

Domain Shift Problems under Resource Constraints
リソース制約下におけるドメインシフト問題の研究

by

Yu Mitsuzumi

三鼓 悠

A Doctor Dissertation

博士論文

Submitted to

the Graduate School of Kyoto University

on March 3, 2026

in Partial Fulfillment of the Requirements

for the Degree of Doctor of Informatics

Supervisor: Professor Hisashi Kashima 鹿島 久嗣

ABSTRACT

The domain shift problem, where a model’s performance degrades due to the difference in distribution between the training data (source domain) and the deployment data (target domain), is an ever-present issue in data-driven model development. Domain adaptation and generalization, which effectively transfer knowledge gained in the source domains to the target domains, are promising solutions to this problem and have long been subjects of research. With the recent rapid advancements in deep learning technology, and given its inherently data-driven nature, the importance of those technologies has also grown. On the other hand, advances in deep learning and changes in societal norms have raised problems that cannot be addressed within the conventional framework of domain adaptation and generalization. In particular, the usage of data and model resources is becoming increasingly restricted, such as difficulties in data collection due to personal information protection and the intellectual property perspective, as well as model access limitations due to the concealed nature of cutting-edge deep learning models. In this doctoral dissertation, we addressed domain shift problems under resource constraints from three perspectives, contributing to the expansion of the applicability of domain adaptation and generalization technologies.

First, to overcome the difficulty of accessing source domain data, we investigated the problem of source-free domain adaptation, adapting a source-pre-trained model to the target domain only with the unlabeled target data. For the theoretical perspective of source-free domain adaptation techniques, which had remained unexplored in existing research, we provided a rational understanding by introducing the theory of self-training based on the expansion assumption. Besides, on the basis of these theoretical insights, we developed two new techniques to improve source-free domain adaptation methods, achieving consistent performance improvements.

Second, to overcome the difficulty of the target data collection compounded by the source data inaccessibility, we investigated the problem of source-free domain generalization, building a model generalizable to unseen target domains without labeled source data. Previous methods lacked flexibility for inputs from diverse domains because they used a common prediction setting for any domain input. To address this issue, we proposed a domain-flexible prediction method to adapt to the diversity of the target domain by constructing a variety of domain-specific configurations and a selector to choose the appropriate configuration for the target domain without using data from either source or target domains.

Third, to overcome access constraints on model resources (such as model parameters, training data, and the use of model outputs for training) like those in concealed models, we established a new problem: model-free domain adaptation, adapting the target concealed model only with the unlabeled target data. To solve this highly challenging problem, we proposed a method that combines transferable visual prompts with insights gained from our investigation on source-free domain adaptation. Specifically, to overcome the constraints on accessing the target model resources, we construct a cross-model transferable prompt between two surrogate models so that the trained prompt becomes effective even on the unseen target model. Furthermore, by incorporating insights from

our research on source-free domain adaptation into the visual prompt training scheme, we enable the visual prompt to acquire domain adaptability without using the model data resources. Besides, we proposed two novel techniques to boost model transferability and domain adaptability of the prompts.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Hisashi Kashima at Kyoto University, for his continuous guidance and helpful advice. His support enabled me to produce valuable research outcomes and make this doctoral dissertation substantial. I would like to express my gratitude to the dissertation committee members, Prof. Akihiro Yamamoto and Prof. Ko Nishino. Their insightful feedback helped improve my doctoral dissertation.

I would also like to express my appreciation to Dr. Akisato Kimura and Dr. Yasuhiro Fujiwara, who served as my supervisors at NTT and also as co-researchers, for their assistance in all aspects. I would like to thank Assoc. Prof. Go Irie at Tokyo University of Science, who initially guided me through research at NTT. I would also like to thank Prof. Atsushi Nakazawa at Okayama University, who was my master's advisor and introduced me to my doctoral advisor. I would like to thank all other researchers who collaborated and discussed with me.

Finally, I would like to thank my family for their acceptance, continuous support, and encouragement.

Contents

1	Introduction	1
1.1	Background	1
1.2	Contributions of This Dissertation	5
2	Understanding and Improving Source-free Domain Adaptation from a Theoretical perspective	9
2.1	Introduction	9
2.2	Related Works	13
2.2.1	Unsupervised Domain Adaptation (UDA)	13
2.2.2	Source-free Domain Adaptation (SFDA)	14
2.3	Understanding SFDA from a Theoretical Perspective	15
2.3.1	Problem Setup	15
2.3.2	Common Feature of SFDA Methods	15
2.3.3	Theoretical Understanding of SFDA	17
2.4	Our Method: Improved SFDA	21
2.4.1	Model Training with Auto-Adjusting Diversity Constraint	21
2.4.2	Augmentation Training with Teacher-Student Framework	24
2.4.3	Training Procedure of Our Method	26
2.5	Experiments	26
2.5.1	Setups	26
2.5.2	Main results	29
2.5.3	Analysis	31
2.6	Conclusion and Limitations	35
2.7	Appendix	35
2.7.1	Formal Definitions and Assumptions	35
2.7.2	Details of Synthetic Data Experiment	37
2.7.3	Additional Results	41
3	Flexible Source-free Domain Generalization via Domain Prompt-Discriminator Collaborative Learning	44
3.1	Introduction	44

3.2	Related Works	48
3.2.1	Domain Generalization (DG)	48
3.2.2	Source-free Domain Generalization (SFDG)	48
3.2.3	Vision-Language Pretrained Models (VLMs)	49
3.2.4	Test-time Adaptation (TTA)	50
3.3	Preliminaries	50
3.3.1	Problem Setup	50
3.3.2	Contrastive language-image pre-training (CLIP)	51
3.4	Our Method: Development Phase: Domain Prompt-Discriminator Collaborative Learning	52
3.4.1	Development Phase: Training Domain-specific Prompts and Domain Discriminator	53
3.4.2	Deployment Phase: Inference and Test-time Adaptation (TTA)	55
3.5	Experiments	57
3.5.1	Setups	57
3.5.2	Main Results	59
3.5.3	Analysis	61
3.6	Conclusion and Limitations	67
3.7	Appendix	68
3.7.1	Detailed Results	68
3.7.2	Hyper-paramter Settings	71
4	Model-free Domain Adaptation for Concealed Multimodal Large-Language Models	73
4.1	Introduction	73
4.2	Related Works	77
4.2.1	Multimodal Large-language Models	77
4.2.2	Domain Adaptation	78
4.2.3	Contribution over Prior Works	79
4.3	Preliminaries	80
4.3.1	Problem Setup	80
4.3.2	Multimodal Large-language Models	80
4.3.3	Visual Prompting	81
4.4	Our Method: Model-Transferable Domain-Adaptable Visual Prompting	82
4.4.1	Cross-Model Visual Prompting with Domain-Adaptation Objective	84
4.4.2	Cross-Model Pseudo Labeling	85
4.4.3	Cross-Model Gradient Alignment	87
4.4.4	Prompt Optimization	88
4.5	Experiments	89

4.5.1	Setups	89
4.5.2	Main Results	90
4.5.3	Analysis	93
4.6	Conclusion and Limitations	97
4.7	Appendix	98
4.7.1	Detailed Descriptions of Proposed Method	98
4.7.2	Detailed Experimental Results	101
4.7.3	Extension of MTDA-VP	104
4.7.4	Broader Impacts	105
5	Conclusion and Future Directions	106
5.1	Conclusion	106
5.2	Future Directions	108
6	List of Publications	110

List of Figures

1.1	Basic Framework and Emerging Difficulties. Conventionally, in domain adaptation, the models are trained on both labeled source domain data and unlabeled target domain data to enhance the recognition performance of the target data. In domain generalization, the models are given multiple source domains so as to be generalizable to various target domains. However, as deep learning technologies advance and societal norms change, several difficulties that cannot be handled in the conventional framework have emerged, such as inaccessibility of source domain data, difficulty in target data collection, and model access limitation. In this dissertation, we investigate domain adaptation and generalization technologies to overcome these emerging difficulties.	3
2.1	Overview of this work. Unsupervised domain adaptation has a theoretical background that has yielded a variety of methods. By contrast, the theoretical perspectives on source-free domain adaptation (SFDA) have not been well explored. Our research motivations are (1) to shed light on the theoretical perspectives of existing SFDA methods, and (2) to propose an improved method based on the theoretical insights.	10
2.2	Model accuracy ($\text{Err}_U(\hat{F})$) and the prediction inconsistency of the ground-truth model (μ) against strength of augmentation (d). . . .	19
2.3	Overview of our method. In model training with auto-adjusting diversity constraint, we train the model by minimizing the discriminability loss $\mathcal{L}_{\text{dis}}$ and the diversity loss $\mathcal{L}_{\text{div}}$ while automatically adjusting the trade-off parameter between discriminability and diversity. In augmentation training with teacher-student framework, we train a learnable data augmentation $\mathcal{A}$ to generate harder samples for the current model F while suppressing the prediction inconsistency of the teacher model F'	22
2.4	Ablation Study. (a) Auto-Adjusting Diversity Constraint (b) Augmentation Training.	31

2.5	Analysis of the coefficient parameter $\lambda_{\text{div}}^{\text{max}}$	33
2.6	Analysis of the interval parameter $\hat{e}$	34
2.7	Analysis of the coefficient parameter $\lambda_{\mathcal{A}}$	34
2.8	Visual Understanding of Expansion. (a) Definition of (a, d) -expansion. (b) Example that violates the expansion assumption.	36
2.9	Visualization of the experimental results on SHOT-IM.	38
2.10	Visualization of the experimental results on AaD.	39
2.11	Experimental Results on Synthetic Data with AaD. (a) Model accuracy w/ and w/o discriminability and diversity losses. (b) Model accuracy and prediction inconsistency of ground-truth model against the strength of augmentation.	40
3.1	Comparison between prior source-free domain generalization (SFDG) methods and our method. (a, b) Prior SFDG methods employ a domain-shared configuration (e.g., representation/classifier), which undermines flexibility for the input image domain. (c) Our method achieves flexible predictions by a set of domain-specific prompts to yield various domain-specific predictions and a domain discriminator to appropriately aggregate the predictions for the input image.	46
3.2	Development Phase: Training Domain-specific Prompts and Domain Discriminator. We jointly train domain-specific prompts and a domain discriminator through a simple domain classification task, which encourages the domain-specific prompts to represent various domains and equips the domain discriminator with the ability to recognize the domain of the input feature. We also employ a knowledge guidance loss to ensure CLIP-based classification ability.	52
3.3	Deployment Phase: Inference and Test-time Adaptation. We obtain domain-specific predictions from each learned prompt, and then we aggregate them by weighted averaging with the domain estimation result of the domain discriminator. The recognition accuracies can be enhanced by updating only the domain discriminator’s parameters based on test-time adaptation loss.	55
3.4	Analysis of loss scaling parameter λ_{domain}.	61
3.5	t-SNE plots of text features of domain-specific prompts. The shapes of the markers represent classes, while the colors of them represent domains.	62
3.6	Analysis of the representation capacity of domain-specific prompts.	64
3.7	Analysis of test-time adaptation steps.	66

4.1	(a) MLLMs perform well on familiar domains (e.g., Real), but their performance degrades on unfamiliar domains (e.g., Infographic, Quickdraw). (b) MFDA prohibits the use of any concealed model resources (i.e., source data, model weights, and model outputs) during training but allows access only for testing. To solve MFDA, our method builds visual prompts that encourage the domain adaptation of the concealed MLLMs by cross-model visual prompting with the domain adaptation training objective.	74
4.2	Overview of MTDA-VP. (a) MTDA-VP executes cross-model visual prompt training with surrogate MLLM and CLIP based on the domain adaptation objective to equip the prompts with model transferability and domain adaptability. We introduce (b) CMPL and (c) CMGA to boost domain adaptability and model transferability, respectively. (d) In testing, we just attach the trained prompt to the input image to adapt the concealed MLLM to the target domain.	83
4.3	Data-scale analysis. MFDA-VP constantly showed improved performance from No Adapt. even with small data size.	95
4.4	Analysis on the balance of cross-model training. MFDA-VP showed slight performance degradation when λ_{CLIP} increased while constantly showing improvements from No Adapt.	95
4.5	Analysis on Cross-model Gradient Alignment. CMGA worked well when α is set to small values.	96
4.6	Analysis on Domain Adaptation Training. Our method slightly dropped its performance when λ_{DA} is small, while Ours w/o CMPL largely dropped.	96
4.7	$\mathcal{L}_{\text{FCA}}$ analysis.	102
4.8	Qualitative analysis of prompts.	102

List of Tables

1.1	Summary of Problem Setups.	6
2.1	Model accuracy w/ and w/o discriminability and diversity losses. .	19
2.2	Classification Accuracy (%) on Office-31 (ResNet-50). The best and second best are highlighted in bold and with <u>underline</u> . .	28
2.3	Classification Accuracy (%) on Office-Home (ResNet-50). The best and second best are highlighted in bold and with <u>underline</u> . .	29
2.4	Classwise Accuracy (%) on VisDA2017 (ResNet-101). The best and second best are highlighted in bold and with <u>underline</u> . .	30
2.5	Analysis of Application to SHOT-IM.	32
2.6	Analysis of Augmentation Training (Detailed version of Fig. 2.4b).	41
2.7	Analysis of the coefficient parameter λ_F	42
3.1	Main Results. Our method is compared with state-of-the-art SFDG methods on four domain generalization benchmarks and evaluated on classification accuracy [%].	60
3.2	Analysis of the position of the domain vectors	62
3.3	Analysis of prediction aggregation method.	63
3.4	Analysis of the prefix word of the reference text.	65
3.5	Per domain classification accuracy on PACS.	68
3.6	Per domain classification accuracy on VLCS.	69
3.7	Per domain classification accuracy on Office-Home.	70
3.8	Per domain classification accuracy on DomainNet.	71
4.1	Comparison of Domain Adaptation Setups. MFDA imposes the most challenging constraints compared with other domain adaptation setups.	76
4.2	Classification accuracy [%] on DomainNet. We highlight improved and degraded cases (best viewed in color). Unconcealed cases (i.e., the surrogate model and the concealed model are the same) are excluded from average calculations.	91

4.3	Classification Accuracy [%] on PACS. We highlight improved and degraded cases (best viewed in color). Unconcealed cases (i.e., the surrogate model and the concealed model are the same) are excluded from average calculations.	92
4.4	Comparison and Ablation Study. We note the performance gain [%] from No Adapt. We highlighted in colors the cases where performance is improved and is declined . The best scores are highlighted in bold	93
4.5	Detailed Comparison Results. We highlighted in colors the cases where performance is improved and is declined . The best scores are highlighted in bold	101
4.6	Analysis on the large model gap. We highlighted in colors the cases where performance is improved and is declined	104

Chapter 1

Introduction

1.1 Background

Deep neural networks (DNNs) [90, 55] have demonstrated exceptional performance on a wide range of tasks. They first garnered significant attention from a multitude of researchers following the astounding and historic victory of AlexNet in the 2012 ImageNet competition [87]. Around the same period, the true potential of deep neural networks was unleashed, thanks to the confluence of massive datasets accumulated through global internet communication, dramatic improvements in the computational power of GPUs, and the development of effective training algorithms. The benefits have extended far beyond computer vision [65, 76, 56, 141, 64, 40, 120, 142] to diverse fields such as speech and acoustics [67, 173, 149, 61, 54], natural language processing [119, 160, 175, 131, 37, 16], tabular data [5, 133, 59], and time-series analysis [68, 7, 206, 185, 194, 125]. Today, DNN technology has become indispensable, permeating our daily lives in various forms, including machine translation [150, 57], generative AI [140, 164, 128, 58], and autonomous driving technologies [183]. However, behind this remarkable

progress, we must confront an essentially unavoidable challenge known as domain shift.

Domain shift problem refers to the degradation of a model’s performance that occurs when there is a distributional discrepancy between the training data and the input data in practice [12]. This is a fundamental issue inherent to all training-based models. In real-world applications, this problem manifests in various ways: Autonomous driving AI trained on data from simulators or video games may struggle to recognize traffic signs and pedestrians correctly in the real world because of the visual gap between computer graphics and real images. In smart health applications, significant deviations in biometric data, caused by the individual user characteristics or device differences, can lead to providing inappropriate recommendations to the users, thereby hindering the user experiences. In the auto manufacturing scenes, the precision of anomaly detection systems may decline, as the product properties can vary between factories due to differences in the machines or operation protocols. As these examples show, some form of discrepancy always arises between training data and deployment data. This discrepancy is the root cause of domain shift, and since it is impossible to encompass all the data in the world, this problem is unavoidable.

To address the persistent issue of domain shift, the field of domain adaptation has attracted much attention as one of the promising solutions for this problem. The primary objective of domain adaptation is to effectively transfer knowledge acquired in a label-abundant source domain to a target domain where labeled data is either scarce or entirely absent. A basic domain adaptation framework, as depicted in the center of Fig. 1.1, concurrently executes supervised training on the source data and unsupervised additional training with labeled source data and

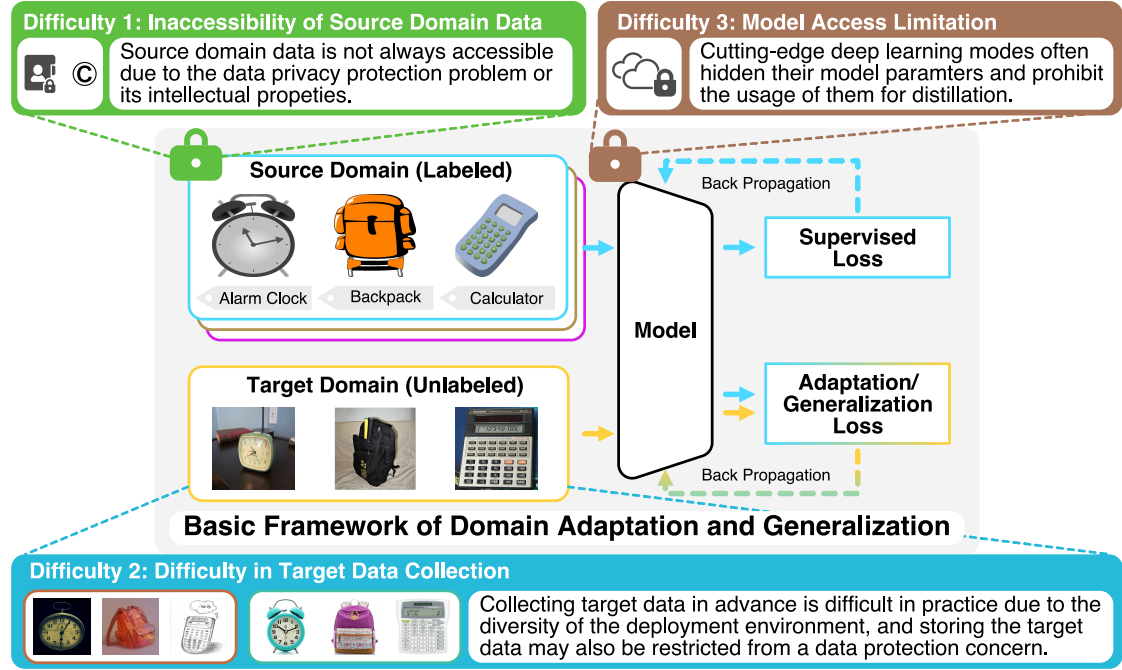


Figure 1.1: **Basic Framework and Emerging Difficulties.** Conventionally, in domain adaptation, the models are trained on both labeled source domain data and unlabeled target domain data to enhance the recognition performance of the target data. In domain generalization, the models are given multiple source domains so as to be generalizable to various target domains. However, as deep learning technologies advance and societal norms change, several difficulties that cannot be handled in the conventional framework have emerged, such as inaccessibility of source domain data, difficulty in target data collection, and model access limitation. In this dissertation, we investigate domain adaptation and generalization technologies to overcome these emerging difficulties.

unlabeled target data to reduce the distribution discrepancy between the source and target domains. This methodology obviates the need for the laborious process of large-scale data acquisition and model retraining for the target domain. Another promising approach is domain generalization. In contrast to domain adaptation, which aims at adapting the model to certain target domains, domain generalization aims to build a model that generalizes to unseen target domains. In domain generalization, instead of using the unlabeled target domains, we train the model

with multiple labeled source domains to obtain generalizable knowledge toward unseen domains. Domain adaptation and generalization technologies demonstrate their effectiveness in various areas, such as computer vision [111, 146, 70, 123], audio and speech processing [36, 50, 52, 151], natural language processing [42, 27, 110, 9], and robotics [15, 167].

On the other hand, the advances in neural network technologies and changing societal norms are demanding that domain adaptation and generalization technologies also adapt to these surrounding trends. As the following examples will show, we are now facing several difficulties that are beyond the scope of the traditional frameworks, which is illustrated in Fig. 1.1.

- **Inaccessibility of source domain data.** The protection of data privacy, exemplified by the EU’s GDPR (General Data Protection Regulation) [45], has now become a global trend. Furthermore, amidst the fierce competition in deep learning technology development, training data is considered as important intellectual property, making it extremely difficult to share readily. This trend restricts access to source domain data, thereby hindering conventional domain adaptation training that depends on both source and target domain data.
- **Difficulty in target data collection.** It is not only the source data that is difficult to access. It is also hard to know with certainty what environment a model will be applied to in advance, which clearly shows the difficulty in collecting target data. Moreover, storing target data on the developer’s side for training purposes can also be constrained by the aforementioned data protection concerns. Furthermore, this problem can occur concurrently with

the issue of inaccessible source data. This leads to a constrained problem where neither source nor target domain data is available — a challenge that falls outside the scope of conventional domain adaptation and generalization paradigms.

- **Model access limitation.** Recently, large-language models (LLMs) have demonstrated remarkable capability for various language tasks, and multi-modal LLMs (MLLMs) have been developed to extend their ability to multi-modal tasks. These cutting-edge deep learning technologies attracted much attention and have been deployed across various services [2, 164, 4]. Whereas this new trend in deep learning technology has given rise to another problem: that is, high-performance foundation models, developed by a few giant tech companies, are provided through APIs, namely, general users cannot directly access model resources such as the internal architecture, parameters, or training data. Besides, using the model outputs for training user-owned models will cause other issues, such as high-cost service charges or license violations for the use of the outputs for model training (e.g., model distillation). The conventional framework requires access to both training data and model parameters, indicating that it is hardly applicable to such “concealed” deep learning models.

1.2 Contributions of This Dissertation

In this doctoral dissertation, we investigate domain adaptation and generalization technologies tailored to the significant changes in deep learning trends. Our work centers on overcoming the emerging difficulties caused by the resource constraints.

Table 1.1: **Summary of Problem Setups.**

Problem Setup	Source Data Accessibility	Target Data Accessibility	Model Resources Accessibility
Domain Adaptation	✓	✓	✓
Domain Generalization	Multiple	✗	✓
Source-free Domain Adaptation	✗	✓	✓
Source-free Domain Generalization	✗	✗	✓
Model-free Domain Adaptation	✗	✓	✗

We summarized the problem setups of each chapter in Tab. 1.1. Note that in our work, we chose a simple image classification task as the benchmark to focus on overcoming resource constraints.

- Chapter 2: Understanding and Improving Source-free Domain Adaptation from a Theoretical Perspective.** To overcome the difficulties in the source data inaccessibility, we tackled the source-free domain adaptation (SFDA) problem, where we adapt the source-trained model only with unsupervised target data. Specifically, we shed light on the theoretical perspective of existing SFDA methods based on a finding that SFDA loss functions comprising discriminability and diversity losses work in the same way as the training objective in the theory of self-training based on the expansion [184], which shows the existence of the target error bound. Additionally, with two novel insights obtained from our theoretical observation, we built an improved SFDA method comprising 1) model training with auto-adjusting diversity constraint and 2) augmentation training with teacher-student framework, yielding a better recognition performance.

- Chapter 3: Flexible Source-free Domain Generalization via Domain Prompt-Discriminator Collaborative Learning.** To overcome the difficulty of the target data collection compounded by the source data inaccessibility, we tackled the source-free domain generalization (SFDG) problem, where we aim to build a model generalizable to unseen target domains without using any actual source data. Previous works tackled this challenging problem by utilizing the generalizability of the large-scale pre-trained vision-language models (VLMs). However, due to their methodological design of employing a domain-shared configuration (e.g., classifier, representation) for data inputs, they hinder the prediction flexibility to the unforeseen diverse target domain data. In contrast, our newly proposed method, domain prompt-discriminator collaborative learning, endows the model with domain flexibility by jointly training the following two modules: 1) domain-specific prompts, which are a set of text inputs destined to have suitable counterparts for the image inputs from various domains in VLM based prediction, and 2) a domain discriminator, which chooses favorable prompts for the prediction of query image. Notably, this training can be carried out in the form of simple domain classification learning. Our method design also opens the door to a simple yet effective test-time adaptation technique that further boosts recognition accuracy.
- Chapter 4: Model-free Domain Adaptation for Concealed Multimodal Large-Language Models.** To overcome the access limitation of the model resources (i.e., training data, model parameters, and model outputs) of cutting-edge MLLMs, we introduced a novel domain adaptation

setup, model-free domain adaptation (MFDA), to investigate whether we can address domain adaptation problems without using any resources of the concealed target models. As a proof of concept for MFDA, we built a method named model-transferable domain-adaptable visual prompting (MTDA-VP). In the training, this method executes cross-model visual prompting on surrogate models with a domain adaptation objective so that the visual prompts simultaneously acquire model transferability and domain adaptability. In the testing, we can adapt the concealed MLLMs to the target domain by just inputting the test images with the trained prompt into the models. Besides, we developed two techniques, cross-model pseudo labeling (CMPL) and cross-model gradient alignment (CMGA), to further enhance model transferability and domain adaptability of the visual prompts.

Chapter 2

Understanding and Improving Source-free Domain Adaptation from a Theoretical perspective

2.1 Introduction

Deep learning has long suffered from the domain-shift problem where models perform well on domains seen in the training phase but struggle with unseen domains. Unsupervised domain adaptation (UDA) is a promising solution: it transfers knowledge learned from a labeled source domain to an unlabeled target domain. UDA methods show their effectiveness in various fields. For example, in computer vision, domain adaptation methods for image classification [111, 198, 108], object detection [26, 146, 214], and segmentation [70, 69, 189] tasks are developed, and they show promising achievements. However, as shown in the upper left of Fig. 2.1, they typically require both source and target domain data, which limits their applicability as this requirement poses privacy concerns about the source data and en-

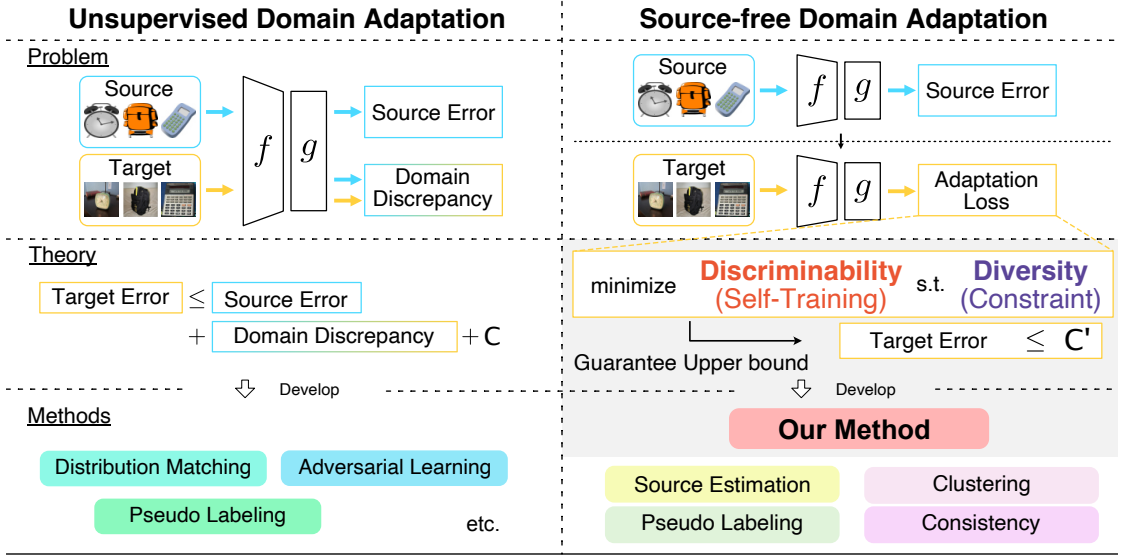


Figure 2.1: **Overview of this work.** Unsupervised domain adaptation has a theoretical background that has yielded a variety of methods. By contrast, the theoretical perspectives on source-free domain adaptation (SFDA) have not been well explored. Our research motivations are (1) to shed light on the theoretical perspectives of existing SFDA methods, and (2) to propose an improved method based on the theoretical insights.

tails computational inefficiency. Recently, researchers have shifted focus to another direction of UDA called source-free domain adaptation (SFDA). SFDA bypasses the above issues by not using raw data from the source data. Instead, as illustrated in the upper right of Fig. 2.1, SFDA performs the training with a source-pre-trained model and unlabeled target data. Various SFDA methods have been proposed, including source estimation-based methods [71, 166, 136, 43, 187], pseudo-labeling-based methods [191, 137, 83], clustering-based methods [102, 99, 91], consistency-learning methods [192, 23, 190, 203], and even without source data, they outperform UDA methods.

Despite these promising achievements, a theoretical understanding of SFDA methods is still absent. UDA studies rely on the theoretical notion that the target

error can be upper-bounded by the source error and the discrepancy between the two domains [117, 11, 202, 204], which is illustrated in the middle left of Fig. 2.1, and develop various approaches such as distribution matching [111, 171, 113], adversarial learning [170, 48, 112, 198], and pseudo-labeling [216, 215, 88, 108]. By contrast, the theoretical analyses of SFDA are either absent or not general enough and cannot form the basis for the development of new methods. Moreover, the theory of UDA is not directly applicable to SFDA due to the inaccessibility of source data.

In this work, we shed light on the theoretical perspective of existing SFDA methods through the theory of self-training based on the expansion assumption [184]. Self-training is an approach that utilizes the current model predictions of the unlabeled data for further training of the model itself, and the expansion assumption states that the data distribution has good continuity within each class. The theory asserts that, under the expansion assumption, there is an upper bound on the target error when the model is trained on the objective with a **self-training** term encouraging prediction consistency among the augmented unlabeled samples and a **constraint** term ensuring prediction diversity. We reveal an interesting correspondence between this training objective and the SFDA training loss. Recent studies [33, 190] have discovered a common feature that most SFDA methods employ the combination of **discriminability** and **diversity** losses: the former improves the model discriminability to the unlabeled target samples while the latter ensures predictions for all classes. We find that the discriminability and diversity losses perform the same respective roles as the self-training term and the constraint term of the theory, which is illustrated in the middle right of Fig. 2.1. Our theoretical observation of SFDA also brings the following new insights: 1) the trade-off

between discriminability and diversity should be adjusted as training progresses, and 2) the upper bound of the target error depends on how we design the data augmentation.

Based on the above insights, we propose an improved SFDA method that alternately performs 1) model training with auto-adjusting diversity constraint and 2) augmentation training with teacher-student framework. In the former training, we update the model on the basis of the discriminability and diversity losses while introducing a novel technique to automatically adjust the trade-off parameter between discriminability and diversity. In the latter training, we introduce a learnable data augmentation and update its parameters by using the predictions of the current model and the teacher model, yielding a tighter upper bound. Experimental results with three benchmark datasets (Office-31 [145], Office-Home [176], VisDA2017 [130]) show the validity of our theoretical analysis and the proposed method.

Our contributions in this chapter are summarized as follows:

- On the basis of the theory of self-training based on the expansion assumption [184], we reveal that a model trained with discriminability and diversity losses will achieve a small target error.
- We propose an improved SFDA method incorporating model training with auto-adjusting diversity constraint and augmentation training with teacher-student framework.

2.2 Related Works

2.2.1 Unsupervised Domain Adaptation (UDA)

On the basis of the theoretical foundation that the target error is upper-bounded by the source error and the distributional discrepancy between the two domains [12, 11, 117, 202, 204], various UDA methods have been developed. Distribution matching approach [111, 171, 113] involves mapping source and target data to a feature space using a neural network. A statistical distance or metric is then computed to quantify the divergence between the two feature distributions. The model is subsequently optimized to minimize this divergence, thereby encouraging domain alignment. Adversarial learning approach [170, 48, 112, 198] employs a domain discriminator tasked with identifying the origin (source or target) of the features extracted from the penultimate layer of the model. The model is trained in an adversarial manner to extract the features that are indistinguishable to the discriminator. This competitive training scheme compels the model to learn domain-invariant representations. In the pseudo-labeling approach [216, 215, 88, 108], the model generates predictions for the unlabeled target data, which are then used as provisional "pseudo-labels." The model is then fine-tuned using this augmented dataset. This process is repeated, iteratively refining the pseudo-labels and adapting the model to the target domain for high-performance prediction. Although the theory of UDA has yielded various methods, it is not applicable to SFDA due to the inaccessibility of the source data. In this work, we instead employed the theory of self-training based on the expansion assumption [184] as a way to understand SFDA methods.

2.2.2 Source-free Domain Adaptation (SFDA)

According to the survey literature [101], SFDA methods can be roughly categorized into four approaches. Source-estimation approaches [99, 71, 166, 136, 43, 187] generate pseudo-source data using a pre-trained model, which transforms the SFDA problem into a conventional UDA problem. Pseudo-labeling approaches [191, 137, 83] assign a class label to each unlabeled target sample using the current model. These methods have a similar concept to the UDA ones, but are different from them in that the pseudo-labeling SFDA methods never utilize the source data. Based on the cluster assumption [174], clustering-based approaches [102, 99, 91] encourage minimizing the uncertainty of the model predictions or performing clustering over the target features. Inspired by the consistency regularization of semi-supervised learning [163, 13, 155], consistency-learning approaches [192, 23, 190, 203] train the model to maximize the prediction consistency regardless of the perturbations on the input data or the model parameters.

Although many successful SFDA methods have been proposed, most of them, except for the source estimation approaches that convert SFDA to conventional UDA setups, do not have a theoretical foundation yet. A few studies [203] have theoretically investigated their methods, but they are not applicable to the others. In this work, we introduce the theory of self-training based on the expansion assumption [184] to give a theoretical perspective on a wide range of SFDA methods, and we propose an improved method based on our theoretical analysis.

2.3 Understanding SFDA from a Theoretical Perspective

In this section, we formally define the SFDA problem, and then, we see the common feature of SFDA methods. Finally, we provide a theoretical analysis of SFDA through [184].

2.3.1 Problem Setup

We are given a model $F_S : \mathcal{X} \rightarrow \mathcal{Y}$ trained on a labeled source domain data D_S and an unlabeled target domain data $D_T = \{x_t^{(i)}\}_{i=1}^{n_t}$. Generally, the model F consists of a feature extractor f and a fully-connected layer-based classifier g . The goal is to train the model F to obtain a target-adapted model $\hat{F}$ that has low target error without using source domain data D_S nor target labels $y_t^{(i)}$.

2.3.2 Common Feature of SFDA Methods

Whilst lacking a precise theoretical background, the following observation [33, 190] provides a key to understanding SFDA methods: the existing SFDA methods have a common feature that their training loss functions can be decomposed into **discriminability** and **diversity** losses. The discriminability loss enhances the model’s discriminability to the unlabeled target samples, while the diversity loss ensures the model has predictions for diverse classes.

For example, SHOT-IM [102], a pioneering work on SFDA, trains the model on the basis of mutual information maximization, i.e., a training loss function comprising conditional entropy minimization (discriminability) and marginal entropy

maximization (diversity):

$$\mathcal{L}_{\text{MIM}} = \underbrace{H(Y|X)}_{\text{discriminability}} - \lambda_{\text{div}} \underbrace{H(Y)}_{\text{diversity}}. \quad (2.1)$$

where λ_{div} represents the trade-off parameter of the loss.

Another example is AaD [190], which trains the model by maximizing the prediction similarities among the local neighborhoods in the feature space (discriminability) while minimizing those of the others (diversity) as follows:

$$\mathcal{L}_{\text{AaD}} = \frac{1}{|\mathbf{B}|} \sum_{i \in \mathbf{B}} \left\{ \underbrace{\frac{1}{|\mathbf{C}_i|} \sum_{\hat{\mathbf{p}} \in \mathbf{C}_i} -\mathbf{p}_i \cdot \hat{\mathbf{p}}}_{\text{discriminability}} + \lambda_{\text{div}} \underbrace{\sum_{j \in \mathbf{B} \setminus \{i\}} \mathbf{p}_i \cdot \mathbf{p}_j}_{\text{diversity}} \right\}, \quad (2.2)$$

where $\mathbf{B}$ is a mini-batch, $\mathbf{p}_i$ is the prediction of sample i , and $\mathbf{C}_i$ is a set of the predictions of K -nearest neighborhoods of sample i on the feature space.

This common feature is widely seen in other methods, such as those using pseudo-labeling or consistency regularization as their own discriminability loss with the above marginal entropy maximization (diversity) [191, 137, 192], and those employing contrastive learning, which maximizes the similarity of positive pairs (discriminability) and minimizes the similarity of negative pairs (diversity) [83, 203, 23].

Now that we have confirmed the discriminability and diversity losses to be the key to the success of SFDA methods, but *why are these losses crucial for the success of SFDA?*

2.3.3 Theoretical Understanding of SFDA

We will answer the above question by introducing the theory of self-training based on the expansion assumption [184]. This theory shows that, under certain assumptions, the model will have a low target error when it is **self-trained** based on prediction consistency, while a **constraint** is imposed to ensure prediction diversity.

Notations. We let $\mathcal{A}$ denote the family of data augmentation and define an augmented sample set an input x as $\mathcal{B}(x) := \{x' \mid \exists A \in \mathcal{A} \text{ s.t. } \|x' - A(x)\| < r\}$, the neighborhoods of x as $\mathcal{N}(x) := \{x' \mid \mathcal{B}(x) \cap \mathcal{B}(x') \neq \emptyset\}$, the neighborhoods of the set V as $\mathcal{N}(V) := \cup_{x \in V} \mathcal{N}(x)$, and prediction inconsistency for a C -class prediction model $F : \mathcal{X} \rightarrow [C]$ on a distribution P as $R_{\mathcal{B}}(F) := \mathbb{E}_P[\mathbf{1}(\exists x' \in \mathcal{B}(x) \text{ s.t. } F(x') \neq F(x))]$.

Assumptions. Before explaining the main theorem that is the key to our theoretical analysis, we must make two assumptions: *Expansion* and *Separation*.

Expansion assumes that the data of the same class are distributed in a continuous region and that any small region V has a neighborhood region of the same class larger than V . Concretely, $P_i(\mathcal{N}(V)) \geq dP_i(V)$ for $P_i(V) \leq 1/2$, $d > 1$, where $P_i(V)$ represents the proportion of subset V in the total class i data and d represents an expansion factor that corresponds to the strength of the data augmentation.

Separation assumes that the distribution of different classes is separated and the predictions of the ground-truth model F^* will not be altered by the data augmentation, i.e., $R_{\mathcal{B}}(F^*) < \mu$, where μ represents a negligible value.

Formal statements of the assumptions are given in Sec. 2.7.1.

Theory for Understanding SFDA. With the above assumptions, [184] derives

the following theorem that is key to understanding why existing SFDA methods perform well.

Theorem 1. *Suppose that the above two assumptions hold for some d, μ such that $\min_{y \in [C]} P(\{x : F^*(x) = y\}) > \max\{2/(d-1), 2\}\mu$. Then any minimizer $\hat{F}$ of*

$$\underbrace{\min_F R_{\mathcal{B}}(F)}_{\text{self-training} = \text{discriminability}} \quad \text{subject to} \quad \underbrace{\min_{y \in [C]} \mathbb{E}_P[\mathbf{1}(F(x) = y)] > \max\left\{\frac{2}{d-1}, 2\right\}}_{\text{constraint} = \text{diversity}} R_{\mathcal{B}}(F) \quad (2.3)$$

satisfies

$$\text{Err}_{\mathcal{U}}(\hat{F}) \leq \max\left\{\frac{d}{d-1}, 2\right\} \mu, \quad (2.4)$$

where $\text{Err}_{\mathcal{U}}(\hat{F}) := \min_{\pi: [C] \rightarrow [C]} [\mathbf{1}(\pi(F(x)) \neq F^*(x))]$, and π represents a permutation.

Theorem 1 shows that a model trained with the objective function (2.3) consisting of a self-training term that requires neighborhood predictions to be consistent and a constraint term that assigns a certain portion of predictions to all classes has an upper bound on the target error (2.4).

What we want to highlight is that the SFDA training loss with discriminability and diversity loss acts the same way as the objective (2.3). Specifically, increasing the **discriminability** leads to a reduction in the prediction inconsistency of the **self-training** term, while the **diversity** loss functions in the same way as the **constraint** term that ensures the minor class predictions. This indicates that we can apply Theorem 1 to the SFDA training loss, and thus, the SFDA-trained model also has an upper bound on the target error.

Method	Acc [%]
i) Source only	72.4
ii) Dis only	93.0
iii) Dis + Div	95.3
iv) Dis + Div w/ Decay	95.7

Table 2.1: Model accuracy w/ and w/o discriminability and diversity losses.

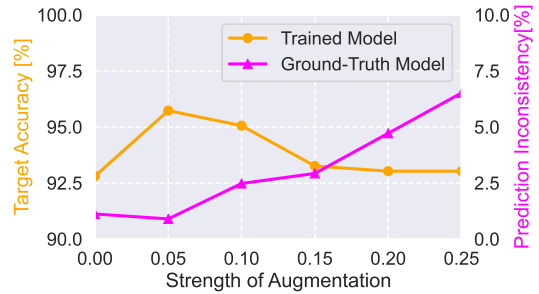


Figure 2.2: Model accuracy ($\text{Err}_U(\hat{F})$) and the prediction inconsistency of the ground-truth model (μ) against strength of augmentation (d).

However, please note that although this finding provides the theoretical understanding behind the success of current mainstream SFDA methods, which has been missing in the literature, it does not purport to solve problems vastly different from those handled in existing studies. Specifically, dealing with types of domain shifts that are distinctively different from those addressed in prior work is beyond the scope of this finding, constituting a limitation of our work. We leave the development of theories and methods for extending the applicability of SFDA methods as a future direction of this work.

Preliminary Experiment with Synthetic Data. We verified this correspondence through an experiment on a two-dimensional synthetic dataset. We employed a variant of the inter-twinning moons 2D dataset, where we simulated the domain shift by rotation. We used the training objective of SHOT-IM [102] and 2D-Gaussian perturbations as the data augmentation. In the experiment, we compared the accuracy of i) Source only, ii) Discriminability (Dis) only, and iii) Discriminability and Diversity (Dis + Div). Moreover, we measured the accuracy of the trained model $\text{Err}_U(\hat{F})$ and the prediction inconsistency of the ground-truth

model corresponding to μ versus the strength of the augmentation corresponding to d . Other details on the experimental settings are described in Sec. 2.7.2.

The results are summarized in Tab. 2.1 and Fig. 2.2. Tab. 2.1 shows that the accuracy is improved by incorporating the diversity loss in addition to the discriminability loss, which establishes the necessity of the diversity loss to function as the constraint of the objective (2.3). The prediction accuracy of the trained model in Fig. 2.2 aligns with the upper bound (2.4); namely, as shown in Fig. 2.2, by keeping the prediction inconsistency of the ground-truth model μ low and increasing the strength of the data augmentation d , the model can achieve a low target error.

Theoretical Insights. Besides, Theorem 1 provides two insights for the further improvement of SFDA methods.

First, the weight of the diversity loss λ_{div} should be adjusted as training progresses, and this can be done by controlling the value of the discriminability loss. The right-hand side (RHS) of the constraint in the objective (2.3) includes $R_{\mathcal{B}}(F)$, which decreases during the training. This indicates that it is more reasonable to let the constraint decay along with $R_{\mathcal{B}}(F)$, i.e., the discriminability loss. The results shown in Tab. 2.1 also demonstrate that decaying λ_{div} (Dis + Div w/ Decay) brings a better result. However, most of the existing SFDA methods fix λ_{div} , which would be suboptimal. Although AaD [190] exceptionally uses a manually designed scheduler, tuning the scheduler is laborious.

Second, the upper bound of the target error depends on the parameters d and μ , which are relevant to the data augmentation properties. This indicates that how we design the data augmentation is a critical factor in training models with better accuracy. However, the prior studies on SFDA [203, 23] have paid less attention to it and have used pre-defined data augmentations [31, 32], which may not be

optimal for SFDA.

2.4 Our Method: Improved SFDA

Fig. 2.3 shows the overview of our method, which has three major components: a prediction model $F = g \circ f$, a teacher model $F' = g' \circ f'$, and a learnable augmentation $\mathcal{A}$. Note that we employed differential data augmentation [63] to make $\mathcal{A}$ trainable in the same way as the training of standard neural networks. On the basis of the above insights, we developed an improved SFDA method comprising 1) model training with auto-adjusting diversity constraint, and 2) augmentation training with teacher-student framework.

2.4.1 Model Training with Auto-Adjusting Diversity Constraint

We update F upon the modified discriminability and diversity losses of AaD [190], coupled with a novel technique to automatically adjust the trade-off parameter between discriminability and diversity.

Discriminability and Diversity Losses. Considering that the prediction inconsistency $R_{\mathcal{B}}(F)$ in the self-training term is originally calculated among data-augmented samples, we modify the training loss (2.2) so as to calculate the prediction dissimilarity among data-augmented samples. Specifically, the discriminability and diversity losses are formally defined as

$$\mathcal{L}_{\text{dis}} = \frac{1}{M|\mathbf{B}||\mathbf{C}_i|} \sum_{i \in \mathbf{B}} \sum_{\hat{\mathbf{p}} \in \mathbf{C}_i} \sum_{m=1}^M (1 - \mathbf{p}_i^m \cdot \hat{\mathbf{p}}), \quad (2.5)$$

$$\mathcal{L}_{\text{div}} = \frac{1}{M|\mathbf{B}|} \sum_{i \in \mathbf{B}} \sum_{j \in \mathbf{B} \setminus \{i\}} \sum_{m=1}^M \mathbf{p}_i^m \cdot \mathbf{p}_j, \quad (2.6)$$

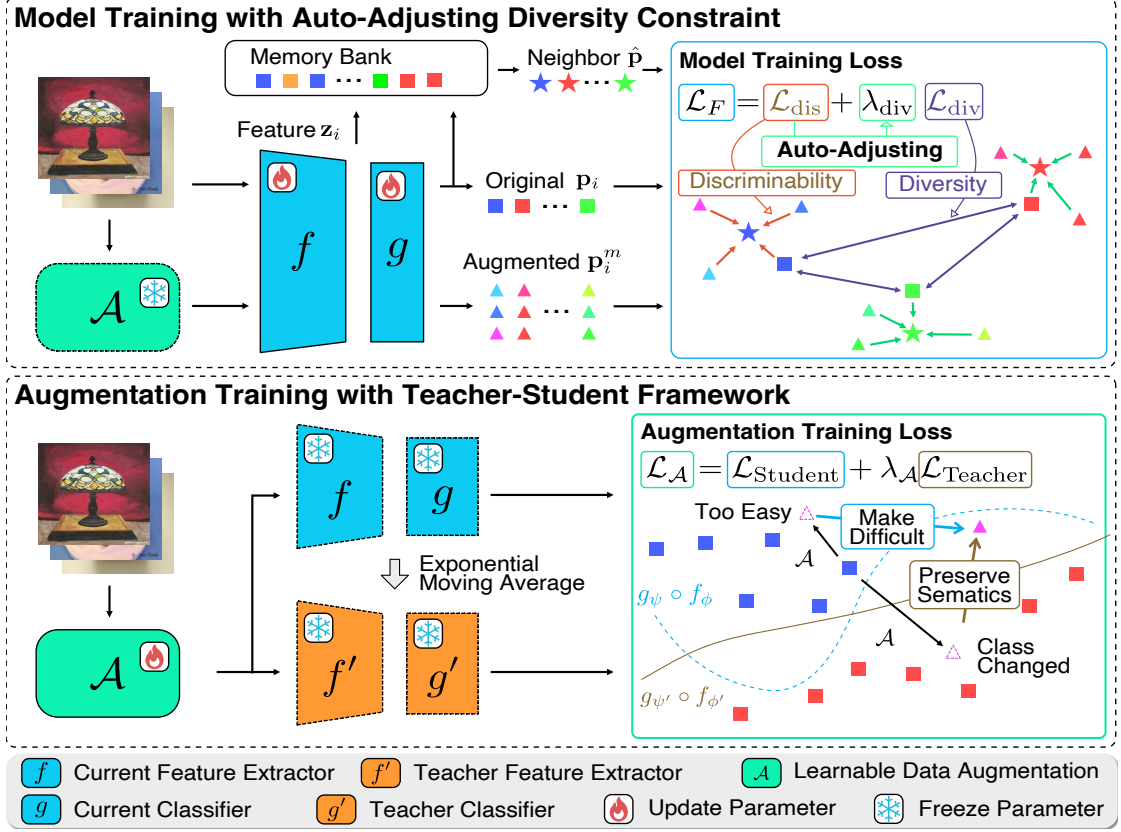


Figure 2.3: **Overview of our method.** In model training with auto-adjusting diversity constraint, we train the model by minimizing the discriminability loss $\mathcal{L}_{\text{dis}}$ and the diversity loss $\mathcal{L}_{\text{div}}$ while automatically adjusting the trade-off parameter between discriminability and diversity. In augmentation training with teacher-student framework, we train a learnable data augmentation A to generate harder samples for the current model F while suppressing the prediction inconsistency of the teacher model F' .

where $\mathbf{p}_i^m = F(A_m(x_t^{(i)}))$, A_m is the m -th augmentation sampled from $\mathcal{A}$, M is the number of augmentations. To retrieve the K -nearest neighbors $\mathbf{C}_i$ efficiently, we build a memory bank that stores the feature vectors $\mathbf{z}_i = f(x_t^{(i)})$ and predictions $\mathbf{p}_i = F(x_t^{(i)})$ of all target samples in D_T .

Auto-Adjusting Diversity Constraint. As shown in Sec. 2.3.3, λ_{div} should be adjusted as the training progresses. Specifically, the RHS of the constraint

of the objective (2.3) involves the prediction inconsistency $R_{\mathcal{B}}(F)$, which will get smaller as the training progresses. Moreover, the discriminability loss $\mathcal{L}_{\text{dis}}$ functions the same way as $R_{\mathcal{B}}(F)$. This means that we can easily control λ_{div} with $\mathcal{L}_{\text{dis}}$. Accordingly, the auto-adjusting λ_{div} can be simply expressed as

$$\lambda_{\text{div}} = \lambda_{\text{div}}^{\max} \mathcal{L}_{\text{dis}}, \quad (2.7)$$

where $\lambda_{\text{div}}^{\max}$ determines the maximum size of λ_{div} . Note that we apply the stop-gradient operation to λ_{div} .

Model Training. The training loss of the model using the learnable augmentation $\mathcal{A}$ is

$$\mathcal{L}_F = \mathcal{L}_{\text{dis}} + \lambda_{\text{div}} \mathcal{L}_{\text{div}}. \quad (2.8)$$

However, we find in an early study that the learnable data augmentation $\mathcal{A}$ (whose details will be described in Sec. 2.4.2) yields some heavy augmentations (e.g., `rotation`, `invert`, etc.), which may impair the model training. Inspired by [8], we stabilize the training by incorporating a loss $\mathcal{L}'_F$ calculated among samples with weak augmentations (e.g., `random_clip`, `random_flip`).

In summary, the total loss of the model training is

$$\min_F \mathcal{L}_F^{\text{Total}} = \lambda_F \mathcal{L}_F + (1 - \lambda_F) \mathcal{L}'_F. \quad (2.9)$$

where λ_F is a hyper-parameter to control the loss balance.

2.4.2 Augmentation Training with Teacher-Student Framework

As discussed in Sec. 2.3.3, the upper bound of the target error depends on how the data augmentation is designed. Motivated by [161], we update the learnable data augmentation $\mathcal{A}$ in the teacher-student framework to get a tighter bound.

Learnable Data Augmentation. $\mathcal{A}$ consists of L different augmentations $A^{(l)}$ ($l = 1, 2, \dots, L$). A single augmentation consists of N consecutive transformation operations $O_1^{(l)}, \dots, O_N^{(l)}$. Each operation includes affine transformations (e.g., `shear_x`) and color enhancing operations (e.g., `contrast`), and it has a magnitude parameter $m_n^{(l)} \in [0, 1]$ to control the transformation strength and a probability parameter $p_n^{(l)} \in [0, 1]$ to control whether to apply the operation. Following [63], we implement these parameters as differentiable and updatable by gradient descent.

Augmentation Training. We train $\mathcal{A}$ to make the bound tighter based on the observation that the upper bound of the target error (2.4) becomes tighter as 1) the strength of the augmentation d gets larger and 2) the prediction inconsistency of the ground-truth model μ gets smaller. Since we cannot actually access the ground-truth model F^* , we use the teacher-student framework and assign the teacher model to be a proxy for F^* ; 1) we increase d by training $\mathcal{A}$ to augment samples that are harder for the current (student) model to predict, and 2) we decrease μ by training $\mathcal{A}$ to augment samples that are recognizable to the teacher model.

More specifically, 1) we train $\mathcal{A}$ to maximize the prediction entropy of the current model:

$$\mathcal{L}_{\text{Student}} = -\frac{1}{|\mathbf{B}|} \sum_{i \in \mathbf{B}} \sum_{c=1}^C p_i[c] \log(1 - p_i[c]), \quad (2.10)$$

where $p_i[c] = F(A(x_t^{(i)}))[c]$ is the prediction probability of sample i for class c , and

A is a augmentation operation randomly sampled from $\mathcal{A}$. Following [161], we employ a non-saturating prediction entropy instead of the naive one.

Whereas, 1) we train $\mathcal{A}$ to minimize the prediction entropy of the teacher model:

$$\mathcal{L}_{\text{Teacher}} = -\frac{1}{|\mathbf{B}|} \sum_{i \in \mathbf{B}} \sum_{c=1}^C p'_i[c] \log p'_i[c], \quad (2.11)$$

where $p'_i[c] = F'(A(x_t^{(i)}))[c]$.

The total loss of the augmentation training is

$$\min_{\mathcal{A}} \mathcal{L}_{\mathcal{A}} = \mathcal{L}_{\text{Student}} + \lambda_{\mathcal{A}} \mathcal{L}_{\text{Teacher}}, \quad (2.12)$$

where $\lambda_{\mathcal{A}}$ is a coefficient hyper-parameter.

With regard to the teacher model configuration, we update the teacher model in parallel with model training using the exponential moving average strategy [163]:

$$F' \leftarrow (1 - \beta)F + \beta F', \quad (2.13)$$

where β is a momentum parameter.

Discussion. Our method is different from the teacher-student-based SFDA methods [23, 201] in that we use the teacher-student framework for training data augmentation. Our approach may appear similar to the prior work [168] but fundamentally differs in that ours treats the teacher model as a proxy of the ground truth model based on our theoretical insights.

Algorithm 1 Training Procedure of Our Method.

Input: Source-trained model F_s , Target domain data D_T

Output: Target-adapted model $\hat{F}$

```
1: for  $e$  in  $\{1 \cdots E\}$  do
2:   for Mini-batch  $\mathbf{B}$  in  $D_T$  do                                     ▷ Model Training
3:     Calculate loss  $\mathcal{L}_F^{\text{Total}}$  on  $\mathbf{B}$ 
4:     Update  $F$  to minimize  $\mathcal{L}_F^{\text{Total}}$ 
5:     Update  $F'$  based on (2.13)
6:   end for
7:   if  $e \equiv 0 \pmod{\hat{e}}$  then
8:     for Mini-batch  $\mathbf{B}$  in  $D_T$  do                                     ▷ Augmentation Training
9:       Calculate loss  $\mathcal{L}_A$  on  $\mathbf{B}$ 
10:      Update  $\mathcal{A}$  to minimize  $\mathcal{L}_A$ 
11:    end for
12:  end if
13: end for
```

2.4.3 Training Procedure of Our Method

As shown Algorithm 1, our method alternately performs model training and data augmentation training. To control the speed of these pieces of training, we set a parameter $\hat{e}$ that determines the interval of the augmentation training.

2.5 Experiments

We experimentally compared the performance of our method and the existing SFDA methods on three benchmark datasets, and we examined the validity of our theoretical insights through further analyses.

2.5.1 Setups

Datasets. We used three benchmark datasets: Office-31 [145] is a small-scale dataset, which consists of 31 categories and three domains (**A**mazon, **W**ebcam, and **D**slr). Office-Home [176] is a moderate-scale dataset, which consists of 65 cate-

gories and four domains (**Art**, **Clipart**, **Product** and **Real world**). VisDA2017 [130] is a large-scale synthetic-to-real adaptation benchmark dataset with 12 categories. For Office-31 and Office-Home, we evaluated the accuracy on all source-target combinations, while we computed the average of per-class accuracies for VisDA2017.

Network Architectures & Augmentation Implementations. We used ResNet-50 [65] as the backbone network in the Office-31 and Office-Home experiments, and ResNet-101 [65] in VisDA2017. All of the networks were pre-trained on Imagenet [35]. Following [102], we replaced the output layer of the backbone network with the following networks: a *fully-connected layer* $\rightarrow$ *batch normalization* [79] $\rightarrow$ *fully-connected layer with weight normalization* [148]. We implemented the learnable data augmentation $\mathcal{A}$ with a public library of differentiable data augmentation¹. We set L to 25, and N to two in all of the experiments.

Source Training. We used Nesterov SGD with a mini-batch size of 64 as the optimization algorithm on all three datasets.

For Office-31 and Office-Home, we set the learning rate η to $1e-2$ for the last replaced layers and $1e-3$ for the backbone layers, momentum to 0.9, and weight decay to $5e-4$. We used a standard cross-entropy loss with label smoothing for training. The label smoothing parameter was set to 0.1. We trained the model for 50 epochs.

For VisDA2017, we set the initial learning rate η_{init} to $1e-3$ for the last replaced layers and $1e-4$ for the backbone layers, momentum to 0.9, and weight decay to $1e-3$. The learning rate η was scheduled as; $\eta = \eta_{\text{init}}(1 + 10p)^{-0.75}$, where p was linearly increased from 0.0 to 1.0 throughout the training. We used the training losses of Office-31 and Office-Home. We trained the model for 10 epochs.

¹<https://github.com/moskomule/dda>

Table 2.2: **Classification Accuracy (%) on Office-31 (ResNet-50)**. The best and second best are highlighted in **bold** and with underline.

Method	A $\rightarrow$ D	A $\rightarrow$ W	D $\rightarrow$ W	W $\rightarrow$ D	D $\rightarrow$ A	W $\rightarrow$ A	Avg.
3C-GAN [99]	92.7	93.7	98.5	99.8	75.3	77.8	89.6
SHOT [102]	94.0	90.1	98.4	<u>99.9</u>	74.7	74.3	88.6
VDM-DA [166]	94.1	93.2	98.0	100.0	75.8	77.1	89.7
A ² Net [187]	94.5	94.0	99.2	100.0	<u>76.7</u>	76.1	90.1
NRC [191]	96.0	90.8	99.0	100.0	75.3	75.0	89.4
CPGA [136]	94.4	94.1	98.4	99.8	76.0	76.6	89.9
CoWA-JMDS [91]	94.4	95.2	98.5	99.8	76.2	<u>77.6</u>	<u>90.3</u>
C-SFDA [83]	<u>96.2</u>	93.9	98.8	99.7	77.3	77.9	90.5
AaD [190]	96.4	92.1	<u>99.1</u>	100.0	75.0	76.5	89.9
Improved SFDA	95.3	<u>94.2</u>	98.3	<u>99.9</u>	76.4	77.5	<u>90.3 (+0.4)</u>

Target Training. We used the Nesterov SGD for training F and AdamW [114] for training $\mathcal{A}$ with mini-batch size 64.

For Office-31 and Office-Home, we set Nesterov SGD parameters as follows: the learning rate η for the second last layer to $3\text{e}-3$ and for the backbone layers to $3\text{e}-4$, momentum to 0.9, and weight decay to $5\text{e}-4$. We fixed the last layer during the training, which yielded better results. We set the parameters of AdamW as follows: the learning rate η^{aug} to $5\text{e}-4$, the momentum parameters to (0.9, 0.999), and the weight decay parameter to $1\text{e}-2$. The number of the training epoch E was set to 100, and the interval $\hat{e}$ was set to three. The other parameters were set as follows; $\lambda_{\text{div}}^{\text{max}}$ to 0.4 for Office-31 and 0.75 for Office-Home, λ_F to 0.5, $\lambda_{\mathcal{A}}$ to 1.0, K to two, M to three, and β to 0.99. We initialized $\mathcal{A}$ with AutoAugment [31] Imagenet Policies.

For VisDA2017, we set the Nesterov SGD parameters as follows: the initial learning rate η_{init} for the last two layers to $2.5\text{e}-3$ and for the backbone layers to $2.5\text{e}-4$, momentum to 0.9, and weight decay to $1\text{e}-4$. We set the parameters of AdamW as follows: initial learning rate $\eta_{\text{init}}^{\text{aug}}$ to $2\text{e}-3$, the momentum parameters

Table 2.3: **Classification Accuracy (%) on Office-Home (ResNet-50)**. The best and second best are highlighted in **bold** and with underline.

Method	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw
SHOT [102]	57.1	78.1	81.5	68.0	78.2	78.1
A ² Net [187]	58.4	79.0	82.4	67.5	79.3	78.9
G-SFDA [192]	57.9	78.6	81.0	66.7	77.2	77.2
NRC [191]	57.7	80.3	82.0	68.1	79.8	78.6
CPGA [136]	59.3	78.1	79.8	65.4	75.5	76.4
U-SFAN [144]	57.8	77.8	81.6	67.9	77.3	79.2
CoWA-JMDS [91]	56.9	78.4	81.0	69.1	<u>80.0</u>	79.9
DaC [203]	59.1	79.5	81.2	<u>69.3</u>	78.9	79.2
C-SFDA [83]	<u>60.3</u>	<u>80.2</u>	82.9	<u>69.3</u>	80.1	78.8
AaD [190]	59.3	79.3	<u>82.1</u>	68.9	79.8	79.5
Improved SFDA	60.7	78.9	82.0	69.9	79.5	<u>79.7</u>

Method	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg.
SHOT [102]	67.4	54.9	82.2	73.3	58.8	84.3	71.8
A ² Net [187]	68.0	56.2	82.9	<u>74.1</u>	60.5	85.0	72.8
G-SFDA [192]	65.6	56.0	82.2	72.0	57.8	83.4	71.3
NRC [191]	65.3	56.4	83.0	71.0	58.6	85.6	72.2
CPGA [136]	65.7	58.0	81.0	72.0	64.4	83.3	71.6
U-SFAN [144]	67.2	54.7	81.2	73.3	60.3	83.9	71.9
CoWA-JMDS [91]	<u>67.7</u>	57.2	82.4	72.8	60.5	84.5	72.5
DaC [203]	67.4	56.4	82.4	74.0	61.4	84.4	72.8
C-SFDA [83]	67.3	<u>58.1</u>	83.4	73.6	<u>61.3</u>	<u>86.3</u>	73.5
AaD [190]	67.2	57.4	<u>83.1</u>	72.1	58.5	85.4	72.7
Improved SFDA	67.1	58.8	82.3	74.2	<u>61.3</u>	86.4	<u>73.4 (+0.7)</u>

to $(0.9, 0.999)$, and the weight decay parameter to $1e-2$. η and η^{aug} are scheduled as in the source training. The training epoch E was set to 100 and the interval $\hat{e}$ to three. The other parameters were set as follows; $\lambda_{\text{div}}^{\text{max}}$ to 0.08, λ_F to 0.2, λ_A to 1.0, K to five, M to three, and β to 0.99. We randomly initialized $\mathcal{A}$.

2.5.2 Main results

We evaluated the performance of our method by taking the average score of three different runs for all benchmarks.

Table 2.4: **Classwise Accuracy (%) on VisDA2017 (ResNet-101).** The best and second best are highlighted in **bold** and with underline.

Method	plane	bcycl	bus	car	horse	knife
3C-GAN [99]	94.8	73.4	68.8	74.8	93.1	95.4
SHOT [102]	94.3	88.5	80.1	57.3	93.1	94.9
VDM-DA [166]	96.9	89.1	79.1	66.5	95.7	96.8
A ² Net [187]	94.0	87.8	85.6	66.8	93.7	95.1
G-SFDA [192]	96.1	83.3	85.5	74.1	97.1	95.4
NRC [191]	96.8	<u>91.3</u>	82.4	62.4	96.2	95.9
CPGA [136]	95.6	89.0	75.4	64.9	91.7	97.5
U-SFAN [144]	-	-	-	-	-	-
AdaContrast [23]	97.0	84.7	84.0	77.3	96.7	93.8
CoWA-JMDS [91]	96.2	89.7	83.9	73.8	96.4	<u>97.4</u>
DaC [203]	96.6	86.8	<u>86.4</u>	78.4	96.4	96.2
C-SFDA [83]	97.6	88.8	86.1	72.2	<u>97.2</u>	94.4
AaD [190]	97.4	90.5	80.8	76.2	97.3	96.1
Improved SFDA	<u>97.5</u>	91.4	87.9	79.4	<u>97.2</u>	97.2

Method	mcycl	person	plant	sktbrd	train	truck	Per-class
3C-GAN [99]	88.6	84.7	89.1	84.7	83.5	48.1	81.6
SHOT [102]	80.7	80.3	91.5	89.1	86.3	58.2	82.9
VDM-DA [166]	85.4	83.3	96.0	86.6	89.5	56.3	85.1
A ² Net [187]	85.8	81.2	91.6	88.2	86.5	56.0	84.3
G-SFDA [192]	89.5	79.4	95.4	92.9	89.1	42.6	85.4
NRC [191]	86.1	80.6	94.8	94.1	90.4	59.7	85.9
CPGA [136]	89.7	83.8	93.9	93.4	87.7	69.0	86.0
U-SFAN [144]	-	-	-	-	-	-	82.7
AdaContrast [23]	91.9	<u>84.8</u>	94.3	93.1	94.1	47.9	86.8
CoWA-JMDS [91]	89.3	86.8	94.6	92.1	88.7	53.8	86.9
DaC [203]	93.6	83.8	96.8	95.1	89.6	50.0	87.3
C-SFDA [83]	92.1	84.7	93.0	90.7	<u>93.1</u>	63.5	87.8
AaD [190]	89.8	82.9	95.5	93.0	92.0	<u>64.7</u>	<u>88.0</u>
Improved SFDA	<u>92.2</u>	83.0	<u>96.4</u>	<u>94.2</u>	91.1	53.0	88.4 (+0.4)

Result on Office-31. The results are shown in Tab. 2.2. Our method improved accuracy by 0.4% on average compared with the baseline AaD [190]. Furthermore, ours was comparable in accuracy to the second-best method, CoWA-JMDS [91], and only 0.2% off the best method, C-SFDA [83].

Result on Office-Home. The results are shown in Tab. 2.3. Ours improved

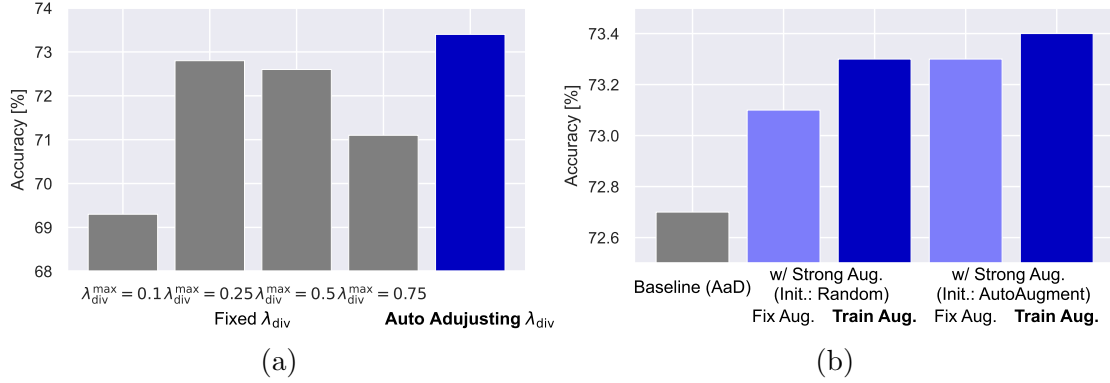


Figure 2.4: **Ablation Study.** (a) Auto-Adjusting Diversity Constraint (b) Augmentation Training.

accuracy by 0.7% on average compared with AaD [190] and was second best. The accuracy difference from the best method, C-SFDA [83], was merely 0.1%.

Result on VisDA2017. The results are shown in Tab. 2.4. Ours improved accuracy by 0.4% compared with AaD [190]. The average of per-class accuracy reached 88.4%, which was the best among the compared methods.

2.5.3 Analysis

Ablation study. Using Office-Home, we analyzed the effectiveness of the proposed two components.

Model Training with Auto-Adjusting Diversity Constraint. We validated the auto-adjusting diversity constraint strategy by comparing our method with a variant that uses a fixed λ_{div} , i.e., $\lambda_{div} = \lambda_{div}^{max}$. Considering that the optimal λ_{div}^{max} for this “Fixed λ_{div} ” may differ from that of our method, we performed experiments with several values of λ_{div}^{max} . The results are shown in Fig. 2.4a. Our method outperformed all fixed λ_{div} variants, which indicates the validity of our proposed techniques. We also obtained the same results from the synthetic data experiment,

Table 2.5: **Analysis of Application to SHOT-IM.**

	Office-31	Office-Home	VisDA2017
SHOT-IM [102]	87.3	70.5	80.4
Improved SFDA	88.6 (+1.3)	71.2 (+0.7)	83.3 (+2.9)

which is described in Sec. 2.7.2.

Augmentation Training with Teacher-Student Framework. We validated the effectiveness of our augmentation training strategy by comparing its performance with that of a fixed augmentation variant. The results are shown in Fig. 2.4b. Firstly, by incorporating stronger augmentations to the baseline, we confirmed the performance improvement, which is in line with our theoretical observation that reasonably strong data augmentation contributes to minimizing target error. Additionally, we confirmed that the performance improvement can be achieved by appropriately controlling the strength of the data augmentation through training. When starting with the randomly initialized data augmentation, our augmentation training yielded a larger accuracy gain from the base method (ΔAaD), 1.5 times greater than without augmentation training. The accuracy gain is slightly smaller when the augmentation is initialized with AutoAugment policies, but ours still yielded better results. More detailed results are provided in Sec. 2.7.3.

Applicability to other SFDA methods. Although we based our method on the implementation of AaD, the theoretical insights and techniques we have developed here should be applicable to many other SFDA methods. To confirm this, we empirically verified the applicability of our proposed techniques to another SFDA method, SHOT-IM [102]. Here, we used the same hyper-parameters as in Sec. 2.5.1, except for the following points; we set $\lambda_{\text{div}}^{\text{max}}$ to 0.7 and η to $2\text{e}-3$ for Office-31 and Office-Home, η_{init} to $1\text{e}-3$ and $\lambda_{\text{div}}^{\text{max}}$ to 0.8 for VisDA2017.

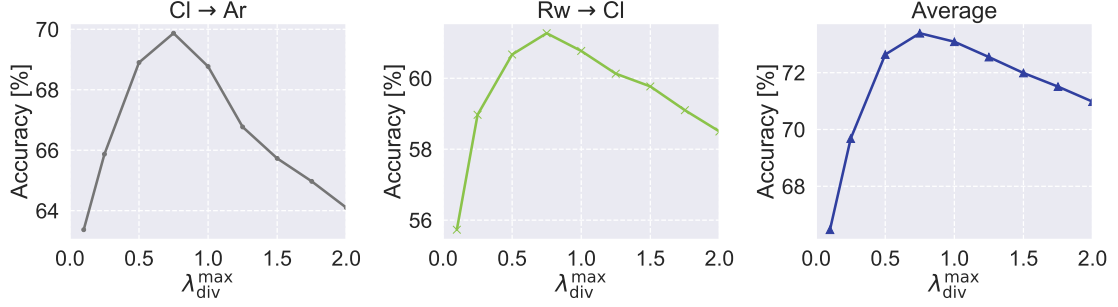


Figure 2.5: **Analysis of the coefficient parameter** $\lambda_{\text{div}}^{\text{max}}$

The results are shown in Tab. 2.5. We can see a steady improvement in all the benchmarks, which demonstrates the effectiveness of our techniques for SHOT-IM. Moreover, since many of the existing SFDA methods are built upon SHOT-IM, this result implies that our method is valid for a wider range of SFDA methods.

Hyper-parameter analysis. We conducted experiments using Office-Home to analyze the effect of the parameter $\lambda_{\text{div}}^{\text{max}}$ in Model Training with Auto-Adjusting Diversity Constraint, and on the parameters $\hat{e}$ and $\lambda_{\mathcal{A}}$ in Augmentation Training with Teacher-Student Framework.

Analysis of $\lambda_{\text{div}}^{\text{max}}$ (Fig. 2.5) $\lambda_{\text{div}}^{\text{max}}$ controls the strength of the prediction diversity constraint. We varied $\lambda_{\text{div}}^{\text{max}}$ from 0.0 to 2.0 and evaluated the accuracy. $\lambda_{\text{div}}^{\text{max}}$ was optimal at 0.75, and the accuracy deteriorated when it was larger or smaller than the optimal value. In particular, the accuracy deteriorated more sharply when it took a smaller value than a larger value. This result is in line with our theoretical analysis. Specifically, if $\lambda_{\text{div}}^{\text{max}}$ is too small, the diversity loss will not play the role of constraining the objective (2.3) sufficiently, and thus we can not obtain the upper bound for the target error. However, if it is too large, the diversity loss becomes an excessive constraint, and that reduces the accuracy.

Analysis of $\hat{e}$ (Fig. 2.6). $\hat{e}$ controls the frequency of the augmentation training. We

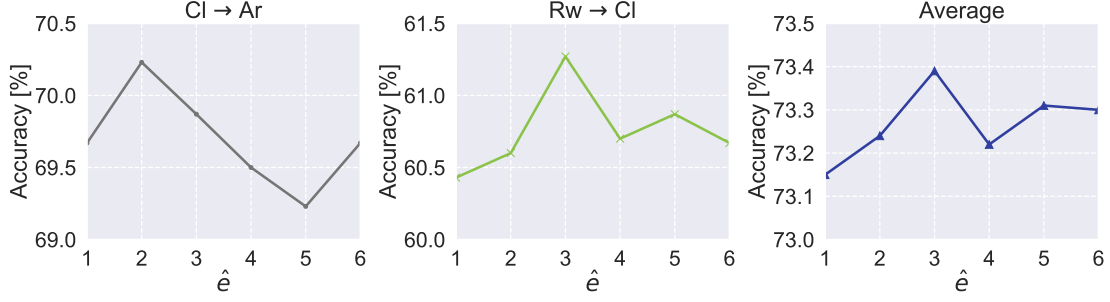


Figure 2.6: **Analysis of the interval parameter $\hat{e}$**

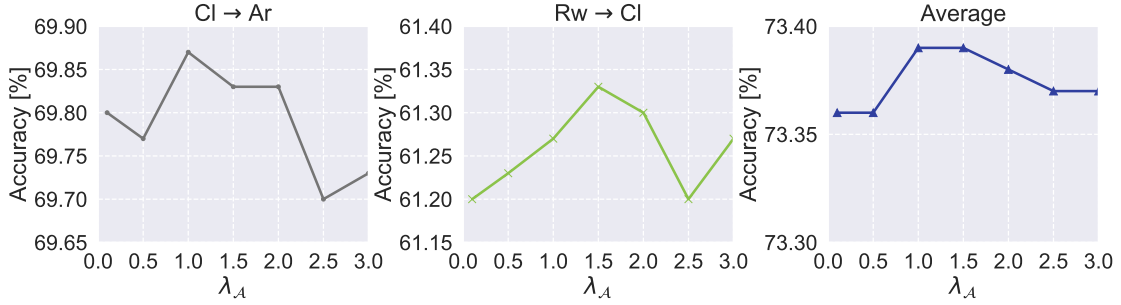


Figure 2.7: **Analysis of the coefficient parameter λ_A**

analyzed the effect of $\hat{e}$ by varying it from one to six. The optimal value of $\hat{e}$ was two or three and the performance was lower when it was set to a larger or smaller value than this. Our augmentation training will not demonstrate its validity unless the current model and the teacher model are different to some extent. When $\hat{e}$ is small, the teacher and the current model are too close to exhibit the full potential of the augmentation training, while when $\hat{e}$ is large, the data augmentation is not trained well enough.

Analysis of λ_A (Fig. 2.7). λ_A controls the effect of $\mathcal{L}_{\text{Student}}$ and $\mathcal{L}_{\text{Teacher}}$, where $\mathcal{L}_{\text{Student}}$ increases d of the bound (2.4) by encouraging the augmentation to generate more difficult samples while $\mathcal{L}_{\text{Teacher}}$ decreases μ of the bound (2.4) by encouraging the augmentation to keep the semantics of the data. We analyzed how our method performed while varying λ_A from 0.0 to 3.0. The optimal value of λ_A was 1.0 to

1.5. If the balance is not proper, it will cause a decrease in d or an increase in μ , resulting in the prediction model having poor accuracy. The results thus demonstrate that our method is consistent with our theoretical insights.

2.6 Conclusion and Limitations

We shed light on the theoretical perspective of existing SFDA methods through the theory of self-training based on the expansion assumption [184]. Our finding that the SFDA training loss with discriminability and diversity functions the same way as the training objective of the theory not only provided a way to understand existing SFDA methods but also yielded two novel techniques for improving the performance of SFDA methods. The experimental results and in-depth analysis justified the validity of our theoretical insights and proposed method. We expect that this work will encourage further development of SFDA research.

Limitations. On the other hand, since this study aims to understand existing SFDA methods, we leave one limitation. That is, we cannot take into account how well the source model originally performs on the target domain, which is also not considered in existing SFDA research. One of our future works is to develop a method that overcomes this weakness and advances SFDA to be more practical.

2.7 Appendix

2.7.1 Formal Definitions and Assumptions

This section is a supplement to the assumptions in Sec. 2.3.3. To describe the assumptions, we first introduce the formal definition of “expansion”.

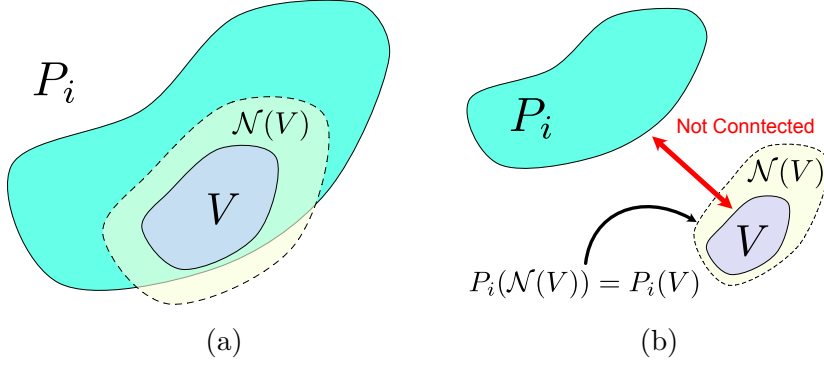


Figure 2.8: **Visual Understanding of Expansion.** (a) Definition of (a, d) -expansion. (b) Example that violates the expansion assumption.

Definition 1 ((a, d) -expansion). *We say that the class-conditional distribution P_i satisfies (a, d) -expansion if for all $V \subseteq \mathcal{X}$ with $P_i(V) \leq a$, the following holds:*

$$P_i(\mathcal{N}(V)) \geq \min\{dP_i(V), 1\} \quad (2.14)$$

If P_i satisfies (a, d) -expansion for all $\forall i \in [C]$, then we say the distribution P satisfies (a, d) -expansion.

Fig 2.8a illustrate the visual understanding of (a, d) -expansion.

Using this definition, the formal statements of *expansion* and *separation* assumptions are described as follows.

Assumption 1 (Expansion requirement for unsupervised learning). *We assume that P satisfies $(1/2, d)$ -expansion on $\mathcal{X}$ for $d > 1$.*

Assumption 2 (Separation). *We assume P is $\mathcal{B}$ -separated with probability $1 - \mu$ by the ground-truth prediction model F^* , as follows: $R_{\mathcal{B}}(F^*) \leq \mu$.*

The *expansion* assumption states that the data distribution of the same class is in a continuous region. This is easier to understand if we look at an example that

does not satisfy the assumption. As shown in Fig 2.8b, if the assumption does not hold, there is a small subset V that does not expand with the data augmentation, i.e., $P(V) = P(\mathcal{N}(V))$. This means that the data of the same class are divided into multiple different regions. Conversely, if the assumption holds, the data of the same class are distributed within one region.

The *separation* assumption states that the data of different classes are separated, and the predictions of the ground-truth model among the augmented samples are consistent. This assumption also requires the data augmentation to maintain the semantics of the data.

2.7.2 Details of Synthetic Data Experiment

In this section, we describe the details of the experiments with the synthetic data in Sec. 2.3.3. We also describe a similar experiment we conducted using another SFDA method, AaD [190] to confirm the validity of our theoretical analysis.

Experimental Setups and Detailed Results.

Dataset. For the dataset construction, we basically refer to the setting in [190], but we slightly change a few points to see the results more clearly. We construct the source domain data with two inter-twinning moons, in which each moon contains 300 samples. We treat each moon as a discrete class of data. We generate the target domain data by rotating the source domain data by 35° and randomly removing half of the data of one class. The generated dataset is visualized in the leftmost column of Fig. 2.9.

Network Architecture and Training Configuration. The model was a network with three fully connected layers ($2 \xrightarrow{FC} 16 \xrightarrow{FC} 16 \xrightarrow{FC} 2$) as the model. We train it

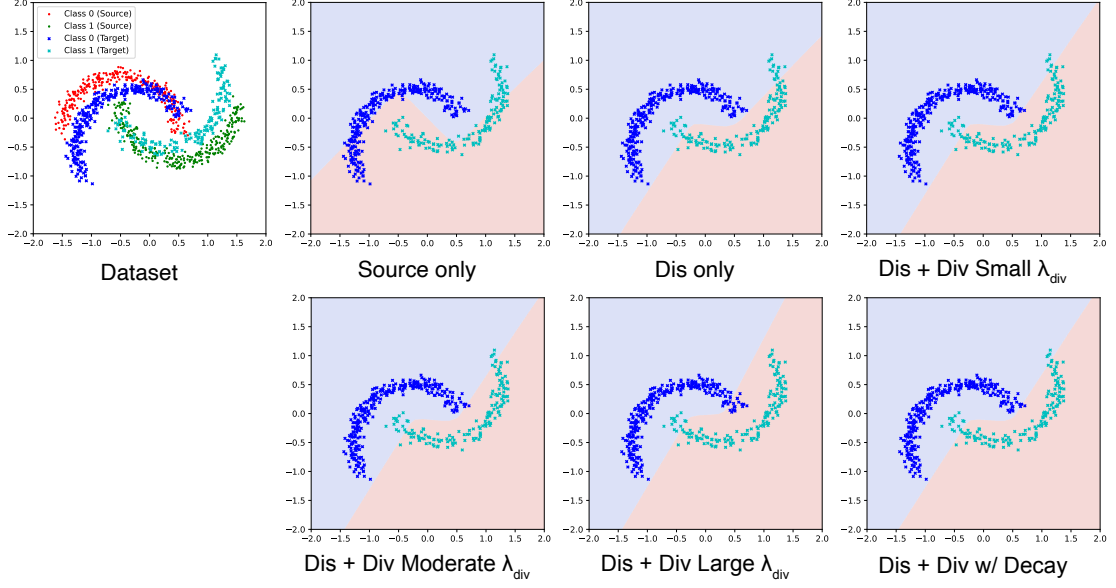


Figure 2.9: **Visualization of the experimental results on SHOT-IM.**

with source domain data by using SGD with a learning rate of $1e-2$, a momentum parameter of 0.9, and a batch size of 64 for 100 epochs. We treated this source-trained model as a source-only one. We subsequently trained the model with the target domain data based on the SHOT-IM loss using SGD with a learning rate of $1e-3$, a momentum parameter of 0.9, and a batch size of 64 for 500 epochs. We trained the Dis-only model without the diversity loss of SHOT-IM. For the training the Div+Dis model, we set λ_{div} to 0.1, while for training the Div+Dis w/ Decay model, we set λ_{div} to $3.0 \cdot \mathcal{L}_{dis}$. We set the variance parameter of 2-D Gaussian perturbation used for the data augmentation to 0.05 in the experiment of Tab. 2.1, while we varied this parameter to obtain Fig. 2.2.

Detailed results of Tab. 2.1. For further analysis, we additionally experimented with Dis + Div w/o Decay by varying the weight of the diversity loss. Specifically, we set the weight λ_{div} to 0.05 for “small λ_{div} ”, 0.1 for “moderate λ_{div} ”, and 0.3 for “large λ_{div} ”. The visualized results for Tab. 2.1 are shown in Fig. 2.9. We can see

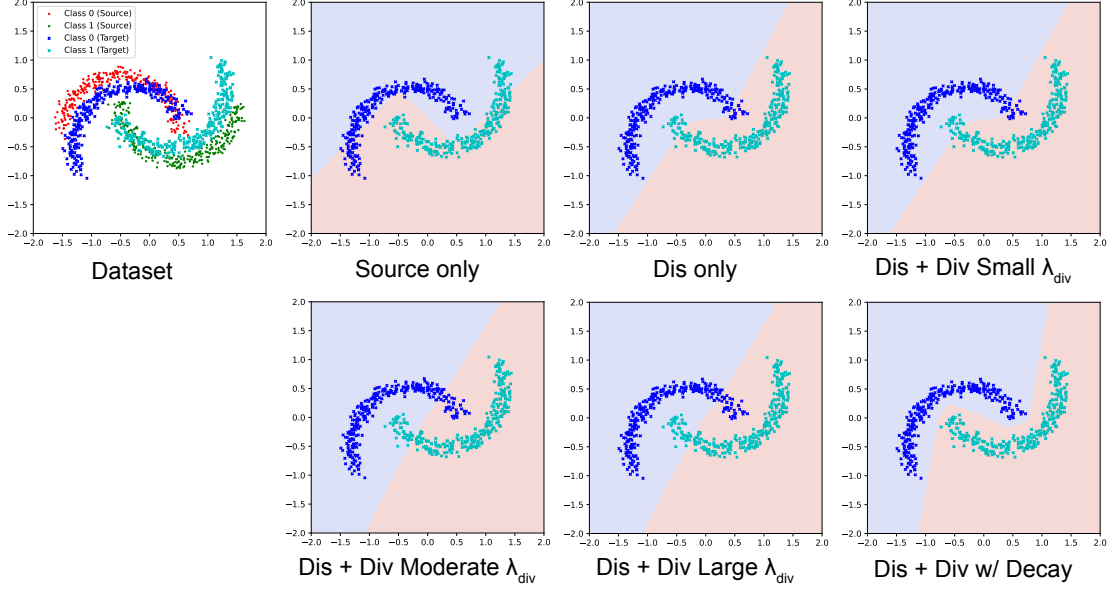


Figure 2.10: **Visualization of the experimental results on AaD.**

that a better decision boundary is obtained when the model is trained with both discriminability and diversity losses.

Additional results using AaD.

Dataset. For the dataset construction, we used the same setting in [190]. We constructed the source domain data with two inter-twinning moons, in which each moon contained 300 samples, and generated the target domain data by rotating the source domain data by 30° . The generated dataset is illustrated in the leftmost column of Fig. 2.10.

Network Architecture and Training Configuration. We used the same network architecture and source training setup as in the experiment with SHOT-IM. We trained the model with the target domain data based on the AaD loss by using SGD with a learning rate of $1e-2$, momentum parameter of 0.9, and batch size of 64 for 500 epochs. We trained the Dis-only model without the diversity loss

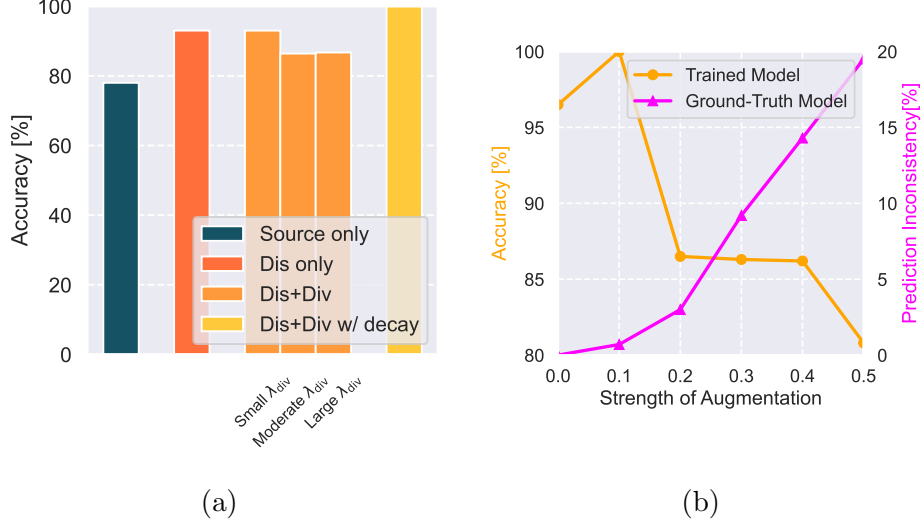


Figure 2.11: **Experimental Results on Synthetic Data with AaD.** (a) Model accuracy w/ and w/o discriminability and diversity losses. (b) Model accuracy and prediction inconsistency of ground-truth model against the strength of augmentation.

of AaD. For training the Div+Dis models, we conducted experiments with several coefficient parameters λ_{div} , namely, we set λ_{div} to 0.1 for small λ_{div} , 0.75 for moderate λ_{div} , and 1.5 for large λ . For training the Div+Dis w/ Decay model, we set the coefficient parameter $\lambda_{div} = 1.5 \cdot \mathcal{L}_{dis}$. We set the parameter of the 2-D Gaussian perturbation used for the data augmentation to 0.1 in the experiments, while we varied this parameter in the analysis.

Results. The qualitative and quantitative results are shown in Fig. 2.10 and Fig. 2.11. They were mostly consistent with the result of the SHOT-IM experiment. As shown in Fig. 2.10 and Fig. 2.11a, the Dis + Div w/ Decay model performed the best, which is in line with the training objective (2.3) In addition, as shown in Fig. 2.11b, the model accuracy grew as the strength of the augmentation grew large to some extent, but if the strength grew too large, the model accuracy deteriorated as the prediction inconsistency of the ground-truth model increased. The

Table 2.6: **Analysis of Augmentation Training (Detailed version of Fig. 2.4b).**

Init. Aug	Aug Training	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw
Random	✓	60.1	78.3	81.8	70.3	80.2	79.4
		60.1	78.6	82.0	69.8	80.2	79.9
AutoAugment	✓	60.5	79.3	82.1	69.5	79.3	79.6
		60.7	78.9	82.0	69.9	79.5	79.7

Init. Aug	Aug Training	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg.	ΔAaD
Random	✓	66.6	57.9	82.5	73.8	60.5	86.1	73.1	+ 0.4
		67.1	58.3	82.1	74.1	60.6	86.4	73.3	+ 0.6
AutoAugment	✓	67.6	58.2	82.2	74.3	60.7	86.3	73.3	+ 0.6
		67.1	58.8	82.3	74.2	61.3	86.4	73.4	+ 0.7

only difference was that in AaD, simply adding the diversity loss did not produce an improvement, which was the reproduced results originally reported [190]. We guess that this was because it is inherently important to decay the coefficient of the diversity loss along with the training objective (1). The gradient magnitude of the diversity loss of AaD does not decrease even as the training progresses, so in the latter stages of training, it tends to function excessively as a constraint. Therefore, it is important to explicitly implement the decay mechanism to reduce the effect of the constraint. On the other hand, simply adding a marginal entropy minimization loss in the case of SHOT-IM did not cause large performance drops. We guess this might be because the diversity loss of SHOT-IM implicitly has some decaying mechanism.

2.7.3 Additional Results

Here, we describe the results of the experiments in more detail and provide additional analyses we could not include in the main paper.

Detailed Results of Analysis.

Table 2.7: **Analysis of the coefficient parameter λ_F**

λ_F	Per-class Avg.
0.0 ($\mathcal{L}'_F$ only)	87.6
0.2 (Our setting)	88.4
0.5 (Simply combining $\mathcal{L}_F$ and $\mathcal{L}'_F$)	87.8
1.0 ($\mathcal{L}_F$ only)	84.6

Ablation study of Augmentation Training. Note that we modified a few hyperparameters ($\eta^{\text{aug}} = 4\text{e-}3$, $\lambda_{\mathcal{A}} = 0.25$) when we conducted the analysis with randomly initialized data augmentation. Tab. 2.6 shows the detailed results of the ablation study on augmentation training. We can see that the performance with augmentation training is better in 10 out of 12 pairs in the randomly initialized case and 8 out of 12 pairs in the AutoAugment initialized case, demonstrating the validity of our technique.

Further Analysis of Our Method.

Analysis of the coefficient parameter λ_F . As we mentioned in Sec. 2.4.1, we found in the early study that the heavy augmentations sampled from $\mathcal{A}$ degrade performance, particularly when we use VisDA2017. We can mitigate this issue by using the training loss calculated only with weakly augmented samples. Here, λ_F controls the balance between the losses of the heavily augmented samples ($\mathcal{L}_F$) and the weakly augmented samples ($\mathcal{L}'_F$).

In this analysis, we confirmed the validity of using both $\mathcal{L}_F$ and $\mathcal{L}'_F$ by varying the value of λ_F . We used VisDA2017 for the evaluation, and the other settings were the same as described in Sec. 2.5.1.

The results are shown in Tab. 2.7. When we finely tuned λ_F , the performance of our method reached 88.4%. Using only heavy data augmentations ($\lambda_F = 1.0$) resulted in lower accuracy than when not using them ($\lambda_F = 0.0$). This indicates

that using complicated data augmentations does not always improve accuracy. Simply combining $\mathcal{L}_F$ and $\mathcal{L}'_F$ ($\lambda_F = 0.5$) increases accuracy, but by adjusting λ_F appropriately, we can obtain an even greater improvement. Note that, on Office-31 and Office-Home, our method reached sufficient accuracy without any special tuning, so we set λ_F to 0.5 in these cases.

Chapter 3

Flexible Source-free Domain

Generalization via Domain

Prompt-Discriminator Collaborative

Learning

3.1 Introduction

The generalizability of deep neural networks (DNNs) is of great importance in achieving trustworthy AI models. DNNs can demonstrate high recognition accuracy on data from the same distribution as the development phase; however, their performance severely degrades when the data distribution is significantly shifted. One of the pragmatic directions to address this issue is domain generalization (DG), which is an extended framework of domain adaptation (DA). In contrast to DA, which assumes a certain target domain, the goal of DG is to build a generalizable model to unseen target domains based on multiple labeled source domains. Various approaches have been proposed for DG, such as domain align-

ment [123, 96, 100], meta-learning [94, 10, 41], data augmentation [177, 152, 188].

Another emerging aspect regarding the generalizability of DNNs is large-scale pre-trained models, which are often named as “foundation models. In the field of computer vision, large-scale vision-language models (VLMs) have attracted much attention. VLMs built upon web-scale image-language pair data have been reported to achieve high recognition accuracy for several downstream tasks without task-oriented training data [139, 81]. Furthermore, VLM can improve the recognition performance for the target tasks by using few-shot training samples and a small number of trainable parameters [209, 208, 49, 197], or even without using any training samples by using external large models [118, 134, 84].

This notable progress in VLMs has led to the proposal of a new paradigm for DG, called source-free domain generalization (SFDG) [29, 126, 162, 195]. In SFDG, a model must be built without using any actual data from either the source domains or the target domains. This new problem setup successfully reduces the two difficulties in the previous setup: the selection of suitable source domains and the collection of the annotated data from those domains. While this problem is seemingly difficult to solve, existing SFDG methods [29, 126, 162, 195] feasibly work it out by using large-scale pre-trained VLMs (e.g., CLIP [139]). For example, domain unified prompt representation generator (DUPRG) [126] (Fig. 3.1 (a)) prepares a domain-unified representation by aggregating the CLIP encoded features of various hand-crafted domain-specific prompts and classifies based on the similarity score between the domain-unified representation and the input image features. PromptStyler [29] (Fig. 3.1 (b)) synthesizes diverse domain samples by a self-supervised prompt learning with a CLIP pre-trained text encoder. Then, PromptStyler utilizes those samples for training a classifier, which can be readily

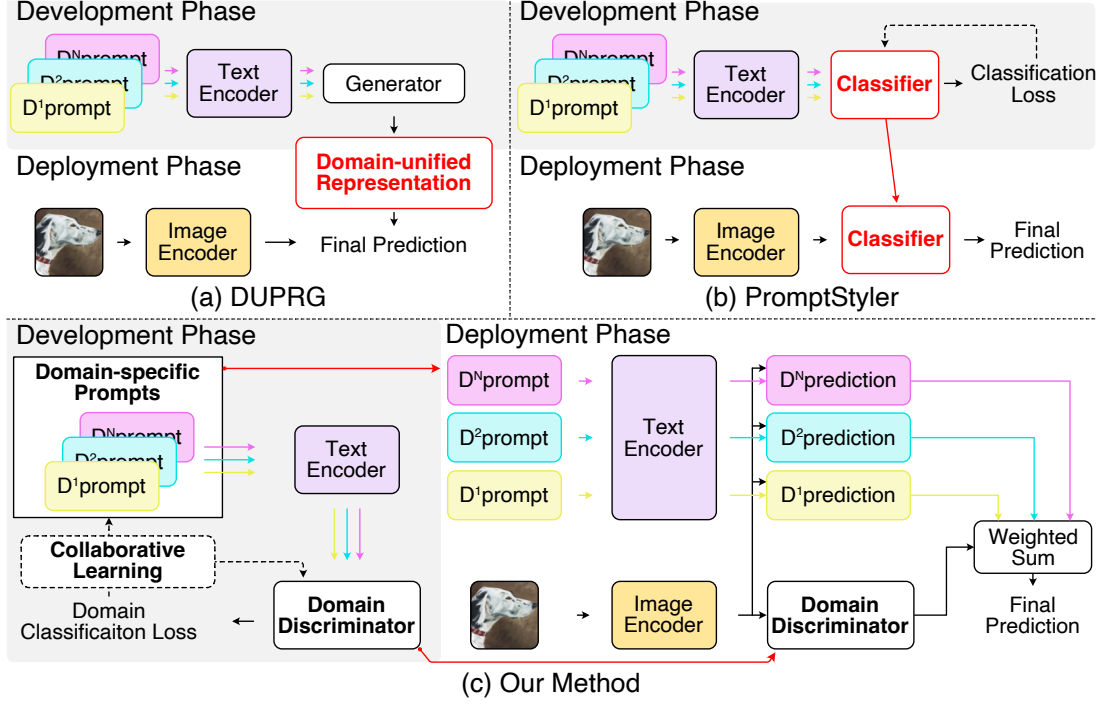


Figure 3.1: **Comparison between prior source-free domain generalization (SFDG) methods and our method.** (a, b) Prior SFDG methods employ a domain-shared configuration (e.g., representation/classifier), which undermines flexibility for the input image domain. (c) Our method achieves flexible predictions by a set of domain-specific prompts to yield various domain-specific predictions and a domain discriminator to appropriately aggregate the predictions for the input image.

applicable to image classification.

However, these techniques do not fully exploit the potential of VLMs for DG due to structural constraints that use the domain-shared configuration (e.g., representation/classifier) (Fig. 3.1 (a, b)). It is intuitive to obtain higher accuracy by flexibly changing the configuration to match the input’s domain rather than fixing the same configuration for all domains. Indeed, some studies [211, 116] have achieved favorable results on conventional DG in a similar concept. To this end, we need to prepare the following two components in a source-free manner: 1) a va-

riety of domain-specific configurations and 2) a selector to choose the appropriate configuration for the target domain.

We propose domain prompt-discriminator collaborative learning (Fig. 3.1 (c)), a simple yet effective method that realizes domain-flexible prediction by training the following two components: 1) a set of domain-specific prompts and 2) a domain discriminator. The domain-specific prompts are trained to comprise a group of text prompts representing various domains. These prompts are utilized to obtain domain-specific predictions, which are the prediction candidates assuming that the input data belongs to the respective domain. Meanwhile, the domain discriminator is trained to estimate the domain of the input data. Due to this training, the domain discriminator can choose appropriate predictions among a group of domain-specific predictions. Notably, we demonstrate that these two pieces of training can be consolidated into one simple training, domain classification learning, only with the CLIP text encoder. In the deployment phase, thanks to the joint vision-language pre-trained representation of CLIP, the domain discriminator can estimate the domain by inputting the CLIP-encoded image feature. The final prediction is obtained by weighting the domain-specific predictions by the domain discriminator outcome. Owing to our method design, we can easily incorporate a simple test-time adaptation that updates only the domain discriminator’s parameters to alleviate the image-text modality gap [104] and improve the performance further.

Our contributions in this chapter are summarized as follows:

- We propose domain prompt-discriminator collaborative learning and demonstrate for the first time the effectiveness of domain-flexible predictions in

source-free domain generalization.

- Our method of training domain-specific prompts and domain discriminator can be implemented through a simple domain classification learning using only the CLIP text encoder.
- Our method design also provides a simple and effective test-time adaptation technique that further boosts the domain generalization performance.

3.2 Related Works

3.2.1 Domain Generalization (DG)

DG is a problem that aims to train a model on multiple source domains to achieve satisfactory recognition performance on unseen target domains. As described in the survey [207], numerous methods have been proposed for DG, such as domain alignment [123, 96, 100], meta-learning [94, 10, 41], data augmentation [177, 152, 188], ensemble learning [211, 116, 20], self-supervised learning [19, 181], learning disentangled representations [95, 22, 78], regularization strategies [77, 86, 21], and reinforcement learning [210, 62, 89]. In recent years, some studies [182, 135] have pointed out that conventional DG is data inefficient in that it requires training datasets from multiple domains, and they have proposed single-source domain generalization, where they train the model with only single-domain training data.

3.2.2 Source-free Domain Generalization (SFDG)

SFDG goes one step further from single-source domain generalization and aims to train a model that generalizes to unseen target domains without using any source

domain data. This problem setup is different from source-free domain adaptation (SFDA) [47, 97, 101] in that SFDA adapts the model to a certain target domain while SFDG does not define a specific target, which brings out efficient adaptivity to the various domains. Existing SFDG methods [29, 126, 162, 195] make effective use of the zero-shot generalizability of vision-language pre-trained models (VLMs) to address the absence of source training data. For instance, PromptStyler [29] synthesizes diverse domain examples in the joint vision-language feature space with a CLIP text encoder. They are utilized to train a linear classifier, which can be readily applied to image classification with the CLIP image encoder. DUPRG [126] constructs a domain-unified representation from various hand-crafted domain-specific prompts using a cosine auto-encoder or mean pooling.

Despite these pioneering works, they do not fully exploit the potential of VLMs. This is because they employ a domain-shared configuration (i.e., classifier and representation) for data from any domain, which leads to decreased flexibility for image inputs from various domains. In contrast, our method trains the following two components for domain-flexible prediction: (1) domain-specific prompts that serve as the basis of the prediction group that covers various domains, and (2) a domain discriminator that chooses favorable predictions for the input.

3.2.3 Vision-Language Pretrained Models (VLMs)

Modern VLMs pre-trained with massive image-text pairs [139, 81] demonstrate considerable zero-shot generalizability on downstream vision tasks. To improve the performance of VLMs for downstream tasks, few-shot training approaches, such as learnable text prompts [209, 208, 17] or adapter modules [49, 197], have been proposed. Some domain adaptation studies [51] employ these techniques to achieve

a lightweight adaptation to the target domain. Very recent studies [118, 134, 143, 84, 172] have improved recognition accuracies even without any task-specific data by utilizing external large models such as large-language models [169, 16] and diffusion models [142]. We exploit the high generalizability of VLMs to solve SFDG effectively.

3.2.4 Test-time Adaptation (TTA)

TTA is an emerging paradigm that aims to update a pre-trained model with test samples on the fly to adapt the model to test data distributions. Test-time training approach introduces self-supervised auxiliary tasks (e.g., rotation prediction [159], contrastive learning [109]) to update the model during both development and deployment phases. Wang et al. proposed Tent [178], a simple and efficient TTA that only updates the batch-normalization parameters of the model based on the entropy minimization loss. Test-time prompt tuning (TPT) [154] leverages VLMs and only updates learnable text prompts with a few parameters while freezing a large number of VLM parameters. Our method incorporates TTA techniques and only updates a small part of the model, i.e., the domain discriminator, to alleviate the image-text modality gaps and boost the overall performance.

3.3 Preliminaries

3.3.1 Problem Setup

The conventional DG aims to train models that generalize to unseen target domains using multiple labeled source domains. We train a model $F : \mathcal{X} \rightarrow \mathcal{Y}$ using a set of source domains $\mathcal{D}_S = \{D_{S_m}\}_{m=1}^M$ and evaluate the trained model on unseen

target domains $\mathcal{D}_T = \{D_{T_n}\}_{n=1}^N$. Note that we cannot access the target domains $\mathcal{D}_T$ during the development phase. SFDG imposes an additional, more challenging restriction where we do not have access to the source domains $\mathcal{D}_S$ at all. The only information we are given is about the task information to be solved. In this work, we focus on the image classification task. In this regard, we are only given the set of class names $\{\langle \text{class} \rangle_y\}_{y=1}^C$ that we want to discriminate. Following [29, 126], we use CLIP [139] to address this problem.

3.3.2 Contrastive language-image pre-training (CLIP)

CLIP, consisting of text and image encoders (E_{text} and E_{image}), is trained on web-scale text-image pair data and demonstrates strong zero-shot generalizability. CLIP predicts the class of the input image x using the feature similarity scores between the image and the hand-crafted text prompts t_y of class labels $y \in \{1, 2, \dots, C\}$ (e.g., ‘a photo of a $\langle \text{class} \rangle_y$ ’). Formally, we can obtain the prediction as

$$y^* = \arg \max_y \frac{\exp(\text{sim}(E_{\text{image}}(x), E_{\text{text}}(t_y))/\tau)}{\sum_{y'=1}^C \exp(\text{sim}(E_{\text{image}}(x), E_{\text{text}}(t_{y'}))/\tau)}, \quad (3.1)$$

where $\text{sim}(\cdot)$ is a cosine similarity score function and τ is a learned temperature parameter by CLIP training.

Prompt tuning approaches [209, 85] further improve the performance by training the text prompts on the word embedding space instead of using hand-crafted text prompts. These methods optimize learnable text prompts $t_y = [l_1, l_2, \dots, l_K, c_y]$ with learnable vectors $\{l_k\}_{k=1}^K$ and a class embedding vector c_y with a few training samples.

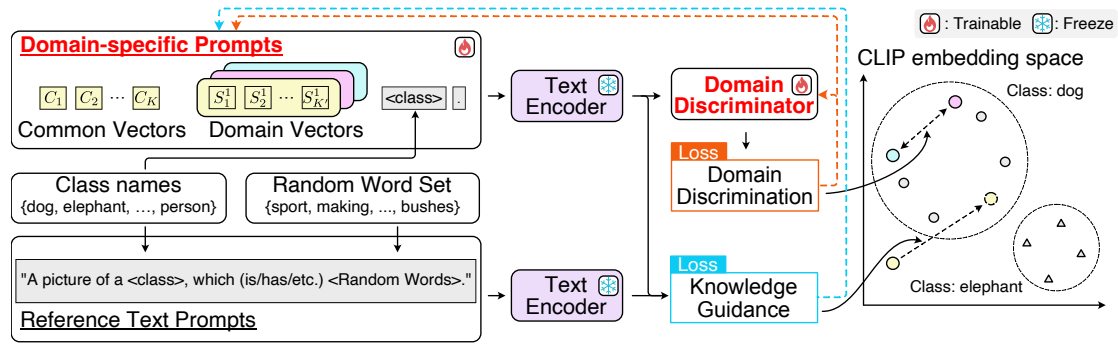


Figure 3.2: **Development Phase: Training Domain-specific Prompts and Domain Discriminator.** We jointly train domain-specific prompts and a domain discriminator through a simple domain classification task, which encourages the domain-specific prompts to represent various domains and equips the domain discriminator with the ability to recognize the domain of the input feature. We also employ a knowledge guidance loss to ensure CLIP-based classification ability.

3.4 Our Method: Development Phase: Domain Prompt-Discriminator Collaborative Learning

The overview of our method is shown in Fig. 3.2 and Fig. 3.3. In the development phase (Fig. 3.2), we train two components: 1) a set of domain-specific prompts and 2) a domain discriminator. The former is trained to comprise a group of diverse text prompts with domain diversity, while the latter is trained to estimate the domain of its input. This training enables us to perform domain flexible prediction at the deployment phase (Fig. 3.3). Specifically, the domain-specific prompts generate per-domain predictions based on each prompt, and the domain discriminator selects appropriate predictions from those domain-specific predictions.

3.4.1 Development Phase: Training Domain-specific Prompts and Domain Discriminator

Domain-specific Prompts. Domain-specific prompts are trained to comprise a group of prompts representing various domains to support input from any domain. The prompt of domain m and class y is composed of three parts: learnable domain vectors $\{S_1^m, \dots, S_{K'}^m\}$, learnable common vectors $\{C_1, \dots, C_K\}$, and a class vector c_y . The domain vectors and the class vector are selected according to domain and class, whereas the common vectors are used universally for any classes and domains. The domain vectors serve to represent words that correspond to styles or environments of images, such as “painting” or “sketch” while common vectors serve to express parts that are commonly used to make a sentence (e.g., “a photo of a ...”). To sum up, the domain-specific prompt of domain m and class y is constructed as $t_{m,y} = [C_1, \dots, C_K, S_1^m, \dots, S_{K'}^m, c_y]$.

Domain Discriminator. The role of the domain discriminator D is to output a probability distribution representing which domain the input data belongs to. This is utilized in the deployment phase to choose favorable predictions from the prediction candidates based on each domain-specific prompt. In the development phase, D is concatenated with frozen CLIP text encoder E_{text} and trained to discriminate each domain-specific prompt given as input. The trained D accepts image feature inputs and can estimate the domain of the input image in the deployment phase because the CLIP embedding space is shared between image and text.

Training Loss. The training loss of our method is composed of two parts: a domain discrimination loss $\mathcal{L}_{\text{domain}}$ and a knowledge guidance loss $\mathcal{L}_{\text{Kg}}$.

The domain discrimination loss $\mathcal{L}_{\text{domain}}$ is the main loss for collaboratively

training the domain-specific prompts and the domain discriminator. Specifically, $\mathcal{L}_{\text{domain}}$ can be calculated as

$$\mathcal{L}_{\text{domain}} = \mathbb{E}_{t_{m,y}} [\ell_{ce}(D(E_{\text{text}}(t_{m,y})), m)], \quad (3.2)$$

where ℓ_{ce} is a cross entropy loss between prediction $D(E_{\text{text}}(t_{m,y}))$ and the domain label m . This simple classification loss has a synergistic effect on the training of both the domain discriminator and the domain-specific prompts. This loss equips the domain discriminator with the ability to estimate the domain of the input image in the deployment phase. Meanwhile, the domain-specific prompts are trained so that the domain discriminator can easily distinguish them, which yields diverse domain prompts. Hence, this approach makes it possible to simultaneously and efficiently prepare diverse domain-specific predictions and a domain prompt selection module.

However, only training with $\mathcal{L}_{\text{domain}}$ does not yield satisfactory results. This is because $\mathcal{L}_{\text{domain}}$ does not bring the textual information necessary for the CLIP-based classification to the domain-specific prompts. Inspired by ProText [84], we introduce a knowledge guidance loss $\mathcal{L}_{\text{Kg}}$ [193] using reference text prompts $\hat{t}_y$ to satisfy this requirement in a source-free manner. ProText trains learnable text prompts from the class-specific descriptions and captions as reference text prompts generated from large-language models [16, 169]. In addition, we incorporate the insight of WaffleCLIP [143]. WaffleCLIP shows that class-specific descriptions constructed with random words are as effective as LLM-generated prompts in zero-shot CLIP classification. In our method, we replace LLM-generated reference text prompts of ProText with random-word-generated prompts. This can successfully

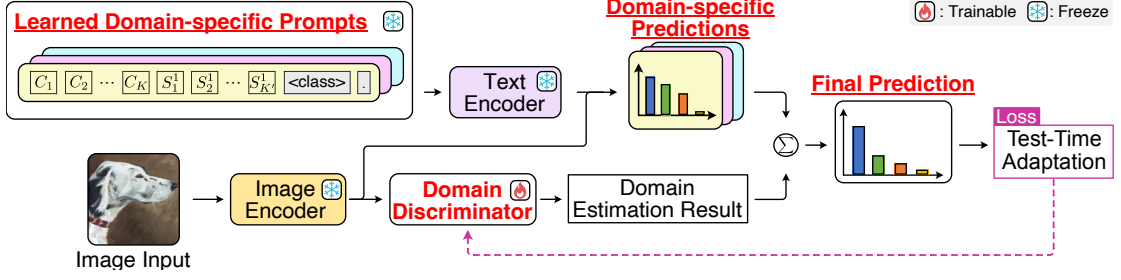


Figure 3.3: **Deployment Phase: Inference and Test-time Adaptation.** We obtain domain-specific predictions from each learned prompt, and then we aggregate them by weighted averaging with the domain estimation result of the domain discriminator. The recognition accuracies can be enhanced by updating only the domain discriminator’s parameters based on test-time adaptation loss.

reduce the time of engineering LLM prompts and the cost of LLM service usage.

Formally, our knowledge guidance loss $\mathcal{L}_{Kg}$ is calculated as

$$\mathcal{L}_{Kg} = \mathbb{E}_{t_{m,y}, \hat{t}_y} \|E_{\text{text}}(t_{m,y}) - E_{\text{text}}(\hat{t}_y)\|_2^2 \times C_{Kg}, \quad (3.3)$$

where C_{Kg} is a scaling parameter to control the size of the mean squared loss. We fixed C_{Kg} to 1000 following the ProText original settings [84].

In summary, the total training loss $\mathcal{L}_{\text{total}}$ of our method is calculated as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{Kg} + \lambda_{\text{domain}} \mathcal{L}_{\text{domain}}, \quad (3.4)$$

where, λ_{domain} is a hyper-parameter to control the training weight of $\mathcal{L}_{\text{domain}}$.

3.4.2 Deployment Phase: Inference and Test-time Adaptation (TTA)

We ensemble the domain-specific predictions based on each learned prompt with the domain discriminator outcome for the image input. Specifically, we obtain the

logits of the domain-specific class likelihood of image $x^{(i)}$ as

$$l_{m,y}^{(i)} = \text{sim}(E_{\text{image}}(x^{(i)}), E_{\text{text}}(t_{m,y})), \quad (3.5)$$

and we estimate which domain the image belongs to as $\mathbf{w} = \sigma(D(E_{\text{image}}(x^{(i)})))$, where $\sigma(\cdot)$ denotes a softmax function. Then, using $l_{m,y}$ and $\mathbf{w}$, we obtain the final class prediction as

$$\hat{y}^{(i)} = \arg \max_y \sum_{m=1}^M w_m l_{m,y}^{(i)}. \quad (3.6)$$

However, we found that directly using the domain discriminator as it was trained in the development phase could not fully exploit the potential of our domain-flexible prediction scheme. This is because the domain discriminator is trained only on the text modality, not on the image modality. It suffers from the modality gap issue between image and text [104], and it cannot satisfactorily estimate the domain from the image feature input. Fortunately, this can be easily alleviated by our method design. We introduce a simple TTA technique that only updates the parameters of the domain discriminator based on the image inputs. Similar to SFDA, we cannot access the labeled source data in TTA. Based on our findings in Chapter 2, we design the TTA loss to have both discriminability and diversity losses. Actually, we employed the mutual information

maximization [74, 102] and calculated the TTA loss as

$$\begin{aligned}\mathcal{L}_{\text{TTA}} = & -\frac{1}{B} \sum_{i=1}^B \sum_{y=1}^C p_y^{(i)} \log p_y^{(i)} \\ & + \lambda_{\text{TTA}} \sum_{y=1}^C \left(\frac{1}{B} \sum_{i=1}^B p_y^{(i)} \right) \log \left(\frac{1}{B} \sum_{i=1}^B p_y^{(i)} \right),\end{aligned}\quad (3.7)$$

where $p_y^{(i)} = \exp(\frac{\sum_{m=1}^M w_m l_{m,y}^{(i)}}{\tau}) / \sum_{y'}^C \exp(\frac{\sum_{m=1}^M w_m l_{m,y'}^{(i)}}{\tau})$, B represents the batch size, and λ_{TTA} is a hyper-parameter.

3.5 Experiments

3.5.1 Setups

We follow the evaluation setups as described in [29].

Datasets. We used four domain generalization benchmark datasets. PACS [95] is a small-scale benchmark consisting of seven categories and four domains (Art Painting, Cartoon, Photo, Sketch). VLCS [46] is a small-scale benchmark consisting of four distinct datasets (Caltech101, LabelMe, SUN09, Pascal VOC) and includes five categories. Office-Home [176] is a moderate-scale benchmark consisting of 65 categories and four domains (Art, Clipart, Product, Real World). DomainNet [129] is a large-scale benchmark consisting of 345 categories and six domains (Clipart, Infograph, Painting, Quickdraw, Real, Sketch).

Compared Methods. We compare our method with the zero-shot CLIP baselines (ZS-CLIP) [139] and the prior SFDG methods, DUPRG [126] and Prompt-Style [29]. ZS-CLIP includes two variants where ZS-CLIP (C) uses a single word, “<class>,” and ZS-CLIP (PC) uses a more detailed text template, “a photo of a

$\langle \text{class} \rangle$,” as their text prompts. The paper [126] introduced two types of implementations. DUPRG-MP aggregates domain-specific prompt representations by Mean-Pooling, and DUPRG-CAE trains a Cosine AutoEncoder to aggregate the prompt representations.

Implementation Details. We employ CLIP [139] as the pre-trained vision language model. We conduct experiments with three different image encoder backbones: ResNet-50 (RN50), ViT-B/16, and ViT-L/14. With regard to the text encoders, we use the one jointly trained with the corresponding image encoders. The dimension of the text and image feature dimension D_{CLIP} is set to 1024, 512, and 768 when we use the ResNet-50, ViT-B/16, and ViT-L/14 backbones, respectively. Note that we freeze the CLIP model parameters throughout the experiments.

In the development phase, we train the other trainable parameters using the AdamW optimizer [114]. We set the number of domain-specific prompts M to 200 except for VLCS cases to 40, the length of common vectors K to 4, and the length of domain vectors K' to 4. The dimension of each learnable vector D_{word} is 512. The domain discriminator is MLP with two fully connected layers ($\text{FC}(D_{\text{CLIP}}, 128) \rightarrow \text{ReLU} \rightarrow \text{FC}(128, M)$). Regarding the optimizer’s parameters, we set the learning rate to 0.03, the weight decay to 0.1, and the momentum parameters to (0.9, 0.98). We employ the cosine scheduler to control the learning rate. The batch size is set to 256. We generate the reference text $\hat{t}_y$ as follows: “a picture of a $\langle \text{class} \rangle_y$, which $\langle \text{verb} \rangle \langle \text{random words} \rangle$.”, where “ $\langle \text{verb} \rangle$ ” is randomly selected from “is” or “has” and “ $\langle \text{random words} \rangle$ ” consists of two to four words randomly selected from the pre-defined random word soup in [143]. In each training epoch, we generate N_{ref} reference texts for each class, where we set N_{ref} to 100 for PACS, 160 for VLCS, 20 for OfficeHome, and 5 for DomainNet. The total

number of training epochs is set to 50 for PACS, 80 for VLCS, 20 for OfficeHome, and 15 for DomainNet. The parameter λ_{domain} is set to 0.3.

In the deployment phase, we set the batch size to 256. When using the test time adaptation technique, we employ the AdamW optimizer with the learning rate set to $1\text{e-}3$, the momentum parameters set to $(0.9, 0.999)$, and the weight decay parameter set to $1\text{e-}2$. We set the test-time adaptation steps to 50, $\lambda_{\text{TTA}} = 1.0$ for PACS, VLCS, Office-Home, and $\lambda_{\text{TTA}} = 2.0$ for DomainNet. We describe the details of parameter settings and computational resources in Sec. 3.7.2.

3.5.2 Main Results

We conduct three different random seed runs for each benchmark task and report the averaged accuracies and standard errors of the mean. The results are summarized in Tab. 3.1. Our method achieved state-of-the-art results in almost all cases except two (OfficeHome with ResNet50 and PACS with ViT-B/16). Notably, our method outperformed the prior SFDG methods even without our test-time adaptation technique, which demonstrates that our method can learn more effective prompts for domain generalization. Furthermore, our test-time adaptation technique constantly improved the recognition accuracy, demonstrating its effectiveness. Looking at the results for individual datasets, although performance improvements on PACS were generally small, our method achieved significant gains on DomainNet. We attribute this to the fact that PACS is a relatively easy dataset where existing methods already achieve very high accuracy, leaving little room for further improvement. In contrast, DomainNet was the most challenging dataset in our experiments, allowing the proposed method to demonstrate substantial performance gains.

Table 3.1: **Main Results.** Our method is compared with state-of-the-art SFDG methods on four domain generalization benchmarks and evaluated on classification accuracy [%].

ResNet-50 with pre-trained weight from CLIP					
Method	PACS	VLCS	OfficeHome	DomainNet	Average
ZS-CLIP (C) [139]	90.6 $\pm$ 0.0	76.0 $\pm$ 0.0	68.6 $\pm$ 0.0	45.6 $\pm$ 0.0	70.2
ZS-CLIP (PC) [139]	90.7 $\pm$ 0.0	80.1 $\pm$ 0.0	72.0 $\pm$ 0.0	46.2 $\pm$ 0.0	72.3
DUPRG-MP [126]	93.0	82.5	75.3	48.1	74.7
DUPRG-CAE [126]	93.1	81.2	73.0	48.4	73.9
PromptStyler [29]	93.2 $\pm$ 0.0	82.3 $\pm$ 0.1	73.6 $\pm$ 0.1	49.5 $\pm$ 0.0	74.7
Ours w/o TTA	93.5 $\pm$ 0.1	83.3 $\pm$ 0.1	74.8 $\pm$ 0.1	49.9 $\pm$ 0.0	75.3
Ours w/ TTA	93.9 $\pm$ 0.1	83.6 $\pm$ 0.1	75.2 $\pm$ 0.0	50.3 $\pm$ 0.1	75.7

ViT-B/16 with pre-trained weight from CLIP					
Method	PACS	VLCS	OfficeHome	DomainNet	Average
ZS-CLIP (C) [139]	95.7 $\pm$ 0.0	76.4 $\pm$ 0.0	79.9 $\pm$ 0.0	57.8 $\pm$ 0.0	77.5
ZS-CLIP (PC) [139]	96.1 $\pm$ 0.0	82.4 $\pm$ 0.0	82.3 $\pm$ 0.0	57.7 $\pm$ 0.0	78.6
DUPRG-MP [126]	96.5	83.1	82.5	59.2	80.3
DUPRG-CAE [126]	97.1	83.9	83.6	59.6	81.1
PromptStyler [29]	97.2 $\pm$ 0.1	82.9 $\pm$ 0.0	83.6 $\pm$ 0.0	59.4 $\pm$ 0.0	80.8
Ours w/o TTA	96.9 $\pm$ 0.0	83.4 $\pm$ 0.1	84.0 $\pm$ 0.1	60.4 $\pm$ 0.1	81.2
Ours w/ TTA	97.0 $\pm$ 0.1	83.6 $\pm$ 0.1	84.2 $\pm$ 0.1	60.5 $\pm$ 0.1	81.3

ViT-L/14 with pre-trained weight from CLIP					
Method	PACS	VLCS	OfficeHome	DomainNet	Average
ZS-CLIP (C) [139]	97.6 $\pm$ 0.0	77.5 $\pm$ 0.0	85.9 $\pm$ 0.0	63.3 $\pm$ 0.0	81.1
ZS-CLIP (PC) [139]	98.5 $\pm$ 0.0	82.4 $\pm$ 0.0	86.9 $\pm$ 0.0	64.0 $\pm$ 0.0	83.0
PromptStyler [29]	98.6 $\pm$ 0.0	82.4 $\pm$ 0.2	89.1 $\pm$ 0.0	65.5 $\pm$ 0.0	83.9
Ours w/o TTA	98.6 $\pm$ 0.0	82.6 $\pm$ 0.2	89.3 $\pm$ 0.0	66.3 $\pm$ 0.0	84.2
Ours w/ TTA	98.8 $\pm$ 0.0	82.9 $\pm$ 0.1	89.5 $\pm$ 0.1	66.6 $\pm$ 0.0	84.4

Regarding network architectures, the effectiveness of our method is particularly pronounced when using ResNet. It is likely that the ResNet architecture, being more vulnerable to domain shifts than Vision Transformers, benefited more from our method. Although the difference from existing methods appeared small in some dataset-network pairs, this is due to averaging results across domains; we can see that the proposed method significantly outperforms existing ones in specific domains. Please refer to Sec. 3.7.1. for details.

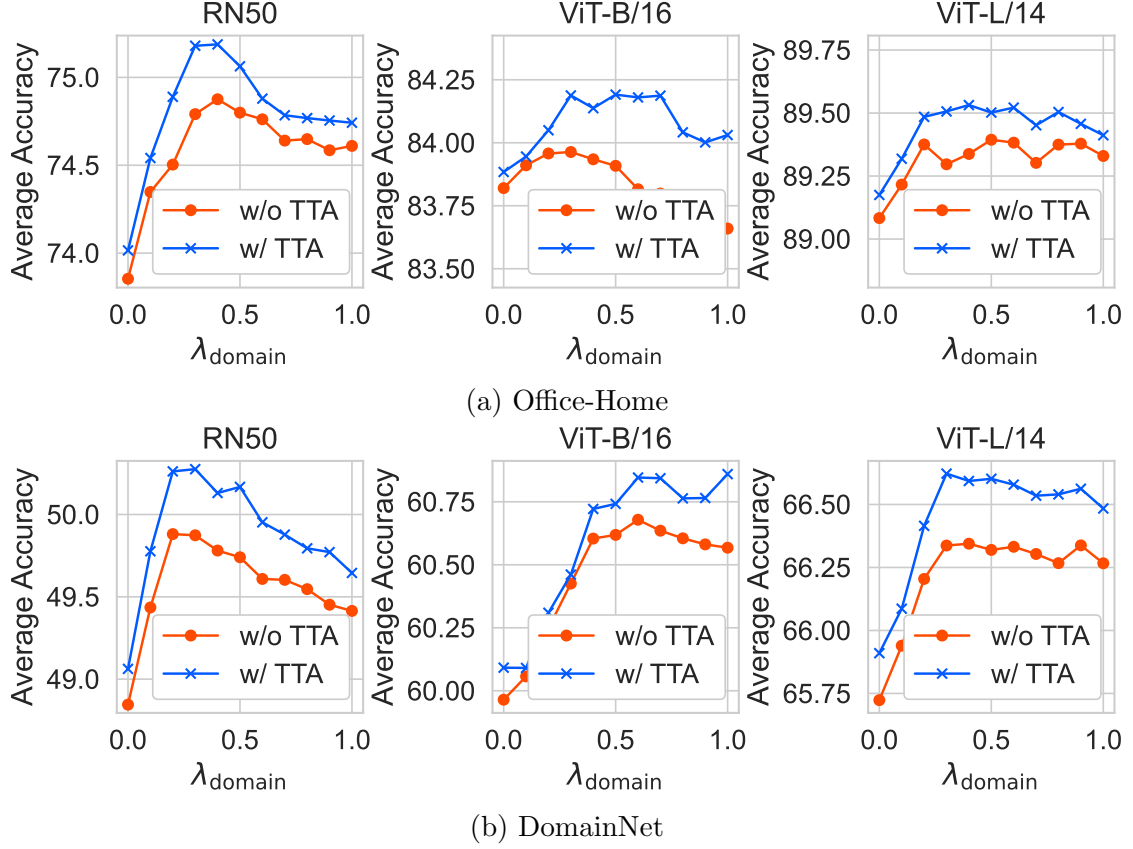


Figure 3.4: **Analysis of loss scaling parameter λ_{domain} .**

3.5.3 Analysis

Analysis of training domain-specific prompts. Our method aims to provide generalization ability to unseen target domains by preparing a variety of domain-specific prompts, to which the training loss $\mathcal{L}_{\text{domain}}$ contributes. To validate the training of domain-specific prompts, we analyzed the effect of $\mathcal{L}_{\text{domain}}$ by varying the magnitude of the coefficient parameter λ_{domain} . We conducted experiments on the Office-Home and DomainNet datasets with all three network architectures described in Sec. 3.5.1. The results are shown in Fig. 3.4. We found that the performance improved as λ_{domain} increased until $\lambda_{\text{domain}} = 0.3$ in all cases. The

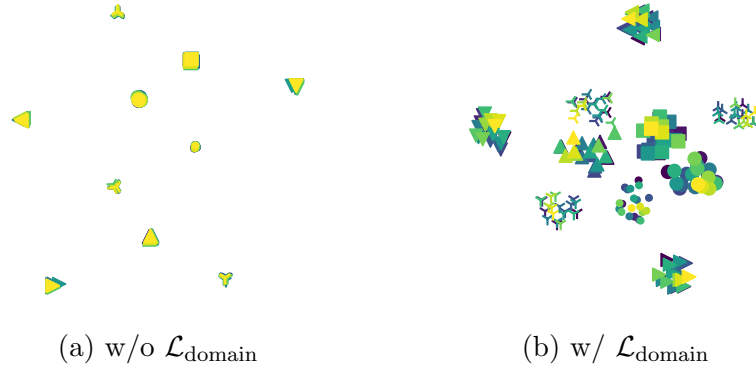


Figure 3.5: **t-SNE plots of text features of domain-specific prompts.** The shapes of the markers represent classes, while the colors of them represent domains.

Table 3.2: **Analysis of the position of the domain vectors**

Office-Home			
Position of Domain Vectors	ResNet-50	ViT-B/16	ViT-L/14
Former	74.6	84.5	89.5
Latter (Ours)	75.2	84.2	89.5
DomainNet			
Position of Domain Vectors	ResNet-50	ViT-B/16	ViT-L/14
Former	50.0	60.7	66.5
Latter (Ours)	50.3	60.5	66.6

results demonstrated the effectiveness of the loss $\mathcal{L}_{\text{domain}}$ and suggested the usefulness of training prompts in different domains. Note that decreasing λ_{domain} closely resembles training the prompts only with $\mathcal{L}_{\text{Kg}}$, which means the training becomes the simple combination of the existing CLIP-based classification methods, Pro-Text [84] and WaffleCLIP [143]. In the experimental results, the accuracy was the lowest when λ_{domain} was the smallest, indicating that merely using the existing methods is not sufficient for obtaining domain generalization.

We conducted a qualitative analysis of whether $\mathcal{L}_{\text{domain}}$ actually contributes to training diverse domain-specific prompts. Specifically, we visualized the t-SNE plots of the encoded features of the trained domain-specific prompts. For visibility,

Table 3.3: **Analysis of prediction aggregation method.**

Aggregation Method	Office-Home		
	ResNet-50	ViT-B/16	ViT-L/14
Eq-Aggre	74.4	84.1	89.3
Domain Discriminator	74.8	84.0	89.3
Domain Discriminator + TTA	75.2	84.2	89.5

Aggregation Method	DomainNet		
	ResNet-50	ViT-B/16	ViT-L/14
Eq-Aggre	49.7	60.3	66.2
Domain Discriminator	49.9	60.4	66.3
Domain Discriminator + TTA	50.3	60.5	66.6

we randomly sample combinations of 10 classes and 20 domains out of all prompts. The results of DomainNet with ResNet-50 are shown in Fig. 3.5. Without $\mathcal{L}_{\text{domain}}$, the prompt features were tightly clustered, undermining the prompt diversity. Conversely, with $\mathcal{L}_{\text{domain}}$, the prompt features were loosely clustered on each class, indicating that the prompts successfully learned diverse domains.

We conducted the analysis of the design of the domain-specific prompts by ablating the position of the domain vectors in the prompts using Office-Home and DomainNet benchmarks. In our default design, we put the position of the common vectors former and that of the domain vectors latter. Tab. 3.2 shows the results compared to those in the reversed order. Our default design of the domain-specific prompts performed slightly better, but we found that the difference was negligible.

Analysis of the aggregation method of domain-specific predictions. Our method aggregates the predictions based on each domain-specific prompt by weighting with the domain estimation outcome of the domain discriminator. To analyze our prediction aggregation method, we compared our method with the method that equally weights each domain-specific prediction (named Eq-Weight). The experimental results on Office-Home and DomainNet are summarized in Tab. 3.3. Our

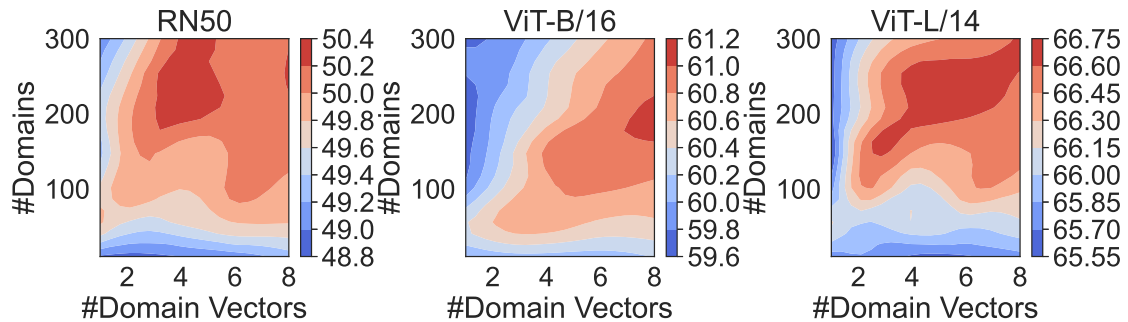


Figure 3.6: **Analysis of the representation capacity of domain-specific prompts.**

method without test-time adaptation (TTA) tended to perform better than Eq-Weight, but the improvement margin was not so large. As discussed in Sec. 3.4.2, the modality gap between image and text in CLIP embedding can be problematic, and our domain discriminator is no exception. Our domain discriminator is trained solely on text modality during the development phase and has never seen any image data until the deployment phase. Hence, directly using our domain discriminator as is failed to yield satisfactory results. Our TTA technique, which updates the parameters of the domain discriminator with image inputs, mitigated the modality gap problem, and indeed, our method with TTA greatly improved recognition accuracy.

Analysis of the representation capacity of domain-specific prompts. In addition to training a variety of domain-specific prompts with the training loss $\mathcal{L}_{\text{domain}}$, we must also provide enough capacity for the prompts to represent the various domains. The number of domains M and the number of domain vectors K' are key factors for sufficient domain representation. To analyze this, we varied M and K' and observed the prediction performance. We conducted the experiment with DomainNet. The results are visualized in Fig. 3.6. As can be seen, both M

Table 3.4: **Analysis of the prefix word of the reference text.**

Prefix word	Office-Home		
	ResNet-50	ViT-B/16	ViT-L/14
“picture” (Ours)	75.2	84.2	89.5
“image”	74.9	83.9	89.4
“photo”	74.9	83.8	89.4

Prefix Word	DomainNet		
	ResNet-50	ViT-B/16	ViT-L/14
“picture” (Ours)	50.3	60.5	66.6
“image”	50.1	60.3	66.4
“photo”	49.9	59.9	66.4

and K' should be large enough to achieve high recognition accuracy. Interestingly, we found that the accuracy degraded when either M or K' is small rather than when both are small. We consider the reason for this as follows. When K' is too small, the domain-specific prompts have limited capacity and cannot discriminately represent each of the large M domains sufficiently. On the other hand, when M is too small, the prompts have too much learning capacity to represent small M domains and learn unnecessary information that interferes with the predictions.

Analysis of the reference text. As described in Sec. 3.5.1, we used the prefix “a picture of a” in the reference text instead of the frequently used phrase, “a photo of a”. This aims to increase the effectiveness of our method by not including domain-related words in the prefix. We validated this by changing the prefix word (e.g., “photo”, “picture”) in the reference text with Office-Home and DomainNet benchmarks and showed the results in Tab. 3.4. We found that our method performed better when we used a word that contained fewer domain characteristics, such as “picture” and “image”, rather than a word that had more domain information, like “photo”. This result validated the effectiveness of the design of our reference text.

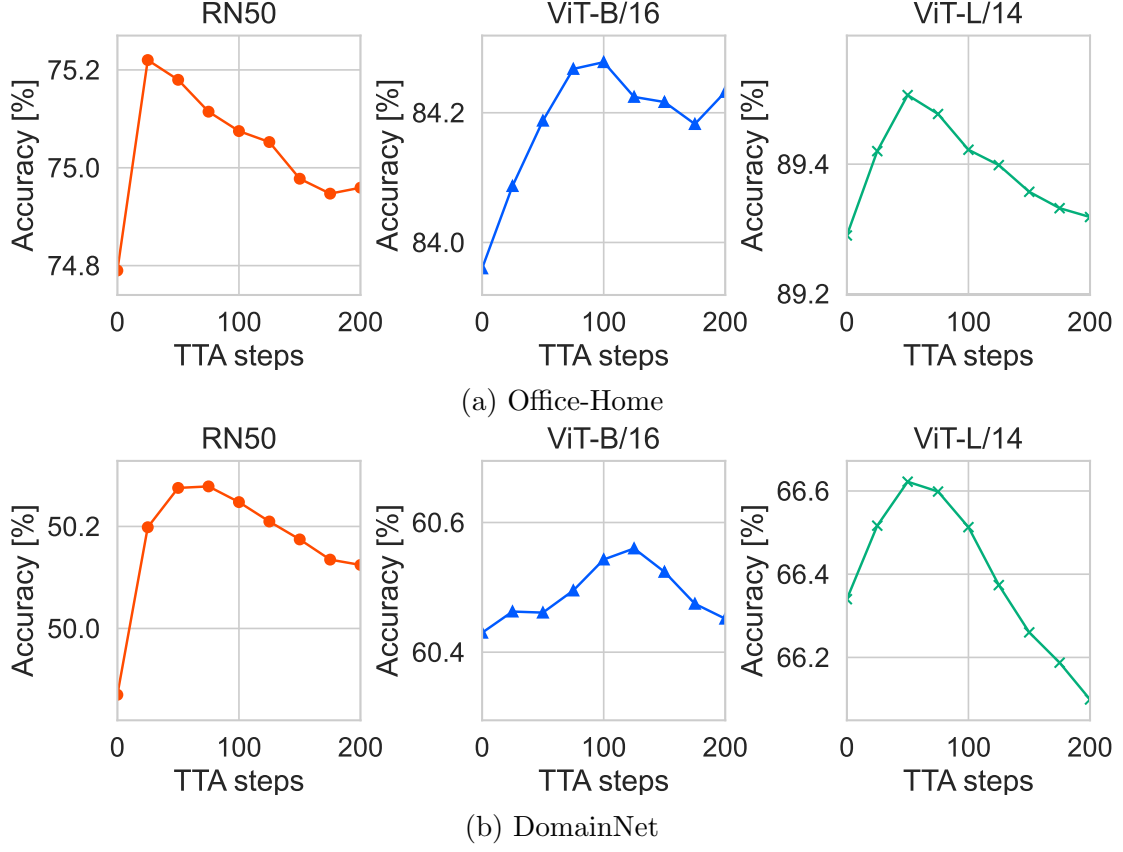


Figure 3.7: **Analysis of test-time adaptation steps.**

Analysis of the test-time adaptation (TTA). Finally, To analyze the applicability of our TTA technique to practical situations, we conducted an analysis of the number of TTA steps. The results of the TTA step analysis on OfficeHome and DomainNet are summarized in Fig. 3.7. The performance rapidly increased in small steps and then gradually decreased. This indicates that our TTA technique does not require long updates and is suitable for practical inference.

3.6 Conclusion and Limitations

In this work, we tackled the source-free domain generalization (SFDG) problem, where a model generalized to unseen target domains must be trained without using actual source domain data. We proposed Domain Prompt-Discriminator Learning, a novel SFDG method that differs from the existing approach of preparing a domain-shared configuration and greatly improves recognition accuracy by jointly training a variety of domain-specific prompts and a domain discriminator that allows domain-flexible predictions. In addition, we demonstrated that our method can be implemented in the very simple form of domain classification learning. We believe that our efficient methodology will encourage the future development of SFDG technology.

Limitations. Though our method demonstrates considerable progress in SFDG, there still remains a modality gap issue despite mitigating it to some extent through test-time adaptation. SFDG methods, including those previously proposed, do not use image data in the development phase, so they cannot inherently avoid the modality gap between image and text representations. To address this issue fundamentally, we need to bring image data into the training, but this would severely undermine the data efficiency of SFDG. A promising solution would be to incorporate generative models (e.g., diffusion models) to synthesize pseudo-image data. The fusion of source-free domain generalization and generative models is a potential future direction of our research that may be worth exploring.

Table 3.5: Per domain classification accuracy on PACS.

ResNet-50 with pre-trained weights from CLIP					
Method	Art Painting	Cartoon	Photo	Sketch	Avg.
ZS-CLIP (C) [139]	88.9±0.0	94.4±0.0	99.3±0.0	79.8±0.0	90.6
ZS-CLIP (PC) [139]	90.8±0.0	93.3±0.0	99.4±0.0	79.3±0.0	90.7
PromptStyler [29]	93.7±0.1	94.7±0.2	99.4±0.0	84.9±0.1	93.2
Ours w/o TTA	94.3±0.1	96.0±0.0	99.6±0.0	83.0±0.4	93.5
Ours w/ TTA	94.5±0.2	96.1±0.0	99.6±0.0	85.4±0.2	93.9
ViT-B/16 with pre-trained weights from CLIP					
Method	Art Painting	Cartoon	Photo	Sketch	Avg.
ZS-CLIP (C) [139]	96.4±0.0	98.9±0.0	99.9±0.0	87.7±0.0	95.7
ZS-CLIP (PC) [139]	97.2±0.0	99.1±0.0	99.9±0.0	88.2±0.0	96.1
PromptStyler [29]	97.6±0.1	99.1±0.1	99.9±0.0	92.3±0.3	97.2
Ours w/o TTA	97.7±0.0	99.1±0.0	99.9±0.0	90.9±0.1	96.9
Ours w/ TTA	97.8±0.0	99.2±0.0	99.9±0.0	91.1±0.2	97.0
ViT-L/14 with pre-trained weights from CLIP					
Method	Art Painting	Cartoon	Photo	Sketch	Avg.
ZS-CLIP (C) [139]	97.2±0.0	99.5±0.0	99.9±0.0	93.8±0.0	97.6
ZS-CLIP (PC) [139]	99.0±0.0	99.7±0.0	99.9±0.0	95.5±0.0	98.5
PromptStyler [29]	99.1±0.0	99.7±0.0	100.0±0.0	95.5±0.1	98.6
Ours w/o TTA	98.8±0.1	99.6±0.0	100.0±0.0	96.0±0.0	98.6
Ours w/ TTA	99.1±0.0	99.6±0.0	100.0±0.0	96.2±0.0	98.8

3.7 Appendix

3.7.1 Detailed Results

In this section, we present our main results in further detail (Tab. 3.1).

Results of PACS experiments (Tab. 3.5). Among the benchmarks used in our work, PACS resulted in the smallest performance gain from the compared methods. In particular, the results with the ViT-L/14 backbone only show an improvement of 0.3 points at most over ZS-CLIP (PC). This may be because PACS is too easy for CLIP pre-trained models, and the performance is almost saturated.

Results of VLCS experiments (Tab. 3.6). Compared to the PACS results,

Table 3.6: Per domain classification accuracy on VLCS.

ResNet-50 with pre-trained weights from CLIP					
Method	Caltech	LabelMe	SUN09	VOC2007	Avg.
ZS-CLIP (C) [139]	99.2±0.0	62.4±0.0	69.0±0.0	73.5±0.0	76.0
ZS-CLIP (PC) [139]	99.4±0.0	65.0±0.0	71.7±0.0	84.2±0.0	80.1
PromptStyler [29]	99.5±0.0	71.2±0.2	72.0±0.0	86.5±0.3	82.3
Ours w/o TTA	99.5±0.0	71.9±0.1	74.6±0.3	87.0±0.1	83.3
Ours w/ TTA	99.5±0.1	72.2±0.1	75.5±0.2	87.1±0.1	83.6
ViT-B/16 with pre-trained weights from CLIP					
Method	Caltech	LabelMe	SUN09	VOC2007	Avg.
ZS-CLIP (C) [139]	99.7±0.0	61.8±0.0	70.1±0.0	73.9±0.0	76.4
ZS-CLIP (PC) [139]	99.9±0.0	68.9±0.0	74.8±0.0	85.9±0.0	82.4
PromptStyler [29]	99.9±0.0	71.5±0.3	73.9±0.2	86.3±0.1	82.9
Ours w/o TTA	99.9±0.0	70.5±0.3	76.8±0.2	86.6±0.1	83.4
Ours w/ TTA	99.9±0.0	70.7±0.2	77.0±0.1	86.6±0.1	83.6
ViT-L/14 with pre-trained weights from CLIP					
Method	Caltech	LabelMe	SUN09	VOC2007	Avg.
ZS-CLIP (C) [139]	99.9±0.0	59.3±0.0	71.0±0.0	79.9±0.0	77.5
ZS-CLIP (PC) [139]	99.9±0.0	70.9±0.0	72.9±0.0	86.0±0.0	82.4
PromptStyler [29]	99.9±0.0	71.1±0.7	71.8±1.0	86.8±0.0	82.4
Ours w/o TTA	99.9±0.0	67.5±0.5	76.3±0.2	86.7±0.2	82.6
Ours w/ TTA	99.9±0.0	68.6±0.3	76.5±0.2	86.7±0.1	82.9

VLCS had a larger gain with our method. The major performance gain was from the SUN09 domain, and there were sufficient improvements in other domains, though there were some exceptional cases. From these results, we confirmed that our method can be highly effective in cases where the performance of the baseline, ZS-CLIP (PC), is poor.

Results of Office-Home experiments (Tab. 3.7). In the Office-Home experiments, our method showed a significant improvement over existing methods in the Clipart domain. Among the four domains in OfficeHome, clipart is the most distant from the real image domain, which requires searching for an appropriate domain for accurate recognition. Our method successfully trained a wide variety

Table 3.7: Per domain classification accuracy on Office-Home.

ResNet-50 with pre-trained weights from CLIP					
Method	Art	Clipart	Product	Real World	Avg.
ZS-CLIP (C) [139]	69.9±0.0	46.8±0.0	77.7±0.0	79.8±0.0	68.6
ZS-CLIP (PC) [139]	71.7±0.0	52.0±0.0	81.6±0.0	82.6±0.0	72.0
PromptStyler [29]	73.4±0.1	52.4±0.2	84.3±0.1	84.1±0.1	73.6
Ours w/o TTA	73.6±0.2	55.1±0.1	85.5±0.2	84.9±0.1	74.8
Ours w/ TTA	74.0±0.1	55.3±0.3	86.2±0.1	85.2±0.1	75.2
ViT-B/16 with pre-trained weights from CLIP					
Method	Art	Clipart	Product	Real World	Avg.
ZS-CLIP (C) [139]	80.7±0.0	64.6±0.0	86.3±0.0	88.0±0.0	79.9
ZS-CLIP (PC) [139]	82.7±0.0	67.6±0.0	89.2±0.0	89.7±0.0	82.3
PromptStyler [29]	83.8±0.1	68.2±0.0	91.6±0.1	90.7±0.1	83.6
Ours w/o TTA	83.8±0.1	70.1±0.1	91.1±0.1	90.8±0.0	84.0
Ours w/ TTA	83.8±0.1	70.6±0.0	91.4±0.1	91.0±0.0	84.2
ViT-L/14 with pre-trained weights from CLIP					
Method	Art	Clipart	Product	Real World	Avg.
ZS-CLIP (C) [139]	86.2±0.0	73.3±0.0	92.0±0.0	92.2±0.0	85.9
ZS-CLIP (PC) [139]	87.2±0.0	73.8±0.0	93.0±0.0	93.4±0.0	86.9
PromptStyler [29]	89.1±0.1	77.6±0.1	94.8±0.1	94.8±0.0	89.1
Ours w/o TTA	88.5±0.1	79.4±0.0	95.0±0.1	94.3±0.1	89.3
Ours w/ TTA	88.9±0.2	79.5±0.2	95.3±0.1	94.4±0.0	89.5

of domain-specific prompts in the development phase, and the prompts covered a larger range of domains than those of existing methods. Therefore, we consider that our method was able to address data from difficult domains such as Clipart.

Results of DomainNet experiments (Tab. 3.8). Compared to the other three benchmarks, the results on DomainNet showed that our method yielded large improvements in high-performance backbone networks (ViT-B/16, ViT-L/14). This may be because existing methods did not fully exploit the true potential of these networks. Our method effectively overcame this limitation by facilitating domain-flexible prediction. Furthermore, the experimental results show that test-time adaptation (Domain Discriminator + TTA) can amplify the benefits of our

Table 3.8: **Per domain classification accuracy on DomainNet.**

Method	ResNet-50 with pre-trained weights from CLIP						
	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	Avg.
ZS-CLIP (C) [139]	53.1±0.0	39.2±0.0	52.7±0.0	6.3±0.0	75.2±0.0	47.1±0.0	45.6
ZS-CLIP (PC) [139]	53.6±0.0	39.6±0.0	53.4±0.0	5.9±0.0	76.6±0.0	48.0±0.0	46.2
PromptStyler [29]	57.9±0.0	44.3±0.0	57.3±0.0	6.1±0.1	79.5±0.0	51.7±0.0	49.5
Ours w/o TTA	57.7±0.1	44.6±0.1	57.1±0.0	8.2±0.1	79.4±0.1	52.3±0.1	49.9
Ours w/ TTA	57.9±0.2	45.2±0.1	57.7±0.1	8.6±0.1	79.9±0.1	52.4±0.1	50.3

Method	ViT-B/16 with pre-trained weights from CLIP						
	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	Avg.
ZS-CLIP (C) [139]	70.7±0.0	49.1±0.0	66.4±0.0	14.8±0.0	82.7±0.0	63.1±0.0	57.8
ZS-CLIP (PC) [139]	71.0±0.0	47.7±0.0	66.2±0.0	14.0±0.0	83.7±0.0	63.5±0.0	57.7
PromptStyler [29]	73.1±0.0	50.9±0.0	68.2±0.1	13.3±0.1	85.4±0.0	65.3±0.0	59.4
Ours w/o TTA	73.4±0.1	53.5±0.1	68.4±0.1	15.6±0.1	85.4±0.0	66.4±0.1	60.4
Ours w/ TTA	73.3±0.1	53.5±0.2	68.3±0.1	16.0±0.2	85.4±0.0	66.3±0.1	60.5

Method	ViT-L/14 with pre-trained weights from CLIP						
	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	Avg.
ZS-CLIP (C) [139]	78.2±0.0	53.0±0.0	70.7±0.0	21.6±0.0	86.0±0.0	70.3±0.0	63.3
ZS-CLIP (PC) [139]	79.2±0.0	52.4±0.0	71.3±0.0	22.5±0.0	86.9±0.0	71.8±0.0	64.0
PromptStyler [29]	80.7±0.0	55.6±0.1	73.8±0.1	21.7±0.0	88.2±0.0	73.2±0.0	65.5
Ours w/o TTA	80.7±0.1	58.1±0.1	73.7±0.0	23.5±0.0	88.3±0.0	73.8±0.0	66.3
Ours w/ TTA	80.7±0.0	58.5±0.1	74.2±0.0	24.3±0.1	88.3±0.0	73.7±0.0	66.6

methodological design.

3.7.2 Hyper-paramter Settings

We set the default values of the hyper-parameters by referring to previous studies and made adjustments as necessary. The AdamW optimizer parameters used in the development phase were determined from the ProText settings [84]. For the configuration of domain-specific prompts, the length of common vectors K was set to 4 based on the one employed in ProText, and the length of the domain vector was set to the same value. The number of domains M is set to 200 for the default setting. This is larger than the value utilized in PromptStyler [29],

but this has little effect on the training time of our method compared to Prompt Styler, where the training time increases linearly with the number of domains. We exceptionally changed M to 40 for VLCS experiments, where we found that our method performed well. This is because VLCS is composed of datasets from the real-world domain, and there was no need to provide domain-specific prompts for many domains. The number of training epochs and the number of reference texts N_{ref} were determined by considering the number of total training iterations. Prompt learning techniques are known to be prone to overfitting [115], and our work avoids this problem by adjusting the number of total training iterations. For the other parameters of our method, we set λ_{domain} to 0.3, λ_{TTA} to 1.0, and test-time adaptation steps to 50 as the default. Our method worked effectively in most cases without any laborious tuning. Exceptionally, we tuned λ_{TTA} to 2.0 in the DomainNet experiments. This is because when $\lambda_{\text{TTA}} = 1.0$, the test-time adaptation had a smaller effect than the other datasets.

Chapter 4

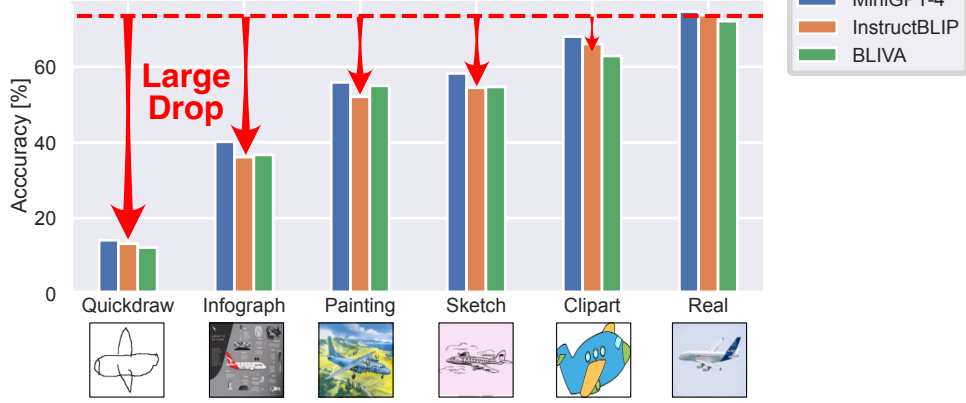
Model-free Domain Adaptation for Concealed Multimodal Large-Language Models

4.1 Introduction

One of the most innovative breakthroughs in recent deep learning research is the emergence of large-language models (LLMs) [16, 30, 169]. LLMs have demonstrated remarkable capability for a wide variety of language tasks. Following this breakthrough, multimodal LLMs (MLLMs) [1, 107, 213] have been developed to extend their ability to multimodal tasks. These models serve as “foundation models” [14, 3] and demonstrate remarkable zero-shot capability for a variety of tasks, such as image understanding, reasoning, and generation. Thus, they attracted much attention not only from academy but also from industry and have been deployed across various services [140, 164, 4].

Despite this remarkable progress, it has been reported that MLLMs often strug-

(a) Performance of MLLM on Several Domains



(b) Model-free Domain Adaptation & Our Solution

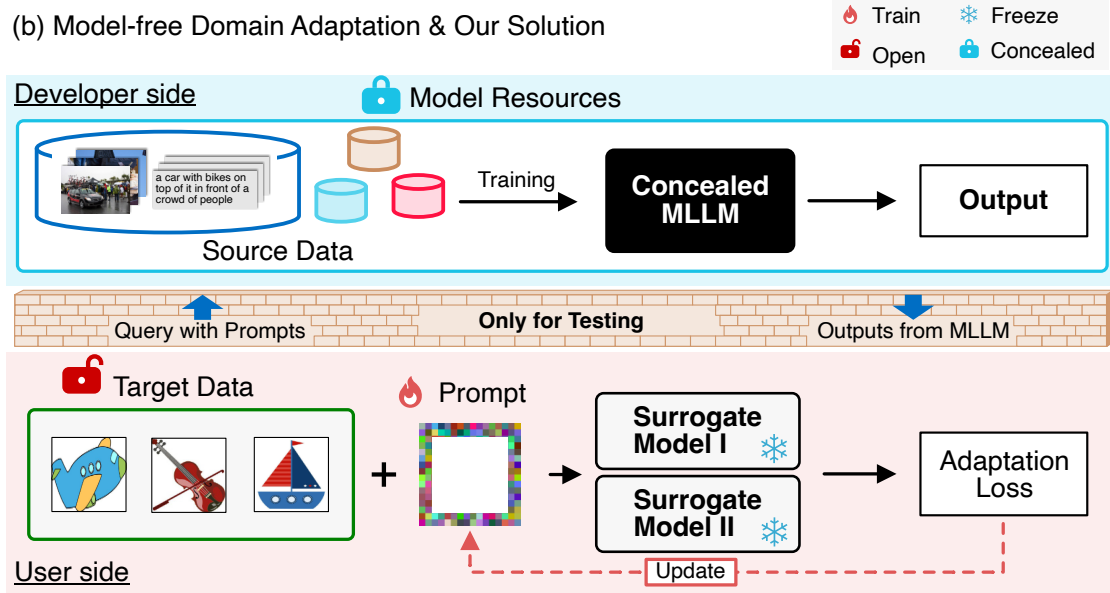


Figure 4.1: (a) MLLMs perform well on familiar domains (e.g., Real), but their performance degrades on unfamiliar domains (e.g., Infographic, Quickdraw). (b) MFDA prohibits the use of any concealed model resources (i.e., source data, model weights, and model outputs) during training but allows access only for testing. To solve MFDA, our method builds visual prompts that encourage the domain adaptation of the concealed MLLMs by cross-model visual prompting with the domain adaptation training objective.

gle to maintain high performance when they are applied to the data from domains unfamiliar to the models [199]. As shown in Fig. 4.1 (a), we empirically confirmed that MLLMs significantly degraded the recognition performance in certain domains (e.g., “Quickdraw” and “Infograph”), which were rarely seen during training, compared with familiar domains (e.g., “Real”). One promising approach to address this issue is domain adaptation [145]. Domain adaptation is aimed at transferring knowledge learned from a source training domain to a specific target domain. A more practical form of domain adaptation called source-free domain adaptation (SFDA) [102] has emerged, where we use a source pre-trained model instead of raw source data. This opens up new possibilities of domain adaptation for MLLMs, which are built upon a huge amount of private source data.

Unfortunately, one challenge still remains: the cutting-edge MLLMs rarely release their raw models, and the users can access the models only through API services, which we call “concealed MLLMs”. Most domain adaptation methods require a raw target model, which makes it difficult to apply them to concealed MLLMs. Black-box domain adaptation (BBDA) methods [103, 196, 28] may seem promising to this problem since they use only the model outputs for training. However, such usage will cause other issues, such as high-cost service charges or license violations for the use of the outputs for model training.

These facts bring us a simple question: *Can we achieve domain adaptation without using such resources of the concealed models?* To investigate this, we introduce a novel domain adaptation setup, **model-free domain adaptation (MFDA)**. During training, MFDA prohibits us from using any model resources, such as, source data, model parameters, or model outputs from the concealed MLLM, which is illustrated in Fig. 4.1 (b). As summarized in Tab. 4.1, MFDA is

Table 4.1: **Comparison of Domain Adaptation Setups.** MFDA imposes the most challenging constraints compared with other domain adaptation setups.

Problem Setup	Source Data	Model Parameters	Model Outputs
Domain Adaptation [145]	✓	✓	✓
Source-free DA [102]	✗	✓	✓
Black-box DA [103]	✗	✗	✓
Model-free DA	✗	✗	✗

most challenging compared with the existing domain adaptation setups.

As a proof of concept for MFDA, we built a novel method named **model-transferable domain-adaptable visual prompting (MTDA-VP)**. To overcome the difficulty in directly updating the model parameters, we use a visual prompting (VP) [6] technique, which can treat the image pixel space as training parameters. Since the image pixel space is inherently shared with the target concealed MLLMs even if the model parameters are concealed, we can address the parameter inaccessibility problem. MTDA-VP is based on the transferable VP (TVP) technique [200]. TVP builds visual prompts using multiple models so that the prompts can carry transferable knowledge across various unseen models. In our method, we employ two surrogate models and train visual prompts that have efficacy not only for those seen surrogate models but also for the unseen concealed MLLMs. However, a direct application of TVP cannot resolve MFDA for two reasons: 1) TVP is a fully supervised algorithm, while we only have access to unsupervised target data, and 2) TVP is not fundamentally designed for domain adaptation. To equip the visual prompts with the domain adaptability in an unsupervised manner, we simultaneously incorporate the insights of SFDA [102, 190] and our findings in Chapter 2. We also developed two novel strategies to boost the abilities of the visual prompts. First, **cross-model pseudo labeling (CMPL)**

enriches the domain adaptation training by generating high-quality pseudo-labels on the basis of the label propagation algorithm [205, 80]. We propagate the label information from reliable samples, in the sense that the predictions of the two surrogate models agree, to other unreliable samples. Second, **cross-model gradient alignment (CMGA)** promotes the model transferability of the prompts by incorporating the gradient-alignment technique [153]. CMGA aligns the loss gradients between two surrogate models and discovers a better direction for training across multiple models.

Our contributions in this chapter are summarized as follows:

- We introduced a novel setup, MFDA, to investigate whether we can adapt the concealed MLLMs to the target domain without using any of their resources.
- We built MTDA-VP, which executes cross-model visual prompting with a domain adaptation objective to equip the visual prompts with both model transferability and domain adaptability.

4.2 Related Works

4.2.1 Multimodal Large-language Models

LLMs [16, 30, 169], which are built with massive text corpora and a simple training strategy of next-token prediction, have not only demonstrated remarkable performance across a wide range of language tasks but also acquired generalization capability even for unseen tasks during the training phase. This remarkable capability of LLMs has encouraged researchers to develop MLLMs [1, 107, 213] that can accept the input of non-language data, such as images, video, and audio. Since

the pioneering MLLM model was released by OpenAI [1], many high-performance models [164, 2, 4] have been deployed. In the open-source domain, a number of models [98, 34, 213, 75, 107, 106, 169, 179, 165] have been made available, fostering a highly active field of research.

Due to the vast number of parameters that compose MLLMs, directly fine-tuning the whole models for specific tasks or data is impractical from both computational cost and storage perspectives. To address this issue, parameter-efficient fine-tuning (PEFT) methods, which train only a small number of additional parameters, have gained attention. Adapters [72, 49, 197] add a small number of parameters to the model and only train these additional parameters, enabling efficient adaptation to downstream tasks. Low-rank Adaptation (LoRA) [73] significantly reduces the number of trainable parameters by inserting trainable low-rank matrices into each layer of the transformer [175]. Prompt-tuning [93, 209, 208, 82, 6, 200] introduces trainable parameters in the input space of the model, which enables flexible adaptation regardless of the model architectures.

4.2.2 Domain Adaptation

Domain adaptation [145] leverages the knowledge of the source domain with rich supervision to the poorly supervised target domain. Unsupervised domain adaptation (UDA), which requires no supervision for the target data, has long been studied. Distribution-matching methods [171, 111, 156, 202, 124] and adversarial-learning methods [170, 48, 147, 25, 198, 24] minimize the distributional discrepancy between source and target domains, while pseudo-labeling methods [216, 215, 88, 108] enrich task information by generating high-quality pseudo labels from the model outputs. Several other methods have also been proposed, including repre-

sentation learning methods [18, 92] and reconstruction-based methods [53].

SFDA is a newer domain adaptation setup in which we have access to a source-pretrained model instead of raw source domain data. SFDA avoids the pragmatic issues of UDA, such as the privacy of the source data and computational inefficiency of re-training with source data. As described in the survey [101], SFDA has been actively studied and various approaches have been proposed, such as source estimation [99, 71, 166, 136, 187], pseudo labeling [191, 83, 201, 105, 138, 38], clustering [102, 91], consistency [192, 23, 190, 203, 121], and self-supervised learning [159, 109]. BBDA [103, 196, 28] was proposed as a variant of SFDA and trains an on-premise model for the target domain with the outputs from the black-box source models.

4.2.3 Contribution over Prior Works

This work aims to develop a domain adaptation method that requires no model resources (i.e., source data, model parameters, model outputs) from the concealed MLLMs. Viewed in this light, some black-box-based algorithms (e.g., BBDA [103, 196, 28], Black box Tuning [158, 157], BlackVIP [127]) seem promising; however, these methods do not meet the MFDA requirement because they need the model outputs for training, which may cause other issues, such as MLLM license restrictions for the use of model outputs for training (e.g., model distillation) or high-cost service charges. In contrast, our proposed MTDA-VP does not use any model resources until the test phase. Instead, we use unlabeled target data and surrogate models and build visual prompts that have model transferability and domain adaptability.

4.3 Preliminaries

4.3.1 Problem Setup

The goal with MFDA is to maximize the accuracy of a concealed MLLM $F_{\text{concealed}}$ on data from an unlabeled target domain $D_T = \{\mathbf{X}_i\}_{i=1}^{N_t}$ without using the model $F_{\text{concealed}}$ itself, the outputs from the model, or the source data D_S , where $F_{\text{concealed}}$ is trained upon the source data D_S , and D_S includes the text-only data used in the base LLM training and the image-text pairs used for the multimodal extension. Since this work is the first to tackle MFDA setup, we focus on the simplest image classification task, leaving other tasks, such as image captioning or image reasoning, for our future work.

4.3.2 Multimodal Large-language Models

MFDA-VP employs an openly accessible, off-the-shelf MLLM as one of its surrogate models, hereinafter referred to as the “surrogate MLLM”. The surrogate MLLM $F_{\text{surrogate}}$ is structured by equipping a text-only base LLM h , such as Llama [169], with a vision encoder f and a projector g that maps image features into the word embedding space of h . When $F_{\text{surrogate}}$ receives an image $\mathbf{X} \in \mathbb{R}^{H \times W \times 3}$ and an instruction text $\mathbf{t} \in \mathbb{V}^L$ as inputs (e.g., “This is a photo of a”) where H and W denote the height and width of the image, respectively, $\mathbb{V}$ denotes the vocabulary set, and L denotes the text length, it autoregressively generates the output as,

$$o_l \sim F_{\text{surrogate}}(o_l | \mathbf{X}, \mathbf{t}, o_{<l}) = h(o_l | g(f(\mathbf{X})), \mathbf{t}, o_{<l}), \quad (4.1)$$

where $\{o_l \in \mathbb{V}\}_{l=1}^{L_o}$ is a sequence of output text tokens. Then, the classification likelihood is calculated as

$$p_{\text{MLLM}}(y|\mathbf{X}, \mathbf{t}) = \prod_{l=1}^{L_y} F_{\text{surrogate}}(o_l^y|\mathbf{X}, \mathbf{t}, o_{<l}^y), \quad (4.2)$$

where $\{o_l^y \in \mathbb{V}\}_{l=1}^{L_y}$ is a text sequence of the class y .

4.3.3 Visual Prompting

VP [6] adds learnable prompts in the image pixel space and trains them for the downstream tasks. Specifically, VP downsizes the input images and pads them with learnable pixel-level prompts so that the padded images recover the original image size, which is formulated as

$$\text{VP}(\mathbf{X}, \boldsymbol{\delta}, p) = \text{Resize}(\mathbf{X}, (H - 2p, W - 2p)) + \mathbf{M}_p \odot \boldsymbol{\delta}, \quad (4.3)$$

where $\boldsymbol{\delta}$ denotes the trainable prompts, p the padding size, $\mathbf{M}_p$ a mask tensor that is set to 1 for the p pixels from the four sides and 0 for the others to constrain the prompting pixel space, and $\odot$ represents Hadamard Product. The visual prompts are optimized on the basis of the training objective. For example, the classification objective with a training dataset D is formulated as

$$\min_{\boldsymbol{\delta}} \mathbb{E}_{(\mathbf{X}, y) \sim D} [-\log p_{\text{MLLM}}(y|\text{VP}(\mathbf{X}, \boldsymbol{\delta}, p), \mathbf{t})]. \quad (4.4)$$

Transferable VP (TVP) [200] aims to build visual prompts that can be applicable to various unseen models, which is based on the insight of the transferability of adversarial examples [122, 212, 39]. However, the authors of TVP found that

visual prompts trained simply on the task objective perform poorly on the different models due to the loss of general visual knowledge and model-shared task knowledge. TVP addresses this issue by using a cross-model training strategy using an off-the-shelf MLLM and CLIP [139] and introducing two additional training losses: the feature consistency alignment (FCA) to maintain the general knowledge of the input image and the task semantic enrichment (TSE) loss to enhance the task-related semantics across the models.

Although TVP appears to be a possible solution to MFDA, it cannot be directly applied for two reasons. One is that TVP is a fully *supervised* algorithm, which makes it inherently impossible to apply TVP to the fully *unsupervised* MFDA setup. Another is that TVP is *NOT* designed for the domain-shift problem. TVP may manage the absence of the supervised information by substituting the model outputs as pseudo labels. However, we have experimentally confirmed that the domain-shift problem cannot be solved by simple pseudo-labeling (see Sec. 4.5.3). We will describe how MTDA-VP solves these issues in the next section.

4.4 Our Method: Model-Transferable Domain-Adaptable Visual Prompting

Fig. 4.2 illustrates an overview of MTDA-VP. During training, MTDA-VP builds visual prompts with a surrogate MLLM and CLIP instead of the concealed MLLM, and during testing, we query the concealed MLLM with the prompted images. To achieve domain adaptation of the concealed MLLM, the visual prompts should acquire both domain adaptability to the target domain and model transferability to the concealed MLLM. To obtain both capabilities, we incorporate TVP [200]

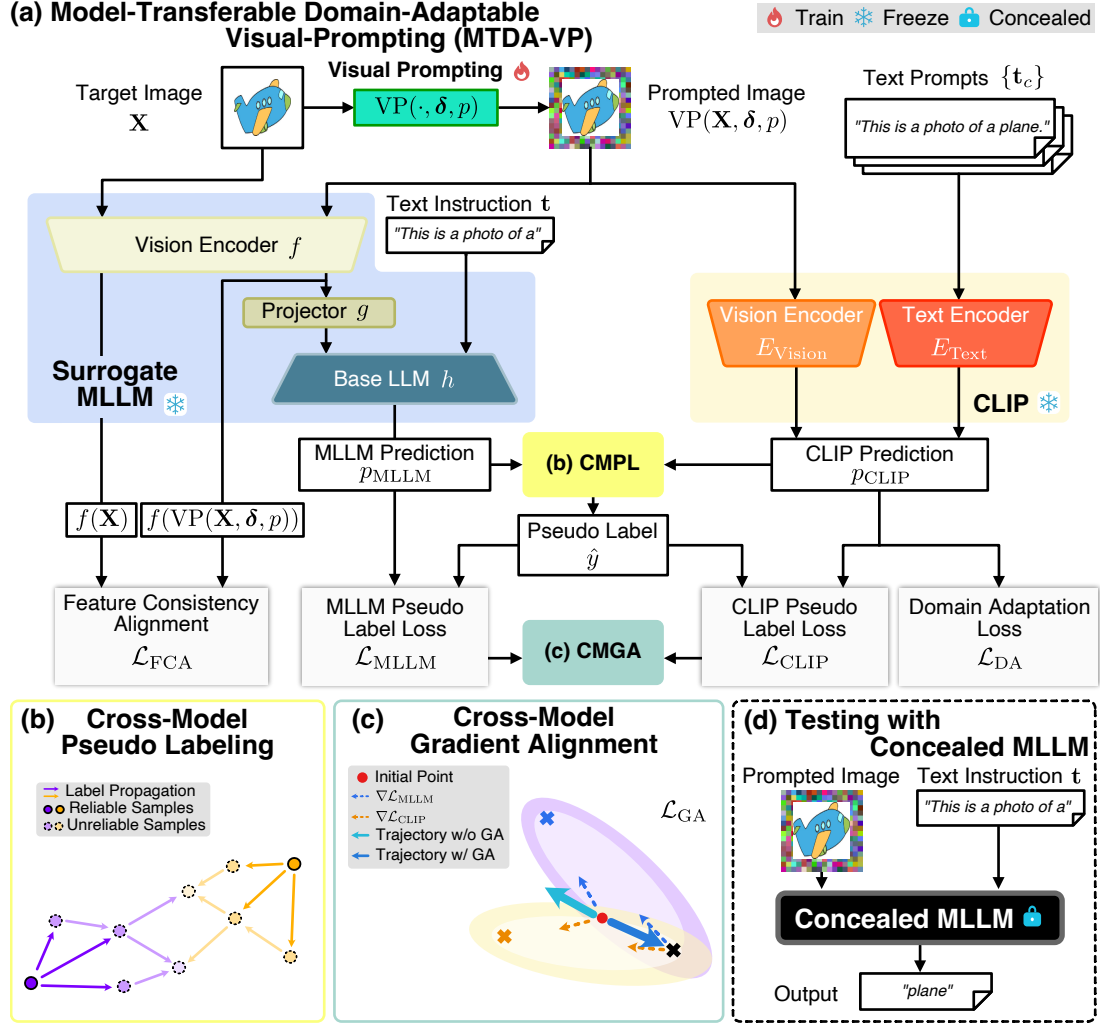


Figure 4.2: **Overview of MTDA-VP.** (a) MTDA-VP executes cross-model visual prompt training with surrogate MLLM and CLIP based on the domain adaptation objective to equip the prompts with model transferability and domain adaptability. We introduce (b) CMPL and (c) CMGA to boost domain adaptability and model transferability, respectively. (d) In testing, we just attach the trained prompt to the input image to adapt the concealed MLLM to the target domain.

based cross-model prompting strategy and optimize the visual prompts on the domain adaptation objective, respectively. We also enhance these two abilities using CMPL and CMGA.

4.4.1 Cross-Model Visual Prompting with Domain-Adaptation Objective

To empower domain adaptability to the visual prompts without any access to the labeled source data, we incorporate SFDA training into our method. According to the insights from the previous SFDA study [190] and our investigation in Chapter 2, the combination of discriminability and diversity losses is a critical factor for training only with the unlabeled target data. The discriminability loss enhances the prediction discriminability of the model to the target samples, while the diversity loss prevents model collapse by ensuring prediction diversity. Implementing these losses for visual prompting on the surrogate MLLM may seem trivial, but it is not actually simple. This is because diversity loss requires the likelihood calculation for all classes, but the autoregressive MLLM needs to calculate the likelihood for each class one by one, which incurs a huge computational cost. Fortunately, we have employed two surrogate models, so we pay attention to the other surrogate model, CLIP, to implement these training losses.

CLIP consists of a vision encoder E_{vision} and a text encoder E_{text} that project the image and text features onto the joint feature space. The prediction likelihood of the image $\mathbf{X}$ is calculated as

$$p_{\text{CLIP}}(y|\mathbf{X}) = \sigma(\text{cossim}(E_{\text{vision}}(\mathbf{X}), E_{\text{text}}(\mathbf{t}_y))/\tau_{\text{CLIP}}), \quad (4.5)$$

where $\sigma(\cdot)$ is a softmax function, $\text{cossim}(\cdot, \cdot)$ is a cosine similarity function, $\mathbf{t}_y$ is a text description tied to the class y (e.g., “This is a photo of a [class]”), and τ_{CLIP} is the CLIP learned temperature parameter. This prediction calculation on CLIP can be efficiently computed for all classes at once and successfully mitigates the

aforementioned issue. We then use the prediction results of CLIP for calculating conditional entropy minimization loss as the discriminability loss and marginal entropy maximization loss as the diversity loss, respectively, following a previous SFDA method, SHOT [102]. Formally, these losses are formulated as

$$\mathcal{L}_{\text{DA}} = \mathbb{E}_{\mathbf{X} \sim D_T} [H(p_{\text{CLIP}}(y|\text{VP}(\mathbf{X}, \boldsymbol{\delta}, p)))] - H(\mathbb{E}_{\mathbf{X} \sim D_T} [p_{\text{CLIP}}(y|\text{VP}(\mathbf{X}, \boldsymbol{\delta}, p))]), \quad (4.6)$$

where $H(\cdot)$ is an entropy function.

4.4.2 Cross-Model Pseudo Labeling

Inspired by the pseudo-labeling SFDA methods [191, 137, 83], we generate pseudo labels for the unlabeled target data to enhance the domain adaptability. The naive solution is to use the predictions from the model directly as pseudo labels; however, the predictions of MLLMs in the early adaptation stage contain a large amount of noise due to the domain shifts. CMPL addresses this issue by introducing two ideas based on the cross-model prompting strategy. Firstly, we employ a filtering technique to get reliable predictions based on the prediction agreements between the surrogate MLLM and CLIP, which mitigates the negative effect of the noise in the early MLLM’s predictions. Secondly, we incorporate a label propagation algorithm [205, 80] to propagate the label information from the reliable samples to all the other target samples. Specifically, CMPL treats the reliable samples as labeled data points and the other samples as unlabeled data points and propagates the label information from the labeled to the unlabeled data points on the basis of the sample similarities.

The prediction results from the two models are¹

$$\hat{y}_{\text{MLLM}_i} = \underset{y}{\operatorname{argmax}} p_{\text{MLLM}}(y|\text{VP}(\mathbf{X}_i, \boldsymbol{\delta}, p), \mathbf{t}), \quad (4.7)$$

$$\hat{y}_{\text{CLIP}_i} = \underset{y}{\operatorname{argmax}} p_{\text{CLIP}}(y|\text{VP}(\mathbf{X}_i, \boldsymbol{\delta}, p)). \quad (4.8)$$

On the basis of the prediction results, we split the dataset D_T into the labeled set D_L and the unlabeled set D_U . When the predictions from the two models are the same, the data sample is assigned to D_L ; otherwise, to D_U . Formally,

$$D_L = \{(\mathbf{X}_i, \hat{y}_{\text{MLLM}_i}) | \hat{y}_{\text{MLLM}_i} = \hat{y}_{\text{CLIP}_i}\}, \quad (4.9)$$

$$D_U = \{\mathbf{X}_i | \hat{y}_{\text{MLLM}_i} \neq \hat{y}_{\text{CLIP}_i}\}. \quad (4.10)$$

We prepare the set of visual features as $V = \{\mathbf{v}_i\}_{i=1}^N$, where $\mathbf{v}_i = f(\text{VP}(\mathbf{X}_i, \boldsymbol{\delta}, p))$, for sample similarity calculations. We then apply the label propagation algorithm using D_L , D_U , and V , and obtain a pseudo-labeled set as

$$D_{\text{PL}} = \{(\mathbf{X}_i, \hat{\mathbf{p}}_i)\}_{i=1}^N = \mathcal{A}_{\text{LP}}(D_L, D_U, V), \quad (4.11)$$

where $\mathcal{A}_{\text{LP}}$ represents the label propagation process, and $\hat{\mathbf{p}}_i$ is a soft pseudo label. We use the conjugated gradient-based method [80] for $\mathcal{A}_{\text{LP}}$. The details of the label propagation algorithm are provided in Sec. 4.7.1.

¹To reduce the high computational cost of MLLM inferences, we use the efficient prediction technique described in Sec. 4.7.1.

4.4.3 Cross-Model Gradient Alignment

Following the TSE loss of TVP, MTDA-VP calculates pseudo-label losses on two models to learn model-shared task knowledge. The pseudo-label-based training losses are represented as

$$\mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta}) = \mathbb{E}_{(\mathbf{X}, \hat{\mathbf{p}}) \sim D_{\text{PL}}} [-w \cdot \log p_{\text{MLLM}}(\hat{y} | \text{VP}(\mathbf{X}, \boldsymbol{\delta}, p), \mathbf{t})], \quad (4.12)$$

$$\mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta}) = \mathbb{E}_{(\mathbf{X}, \hat{\mathbf{p}}) \sim D_{\text{PL}}} [-w \cdot \log p_{\text{CLIP}}(\hat{y} | \text{VP}(\mathbf{X}, \boldsymbol{\delta}, p))], \quad (4.13)$$

where, $\hat{y}$ is a hard pseudo label $\hat{y} = \arg \max_y \hat{\mathbf{p}}[y]$, w is a weight value obtained by $w = 1 - H(\hat{\mathbf{p}})/\log(C)$, and C is the number of classes.

However, there remains one problem: the loss-gradient conflict between the two models may discourage the model transferability of the prompts. We address this by introducing a gradient-alignment technique originally proposed in domain generalization studies [153, 180, 132]. These works proposed to align the loss gradients calculated on different domains so that the trained model becomes transferable across different domains. In this analogy, we align the loss gradients calculated on different models so that the visual prompts become transferable across different models. Formally, we introduce the following regularization term

$$\mathcal{L}_{\text{GA}}(\boldsymbol{\delta}) = -\frac{\nabla \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta})}{\|\nabla \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta})\|_2} \cdot \frac{\nabla \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta})}{\|\nabla \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta})\|_2}. \quad (4.14)$$

While Eq. (4.14) mitigates the gradient-conflict problem, it requires high computational costs of the second-order derivatives to minimize Eq. (4.14) by gradient descent. Following previous studies [153, 180, 132], we avoid this issue by using an implicit gradient-alignment technique. Specifically, we alter the loss functions

(4.12), (4.13) as

$$\mathcal{L}_{\text{MLLM-GA}}(\boldsymbol{\delta}) = \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta} - \alpha \nabla \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta}) / G_{\text{norm}}), \quad (4.15)$$

$$\mathcal{L}_{\text{CLIP-GA}}(\boldsymbol{\delta}) = \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta} - \alpha \nabla \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta}) / G_{\text{norm}}), \quad (4.16)$$

where α is a coefficient parameter, and $G_{\text{norm}} = \|\nabla \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta})\|_2 \cdot \|\nabla \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta})\|_2$. These losses implicitly incorporate Eq. (4.14). For example, we can carry out the first-order Taylor expansion as follows:

$$\begin{aligned} & \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta} - \alpha \nabla \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta}) / G_{\text{norm}}) \\ &= \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta}) - \alpha \nabla \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta}) \cdot \nabla \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta}) / G_{\text{norm}} \\ & \quad + O(\alpha \nabla \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta}) / G_{\text{norm}}) \\ &\approx \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta}) - \alpha \nabla \mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta}) \cdot \nabla \mathcal{L}_{\text{CLIP}}(\boldsymbol{\delta}) / G_{\text{norm}}. \end{aligned} \quad (4.17)$$

The second term of Eq. (4.17) is equal to Eq. (4.14), which indicates that our modified loss $\mathcal{L}_{\text{MLLM-GA}}(\boldsymbol{\delta})$ simultaneously functions as both $\mathcal{L}_{\text{MLLM}}(\boldsymbol{\delta})$ and $\mathcal{L}_{\text{GA}}(\boldsymbol{\delta})$.

4.4.4 Prompt Optimization

In addition to the training losses we have described, we use FCA to reduce the general knowledge loss, which was originally proposed in TVP [200]. FCA imposes consistency regularization between the original image features and the prompted image features. Formally, we use the following loss function as

$$\mathcal{L}_{\text{FCA}} = \mathbb{E}_{\mathbf{X} \sim D_T} \left[\frac{1}{2} \|f(\mathbf{X}) - f(\text{VP}(\mathbf{X}, \boldsymbol{\delta}, p))\|_2^2 \right]. \quad (4.18)$$

In summary, our training objective is

$$\min_{\boldsymbol{\delta}} \mathcal{L}(\boldsymbol{\delta}) = \min_{\boldsymbol{\delta}} \mathcal{L}_{\text{MLLM-GA}}(\boldsymbol{\delta}) + \lambda_{\text{CLIP}} \mathcal{L}_{\text{CLIP-GA}}(\boldsymbol{\delta}) + \lambda_{\text{DA}} \mathcal{L}_{\text{DA}}(\boldsymbol{\delta}) + \lambda_{\text{FCA}} \mathcal{L}_{\text{FCA}}(\boldsymbol{\delta}), \quad (4.19)$$

where λ_{CLIP} , λ_{DA} , and λ_{FCA} are hyper-parameters. On the basis of the insight of the previous VP work [186], we apply the normalization to the gradient of the total loss and update the visual prompts by

$$\boldsymbol{\delta} \leftarrow \boldsymbol{\delta} - \eta \frac{\nabla \mathcal{L}(\boldsymbol{\delta})}{\|\nabla \mathcal{L}(\boldsymbol{\delta})\|_2}, \quad (4.20)$$

where η is the learning rate.

4.5 Experiments

4.5.1 Setups

Datasets. We used two benchmark datasets. DomainNet [145] is a large-scale dataset consisting of 345 categories and 6 domains (Clipart, Infograph, Painting, Quickdraw, Real, Sketch). We used the randomly sampled 90% of the training splits for training, the remaining 10% for validation, and all test splits for testing. PACS [95] is a small-scale dataset consisting of 7 categories and 4 domains (Art Painting, Cartoon, Photo, Sketch). We followed the official training, validation, and test splits. We calculated the classification accuracy for each domain and evaluated the performance of MTDA-VP on the average of the per-domain accuracy.

Models. In our experiments, we employed three MLLMs, MiniGPT-4 [213],

Instruct-BLIP [34], and BLIVA [75], and simulated MFDA setups. Specifically, we conducted two patterns of experiments: 1) we chose MiniGPT-4 as the surrogate MLLM for training visual prompts and transferred the prompts to the other two models (InstructBLIP and BLIVA) as the concealed models, and 2) we chose InstructBLIP as the surrogate MLLM and the others (MiniGPT-4 and BLIVA) as the concealed models. For the CLIP model used in MTDA-VP, we chose one with the ViT-B/32 vision encoder.

Implementation Details. We set the learning rate η to 1.0 for DomainNet and 0.5 for PACS, and η was controlled with the cosine decay scheduler. We tuned λ_{FCA} to $3\text{e}-3$, the size of VP p to 30 with reference to the original TVP setting [200]. We set λ_{CLIP} to 0.5, λ_{DA} to 1.0 and α to $1\text{e}-2$. We trained the VP for 5 epochs with a batch size of 16. The other details can be found in Sec. 4.7.1.

4.5.2 Main Results

We conducted three random-seed runs and reported the averaged accuracies with the standard errors of the means.

Result on DomainNet (Tab. 4.2). MTDA-VP improved the accuracy from No Adapt. results with significant margins in almost all cases. MTDA-VP achieved significant improvements, particularly in the Quickdraw domain. The Quickdraw domain was an unfamiliar domain for MLLMs, which led to poor baseline performance. MTDA-VP effectively improved performance in such unfamiliar domains. In contrast, the Real domain was already familiar to MLLMs, so it showed smaller improvements in the Real domain, where the improvements were less than 1.00 in most cases.

Table 4.2: **Classification accuracy [%] on DomainNet.** We highlight **improved** and **degraded** cases (best viewed in color). Unconcealed cases (i.e., the surrogate model and the concealed model are the same) are excluded from average calculations.

Concealed MLLM	Method	<i>Surrogate MLLM: MiniGPT-4</i>						Average
		Clipart	Infograph	Painting	Quickdraw	Real	Sketch	
<i>MiniGPT-4</i> *	No Adapt.	68.00	40.20	55.90	14.20	74.62	58.27	51.87
	MTDA-VP	73.34 $\pm$ 0.05	43.76 $\pm$ 0.15	64.80 $\pm$ 0.31	23.46 $\pm$ 0.22	80.75 $\pm$ 0.30	64.93 $\pm$ 0.10	58.51 $\pm$ 0.18
	Gain Δ	+ 5.34	+ 3.56	+ 8.90	+ 9.26	+ 6.13	+ 6.66	+ 6.64
InstructBLIP	No Adapt.	66.04	36.16	52.09	13.26	73.64	54.48	49.28
	MTDA-VP	70.12 $\pm$ 0.38	40.05 $\pm$ 0.56	59.42 $\pm$ 0.98	20.28 $\pm$ 0.42	74.24 $\pm$ 0.82	58.92 $\pm$ 0.13	53.84 $\pm$ 0.28
	Gain Δ	+ 4.08	+ 3.89	+ 7.33	+ 7.02	+ 0.60	+ 4.44	+ 4.56
BLIVA	No Adapt.	62.87	36.73	55.00	12.24	72.07	54.71	48.94
	MTDA-VP	66.80 $\pm$ 0.99	37.38 $\pm$ 0.80	56.03 $\pm$ 0.64	19.77 $\pm$ 0.70	71.75 $\pm$ 0.40	57.37 $\pm$ 0.39	51.52 $\pm$ 0.38
	Gain Δ	+ 3.93	+ 0.65	+ 1.03	+ 7.53	- 0.32	+ 2.66	+ 2.58
Average	No Adapt.	64.45	36.44	53.55	12.75	72.85	54.59	49.11
	MTDA-VP	68.46 $\pm$ 0.68	38.71 $\pm$ 0.68	57.72 $\pm$ 0.62	20.03 $\pm$ 0.46	72.99 $\pm$ 0.59	58.15 $\pm$ 0.22	52.68 $\pm$ 0.29
	Gain Δ	+ 4.01	+ 2.27	+ 4.18	+ 7.28	+ 0.14	+ 3.55	+ 3.57
Concealed MLLM	Method	<i>Surrogate MLLM: InstructBLIP</i>						Average
		Clipart	Infograph	Painting	Quickdraw	Real	Sketch	
MiniGPT-4	No Adapt.	68.00	40.20	55.90	14.20	74.62	58.27	51.87
	MTDA-VP	68.99 $\pm$ 0.34	41.43 $\pm$ 0.03	59.49 $\pm$ 0.92	19.26 $\pm$ 0.19	74.47 $\pm$ 0.55	59.88 $\pm$ 0.16	53.92 $\pm$ 0.18
	Gain Δ	+ 0.99	+ 1.23	+ 3.59	+ 5.06	- 0.15	+ 1.61	+ 2.06
<i>InstructBLIP</i> *	No Adapt.	66.04	36.16	52.09	13.26	73.64	54.48	49.28
	MTDA-VP	74.35 $\pm$ 0.10	44.49 $\pm$ 0.12	66.76 $\pm$ 0.21	24.54 $\pm$ 0.06	81.37 $\pm$ 0.22	65.82 $\pm$ 0.25	59.61 $\pm$ 0.05
	Gain Δ	+ 8.31	+ 8.33	+ 14.67	+ 11.28	+ 8.09	+ 11.34	+ 10.34
BLIVA	No Adapt.	62.87	36.73	55.00	12.24	72.07	54.71	48.94
	MTDA-VP	68.40 $\pm$ 0.99	40.10 $\pm$ 0.28	60.61 $\pm$ 0.78	21.07 $\pm$ 0.28	76.19 $\pm$ 1.12	60.52 $\pm$ 0.75	54.48 $\pm$ 0.30
	Gain Δ	+ 5.53	+ 3.37	+ 5.61	+ 8.83	+ 4.12	+ 5.81	+ 5.54
Average	No Adapt.	65.44	38.47	55.45	13.22	73.34	56.49	50.40
	MTDA-VP	68.69 $\pm$ 0.59	40.76 $\pm$ 0.13	60.05 $\pm$ 0.85	20.17 $\pm$ 0.18	75.33 $\pm$ 0.83	60.20 $\pm$ 0.32	54.20 $\pm$ 0.20
	Gain Δ	+ 3.26	+ 2.30	+ 4.60	+ 6.95	+ 1.99	+ 3.71	+ 3.80

Results on PACS (Tab. 4.3). As same as in the DomainNet experiments, MTDA-VP showed solid improvement in all cases. MTDA-VP showed the greatest improvement in the Art Painting domain. This was because, in addition to the large domain gap from the Real domain, Art Painting contained rich decorative features. MLLMs tend to recognize the image abstractly [60], which makes it difficult to distinguish semantic features from decorative features. MTDA-VP successfully mitigated the negative effects of the latter features. Meanwhile, the

Table 4.3: **Classification Accuracy [%] on PACS.** We highlight **improved** and **degraded** cases (best viewed in color). Unconcealed cases (i.e., the surrogate model and the concealed model are the same) are excluded from average calculations.

Concealed MLLM	Method	<i>Surrogate MLLM: MiniGPT-4</i>				Average
		Art Painting	Cartoon	Photo	Sketch	
<i>MiniGPT-4*</i>	No Adapt.	85.55	98.51	97.66	96.31	94.51
	MTDA-VP	98.99 $\pm$ 0.07	99.93 $\pm$ 0.01	99.64 $\pm$ 0.09	97.22 $\pm$ 0.07	98.94 $\pm$ 0.05
	Gain Δ	+ 13.44	+ 1.42	+ 1.98	+ 0.91	+ 4.44
InstructBLIP	No Adapt.	73.97	93.52	96.53	93.41	89.36
	MTDA-VP	89.21 $\pm$ 4.33	95.96 $\pm$ 2.04	97.15 $\pm$ 0.95	93.72 $\pm$ 0.61	94.01 $\pm$ 1.66
	Gain Δ	+ 15.24	+ 2.44	+ 0.62	+ 0.31	+ 4.65
BLIVA	No Adapt.	77.15	88.05	93.23	95.09	88.38
	MTDA-VP	88.75 $\pm$ 6.07	96.36 $\pm$ 1.52	99.06 $\pm$ 0.18	95.67 $\pm$ 0.38	94.96 $\pm$ 1.38
	Gain Δ	+ 11.60	+ 8.31	+ 5.83	+ 0.58	+ 6.58
Average	No Adapt.	75.56	90.78	94.88	94.25	88.87
	MTDA-VP	88.98 $\pm$ 3.90	96.16 $\pm$ 0.47	98.10 $\pm$ 0.53	94.70 $\pm$ 0.12	94.49 $\pm$ 1.10
	Gain Δ	+ 13.42	+ 5.38	+ 3.22	+ 0.45	+ 5.62
Concealed MLLM	Method	<i>Surrogate MLLM: InstructBLIP</i>				Average
		Art Painting	Cartoon	Photo	Sketch	
MiniGPT-4	No Adapt.	85.55	98.51	97.66	96.31	94.51
	MTDA-VP	93.10 $\pm$ 1.57	99.30 $\pm$ 0.14	98.72 $\pm$ 0.13	96.84 $\pm$ 0.28	96.99 $\pm$ 0.45
	Gain Δ	+ 7.55	+ 0.79	+ 1.06	+ 0.53	+ 2.48
<i>InstructBLIP*</i>	No Adapt.	73.97	93.52	96.53	93.41	89.36
	MTDA-VP	98.63 $\pm$ 0.10	99.66 $\pm$ 0.07	99.62 $\pm$ 0.10	97.57 $\pm$ 0.32	98.87 $\pm$ 0.11
	Gain Δ	+ 24.66	+ 6.14	+ 3.09	+ 4.16	+ 9.51
BLIVA	No Adapt.	77.15	88.05	93.23	95.09	88.38
	MTDA-VP	95.13 $\pm$ 1.82	99.37 $\pm$ 0.15	99.30 $\pm$ 0.13	95.80 $\pm$ 1.33	97.40 $\pm$ 0.27
	Gain Δ	+ 17.98	+ 11.32	+ 6.07	+ 0.71	+ 9.02
Average	No Adapt.	81.35	93.28	95.44	95.70	91.44
	MTDA-VP	94.12 $\pm$ 1.02	99.34 $\pm$ 0.14	99.01 $\pm$ 0.10	96.32 $\pm$ 0.77	97.20 $\pm$ 0.10
	Gain Δ	+ 12.77	+ 6.06	+ 3.57	+ 0.62	+ 5.75

Sketch domain contained fewer decorative features, which resulted in little performance improvement.

Discussion: Comparison with the case with the model access. Although MFDA aims for domain adaptation under conditions where model access is restricted, this section virtually examines the extent to which our method achieves performance compared to cases where model access is available. We compared the performance when the concealed and surrogate MLLMs differ with the scenario where they are the same (the "Unconcealed" cases in Tab. 4.2 and 4.3). On Do-

Table 4.4: **Comparison and Ablation Study.** We note the performance gain [%] from No Adapt. We highlighted in colors the cases where performance is **improved** and is **declined**. The best scores are highlighted in **bold**.

Methods	Components			Performance Gain of Surrogate MLLM (MiniGPT-4)						
	DA	CMPL	CMGA	C	I	P	Q	R	S	Avg.
1. TVP [†] [200]				- 65.26	- 38.59	- 14.23	- 13.81	+ 5.61	- 56.93	- 30.54
2. TVP [†] + DA	✓			+ 4.94	+ 2.30	+ 8.10	+ 5.44	+ 5.96	+ 5.63	+ 5.40
3. 2. + CMPL	✓	✓		+ 5.40	+ 3.58	+ 9.21	+ 9.24	+ 6.38	+ 6.50	+ 6.72
4. 2. + CMGA	✓		✓	+ 4.76	+ 1.57	+ 7.49	+ 5.10	+ 5.63	+ 5.12	+ 4.95
5. MTDA-VP	✓	✓	✓	+ 5.34	+ 3.56	+ 8.90	+ 9.26	+ 6.13	+ 6.66	+ 6.64

Methods	Components			Performance Gain of Concealed MLLM (InstructBLIP)						
	DA	CMPL	CMGA	C	I	P	Q	R	S	Avg.
1. TVP [†] [200]				- 21.00	- 16.69	- 10.61	- 5.89	+ 0.42	- 37.87	- 15.27
2. TVP [†] + DA	✓			+ 3.16	+ 2.22	+ 2.49	+ 4.13	- 0.09	+ 2.89	+ 2.46
3. 2. + CMPL	✓	✓		+ 3.69	+ 4.91	+ 5.73	+ 5.61	- 0.11	+ 3.77	+ 3.93
4. 2. + CMGA	✓		✓	+ 2.94	- 0.28	+ 2.53	+ 3.90	- 0.08	+ 4.03	+ 2.17
5. MTDA-VP	✓	✓	✓	+ 4.08	+ 3.89	+ 7.33	+ 7.02	+ 0.60	+ 4.44	+ 4.56

mainNet, relative to the model-accessible setting, the score for MiniGPT-4 reached 31%(2.06/6.64), and the score for InstructBLIP reached 44%(4.56/10.34). On PACS, MiniGPT-4 reached 56%(2.48/4.44), and InstructBLIP reached 49%(4.65/9.51). These scores can be seen as a form of transfer efficiency, and maximizing this is considered a crucial factor in MFDA.

4.5.3 Analysis

We conducted deeper analyses of MTDA-VP using the DomainNet dataset and MiniGPT-4/Instruct-BLIP for the surrogate/concealed models, respectively. We also provided other analyses in Sec. 4.7.2.

Comparison and Ablation Study. We verified the effectiveness of each component of MTDA-VP through comparison experiments with baseline methods. Since this is the first study to address MFDA, no existing method is directly applicable to this problem, as discussed in Sec. 4.1 and Sec. 4.2. To verify the effec-

tiveness of our method, we carefully prepared two rationale baselines, $\text{TVP}^\dagger$ and $\text{TVP}^\dagger + \text{DA}$. $\text{TVP}^\dagger$ is a minor modified version of fully-supervised TVP [200] that simply uses the prediction-agreed samples between MLLM and CLIP as pseudo-labeled samples to make it applicable to MFDA. $\text{TVP}^\dagger + \text{DA}$ is a relatively simple method that combines $\text{TVP}^\dagger$ and the SFDA method, SHOT [102], without excessive modifications.

The results are summarized in Tab. 4.4. $\text{TVP}^\dagger$ obviously caused degradations, while $\text{TVP}^\dagger + \text{DA}$ avoided such failures, which shows the necessity of both model-transferability and domain-adaptability for MFDA. Both of our proposed techniques, CMPL and CMGA, were effective, but brought different benefits. CMPL boosts the domain adaptability of the visual prompts. CMPL significantly improved the performance of the surrogate model from +5.40 to +6.72, and it propagated to the concealed model (+2.46 $\rightarrow$ +3.93) by the model transferability of the visual prompts. CMGA contributes to the enhancement of the model transferability. CMGA showed a slight effect on the surrogate model, but largely improved the concealed model (+3.93 $\rightarrow$ +4.56).

Data-Scale Analysis (Fig. 4.3). In real-world problems, many cases are assumed in which it is challenging to collect unlabeled target data in large quantities. In this regard, we examined how the performance of MTDA-VP varied when the amount of unlabeled target data decreased from 100% to 20%. The validity of MTDA-VP was clear even when the amount of data was reduced to 20%, which reinforced the applicability of MTDA-VP to a real-world scenario.

Analysis on Balance of Cross-model Training (Fig. 4.4). We analyzed the balance of the cross-model training between a surrogate MLLM and CLIP

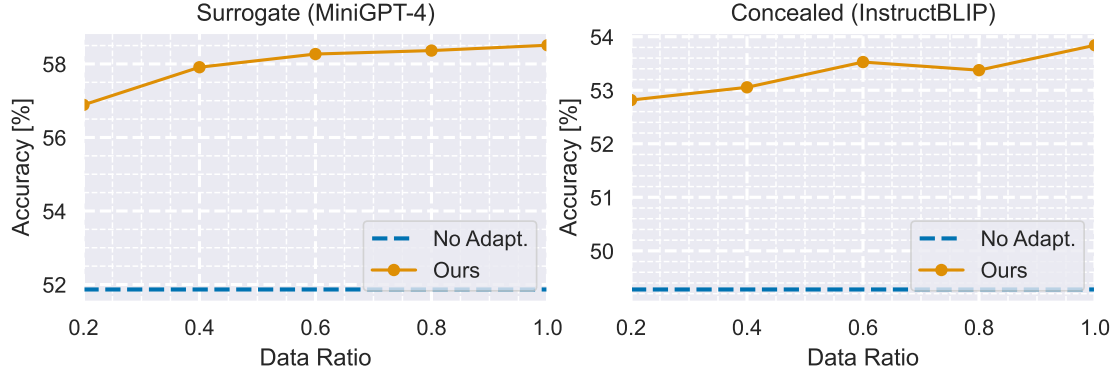


Figure 4.3: **Data-scale analysis.** MFDA-VP constantly showed improved performance from No Adapt. even with small data size.

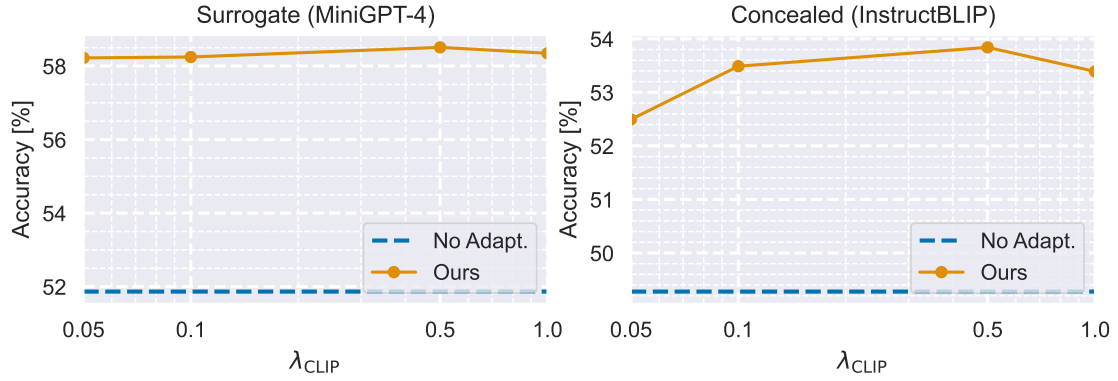


Figure 4.4: **Analysis on the balance of cross-model training.** MFDA-VP showed slight performance degradation when λ_{CLIP} increased while constantly showing improvements from No Adapt.

by changing λ_{CLIP} . Basically, our method showed improvement from No Adapt. regardless of the value of λ_{CLIP} . However, when λ_{CLIP} was small, there was quite a large drop in the concealed model’s performance. This indicates that the cross-model training between two different models is effective in enhancing the transferability of the visual prompts to the unseen models.

Analysis on Cross-model Gradient Alignment (Fig. 4.5). We analyzed the effectiveness of the gradient alignment between a surrogate MLLM and CLIP by changing α . CMGA worked better when α was set to small values, which was

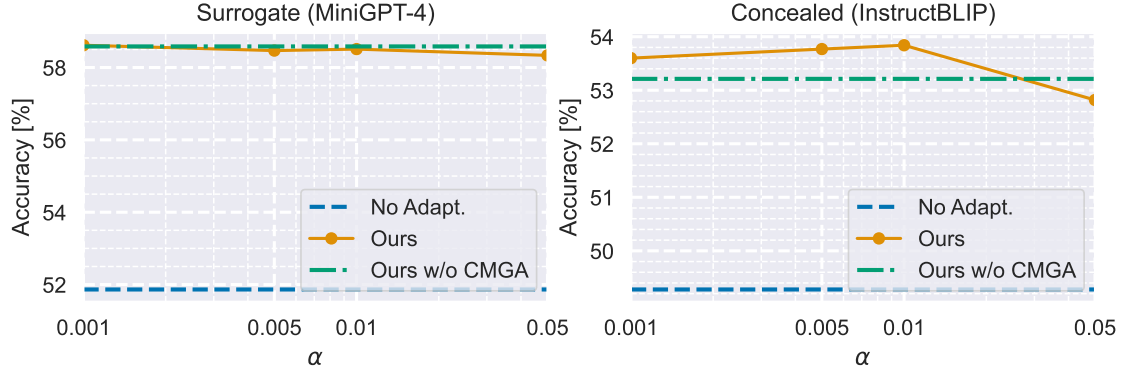


Figure 4.5: **Analysis on Cross-model Gradient Alignment.** CMGA worked well when α is set to small values.

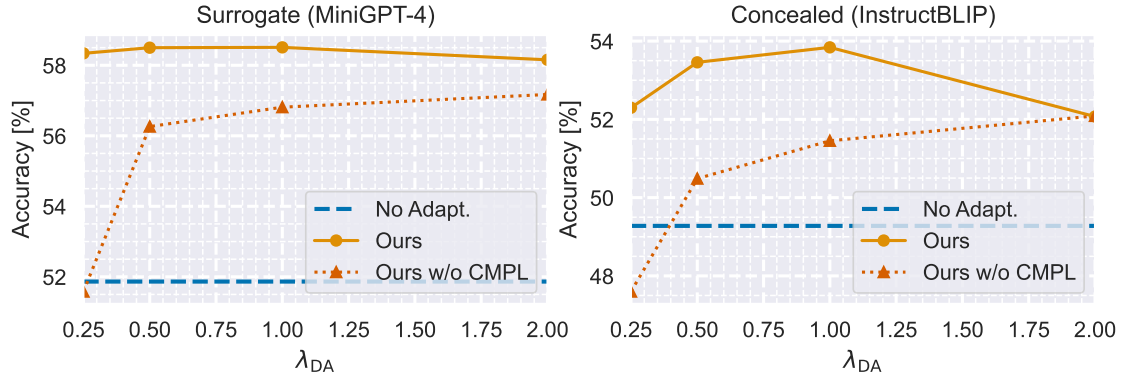


Figure 4.6: **Analysis on Domain Adaptation Training.** Our method slightly dropped its performance when λ_{DA} is small, while Ours w/o CMPL largely dropped.

in line with previous insights [153]. We found that setting a large value for α led to CMGA hindering the main training, resulting in a worse performance than ours w/o CMGA.

Analysis on Domain Adaptation Training (Fig. 4.6). We conducted a deeper analysis of the domain adaptation training by changing λ_{DA} . Our method showed a large performance drop with small λ_{DA} , especially w/o CMPL. As we have seen in the ablation study, DA was effective for avoiding collapse and giving domain-adaptability to the visual prompts. When comparing w/ and w/o

CMPL, ours w/ CMPL showed a milder drop. We guess this was because CMPL had domain adaptation effects similar to $\mathcal{L}_{\text{DA}}$ and reduced the performance drop. Meanwhile, when λ_{DA} was set to 2.0, the performance of ours w/ CMPL decreased. This is due to the fact that the proportion of the losses $\mathcal{L}_{\text{MLLM-GA}}$ and $\mathcal{L}_{\text{CLIP-GA}}$ to the total loss was decreased, leading to the reduced efficacy of CMPL.

4.6 Conclusion and Limitations

To resolve the domain adaptation problem for concealed multimodal large-language models (MLLMs), we investigated a novel and challenging setup called “model-free domain adaptation (MFDA)”, in which no resources are given from the concealed MLLMs. We proposed a pioneering method for solving MFDA, model-transferable domain-adaptable visual prompting (MTDA-VP), which executes cross-model visual prompting on the basis of the domain adaptation objective to equip the visual prompts with both model transferability and domain adaptability. We also developed two techniques to enhance these abilities of visual prompts: cross-model pseudo labeling (CMPL) and cross-model gradient alignment (CMGA). The experimental results and in-depth analysis support the validity of MTDA-VP. We expect that this study will open up a new perspective on domain adaptation.

Limitations. Since this is the first study to tackle MFDA, we focused on investigating whether MFDA is actually solvable, and we verified this with simple classification tasks and under controlled experimental settings. On the other hand, the investigations in more practical situations, such as validation in generative tasks (e.g., VQA) and application to commercial concealed models (e.g., GPT-5, Gemini), remain unexplored. However, even in such cases, our developed MTDA-VP

can be a promising base method for solving the problems, and we hope it will encourage various future research. Exploring extensions to such applications is a promising direction for our future research.

4.7 Appendix

4.7.1 Detailed Descriptions of Proposed Method

Efficient Prediction Technique for Multimodal Large-language Model.

To obtain a prediction from an MLLM F , we need to calculate the classification likelihood for each class y by

$$p_{\text{MLLM}}(y|\mathbf{X}, \mathbf{t}) = \prod_{i=1}^N F(o_i^y|\mathbf{X}, \mathbf{t}, o_{<i}^y). \quad (4.21)$$

However, this class likelihood calculation cannot be conducted in parallel, so it requires huge computational costs as the number of classes increases. To avoid this problem, we limit the calculation to a part of the whole class set by using CLIP prediction results. Specifically, we first obtain the classification likelihood for all classes $\{y\}_{y=1}^C$ as

$$p_{\text{CLIP}}(y|\mathbf{X}) = \sigma(\text{cosim}(E_{\text{vision}}(\mathbf{X}), E_{\text{text}}(\mathbf{t}_y))). \quad (4.22)$$

We then obtain the top- k class predictions of CLIP, i.e., $\text{top-}k_y[p_{\text{CLIP}}(y|X)]$. We obtain the classification results of the MLLM by calculating the likelihood only for

top- $k_y[p_{\text{CLIP}}(y|\mathbf{X})]$ classes; namely,

$$\hat{y}_{\text{MLLM}} = \underset{y' \in \text{top-}k_y[p_{\text{CLIP}}(y|\mathbf{X})]}{\text{argmax}} p_{\text{MLLM}}(y'|\mathbf{X}). \quad (4.23)$$

We set k to 5 during the prompt training. We also used this technique during the test phase by setting k to 10 to facilitate efficient research.

Label Propagation Algorithm. We used the label propagation algorithm described in a previous study [80], from which we basically borrowed the formulations. The label propagation algorithm $\mathcal{A}_{\text{LP}}$ is given a labeled set D_L , an unlabeled set D_U , and the visual feature vector set V as inputs. We first obtain a reliable label matrix Y using D_L and D_U as

$$Y_{ij} := \begin{cases} 1 & \text{if } (\mathbf{X}_i, \hat{y}_{\text{MLLM}_i}) \in D_L \wedge \hat{y}_{\text{MLLM}_i} = j, \\ 0 & \text{otherwise.} \end{cases} \quad (4.24)$$

We calculate a sparse affinity matrix A using $V = \{\mathbf{v}_i\}_{i=1}^N$ as

$$a_{ij} = \begin{cases} [\mathbf{v}_i^\top \mathbf{v}_j]_+^\gamma, & \text{if } i \neq j \wedge \mathbf{v}_i \in \text{NN}_m(\mathbf{v}_j) \\ 0, & \text{otherwise,} \end{cases} \quad (4.25)$$

where NN_m denotes the set of m nearest neighbors and γ denotes the parameter. We set $\gamma = 3$ for the default setting. We calculate a symmetric adjacency matrix W by $W = A + A^\top$ then, calculate the normalized counterpart as $\mathcal{W} = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, where D is a degree matrix $D = \text{diag}(W \mathbf{1}_N)$ and $\mathbf{1}_N$ is the all-ones vector. The

label-propagated results can be obtained by calculating

$$Z = (I - \beta\mathcal{W})^{-1}Y. \quad (4.26)$$

However, calculating the inverse matrix incurs high computational costs. Thus, we use the conjugated gradient method and solve the following problem:

$$(I - \beta\mathcal{W})Z = Y. \quad (4.27)$$

After obtaining the matrix Z , the soft label of sample i is calculated by

$$\hat{\mathbf{p}}_i = \sigma(\mathbf{z}_i/\tau_{\text{LP}}), \quad (4.28)$$

where τ_{LP} is the temperature parameter. We set $\tau_{\text{LP}} = 0.01$ by default.

In addition to the above label propagation algorithm, we use an approach to adjust the sample size. When we apply the label propagation algorithm to the whole dataset for each training epoch, we cannot fully exploit the training results in the label propagation due to the scarce execution frequency. When we apply the label propagation algorithm to each training batch, we cannot obtain satisfactory results due to the too small sample size. Therefore, we apply the label propagation algorithm to each chunk of several batches. We additionally introduce a parameter N_C to adjust the chunk size. Note that the data size N is calculated as $N = N_C \times B$, where B denotes the batch size.

Among the settings of the label propagation algorithm, the tunable hyper-parameters are the nearest neighbor size m and N_C . We set m to 30 and N_C to 300 for DomainNet experiments and m to 10 and N_C to 100 for PACS, considering

Table 4.5: **Detailed Comparison Results.** We highlighted in colors the cases where performance is **improved** and is **declined**. The best scores are highlighted in **bold**.

		<i>Performance of Surrogate MLLM: MiniGPT-4</i>						
Methods		Clipart	Infograph	Painting	Quickdraw	Real	Sketch	Average
0. No Adapt.	Accuracy	68.00	40.20	55.90	14.20	74.62	58.27	51.87
1. TVP [†] [200]	Accuracy	2.74±0.10	1.61±0.52	41.67±19.97	0.39±0.02	80.23±0.14	1.34±0.30	21.06±3.28
	Gain Δ	- 65.26	- 38.59	- 14.23	- 13.81	+ 5.61	- 56.93	- 30.54
2. TVP [†] +DA	Accuracy	72.94 ±0.27	42.50 ±0.20	64.00 ±0.24	19.64 ±0.01	80.58 ±0.11	63.90 ±0.18	57.26 ±0.02
	Gain Δ	+ 4.94	+ 2.30	+ 8.10	+ 5.44	+ 5.96	+ 5.63	+ 5.40
3. 2. + CMPL	Accuracy	73.40 ±0.16	43.78 ±0.04	65.11 ±0.18	23.44±0.24	81.00 ±0.23	64.77±0.14	58.58 ±0.03
	Gain Δ	+ 5.40	+ 3.58	+ 9.21	+ 9.24	+ 6.38	+ 6.50	+ 6.72
4. 2. + CMGA	Accuracy	72.76±0.17	41.77±0.56	63.39±0.41	19.30±0.26	80.25±0.13	63.39±0.33	56.81±0.08
	Gain Δ	+ 4.76	+ 1.57	+ 7.49	+ 5.10	+ 5.63	+ 5.12	+ 4.95
5. MTDA-VP	Accuracy	73.34±0.05	43.76±0.15	64.80±0.31	23.46 ±0.22	80.75±0.30	64.93 ±0.10	58.51±0.18
	Gain Δ	+ 5.34	+ 3.56	+ 8.90	+ 9.26	+ 6.13	+ 6.66	+ 6.64
		<i>Performance of Concealed MLLM: InstructBLIP</i>						
Methods		Clipart	Infograph	Painting	Quickdraw	Real	Sketch	Average
0. No Adapt.	Accuracy	66.04	36.16	52.09	13.26	73.64	54.48	49.28
1. TVP [†] [200]	Accuracy	45.04±4.14	19.47±1.78	41.48±13.39	7.37±0.19	74.06±0.38	16.61±3.16	34.00±2.48
	Gain Δ	- 21.00	- 16.69	- 10.61	- 5.89	+ 0.42	- 37.87	- 15.27
2. TVP [†] +DA	Accuracy	69.20±0.62	38.38±0.51	54.58±0.70	17.39±0.16	73.55±0.31	57.37±0.54	51.74±0.25
	Gain Δ	+ 3.16	+ 2.22	+ 2.49	+ 4.13	- 0.09	+ 2.89	+ 2.46
3. 2. + CMPL	Accuracy	69.73±0.42	41.07 ±0.04	57.82±0.35	18.87±0.17	73.53±0.59	58.25±1.28	53.21±0.23
	Gain Δ	+ 3.69	+ 4.91	+ 5.73	+ 5.61	- 0.11	+ 3.77	+ 3.93
4. 2. + CMGA	Accuracy	68.98±0.41	35.88±2.09	54.62±1.22	17.16±0.40	73.56±0.24	58.51±0.83	51.45±0.42
	Gain Δ	+ 2.94	- 0.28	+ 2.53	+ 3.90	- 0.08	+ 4.03	+ 2.17
5. MTDA-VP	Accuracy	70.12 ±0.38	40.05±0.56	59.42 ±0.98	20.28 ±0.42	74.24 ±0.82	58.92 ±0.13	53.84 ±0.28
	Gain Δ	+ 4.08	+ 3.89	+ 7.33	+ 7.02	+ 0.60	+ 4.44	+ 4.56

the whole dataset size.

4.7.2 Detailed Experimental Results

Comparison Results.

We provide the full version of Tab. 4.4 in Tab. 4.5. A new finding from this table is that the No.1 case had a significantly larger standard error than the others. We observed in the course of conducting our experiments that, in very few cases, the training normally progresses by chance, even in the absence of the domain

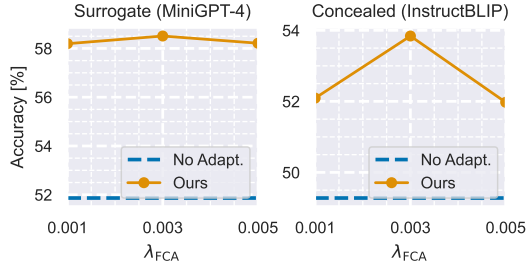


Figure 4.7: $\mathcal{L}_{FCA}$ analysis.

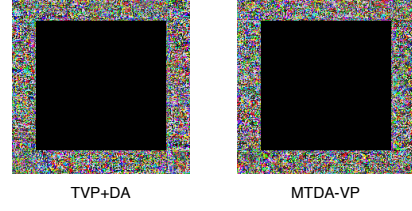


Figure 4.8: Qualitative analysis of prompts.

adaptation training. However, in most cases, the training failed, making it very unstable. This result indicates that the domain adaptation loss makes the training stable and successfully avoids prediction collapse.

A more in-depth look at the effect of CMGA shows that it varied with and without the use of CMPL. Specifically, the performance of the concealed model was slightly reduced by CMGA without CMPL, but CMGA with CMPL demonstrated an improvement in the concealed model. This indicates that a sufficient number of pseudo-labeled samples is required for CMGA to be effective. The samples utilized in the pseudo-label training without CMPL were limited to a part of the whole sample. The role of CMGA is to accelerate the pseudo-label training of two models in those samples, but the shortage of the label information may hinder the training as a whole. In fact, CMGA caused the performance degradation in the surrogate model in this case. Since CMPL provides high-quality pseudo labels for all samples, CMGA successfully demonstrates its full potential with CMPL.

Additional Analysis.

Quantitative Analysis of FCA. We provide additional analysis of our method, i.e., the quantitative analysis of the feature consistency alignment (FCA) and the qualitative analysis of the visual prompts. Since FCA was originally proposed in the

base method, TVP [200], we employed it as a default component. To confirm the effect of FCA on our method, we conducted the quantitative analysis by varying the size of λ_{FCA} . As shown in Fig. 4.7, it was beneficial to employ FCA to enhance the model transferability of the visual prompts, but we found that too strong FCA training might hinder the other training effect of our method.

Qualitative Analysis of the training visual prompts. We qualitatively analyzed the trained visual prompts by actually visualizing them (Fig. 4.8). Unfortunately, we could not find any interpretable difference between the prompts trained on different training objectives. We suppose that since the visual prompting techniques [6] are mechanically the same as the adversarial perturbations, the difference in the visual prompts is not interpretable to humans.

Analysis on the large model gap. The concealed MLLMs in the real-world applications are completely black-box, and there may be a large gap between them and the open-source model in terms of model components, such as the network architectures and training recipes. To verify the effectiveness of our method under such cases, we conducted the experiments by setting Qwen2-VL-7B [179] model as the concealed model, which is newer than the surrogate models (MiniGPT-4 and InstructBLIP) and has a different model structure. The experimental results using DomainNet dataset are shown in Tab. 4.6. The gain becomes smaller than in the smaller model gap cases, but the performance of the concealed model is still improved, demonstrating the effectiveness of our method. In this experiment, we validate the effectiveness of our method for Qwen2-VL by using it as the concealed model, but the experimental result suggests that the reversed pattern may also be effective (i.e., using Qwen2-VL as the surrogate model). Building more robust

Table 4.6: **Analysis on the large model gap.** We highlighted in colors the cases where performance is **improved** and is **declined**.

<i>Surrogate MLLM: MiniGPT-4</i>							
	C	I	P	Q	R	S	Ave.
No Aadpt.	45.97	26.80	38.46	10.76	56.57	40.93	36.58
MTDA-VP	45.76	30.63	42.03	14.05	54.34	42.30	38.19
Gain Δ	-0.21	+3.83	+3.57	+3.29	-2.23	+1.37	+1.60
<i>Surrogate MLLM: Instruct-BLIP</i>							
	C	I	P	Q	R	S	Ave.
No Aadpt.	45.97	26.80	38.46	10.76	56.57	40.93	36.58
MTDA-VP	45.76	30.69	42.15	13.72	54.04	42.56	38.15
Gain Δ	-0.21	+3.89	+3.69	+2.96	-2.53	+1.63	+1.57

algorithms against the model gaps will be our future work. In the practical cases, using an open-source MLLM that is expected to have a small model gap from the target concealed MLLM as the surrogate model could be one solution (e.g., using Gemma [165] as the surrogate MLLM when the concealed model is Gemini [164]).

4.7.3 Extension of MTDA-VP

As we have mentioned in the limitations, the MTDA-VP proposed in this work was verified with simple classification tasks. However, our method possesses sufficient potential as a good base method for solving various other tasks. For example, our method can be applicable to the generative tasks (e.g., VQA) with some minor modifications. Specifically, in CMPL, we use CLIP-Score [66] for quality assessment of the MLLM outputs to obtain reliable pseudo labels, and SFDA training in the CLIP branch can be substituted by image-text contrastive learning (positive pair loss = discriminability, negative pair loss = diversity), which is also employed in training the generative model like BLIP.

4.7.4 Broader Impacts

Our method primarily aims to improve the performance of the concealed MLLMs by adding visual prompts to the query image. From another viewpoint, this method modifies the image to make the models behave differently from the developer’s intent, which also applies to TVP [200] as this is the base of our method. As discussed in adversarial reprogramming (AR) [44], there are concerns that these could lead to ethical issues such as abuse of the models. However, unlike AR, our method (including TVP) does not directly use the model itself, so we consider that it is difficult to manipulate the model as desired, and the risk is small.

Chapter 5

Conclusion and Future Directions

This doctoral dissertation investigated resource constraint problems in domain adaptation and generalization technologies that arise from advances in deep learning techniques and changes in societal norms through the ages. Specifically, our work centered on overcoming the three types of resource constraints: inaccessibility of source domain data, difficulty in target data collection, and model access limitation. In this chapter, we summarize the works for each chapter and discuss the future directions of our work.

5.1 Conclusion

- **Chapter 2.** We tackled the source-free domain adaptation (SFDA) problem — adapting a source-trained model to the target domain only using unlabeled target data — with the view of providing a theoretical understanding of existing methods. We found that the major SFDA loss functions, comprising discriminability and diversity losses, work in the same way as the training objective in the theory of self-training based on the expansion as-

sumption [184]. Based on this observation, we derived two practical insights for the method improvements and proposed a novel SFDA method, Improved SFDA, comprising model training with auto-adjusting diversity constraint and augmentation training with a teacher-student framework. This work not only proposed a new superior SFDA method but also contributed to advancing our following works.

- **Chapter 3.** We tackled the source-free domain generalization (SFDG) problem, which aims to build a model generalizable to unseen target domains only with a pre-trained model. To overcome the rigidity of the existing methods that employ a domain-shared configuration for image inputs from any domain, we proposed domain prompt-discriminator collaborative learning, which simultaneously constructs a variety of domain-specific configurations to cover the wide variety of domains and a selector to choose the appropriate configuration for the input data. The design of our method also derived a simple and effective test-time adaptation technique and boosted the domain generalization performance. In addressing severe data constraints, this work presents a method that is simple yet superior to existing techniques, revealing new ways to leverage pre-trained models.
- **Chapter 4.** We proposed a novel model-free domain adaptation (MFDA) problem — improving the concealed multimodal large-language models without using any model resources (i.e., data used in the model training, model parameters, and model outputs) — to align with the recent trends that the cutting-edge deep learning models rarely provide their parameters and training data. As a proof of concept of MFDA, we built a method named model-

transferable domain-adaptable visual prompting. We combined a transferable visual prompting framework on two surrogate models with the insights of SFDA to construct visual prompts that exhibit domain adaptability to unseen concealed models. In addition, we developed two novel techniques that extend the effectiveness of visual prompting by leveraging a cross-model training strategy. This work has pioneered new directions in domain adaptation technology and produced valuable and foundational results for this field.

In conclusion, this dissertation presented new insights and practical solutions for resource constraint problems from both exploring (SFDA, SFDG) and unexplored perspectives (MFDA). Our findings extended the potential of domain adaptation and generalization technologies and demonstrated significant contributions to addressing the fundamentally unavoidable domain shift problem.

5.2 Future Directions

Finally, we discuss the future directions of resolving domain shift problems under resource constraints. We have already discussed the limitations and future prospects of each work in the respective chapters, so here we focus on the future prospects for this dissertation overall. One is the extension to different tasks and modalities. This research focused on the image classification task to address the resource challenge in each problem. However, even within computer vision tasks, there exist diverse tasks, such as VQA, object detection, dense prediction, etc. Furthermore, domain adaptation and generalization techniques are in high demand not only in computer vision but also across various modalities such as speech and

audio, and language. Building upon the core techniques proposed in this dissertation to tackle task and modality barriers will be a promising future direction of this work. Another direction of our future work is handling different types of resource constraints. This work tackled particularly data and model resource constraints, but other constraints on domain adaptation and generalization can arise when we review the problems from a different point of view. For example, from the viewpoint of device resources, we may be imposed several constraints such as memory usage, execution time, and processor performance, which can emerge in an edge computing scenario. A future prospect is to explore such a new perspective of resource constraints and combine the novel insights and findings in this work to expand the applicability of our proposed methods.

Chapter 6

List of Publications

1. Yu Mitsuzumi, Akisato Kimura, Hisashi Kashima. Understanding and Improving Source-free Domain Adaptation from a Theoretical Perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
2. Yu Mitsuzumi, Akisato Kimura, Hisashi Kashima. Flexible Source-free Domain Generalization via Domain Prompt-Discriminator Collaborative Learning. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2025.
3. 三鼓 悠, 木村 昭悟, 鹿島 久嗣. “ドメイン別プロンプトとドメイン推定器の同時学習によるソースフリースドメイン汎化”. 人工知能学会論文誌, 41(1), 2026.
4. Yu Mitsuzumi, Akisato Kimura, Hisashi Kashima. Model-free Domain Adaptation for Concealed Multimodal Large-Language Models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2026.

Others

1. Yu Mitsuzumi, Go Irie, Daiki Ikami, Takashi Shibata. Generalized Domain Adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
2. Takashi Shibata, Go Irie, Daiki Ikami, Yu Mitsuzumi. Learning with Selective Forgetting. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.
3. Yu Mitsuzumi, Go Irie, Akisato Kimura, Atsushi Nakazawa. Phase Randomization: A data augmentation for domain adaptation in human action recognition. *Pattern Recognition*, Volume 146, 2024.
4. Yu Mitsuzumi, Akisato Kimura, Go Irie, Atsushi Nakazawa. Cross-Action Cross-Subject Skeleton Action Recognition Via Simultaneous Action-Subject Learning With Two-Step Feature Removal. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2024.

References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. arXiv preprint arXiv:2303.08774, 2023.
- [2] Open AI. Introducing gpt-5 - resource — openai academy, 2025. Accessed on October 4th, 2025.
- [3] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. Advances in neural information processing systems, 35:23716–23736, 2022.
- [4] Anthropic. Introducing claude 4 anthropic, 2025. Accessed on October 4th, 2025.
- [5] Sercan Ö Arik and Tomas Pfister. Tabnet: Attentive interpretable tabular learning. In Proceedings of the AAAI conference on artificial intelligence, volume 35, pages 6679–6687, 2021.
- [6] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola.

- Exploring visual prompts for adapting large-scale models. arXiv preprint arXiv:2203.17274, 2022.
- [7] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271, 2018.
- [8] Yalong Bai, Yifan Yang, Wei Zhang, and Tao Mei. Directional self-supervised learning for heavy image augmentations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 16692–16701, 2022.
- [9] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. Advances in neural information processing systems, 31, 2018.
- [10] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. Advances in neural information processing systems, 31, 2018.
- [11] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. Machine learning, 79(1):151–175, 2010.
- [12] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. Advances in neural information processing systems, 19, 2006.

- [13] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. Advances in neural information processing systems, 32, 2019.
- [14] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258, 2021.
- [15] Konstantinos Bousmalis, Alex Irpan, Paul Wohlhart, Yunfei Bai, Matthew Kelcey, Mrinal Kalakrishnan, Laura Downs, Julian Ibarz, Peter Pastor, Kurt Konolige, et al. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In 2018 IEEE international conference on robotics and automation (ICRA), pages 4243–4250. IEEE, 2018.
- [16] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020.
- [17] Adrian Bulat and Georgios Tzimiropoulos. Lasp: Text-to-text optimization for language-aware soft prompting of vision & language models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 23232–23241, 2023.
- [18] Ruichu Cai, Zijian Li, Pengfei Wei, Jie Qiao, Kun Zhang, and Zhifeng Hao. Learning disentangled semantic representation for domain adapta-

- tion. In Proceedings of the 28th International Joint Conference on Artificial Intelligence, pages 2060–2066, 2019.
- [19] Fabio M Carlucci, Antonio D’Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 2229–2238, 2019.
- [20] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. Advances in Neural Information Processing Systems, 34:22405–22418, 2021.
- [21] 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.
- [22] Prithvijit Chattopadhyay, Yogesh Balaji, and Judy Hoffman. Learning to balance specificity and invariance for in and out of domain generalization. In European Conference on Computer Vision, pages 301–318. Springer, 2020.
- [23] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 295–305, 2022.
- [24] Lin Chen, Huaian Chen, Zhixiang Wei, Xin Jin, Xiao Tan, Yi Jin, and Enhong Chen. Reusing the task-specific classifier as a discriminator: Discriminator-free adversarial domain adaptation. In Proceedings of the

- IEEE/CVF conference on computer vision and pattern recognition, pages 7181–7190, 2022.
- [25] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In International conference on machine learning, pages 1081–1090. PMLR, 2019.
- [26] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3339–3348, 2018.
- [27] Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Semi-supervised learning for neural machine translation. In Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers), pages 1965–1974. 2016.
- [28] Florent Chiaroni, Malik Boudiaf, Amar Mitiche, and Ismail Ben Ayed. Simplex clustering via sbeta with applications to online adjustment of black-box predictions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 46(12):9123–9138, 2024.
- [29] Junhyeong Cho, Gilhyun Nam, Sungyeon Kim, Hunmin Yang, and Suha Kwak. Promptstyler: Prompt-driven style generation for source-free domain generalization. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 15702–15712, 2023.

- [30] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. Journal of Machine Learning Research, 24(240):1–113, 2023.
- [31] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 113–123, 2019.
- [32] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. Advances in Neural Information Processing Systems, 33:18613–18624, 2020.
- [33] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Liang Li, Qingming Huang, and Qi Tian. Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 3941–3950, 2020.
- [34] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. Advances in neural information processing systems, 36:49250–49267, 2023.
- [35] 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. IEEE, 2009.

- [36] Jun Deng, Zixing Zhang, Florian Eyben, and Björn Schuller. Autoencoder-based unsupervised domain adaptation for speech emotion recognition. IEEE Signal Processing Letters, 21(9):1068–1072, 2014.
- [37] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), pages 4171–4186, 2019.
- [38] Idit Diamant, Amir Rosenfeld, Idan Achituve, Jacob Goldberger, and Arnon Netzer. De-confusing pseudo-labels in source-free domain adaptation. In European Conference on Computer Vision, pages 108–125. Springer, 2024.
- [39] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 4312–4321, 2019.
- [40] 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. In International Conference on Learning Representations, 2021.
- [41] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. Advances in neural information processing systems, 32, 2019.

- [42] Chunling Du, Haifeng Sun, Jingyu Wang, Qi Qi, and Jianxin Liao. Adversarial and domain-aware bert for cross-domain sentiment analysis. In Proceedings of the 58th annual meeting of the Association for Computational Linguistics, pages 4019–4028, 2020.
- [43] Yuntao Du, Haiyang Yang, Mingcai Chen, Hongtao Luo, Juan Jiang, Yi Xin, and Chongjun Wang. Generation, augmentation, and alignment: A pseudo-source domain based method for source-free domain adaptation. Machine Learning, 113(6):3611–3631, 2024.
- [44] Gamaleldin F Elsayed, Ian Goodfellow, and Jascha Sohl-Dickstein. Adversarial reprogramming of neural networks. In International Conference on Learning Representations, 2019.
- [45] European Parliament and Council of the European Union. General data protection regulation (gdpr), 2016. Regulation (EU) 2016/679.
- [46] Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In Proceedings of the IEEE international conference on computer vision, pages 1657–1664, 2013.
- [47] Yuqi Fang, Pew-Thian Yap, Weili Lin, Hongtu Zhu, and Mingxia Liu. Source-free unsupervised domain adaptation: A survey. Neural Networks, 174:106230, 2024.
- [48] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky.

- Domain-adversarial training of neural networks. Journal of machine learning research, 17(59):1–35, 2016.
- [49] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. International Journal of Computer Vision, 132(2):581–595, 2024.
- [50] Daniel Garcia-Romero, Alan McCree, Stephen Shum, Niko Brummer, and Carlos Vaquero. Unsupervised domain adaptation for i-vector speaker recognition. In Proceedings of Odyssey: The Speaker and Language Recognition Workshop, volume 8, 2014.
- [51] Chunjiang Ge, Rui Huang, Mixue Xie, Zihang Lai, Shiji Song, Shuang Li, and Gao Huang. Domain adaptation via prompt learning. IEEE Transactions on Neural Networks and Learning Systems, 2023.
- [52] Shayan Gharib, Konstantinos Drossos, Emre Cakir, Dmitriy Serdyuk, and Tuomas Virtanen. Unsupervised adversarial domain adaptation for acoustic scene classification. pages 138–142, 2018.
- [53] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In European conference on computer vision, pages 597–613. Springer, 2016.
- [54] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. In Proc. Interspeech 2021, pages 571–575, 2021.

- [55] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. Deep learning, volume 1. MIT press Cambridge, 2016.
- [56] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. Advances in neural information processing systems, 27, 2014.
- [57] Google. Google translator. Accessed on October 11th, 2025.
- [58] Google. Veo - google deepmind. Accessed on October 11th, 2025.
- [59] Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. Advances in neural information processing systems, 34:18932–18943, 2021.
- [60] Chenhui Gou, Abdulwahab Felemban, Faizan Farooq Khan, Deyao Zhu, Jianfei Cai, Hamid Rezatofighi, and Mohamed Elhoseiny. How well can vision language models see image details? arXiv preprint arXiv:2408.03940, 2024.
- [61] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. Conformer: Convolution-augmented transformer for speech recognition. In Proc. Interspeech 2020, pages 5036–5040, 2020.
- [62] Nicklas Hansen and Xiaolong Wang. Generalization in reinforcement learning by soft data augmentation. In 2021 IEEE International Conference on Robotics and Automation (ICRA), pages 13611–13617. IEEE, 2021.

- [63] Ryuichiro Hataya, Jan Zdenek, Kazuki Yoshizoe, and Hideki Nakayama. Faster autoaugment: Learning augmentation strategies using backpropagation. In European conference on computer vision, pages 1–16. Springer, 2020.
- [64] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In Proceedings of the IEEE international conference on computer vision, pages 2961–2969, 2017.
- [65] 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.
- [66] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 7514–7528, 2021.
- [67] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal processing magazine, 29(6):82–97, 2012.
- [68] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.
- [69] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adver-

- serial domain adaptation. In International conference on machine learning, pages 1989–1998. PMLR, 2018.
- [70] Weixiang Hong, Zhenzhen Wang, Ming Yang, and Junsong Yuan. Conditional generative adversarial network for structured domain adaptation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1335–1344, 2018.
- [71] Yunzhong Hou and Liang Zheng. Source free domain adaptation with image translation. arXiv preprint arXiv:2008.07514, 2020.
- [72] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In International conference on machine learning, pages 2790–2799. PMLR, 2019.
- [73] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In International Conference on Learning Representations, 2022.
- [74] Weihua Hu, Takeru Miyato, Seiya Tokui, Eiichi Matsumoto, and Masashi Sugiyama. Learning discrete representations via information maximizing self-augmented training. In International conference on machine learning, pages 1558–1567. PMLR, 2017.
- [75] Wenbo Hu, Yifan Xu, Yi Li, Weiyue Li, Zeyuan Chen, and Zhuowen Tu. Bliva: A simple multimodal llm for better handling of text-rich visual questions. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pages 2256–2264, 2024.

- [76] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4700–4708, 2017.
- [77] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In European conference on computer vision, pages 124–140. Springer, 2020.
- [78] Maximilian Ilse, Jakub M Tomczak, Christos Louizos, and Max Welling. Diva: Domain invariant variational autoencoders. In Medical Imaging with Deep Learning, pages 322–348. PMLR, 2020.
- [79] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning, pages 448–456. PMLR, 2015.
- [80] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 5070–5079, 2019.
- [81] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In International conference on machine learning, pages 4904–4916. PMLR, 2021.

- [82] 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.
- [83] Nazmul Karim, Niluthpol Chowdhury Mithun, Abhinav Rajvanshi, Hang-pang Chiu, Supun Samarasekera, and Nazanin Rahnavard. C-sfda: A curriculum learning aided self-training framework for efficient source free domain adaptation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 24120–24131, 2023.
- [84] Muhammad Uzair Khattak, Muhammad Ferjad Naeem, Muzammal Naseer, Luc Van Gool, and Federico Tombari. Learning to prompt with text only supervision for vision-language models. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 39, pages 4230–4238, 2025.
- [85] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 19113–19122, 2023.
- [86] Daehee Kim, Youngjun Yoo, Seunghyun Park, Jinkyu Kim, and Jaekoo Lee. Selfreg: Self-supervised contrastive regularization for domain generalization. In Proceedings of the IEEE/CVF international conference on computer vision, pages 9619–9628, 2021.
- [87] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, 2012.

- [88] Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In International conference on machine learning, pages 5468–5479. PMLR, 2020.
- [89] Misha Laskin, Kimin Lee, Adam Stooke, Lerrel Pinto, Pieter Abbeel, and Aravind Srinivas. Reinforcement learning with augmented data. Advances in neural information processing systems, 33:19884–19895, 2020.
- [90] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. nature, 521(7553):436–444, 2015.
- [91] Jonghyun Lee, Dahuin Jung, Junho Yim, and Sungroh Yoon. Confidence score for source-free unsupervised domain adaptation. In International conference on machine learning, pages 12365–12377. PMLR, 2022.
- [92] Seunghun Lee, Sunghyun Cho, and Sunghoon Im. Dranet: Disentangling representation and adaptation networks for unsupervised cross-domain adaptation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 15252–15261, 2021.
- [93] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3045–3059, 2021.
- [94] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In Proceedings of the AAAI conference on artificial intelligence, volume 32, 2018.

- [95] 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.
- [96] 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.
- [97] Jingjing Li, Zhiqi Yu, Zhekai Du, Lei Zhu, and Heng Tao Shen. A comprehensive survey on source-free domain adaptation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 46(8):5743–5762, 2024.
- [98] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In International conference on machine learning, pages 19730–19742. PMLR, 2023.
- [99] Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.
- [100] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In European Conference on Computer Vision, pages 647–663. Springer, 2018.

- [101] Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. International Journal of Computer Vision, 133(1):31–64, 2025.
- [102] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In International conference on machine learning, pages 6028–6039. PMLR, 2020.
- [103] Jian Liang, Dapeng Hu, Jiashi Feng, and Ran He. Dine: Domain adaptation from single and multiple black-box predictors. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 8003–8013, 2022.
- [104] Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. Advances in Neural Information Processing Systems, 35:17612–17625, 2022.
- [105] Mattia Litrico, Alessio Del Bue, and Pietro Morerio. Guiding pseudo-labels with uncertainty estimation for source-free unsupervised domain adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 7640–7650, 2023.
- [106] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024.

- [107] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. Advances in neural information processing systems, 36:34892–34916, 2023.
- [108] Hong Liu, Jianmin Wang, and Mingsheng Long. Cycle self-training for domain adaptation. Advances in Neural Information Processing Systems, 34:22968–22981, 2021.
- [109] Yuejiang Liu, Parth Kothari, Bastien Van Delft, Baptiste Bellot-Gurlet, Taylor Mordan, and Alexandre Alahi. Ttt++: When does self-supervised test-time training fail or thrive? Advances in Neural Information Processing Systems, 34:21808–21820, 2021.
- [110] Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. Crossner: Evaluating cross-domain named entity recognition. In Proceedings of the AAAI conference on artificial intelligence, volume 35, pages 13452–13460, 2021.
- [111] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In International conference on machine learning, pages 97–105. PMLR, 2015.
- [112] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. Advances in neural information processing systems, 31, 2018.
- [113] 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.

- [114] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In International Conference on Learning Representations, 2019.
- [115] Chengcheng Ma, Yang Liu, Jiankang Deng, Lingxi Xie, Weiming Dong, and Changsheng Xu. Understanding and mitigating overfitting in prompt tuning for vision-language models. IEEE Transactions on Circuits and Systems for Video Technology, 33(9):4616–4629, 2023.
- [116] Massimiliano Mancini, Samuel Rota Buló, 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.
- [117] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. arXiv preprint arXiv:0902.3430, 2009.
- [118] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. In The Eleventh International Conference on Learning Representations, 2022.
- [119] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, 2013.
- [120] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM, 65(1):99–106, 2021.

- [121] Yu Mitsuzumi, Akisato Kimura, and Hisashi Kashima. Understanding and improving source-free domain adaptation from a theoretical perspective. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 28515–28524, 2024.
- [122] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1765–1773, 2017.
- [123] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In International conference on machine learning, pages 10–18. PMLR, 2013.
- [124] A Tuan Nguyen, Toan Tran, Yarin Gal, Philip Torr, and Atilim Gunes Baydin. Kl guided domain adaptation. In International Conference on Learning Representations, 2022.
- [125] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In The Eleventh International Conference on Learning Representations, 2023.
- [126] Hongjing Niu, Hanting Li, Feng Zhao, and Bin Li. Domain-unified prompt representations for source-free domain generalization. arXiv preprint arXiv:2209.14926, 2022.
- [127] Changdae Oh, Hyeji Hwang, Hee-young Lee, YongTaek Lim, Geunyoung Jung, Jiyoung Jung, Hosik Choi, and Kyungwoo Song. Blackvip: Black-box visual prompting for robust transfer learning. In Proceedings of the

- IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 24224–24235, 2023.
- [128] OpenAI. Sora — openai. Accessed on October 11th, 2025.
- [129] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In Proceedings of the IEEE/CVF international conference on computer vision, pages 1406–1415, 2019.
- [130] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. arXiv preprint arXiv:1710.06924, 2017.
- [131] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (Long Papers), pages 2227–2237, 2018.
- [132] Viet Hoang Phan, Tung Lam Tran, Quyen Tran, and Trung Le. Enhancing domain adaptation through prompt gradient alignment. Advances in Neural Information Processing Systems, 37:45518–45551, 2024.
- [133] Sergei Popov, Stanislav Morozov, and Artem Babenko. Neural oblivious decision ensembles for deep learning on tabular data. In International Conference on Learning Representations, 2020.

- [134] Sarah Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. In Proceedings of the IEEE/CVF international conference on computer vision, pages 15691–15701, 2023.
- [135] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 12556–12565, 2020.
- [136] Zhen Qiu, Yifan Zhang, Hongbin Lin, Shuaicheng Niu, Yanxia Liu, Qing Du, and Mingkui Tan. Source-free domain adaptation via avatar prototype generation and adaptation. In Proceedings of the 30th International Joint Conference on Artificial Intelligence, pages 2921–2927, 2021.
- [137] Sanqing Qu, Guang Chen, Jing Zhang, Zhijun Li, Wei He, and Dacheng Tao. Bmd: A general class-balanced multicentric dynamic prototype strategy for source-free domain adaptation. In European conference on computer vision, pages 165–182. Springer, 2022.
- [138] Sanqing Qu, Tianpei Zou, Lianghua He, Florian Röhrbein, Alois Knoll, Guang Chen, and Changjun Jiang. Lead: Learning decomposition for source-free universal domain adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 23334–23343, 2024.
- [139] 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.
- [140] Partha Pratim Ray. Chatgpt: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. Internet of Things and Cyber-Physical Systems, 3:121–154, 2023.
- [141] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 779–788, 2016.
- [142] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10684–10695, 2022.
- [143] Karsten Roth, Jae Myung Kim, Andrew Koepke, Oriol Vinyals, Cordelia Schmid, and Zeynep Akata. Waffling around for performance: Visual classification with random words and broad concepts. In Proceedings of the IEEE/CVF international conference on computer vision, pages 15746–15757, 2023.
- [144] Subhankar Roy, Martin Trapp, Andrea Pilzer, Juho Kannala, Nicu Sebe, Elisa Ricci, and Arno Solin. Uncertainty-guided source-free domain adaptation. In European conference on computer vision, pages 537–555. Springer, 2022.
- [145] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting vi-

- sual category models to new domains. In European conference on computer vision, pages 213–226. Springer, 2010.
- [146] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 6956–6965, 2019.
- [147] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3723–3732, 2018.
- [148] Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. Advances in neural information processing systems, 29, 2016.
- [149] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. wav2vec: Unsupervised pre-training for speech recognition. In Proc. Interspeech 2019, pages 3465–3469, 2019.
- [150] DeepL SE. Deepl translate. Accessed on October 11th, 2025.
- [151] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In International Conference on Learning Representations, 2018.
- [152] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri,

- Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In International Conference on Learning Representations, 2018.
- [153] Yuge Shi, Jeffrey Seely, Philip Torr, Awni Hannun, Nicolas Usunier, Gabriel Synnaeve, et al. Gradient matching for domain generalization. In International Conference on Learning Representations, 2022.
- [154] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. Advances in Neural Information Processing Systems, 35:14274–14289, 2022.
- [155] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. Advances in neural information processing systems, 33:596–608, 2020.
- [156] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In European conference on computer vision, pages 443–450. Springer, 2016.
- [157] Tianxiang Sun, Zhengfu He, Hong Qian, Yunhua Zhou, Xuan-Jing Huang, and Xipeng Qiu. Bbtv2: Towards a gradient-free future with large language models. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 3916–3930, 2022.

- [158] Tianxiang Sun, Yunfan Shao, Hong Qian, Xuanjing Huang, and Xipeng Qiu. Black-box tuning for language-model-as-a-service. In International Conference on Machine Learning, pages 20841–20855. PMLR, 2022.
- [159] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In International conference on machine learning, pages 9229–9248. PMLR, 2020.
- [160] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. Advances in neural information processing systems, 27, 2014.
- [161] Teppei Suzuki. Techaugment: Data augmentation optimization using teacher knowledge. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10904–10914, 2022.
- [162] Yunlong Tang, Yuxuan Wan, Lei Qi, and Xin Geng. Dpstyler: dynamic promptstyler for source-free domain generalization. IEEE Transactions on Multimedia, 2025.
- [163] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. Advances in neural information processing systems, 30, 2017.
- [164] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805, 2023.

- [165] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. arXiv preprint arXiv:2403.08295, 2024.
- [166] Jiayi Tian, Jing Zhang, Wen Li, and Dong Xu. Vdm-da: Virtual domain modeling for source data-free domain adaptation. IEEE Transactions on Circuits and Systems for Video Technology, 32(6):3749–3760, 2021.
- [167] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS), pages 23–30. IEEE, 2017.
- [168] Devavrat Tomar, Guillaume Vray, Behzad Bozorgtabar, and Jean-Philippe Thiran. Tesla: Test-time self-learning with automatic adversarial augmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 20341–20350, 2023.
- [169] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971, 2023.
- [170] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7167–7176, 2017.

- [171] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. arXiv preprint arXiv:1412.3474, 2014.
- [172] Vishaal Udandaraao, Ankush Gupta, and Samuel Albanie. Sus-x: Training-free name-only transfer of vision-language models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 2725–2736, 2023.
- [173] Aaron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu, et al. Wavenet: A generative model for raw audio. arXiv preprint arXiv:1609.03499, 12:1, 2016.
- [174] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. Machine learning, 109(2):373–440, 2020.
- [175] 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.
- [176] 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.
- [177] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial

- data augmentation. Advances in neural information processing systems, 31, 2018.
- [178] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In International Conference on Learning Representations, 2021.
- [179] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. arXiv preprint arXiv:2409.12191, 2024.
- [180] Pengfei Wang, Zhaoxiang Zhang, Zhen Lei, and Lei Zhang. Sharpness-aware gradient matching for domain generalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3769–3778, 2023.
- [181] Shujun Wang, Lequan Yu, Caizi Li, Chi-Wing Fu, and Pheng-Ann Heng. Learning from extrinsic and intrinsic supervisions for domain generalization. In European Conference on Computer Vision, pages 159–176. Springer, 2020.
- [182] Zijian Wang, Yadan Luo, Ruihong Qiu, Zi Huang, and Mahsa Baktashmotlagh. Learning to diversify for single domain generalization. In Proceedings of the IEEE/CVF international conference on computer vision, pages 834–843, 2021.
- [183] Waymo. Waymo - self-driving cars - autonomous vehicles - ride-hail. Accessed on October 11th, 2025.

- [184] Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data. In International Conference on Learning Representations, 2020.
- [185] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in neural information processing systems, 34:22419–22430, 2021.
- [186] Junyang Wu, Xianhang Li, Chen Wei, Huiyu Wang, Alan Yuille, Yuyin Zhou, and Cihang Xie. Unleashing the power of visual prompting at the pixel level. Transactions on Machine Learning Research, 2024.
- [187] Haifeng Xia, Handong Zhao, and Zhengming Ding. Adaptive adversarial network for source-free domain adaptation. In Proceedings of the IEEE/CVF international conference on computer vision, pages 9010–9019, 2021.
- [188] Zhenlin Xu, Deyi Liu, Junlin Yang, Colin Raffel, and Marc Niethammer. Robust and generalizable visual representation learning via random convolutions. In International Conference on Learning Representations, 2020.
- [189] Jinyu Yang, Chunyuan Li, Weizhi An, Hehuan Ma, Yuzhi Guo, Yu Rong, Peilin Zhao, and Junzhou Huang. Exploring robustness of unsupervised domain adaptation in semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 9194–9203, 2021.
- [190] Shiqi Yang, Shangling Jui, Joost Van De Weijer, et al. Attracting and dispersing: A simple approach for source-free domain adaptation. Advances in Neural Information Processing Systems, 35:5802–5815, 2022.

- [191] Shiqi Yang, Joost Van de Weijer, Luis Herranz, Shangling Jui, et al. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. Advances in neural information processing systems, 34:29393–29405, 2021.
- [192] Shiqi Yang, Yaxing Wang, Joost Van De Weijer, Luis Herranz, and Shangling Jui. Generalized source-free domain adaptation. In Proceedings of the IEEE/CVF international conference on computer vision, pages 8978–8987, 2021.
- [193] Hantao Yao, Rui Zhang, and Changsheng Xu. Visual-language prompt tuning with knowledge-guided context optimization. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 6757–6767, 2023.
- [194] Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In Proceedings of the AAAI conference on artificial intelligence, volume 36, pages 8980–8987, 2022.
- [195] Haoran Zhang, Shuanghao Bai, Wanqi Zhou, Jingwen Fu, and Badong Chen. Promptta: Prompt-driven text adapter for source-free domain generalization. In ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1–5. IEEE, 2025.
- [196] Jingyi Zhang, Jiaying Huang, Xueying Jiang, and Shijian Lu. Black-box unsupervised domain adaptation with bi-directional atkinson-shiffrin memory. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 11771–11782, 2023.

- [197] Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of clip for few-shot classification. In European conference on computer vision, pages 493–510. Springer, 2022.
- [198] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3801–3809, 2018.
- [199] Xingxuan Zhang, Jiansheng Li, Wenjing Chu, Junjia Hai, Renzhe Xu, Yuqing Yang, Shikai Guan, Jiazheng Xu, and Peng Cui. On the out-of-distribution generalization of multimodal large language models. arXiv preprint arXiv:2402.06599, 2024.
- [200] Yichi Zhang, Yinpeng Dong, Siyuan Zhang, Tianzan Min, Hang Su, and Jun Zhu. Exploring the transferability of visual prompting for multimodal large language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 26562–26572, 2024.
- [201] Yixin Zhang, Zilei Wang, and Weinan He. Class relationship embedded learning for source-free unsupervised domain adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 7619–7629, 2023.
- [202] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In International conference on machine learning, pages 7404–7413. PMLR, 2019.

- [203] Ziyi Zhang, Weikai Chen, Hui Cheng, Zhen Li, Siyuan Li, Liang Lin, and Guanbin Li. Divide and contrast: Source-free domain adaptation via adaptive contrastive learning. Advances in Neural Information Processing Systems, 35:5137–5149, 2022.
- [204] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In International conference on machine learning, pages 7523–7532. PMLR, 2019.
- [205] Dengyong Zhou, Olivier Bousquet, Thomas Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. Advances in neural information processing systems, 16, 2003.
- [206] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In Proceedings of the AAAI conference on artificial intelligence, volume 35, pages 11106–11115, 2021.
- [207] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. IEEE transactions on pattern analysis and machine intelligence, 45(4):4396–4415, 2022.
- [208] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 16816–16825, 2022.
- [209] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning

- to prompt for vision-language models. International Journal of Computer Vision, 130(9):2337–2348, 2022.
- [210] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In International Conference on Learning Representations, 2020.
- [211] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain adaptive ensemble learning. IEEE Transactions on Image Processing, 30:8008–8018, 2021.
- [212] Wen Zhou, Xin Hou, Yongjun Chen, Mengyun Tang, Xiangqi Huang, Xiang Gan, and Yong Yang. Transferable adversarial perturbations. In European Conference on Computer Vision, pages 471–486. Springer, 2018.
- [213] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. In International Conference on Learning Representations, 2024.
- [214] Xinge Zhu, Jiangmiao Pang, Ceyuan Yang, Jianping Shi, and Dahua Lin. Adapting object detectors via selective cross-domain alignment. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 687–696, 2019.
- [215] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In Proceedings of the IEEE/CVF international conference on computer vision, pages 5982–5991, 2019.

- [216] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In European Conference on Computer Vision, pages 297–313. Springer, 2018.