. To summarize, further tuning might enhance average accuracy in certain scenarios, it generally leads to a decrease in accuracy and contributes to performance instability. Additionally, this implies that training experts through domain adaptation is more beneficial and effective compared to domain generalization.

4.7. Further Expert Tuning

We carried out further experiments with a focus on generalization strategies, concentrating specifically on the experts rather than solely on the networks. The premise was that further tuning of the experts during the generalization phase would facilitate the sharing of domain-specific knowledge among them. To validate the effect of further tuning, we repeated the training ten times on PACS dataset, each time using a different fixed random seed. The results of this experiment are depicted in Figure 4. In the Picture domain, we observed a slight drop in mean accuracy, although this change was not statistically significant. The Art and Cartoon domains exhibited similar results, with average accuracies decreasing by 0.60% and 1.60%, respectively. Notably, the standard deviation in both these domains increased significantly by 0.40%. In contrast, the Sketch domain showed an improvement, with the average accuracy rising by 0.48%, albeit accompanied by a similar increase in the standard deviation of 0.48%. This indicates that while further tuning of experts can lead to improvements in certain domains, it may also introduce greater variability in performance across different domains.

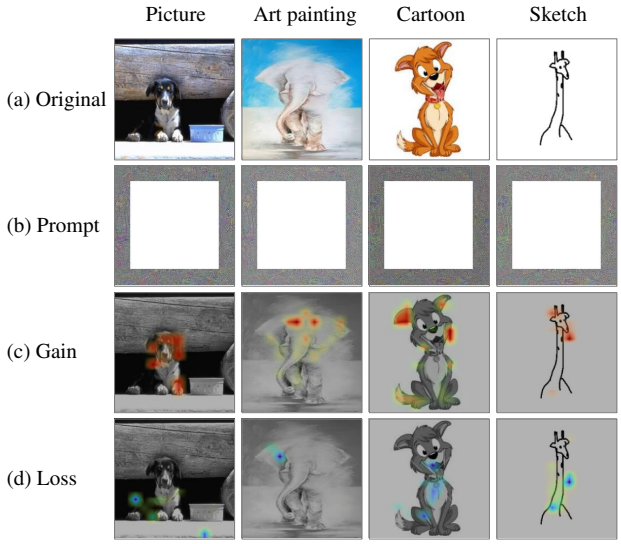
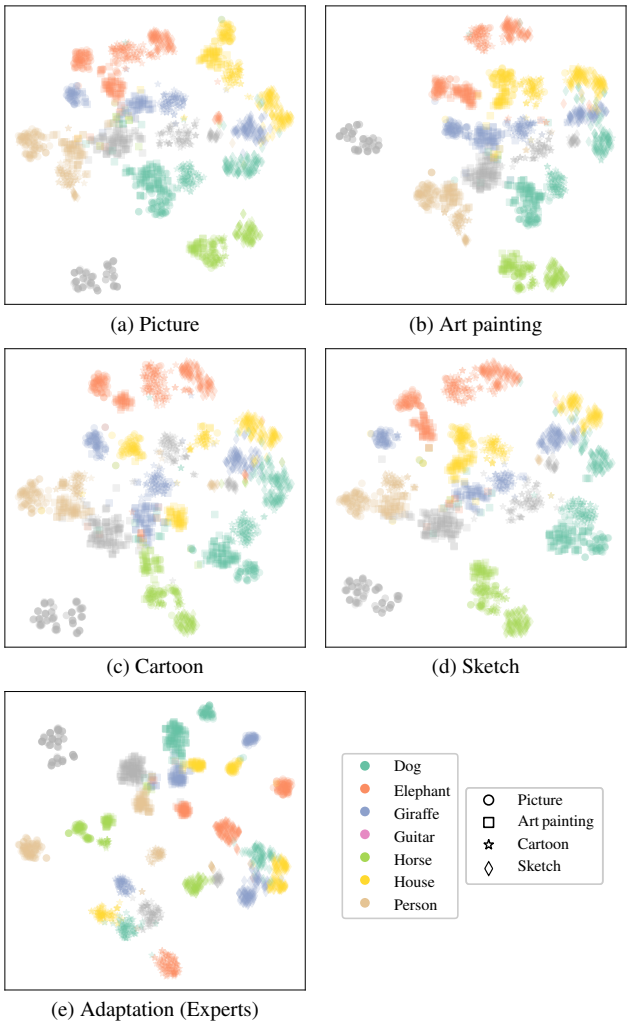


Figure 5. Activation visualization of A2XP using Grad-CAM [39]. (a) shows the input image, (c) and (d) show the relative gain and loss of activation using A2XP prompts in (b), respectively.

4.8. Visualization

To help understand the effects of A2XP on the neural network's focus, we visualized the activation maps. Table 4 demonstrates that while linear probing is generalized in a way, the takes the generalizability even further. This suggests that linear probing without A2XP yields reasonably effective activation maps, and the incorporation of A2XP further refines and improves these activation maps. Consequently, we extended our visualization beyond just the activation maps to include both the gains and losses in activation, as depicted in Figure 5. The prompts shown in the (b) row change the activation maps as much as shown in (c) and (d). The prompts have similar expression because they are from the same experts, but the intensities are different or some of them seem inverted. This means the experts are mixed in different ratios dependent on the target image. They show that A2XP makes the network attend more to the face representation and kills activation on other representations, such as the backgrounds or the body of an animal. Specifically in the Picture domain, (c) shows that it primarily activates the ears of the dog and deactivates the background. In the Sketch domain, it activates representations around the head while it deactivates the background next



(a) Picture (b) Art painting (c) Cartoon (d) Sketch (e) Adaptation (Experts)

Figure 6. t-SNE [43] visualization of correctly classified samples in manifold space. (a)-(d) illustrate the representation achieved through generalization, with Picture, Art Painting, Cartoon, and Sketch as the target domains. (e) depicts the representation of expert adaptation prior to the generalization process.

to the neck and the body, which contains fewer domain-agnostic clues for classification.

Additionally, we visualized the manifold space of the features extracted from the last hidden layer, as shown in Figure 6, to observe how the classes and domains are represented in a 2-dimensional space. Figure 6a–6d shows generalized features are mapped similarly regardless of the target domain. Additionally, samples belonging to the same classes are closely grouped together, even when they originate from different domains. Conversely, as depicted in Figure 6e, samples with the same class label but from different domains are mapped distinctly. It is understandable because the experts are trained independently, and the training does not concern other prompts to be mapped relevantly.

4.9. Space Complexity Analysis

We calculated the space complexity of A2XP compared to DART [19]. DART requires memory proportional to the number of augmentation presets (M) while A2XP requires much less memory space with N expert prompts. Let the number of parameters of the objective network as $S_{\mathcal{N}}$, the big-**O** notation of DART and A2XP are

$$\mathbf{O}_{\text{DART}}(M) = MS_{\mathcal{N}}, \tag{6}$$

$$\mathbf{O}_{\text{A2XP}}(N) = NS_{\mathbf{p}} + S_{\mathcal{N}} + S_{\mathcal{E}} = NS_{\mathbf{p}}, \tag{7}$$

where $S_{\mathbf{p}}$ and $S_{\mathcal{E}}$ denote the number of parameters in a single prompt and the encoders, respectively. This demonstrates a key advantage of our method: its reduced memory usage compared to comparing approaches.

5. Conclusion and Future Works

In this work, we proposed a novel domain generalization method A2XP. A2XP solves the domain generalization problem as a direction regression problem by disentangling it into two steps: domain adaptation and domain generalization. In the domain adaptation step, experts are trained on each source domain to take the place of a hint. In the domain generalization step, a network is trained to mix those experts properly depending on the target images. A2XP does not require changing the architecture or parameters of the objective network, which is the key to keeping the network private. A2XP outperformed state-of-the-art with a limited number of updates in PACS, VLCS datasets and successfully performed not only on the target domain but also on the source domains. We proved this problem definition mathematically based on the likelihood maximization problem. We also justified the effectiveness and characteristics by conducting extensive experimentation.

Our work introduced a remarkable issue of privacy in domain generalization and proposed a powerful domain generalization method, but it also has limitations. A2XP requires well-trained experts for the domain generalization step. However, to the best of our knowledge, some datasets are difficult to adapt with input prompts. And the problems with adaptation techniques must be improved for A2XP to be widely used. We hope that this work encourages more research to solve this issue and improve this novel framework, and this will also be left as our future work.

Acknowledgement. This work was partly supported by a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (NRF-2022R1C1C1008074), and by an Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korean government (MSIT) (No.RS-2022-00155911, Artificial Intelligence Convergence Innovation Human Resources Development (Kyung Hee University)).

References

[1] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, 2022. 2, 5

[2] Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. In *European Conference on Computer Vision*, pages 440–457. Springer, 2022. 2, 5

[3] Prithvijit Chattopadhyay, Yogesh Balaji, and Judy Hoffman. Learning to balance specificity and invariance for in and out of domain generalization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 301–318. Springer, 2020. 2

[4] Bharath Bhushan Damodaran, Benjamin Kellenberger, Remi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 1

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2, 3, 5

[7] 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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 200–216. Springer, 2020. 1, 2

[8] Yingjun Du, Xiantong Zhen, Ling Shao, and Cees GM Snoek. Metanorm: Learning to normalize few-shot batches across domains. In *International Conference on Learning Representations*, 2020. 2

[9] Antonio D'Innocente and Barbara Caputo. Domain generalization with domain-specific aggregation modules. In *Pattern Recognition: 40th German Conference, GCPR 2018, Stuttgart, Germany, October 9-12, 2018, Proceedings 40*, pages 187–198. Springer, 2019. 2

[10] Gamaleldin F Elsayed, Ian Goodfellow, and Jascha Sohl-Dickstein. Adversarial reprogramming of neural networks. *arXiv preprint arXiv:1806.11146*, 2018. 2, 4

[11] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338, 2010. 5

[12] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004. 5

[13] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020. 5

[14] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016. 2, 5

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[16] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 3

[17] Haoshuo Huang, Qixing Huang, and Philipp Krahenbuhl. Domain transfer through deep activation matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 1

[18] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Weiming Zhang, Feifei Wang, Gang Hua, and Nenghai Yu. Diversity-aware meta visual prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10878–10887, 2023. 2, 4, 6

[19] Samyak Jain, Sravanti Addepalli, Pawan Kumar Sahu, Priyam Dey, and R. Venkatesh Babu. Dart: Diversify-aggregate-repeat training improves generalization of neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16048–16059, 2023. 2, 5, 8

[20] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022. 2

[21] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 491–507. Springer, 2020. 1

[22] Aodi Li, Liansheng Zhuang, Shuo Fan, and Shafei Wang. Learning common and specific visual prompts for domain generalization. In *Proceedings of the Asian Conference on Computer Vision*, pages 4260–4275, 2022. 2

[23] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017. 1, 5, 6

[24] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C. Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2018. 2

[25] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639, 2018. 2

[26] Yizhe Li, Yu-Lin Tsai, Chia-Mu Yu, Pin-Yu Chen, and Xuebin Ren. Exploring the benefits of visual prompting in differential privacy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5158–5167, 2023. 2

[27] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *International conference on machine learning*, pages 2208–2217. PMLR, 2017. 1

[28] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 5

[29] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

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

[31] Saeid Motiian, Marco Piccirilli, Donald A. Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5715–5725, 2017. 2

[32] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International conference on machine learning*, pages 10–18. PMLR, 2013.

[33] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8690–8699, 2021. 2, 5

[34] Yingwei Pan, Ting Yao, Yehao Li, Yu Wang, Chong-Wah Ngo, and Tao Mei. Transferrable prototypical networks for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[35] Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. Efficient domain generalization via common-specific low-rank decomposition. In *International Conference on Machine Learning*, pages 7728–7738. PMLR, 2020. 1, 2

[36] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 12(1):145–151, 1999. 5

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 5

[38] Bryan C Russell, Antonio Torralba, Kevin P Murphy, and William T Freeman. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*, 77:157–173, 2008. 5

[39] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 7

[40] Rui Shao, Xiangyuan Lan, Jiawei Li, and Pong C. Yuen. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10023–10031, 2019. 2

[41] Kihyuk Sohn, Huiwen Chang, José Lezama, Luisa Polania, Han Zhang, Yuan Hao, Irfan Essa, and Lu Jiang. Visual prompt tuning for generative transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19840–19851, 2023. 1

[42] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1521–1528. IEEE, 2011. 1, 5

[43] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 8

[44] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999. 5

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

[46] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017. 1, 5, 6

[47] Bailin Wang, Mirella Lapata, and Ivan Titov. Meta-learning for domain generalization in semantic parsing. *arXiv preprint arXiv:2010.11988*, 2020. 1, 2

[48] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip Yu. Generalizing to unseen domains: A survey on domain generalization. *IEEE Transactions on Knowledge and Data Engineering*, 2022. 1

[49] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018. 1

[50] Shujun Wang, Lequan Yu, Kang Li, Xin Yang, Chi-Wing Fu, and Pheng-Ann Heng. Dofe: Domain-oriented feature embedding for generalizable fundus image segmentation on unseen datasets. *IEEE Transactions on Medical Imaging*, 39 (12):4237–4248, 2020. 2

[51] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010. 5

[52] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 1, 2

A2XP: Towards Private Domain Generalization

Supplementary Material

1. Implementation of Generalization

In this section, we present detailed implementation of the *Attention-based Generalization* module in a pseudo-code form from initialization to forwarding Algorithm 1.

Algorithm 1 Generalization Implementation

procedure INIT(self, $\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_i, \cdots, \mathbf{p}_N$)
  self.$\mathcal{E}_{\text{shared}} \leftarrow \text{resnet18\_1k}()$ ▷ Initialize embedders.
  self.$\mathcal{E}_{\text{T}}$, self.$\mathcal{E}_{\text{E}} \leftarrow \text{linear}(), \text{linear}()$
  self.$\mathbf{p}_i \leftarrow \mathbf{p}_i / \|\mathbf{p}_i\|_2 \quad \forall i \in [1, N]$ ▷ Normalize experts.
end procedure

procedure FORWARD(self, $\mathbf{x}_{N+1,j}$)
  $\mathbf{z}_{\mathbf{x}} \leftarrow \text{self.}\mathcal{E}_{\text{T}}(\text{self.}\mathcal{E}_{\text{shared}}(\mathbf{x}_{N+1,j}))$
  $\mathbf{z}_{\mathbf{p}_i} \leftarrow \text{self.}\mathcal{E}_{\text{E}}(\text{self.}\mathcal{E}_{\text{shared}}(\text{self.}\mathbf{p}_i)) \quad \forall i \in [1, N]$
  $\lambda_i \leftarrow \mathbf{z}_{\mathbf{x}} \mathbf{z}_{\mathbf{p}_i}^{\top} \quad \forall i \in [1, N]$ ▷ Calculate attention scores.
  $\mathbf{p}_{N+1,j} \leftarrow \sum_{i=1}^{N} \lambda_i \text{self.}\mathbf{p}_i$
  return $\mathbf{x}_{N+1,j} + \mathbf{p}_{N+1,j}$
end procedure

2. Further Analysis on the Experts

In this section, we conduct further analysis on the expert prompts of A2XP. We analyzed the prompts by changing various components: the size of the experts, the number of experts, the type of prompts, the way to mix the experts.

2.1. Size of the Experts

We analyzed the prompt size in the performance and the memory requirement perspectives (see Figure 1). We empirically found that 30 is the best prompt size among the five sizes and applied it to our method.

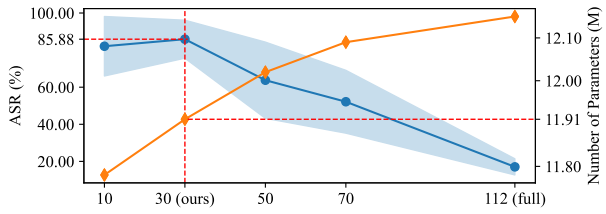


Figure 1. Expert adaptation performance of A2XP with Gaussian initialization. The blue transparent range shows $\mu \pm \sigma$ of ASR.

2.2. Ablation Study on Various Experts

We compared domain generalization performance among various experts.

2.2.1 Various Prompts

Generalization *without* prompts, which is equivalent to linear probing, loses the benefits of linear combination; therefore, it has a lower generalization performance (see Table 1). Utilizing *random* prompts show performance improvement, indicating that prompts can contribute to better generalization performance. Furthermore, we show the effectiveness of our method that enhances generalization performance more efficiently by leveraging *experts* trained from each domain.

	Picture	Art	Cartoon	Sketch	Avg.
Without	86.95	83.11	94.04	86.79	87.72 (-7.35)
Random	98.98	93.85	90.19	88.09	92.78 (-2.29)
Experts	**99.07**	**95.27**	**98.07**	**87.85**	**95.07** (-0.00)

Table 1. Comparison by various prompts.

2.2.2 Number of Experts

As shown in Table 2, using either no experts or just a single expert offers limited generalization potential. In contrast, employing multiple experts, particularly in numbers matching the domain count, broadens the scope to identify the optimal direction for generalization.

# experts	Picture	Art	Cartoon	Sketch	Avg.
0	86.95	83.11	94.04	86.79	87.72 (-7.35)
1	83.75	95.69	86.82	86.49	88.19 (-6.88)
2	97.35	**99.34**	93.41	87.27	94.34 (-0.73)
3 (all)	**99.07**	95.27	**98.07**	**87.85**	**95.07** (-0.00)

Table 2. Comparison by # experts.

2.3. Various Ways to Mix Experts

In Table 3, we compared various methods to mix pre-trained experts for unseen datasets. We demonstrate that our *attention*-based approach outperforms methods that mix experts in *constant* or *random* weights.

	Picture	Art	Cartoon	Sketch	Avg.
Constant	97.82	99.40	94.24	86.99	94.61 (-0.46)
Random	97.65	99.16	93.85	87.05	94.43 (-0.64)
Attention	**99.07**	**95.27**	**98.07**	**87.85**	**95.07** (-0.00)

Table 3. Various ways to mix experts.

3. Further Analysis on the Framework

We analyzed more about the A2XP framework itself in the perspective of how evenly the experts are mixed, how does the objective network architecture affects, and the scalability of A2XP.

3.1. Attention Distribution

When A2XP is applied on the source domain, we expected the attention weights of A2XP emphasize the experts of the source domain. This study analyzes how A2XP attends to different experts depending on the domain of the input images. The violin plots in Figure 2 show the distribution of normalized attention weights in PACS [7] dataset. Each cell shows the distribution of attention weights on each domain. Across all combinations of target and source domains, a significant standard deviation was observed, indicating a wide range of variation in the attention weights. This suggests that the attention weights have a very large range.

	P	A	C	S
P	1.729E-1	**1.330E-2**	3.424E-1	**2.377E-4**
A	4.966E-1	5.752E-2	**4.210E-2**	5.739E-2
C	**2.127E-2**	**1.641E-3**	1.759E-1	**1.797E-2**
S	2.556E-1	2.526E-1	5.566E-1	**2.460E-9**

Table 4. p-values of RM-ANOVA [9] with the normalized attention weights on PACS [7] dataset. Bold styled cells are significant with $p \leq 0.05$.

To be analytic, we performed Repeated Measures-ANalysis Of VAriance (RM-ANOVA) [9] on the normalized attention weights, and the result is in Table 4. Each cell contains the p-value of a combination of the target domain and tested domain. For example, p-value of weights when trained on 'P' and tested on 'A' is 1.330E-2. In this case, the experts are from the 'A,' 'C,' and 'S' domains. The smaller a p-value is, the more the combination showed a significant correlation among weights for experts. The p-values are significant with $p \leq 0.05$ in some cases but not dominant. As a result, A2XP mixes the experts differently depending to the input images, and the mixing ratios are not always similar even if the target and testing domain is the same.

3.2. Various Objective Networks

We are concerned only about CLIP [8]-pretrained Vision Transform (ViT) [3] for the objective network in the main paper. We present another result on a convolutional neural network ResNet50 [4] and ImageNet [2] supervised pretraining to reveal another characteristic of A2XP. The leave-one-domain-out evaluation result is compared in Table 5. The number of updates was limited to 3K for ImageNet and 1K for CLIP pretrained models in the adaptation step. And we initialized the experts by zero before adaptation.

We observed that the experts must be well adapted for all domain from ResNet50 with both ImageNet and CLIP pretraining. Moreover, even if the adaptation was successful, the model itself have to be generalized at the pretext task. Both the average accuracy of the both ResNet50 was lower compared to other existing methods [1, 5]. As a result, A2XP is sensitive to the adaptation method, the objective network architecture, and the pretext task.

		Expert Adaptation				
Architecture	Pretraining	P	A	C	S	Avg.
ResNet50 [4]	ImageNet [2]	92.40	72.36	85.24	66.28	79.07
ResNet50 [4]	CLIP [8]	67.25	52.83	59.98	56.73	59.20
ViT-base [3]	ImageNet [2]	96.95	79.30	92.41	87.94	89.15
ViT-base [3]	CLIP [8]	97.54	73.88	95.52	94.55	90.37
		Attention-based Generalization				
Architecture	Pretraining	P	A	C	S	Avg.
ResNet50 [4]	ImageNet [2]	51.56	49.12	46.25	36.12	45.76
ResNet50 [4]	CLIP [8]	74.31	44.38	42.62	16.34	44.41
ViT-base [3]	ImageNet [2]	81.02	69.53	49.23	31.38	57.79
ViT-base [3]	CLIP [8]	99.07	95.07	98.12	88.22	95.12

Table 5. The result of leave-one-domain-out evaluation using ViT [3] and ResNet50 [4].

3.3. Scalability

We applied our method to larger datasets: Office-Home (Table 6) and DomainNet (Table 7). The results show that A2XP outperforms current methods, validating its applicability across datasets of varying sizes.

	Art	Clipart	Product	Real	Avg.
ERM	48.04	42.27	48.25	47.63	46.55
MIRO	56.49	58.56	43.30	54.43	53.20
A2XP	**77.42**	**65.73**	**81.93**	**83.15**	**77.06**

Table 6. Office-Home evaluation.

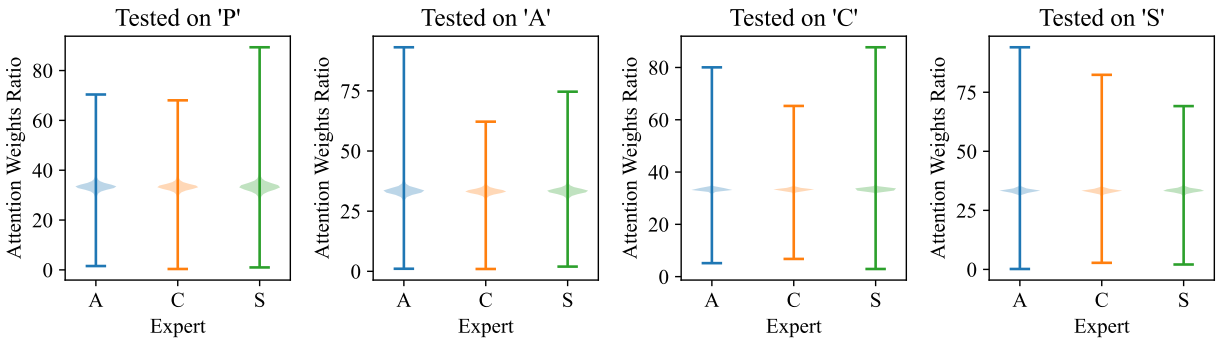
	Clip	Info	Paint	Quick	Real	Sketch	Avg.
ERM	0.32	0.35	0.45	0.39	0.41	0.57	0.41
MIRO	39.31	39.48	40.10	**39.77**	40.59	42.18	40.24
A2XP	**62.88**	**43.58**	**58.99**	13.72	**55.96**	**58.45**	**48.93**

Table 7. DomainNet evaluation.

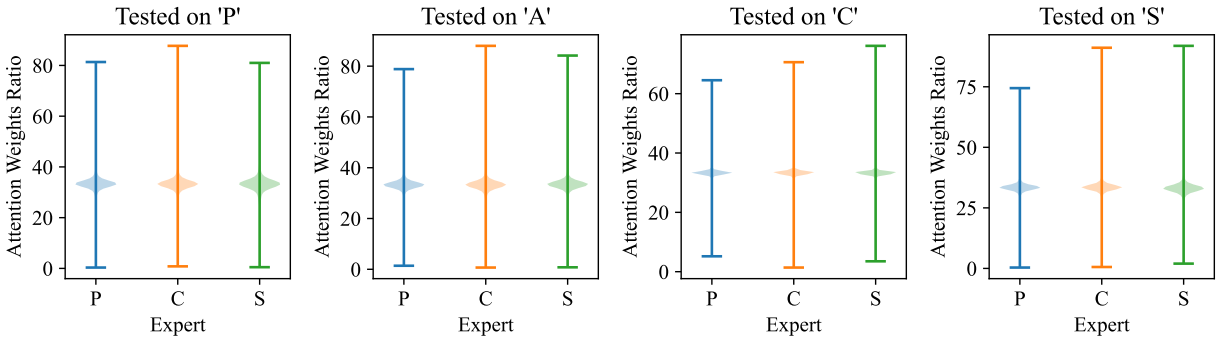
We further investigated the scalability of A2XP for datasets of various sizes by measuring the number of parameters, memory requirements, computational resources (GFLOPs), and the training time (see Table 8). The number of parameters and the memory requirement of A2XP only depends on the number of experts. Training time primarily depends on the number of training samples. From this perspective, we show the practical applicability of A2XP for larger datasets.

Dataset	# classes	# domains	# samples	# params	Mem. load	GFLOPs	Time (s)	Avg. Acc.
PACS	7	4	9,991	11.91M	17817MiB	1.814	2.12	95.07
VLCS	5	4	10,729	11.91M	17777MiB	1.814	2.51	83.15
Office-Home	65	4	15,588	11.91M	17779MiB	1.814	2.87	77.06
DomainNet	345	6	586,575	12.05M	18185MiB	2.539	136.46	48.93

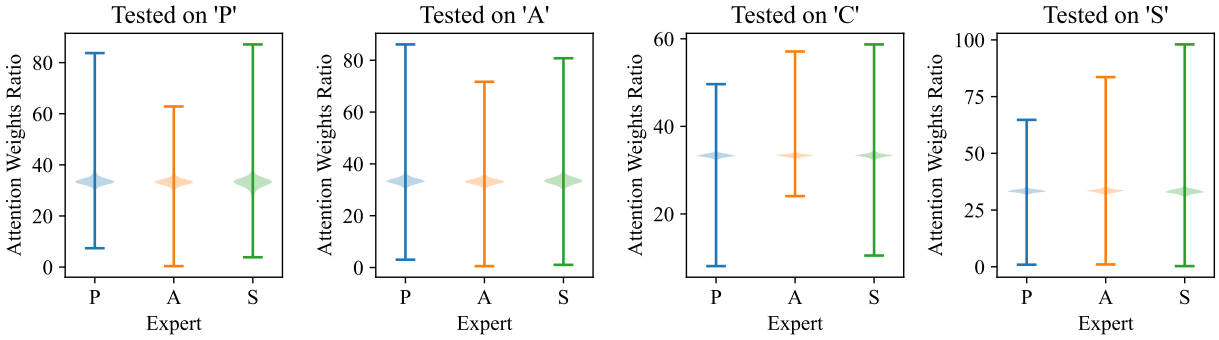
Table 8. Scalability analysis.



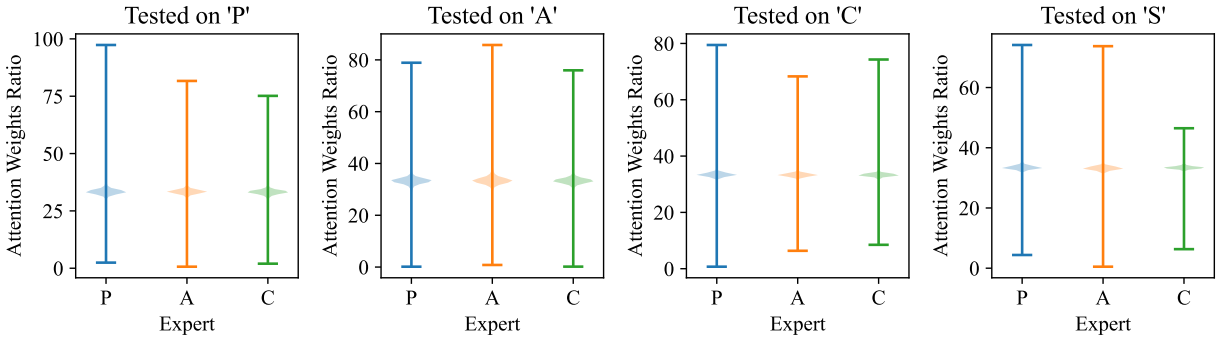
(a) Trained on 'P'



(b) Trained on 'A'



(c) Trained on 'C'



(d) Trained on 'S'

Figure 2. Visualization of normalized attention weights of correctly classified samples from A2XP on PACS [7] dataset.

4. Discussion

4.1. Failure Cases

We found that A2XP struggles to generalize for specific domains such as *Quick* in the DomainNet dataset and *Sketch* domain in the PACS dataset; both have more significant domain shifts from another dataset. Despite limitations in generalizing distinct domains, the performance of our approach still achieved state-of-the-art average accuracy.

4.2. Interpretability

As noted in [6], a visual prompt facilitates domain adaptation by aligning features between the source and target domains. This can be interpreted as our experts are responsible for shifting features towards the target domain's manifolds. In our privacy setting, the alignment target is the manifold characterized by features from the data used to pre-train the model. Consequently, generating an expert for an unseen domain by mixing the experts from other domains can be considered crafting a mapping function to the pre-trained manifold, which we interpret as contributing to enhancing decision-making when using a pre-trained model while keeping it private.

References

[1] Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. In *European Conference on Computer Vision*, pages 440–457. Springer, 2022. 2

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 2

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[5] Samyak Jain, Sravanti Addepalli, Pawan Kumar Sahu, Priyam Dey, and R. Venkatesh Babu. Dart: Diversify-aggregate-repeat training improves generalization of neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16048–16059, 2023. 2

[6] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022. 4

[7] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017. 2, 3

[8] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[9] Edward H Simpson. The interpretation of interaction in contingency tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 13(2):238–241, 1951. 2