INTERPRETABLE VISUAL DOMAIN ADAPTATION FROM FEATURE REPRESENTATION TO MULTI-MODAL SEMANTICS

AN ABSTRACT
SUBMITTED ON THE FIFTH DAY OF AUGUST 2023
TO THE DEPARTMENT OF COMPUTER SCIENCE
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
OF THE SCHOOL OF SCIENCE AND ENGINEERING
OF TULANE UNIVERSITY
FOR THE DEGREE
OF
DOCTOR OF PHILOSOPHY
BY

Taotao Jing

APPROVED: Zhengming Ding

Zhengming Ding, Ph.D.
Director

Jihun Hamm

Jihun Hamm, Ph.D.

Aron Culotta, Ph.D.

Renran Tian

Renran Tian, Ph.D.

Thesis advisor: Professor Zhengming (Allan) Ding Taotao (Scott) Jing

Interpretable Visual Domain Adaptation from Feature Representation to Multi-modal Semantics

Abstract

Transfer learning has revolutionized the field of deep learning, allowing the utilization of pretrained models to address challenges such as limited training data and expensive computational resources. However, the lack of interpretability and transparency in transfer learning methods poses significant obstacles to their practical deployment and trustworthiness. This doctoral dissertation is dedicated to enhancing the transparency and interpretability of visual domain adaptation, a critical task of transfer learning, encompassing feature representation analysis and integration of multimodal semantic knowledge. By addressing the cross-domain shift and providing human-friendly explanations simultaneously, this research aims to provide deeper insights into the transfer learning process and facilitate more interpretable and trustworthy outcomes for real-world applications.

We start by analyzing the distribution of learned **feature representations** in visual domain adaptation tasks with solely visual images available, to gain valuable insights into the transfer of knowledge across different domains. By visualizing the learned features in the domain-invariant feature space, we can observe how the boundaries between task-specific categories align in unsupervised domain adaptation tasks. These insights derived from the analysis contribute to our efforts in addressing partial domain adaptation by measuring the similarities between features and filtering out outlier categories and also support us in tackling fairness issues in imbalanced domain adaptation with limited training data through the utilization of various feature generation strategies.

Moreover, we seek to utilize **high-level semantic knowledge** such as textual descriptions in addition to images to enhance the explanations of domain adaptation. In this regard, we introduce the Semantic-Recovery Open-Set Domain Adaptation (SR-OSDA) problem and propose a solution to recover semantic descriptions for unseen categories in the target domain while accurately identifying seen categories. By combining textual and visual data, we efficiently discover novel target classes and provide human-friendly explanations with semantic attribute prediction.

Furthermore, in order to elucidate the inner workings of convolutional networks for visual feature extraction to enrich the high-level semantic explanation, we propose an interpretable driving decision-making model which employs **learnable concept-based visual prototypes** to identify the crucial regions and objects in ego-view images for driving actions and align the learned semantic prototypes with human annotations to enable interpretable driving decision-making.

Finally, this dissertation presents an Interpretable Novel Target Discovery model that addresses the SR-OSDA problem by combining **interpretation strategies and multimodal semantic knowledge**. The model achieves interpretation through human-friendly multimodal semantic concept-based visual prototypes and analysis of feature representations. The research provides valuable insights for integrating AI systems across domains, promoting transparency, interpretability, and trustworthiness in decision-making. Overall, it contributes to the development of interpretable transfer learning techniques, enhancing the understanding and practical application of deep learning models, and fostering transparent and collaborative human-AI interactions.

INTERPRETABLE VISUAL DOMAIN ADAPTATION FROM FEATURE REPRESENTATION TO MULTI-MODAL SEMANTICS

A DISSERTATION
SUBMITTED ON THE FIFTH DAY OF AUGUST 2023
TO THE DEPARTMENT OF COMPUTER SCIENCE
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
OF THE SCHOOL OF SCIENCE AND ENGINEERING
OF TULANE UNIVERSITY
FOR THE DEGREE
OF
DOCTOR OF PHILOSOPHY
BY

Taotao Jing

APPROVED: Zhengming Ding
Zhengming Ding, Ph.D.
Director

Jihun Hamm
Jihun Hamm, Ph.D.

Aron Culotta, Ph.D.

Renran Tian
Renran Tian, Ph.D.

To my family.

Acknowledgments

The year 2023 marks the significant milestone of my 7th year in the United States since I landed at Boston Logan International Airport in 2016, embarking on a new chapter in my life. Stepping into a foreign country with uncertainty and excitement, I had little idea of what the future held for me. Over the course of these seven years, my journey has taken me from Boston to Indianapolis and then to New Orleans, from pursuing a Master's degree to completing a Ph.D., and from the age of 23 to nearly 30. This path has been long and riddled with challenges, fears, disappointments, and moments of doubt, but it has also been enriched with numerous acts of help, understanding, support, and guidance, which I am deeply grateful for.

My heartfelt gratitude goes to my advisor, Professor Zhengming (Allan) Ding, whose belief in my potential opened the door to my Ph.D. journey. He patiently taught, guided, and mentored me, transforming me from a novice to a confident researcher. Under his guidance, I learned the essence of research, developed professionally, and worked as a team to address various problems and challenges. Together, we weathered the storms of the pandemic and Hurricane Ida, and I am truly grateful to have had Professor Ding as both my Ph.D. and life advisor.

I extend my thanks to all my committee members, Professor Jihun Hamm, Professor Aron Culotta, and Professor Renran Tian, for their unwavering support, valuable time, and insightful feedback, which played a crucial role in the success of my dissertation. Moreover, I am grateful to the numerous professors, mentors, and labmates who generously aided me throughout these years, contributing to my growth as a scholar and as an individual. Special thanks go to my roommate Haifeng and all my labmates, with whom I shared the challenges of living in a foreign country and engaged in stimulating discussions on diverse and fascinating research topics. The intellectual sparks that flew when our minds met were truly enriching.

Lastly, I want to express my profound appreciation to my parents and sister. Their unconditional love and support have been a guiding force throughout my life. Their encouragement and belief in me have been invaluable as I ventured to study for my M.S. and Ph.D. in a country far across the Pacific Ocean. Additionally, I am immensely grateful to my lifelong friends from JUSHUIGE, who have been there for me whenever I needed a listening ear, assistance, and unwavering support. Over the past decade, they have become like family to me, and I credit them for helping me endure and persevere through challenging times. Without the love and support of my family and friends, I wouldn't have been able to overcome obstacles and achieve my academic goals.

Contents

0

Introduction

0.1 Background

The rapid advancements in machine learning (ML) and deep learning (DL) have revolutionized numerous domains, such as computer vision and natural language processing, by showcasing remarkable breakthroughs thanks to the unprecedented capabilities of DL in understanding and extracting information from vast and complex datasets [35, 16]. In recent times, the impressive performances of diffusion models in computer vision and Large Language Models (LLM) in natural language processing have garnered significant attention from both industry and academia to AI-Generated Content (AIGC) [94, 108]. However, the practical applications of those large-scale DL models are hindered by the inherent demands for large amounts of well-annotated training data and expensive computing resources.

For instance, the training of GPT-4 relied on text databases sourced from the internet, including about 570 GB of data, comprising 300 billion words extracted from books, webtexts, Wikipedia, articles, and other written sources available online [94]. Unfortunately, during the early days of the Covid-19 pandemic, the lack of well-annotated chest X-ray data from patients posed a significant challenge, limiting the application of machine-learning techniques in disease detection. Moreover, the training process of the Stable-Diffusion model necessitated the utilization of 256 Nvidia A100 GPUs on Amazon Web Services, amounting to a total of 150,000 GPU-hours [108]. This substantial computational requirement poses a barrier to deploying such large-scale models on edge devices with limited processing power, such as cell phones and VR/AR headsets.

Transfer learning aims to overcome the limitations of machine learning and deep learning by reducing training costs and minimizing reliance on large-scale training data, which can be difficult to obtain [65, 77, 15]. This research area has been significant even before the emergence of deep learning in the last two decades. A key challenge and objective of transfer learning is to address the differences in data distribution between the source data used for training and the target data used for testing, known as "domain shift." Based on the specific problems being addressed, transfer learning methods can be broadly categorized into two branches: **data distribution disparity** and **label space mismatch**.

Domain Adaptation (DA) has become a prominent technique for addressing the disparity in visual data distribution between source and target datasets [68, 137, 163]. This approach offers an effective solution to mitigate the challenges posed by such differences. Within the field of DA, two primary categories of methods have emerged: *feature-based* and *model-based approaches*. Feature-based transfer learning strategies focus on extracting a shared latent embedding space that encompasses both the source and target data. These strategies employ various techniques, including metric-based training or adversarial optimization, to minimize the differences between the two domains in the latent space. Differently, model-based methods capitalize on the knowledge acquired from a trained model to facilitate adaptation to the target domain. Teacher-student architecture has shown to be effective for domain adaptation in various tasks beyond visual image recognition [18, 91, 70, 17].

Moreover, label space mismatch refers to the challenge posed by the different label spaces present in the source training data and the target test data. Zero-shot learning (ZSL) is a common solution for addressing this type of problem. Most ZSL methods can be broadly categorized as *embedding-based* and *generative-based* approaches [64, 101, 166]. Embedding-based methods learn an embedding space that associates low-level visual features of seen classes with their corresponding semantic vectors. This learned projection function is then used to recognize novel classes by measuring the similarity between prototype representations and predicted representations of data samples in the embedding space. Generative-based methods, on the other hand, learn a model to generate images or visual features for unseen classes based on samples from seen classes and semantic representations of both.

It is important to note that both of these two problems may occur in real-world tasks, and transfer learning solutions must address all of them simultaneously. For instance, tasks such as open-set domain adaptation and some generalized zero-shot learning involving more than one dataset present challenges involving both data distribution disparity and label space mismatch [114, 128, 133, 82].

However, although transfer learning has gained numerous interests from both academia and industry in the past decades and achieved impressive progress in various applications of transferring knowledge across different datasets, two critical challenges have not received sufficient exploration yet: **label space mismatch** and **interpretation of the knowledge transfer**.

In most conventional domain adaptation problems, the existence of novel categories in the target domain is not considered. Open-set Domain Adaptation (OSDA) is one of the research problems that the target domain contains novel classes never observed in the source domain, while most OSDA solutions typically group them as one “unknown” category without further exploration [114, 81, 128]. It is worth noting that while Zero-shot Learning (ZSL) and Generalized Zero-shot Learning (GZSL) seek to address the research of new categories not included in the training data, they struggle with handling the data distribution disparity between the source and target domains. Additionally, most of the GZSL methods rely on prior knowledge of the semantics of the novel categories in the target data, which is not always practical in real-life situations [133, 82, 119, 32].

Furthermore, despite the countless transfer learning solutions and techniques proposed to address various problems, transparency remains an open problem in this area. Interpretability is crucial in machine learning and deep learning to enable the practical application of these techniques in real-life problems, particularly in risk-critical areas [52, 28, 83]. However, the motivation for the interpretation of transfer learning has not received enough attention in this regard. An interpretable transfer learning solution that provides trackable explanations and insights into the knowledge transfer process can contribute to building responsible AI systems and efficiently solving real-life problems while benefiting from the advantages of transfer learning. It can also assist researchers in understanding the reasoning behind the decision-making process of the model, enabling them to diagnose errors and gain insights into how to further improve its performance.

This doctoral dissertation aims to enhance the transparency and interpretability of visual domain adaptation tasks. It achieves this by analyzing visual feature representations, integrating multimodal semantic knowledge, and addressing cross-domain data distribution and label space differences. First of all, we analyze the feature representation distribution in a domain-invariant latent space, gaining valuable insights into the domain adaptation process. For instance, by assessing the similarities in features between the source and target domains, we can identify and exclude mismatched categories across the two domains. This allows us to focus solely on aligning the shared classes during domain adaptation. Next, by leveraging textual descriptions of high-level semantic attributes, we discover and explain novel categories while aligning shared categories across domains. Furthermore, to enhance interpretability, we propose an interpretable driving decision-making model that uncovers convolutional networks within visual feature extraction networks. This model learns concept-based visual prototypes aligned with human annotations and can identify action-inducing regions in ego-view images, providing interpretable driving decision predictions. Finally, we combine interpretable strategies and multimodal semantic knowledge to address the SR-OSDA problem, aligning shared categories and discovering novel target classes. This work extends beyond open-set domain adaptation and represents an initial step towards developing comprehensive and transparent transfer learning techniques, unveiling the black-box nature of knowledge transfer across datasets and tasks.

0.2 Definitions and Notations

This section formally defines transfer learning and domain adaptation while presenting categorizations for domain adaptation tasks under different conditions. It also introduces and explains important notations used throughout the dissertation for clarity and convenience.

0.2.1 Transfer Learning

In transfer learning problems, we define a *domain* $\mathcal{D}$ as comprising two key components: a feature space $\mathcal{X}$ and a marginal probability distribution $P(\mathbf{x})$, with $\mathbf{x} \in \mathcal{X}$. When dealing with a specific domain $\mathcal{D} = \{\mathcal{X}, P(\mathbf{x})\}$, a *task* consists of two fundamental elements: a label space $\mathcal{Y}$ and an objective predictive function denoted as $f(\cdot)$, forming $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$. Although the predictive function $f(\cdot)$ is not explicitly observable, it can be learned from the training data. The training data itself is composed of pairs $\{\mathbf{x}_i, y_i\}$, where $\mathbf{x}_i \in \mathbf{X}$ and $y_i \in \mathbf{Y}$. The predictive function $f(\cdot)$ is used to determine the corresponding label y of a new input instance $\mathbf{x}$, and this relationship can also be represented as $P(y|\mathbf{x})$ from a probabilistic perspective.

Let $\mathcal{D}_s$ be the source domain and $\mathcal{D}_t$ be the target domain. The source domain data, $D_s = \{(\mathbf{x}_s^i, y_s^i)\}_{i=1}^{n_s}$, consists of n_s input data $\mathbf{x}_s^i \in \mathcal{X}_s$ and corresponding labels $y_s^i \in \mathcal{Y}_s$, drawn from the source distribution $P_s(\mathbf{x}, y)$. Similarly, the target domain data, $D_t = \{(\mathbf{x}_t^i, y_t^i)\}_{i=1}^{n_t}$, comprises n_t samples with $\mathbf{x}_t^i \in \mathbb{R}^{d_t}$. Here, n_s and n_t represent the number of source and target samples, respectively. Now we give the definition of transfer learning as:

Definition 1 (*Transfer Learning*): In the context of a source domain $\mathcal{D}_s$ and learning task $\mathcal{T}_s$, and a target domain $\mathcal{D}_t$ and learning task $\mathcal{T}_t$, *transfer learning* improves the learning of the target predictive function $f_t(\cdot)$ in $\mathcal{D}_t$ by leveraging knowledge from $\mathcal{D}_s$ and $\mathcal{T}_s$, even when $\mathcal{D}_s \neq \mathcal{D}_t$ or $\mathcal{T}_s \neq \mathcal{T}_t$.

Transfer learning settings can be categorized into three sub-settings: *inductive transfer learning*, *transductive transfer learning*, and *unsupervised transfer learning*, depending on the relationships between the source and target domains and tasks [97]. In inductive transfer learning, the target task differs from the source task, regardless of whether the source and target domains are the same or not. In transductive transfer learning, the source and target tasks are the same, but the domains differ. Lastly, in unsupervised transfer learning, the target task is different from the source task, and there are no labeled data available in both source and target domains during training.

0.2.2 Domain Adaptation

In the context of transfer learning categorization, domain adaptation pertains to transductive transfer learning, where the assumption is that the source and target tasks are identical ($\mathcal{T}_s = \mathcal{T}_t$), while the source and target domains differ ($\mathcal{D}_s \neq \mathcal{D}_t$). The underlying reason for domain divergence may stem from variations in the feature space ($\mathcal{X}_s \neq \mathcal{X}_t$) or distribution shift, where the feature space is the same ($\mathcal{X}_s = \mathcal{X}_t$), but the data distributions differ ($P(\mathbf{x}_s) \neq P(\mathbf{x}_t)$) [97, 139].

Most domain adaptation problems can be categorized based on the availability of labeled target domain data during training:

- **Unsupervised Domain Adaptation (UDA)**: Fully unlabeled target domain data.
- **Semi-supervised Domain Adaptation (SSDA)**: Partially labeled target domain data and labels.
- **Supervised Domain Adaptation**: Fully labeled target domain data and labels.

Moreover, based on the relationship between the label spaces of the source and target domains, $\mathcal{Y}_s$ and $\mathcal{Y}_t$, domain adaptation problems can be categorized as follows:

- **Closed-set Domain Adaptation (CDA)**: Same label space for both source and target domains, $\mathcal{Y}_s = \mathcal{Y}_t$.
- **Open-set Domain Adaptation (OSDA)**: The source domain label space is a proper subset of the target domain label space, $\mathcal{Y}_s \subset \mathcal{Y}_t$.
- **Partial Domain Adaptation (PDA)**: The target domain label space is a proper subset of the source domain label space, $\mathcal{Y}_t \subset \mathcal{Y}_s$.
- **Universal Domain Adaptation (UniDA)**: No prior knowledge of the label spaces, $\mathcal{Y}_s \ ? \ \mathcal{Y}_t$.

This dissertation undertakes a comprehensive investigation into domain adaptation, commencing with closed-set unsupervised domain adaptation to analyze feature representations in the domain-invariant hidden space. The obtained insights serve as a cornerstone for addressing partial domain adaptation through the measurement of feature similarities across domains and the generation of

Table 1: Notations and Descriptions

Notation	Description
$\mathcal{D}_s, \mathcal{D}_t$	source / target domain
$\mathcal{T}_s, \mathcal{T}_t$	source / target tasks
$\mathcal{X}_s, \mathcal{X}_t$	source / target domain data space
$\mathcal{Y}_s, \mathcal{Y}_t$	source / target domain label space
C_s, C_t	number of categories in the source / target domain ($C_s = \lvert\mathcal{Y}_s\rvert, C_t = \lvert\mathcal{Y}_t\rvert$)
$\mathbf{X}_s, \mathbf{X}_t$	source / target data input
n_s, n_t	number of source / target samples
$\mathbf{Y}_s, \mathbf{Y}_t$	source / target labels
$\mathbf{x}_s^i, \mathbf{x}_t^j$	source / target domain instance
$P_s(\mathbf{x}_s), P_t(\mathbf{x_t})$	source / target domain data distribution
$\mathbf{z}_s^i, \mathbf{z}_t^j$	source / target domain embedding features
y_s^i, y_t^j	source / target domain ground-truth label
$f_s(\cdot),\ f_t(\cdot)$	source / target predictive function
$\hat{\mathbf{y}}_s^i, \hat{\mathbf{y}}_t^j$	prediction of the source / target sample

synthetic features to address the challenge of scarce training data and labels. Furthermore, we explore open-set domain adaptation, striving to recover semantical descriptions of novel target domain categories by leveraging multimodal semantic knowledge and an interpretable AI architecture. This research aims to contribute valuable advancements to the field of domain adaptation and enhance model generalization across diverse domains.

0.2.3 Notation and Descriptions

For convenience, a list of notations and their definitions are shown in Table 1.

0.3 Related Work

0.3.1 Unsupervised Domain adaptation (UDA)

Domain adaptation (DA) has been extensively studied recently, which casts light when there are no or limited labels in the target domain and shows very promising performance in different vision applications [163, 67, 78, 149, 62]. *Closed-set* domain adaptation is one of the most explored problems. Specifically, closed-set domain adaptation assumes that the categories in the target domain are already seen in the source domain. Unsupervised domain adaptation (UDA) deals with problems when the source and target domains have identical label spaces [126, 20, 23, 43]. Most UDA solutions seek to mitigate the cross-domain distribution disparity via minimizing the cross-domain marginal and conditional distribution divergence or learning domain-invariant representations in an adversarial manner [126, 46, 12, 127, 162]. With the renaissance of deep neural networks, deep DA methods successfully embed DA into deep learning pipelines by either minimizing an appropriate distribution distance metric [75] or leveraging adversarial technologies to generate domain-invariant representations [113, 15]. The cross-domain distribution discrepancy enlarged by traditional deep learning models can be explicitly alleviated by incorporating various domain alignment strategies at the top layers. To name a few, Domain Adaptation Network (DAN) applies multiple kernel MMD distances on the last three task-specific layers to minimize the distribution difference [74]. Long *et al.* [79] proposed a Joint Adaptation Network (JAN) and joint MMD criterion to solve the problem. Another strategy is to leverage generative adversarial networks (GAN) [34] to couple the cross-domain discrepancy in an adversarial manner [30, 113, 161, 163]. Such techniques aim to train a domain discriminator to differentiate source and target samples, while the feature generator will deceive the domain discriminator, such that the domain-invariant features will be produced. Ganin *et al.* [31] proposed DANN to generate task-specific discriminative while domain-wise indiscriminative features. Tzeng *et al.* [130] presented ADDA for adversarial adaptation.

Both discrepancy and adversarial loss-based methods attempt to match the whole source and target domain distribution completely, neither of them considers the target domain data structure and task-specific decision boundaries. To address this, Saito *et al.* [113] adopted the task-specific category decision boundaries and proposes a model with two classifiers as a discriminator to detect the relationship between the source and target domain data (MCD). By maximizing the prediction results of the two classifiers, the framework is able to screen out target samples that are near the category decision boundaries and far from the source domain support. Following this, Lee *et al.* [62] extended MCD and proposed a novel Wasserstein metric to capture the natural notion of dissimilarity between the outputs of two task-specific classifiers. Most recently, Li *et al.* [67] claimed that label distribution alignment is still not enough and proposed Joint Adversarial Domain Adaptation (JADA) to explore a unified adversarial learning mechanism to align the cross-domain domain-wise and class-wise distribution simultaneously. Unfortunately, existing works seek to maximize the prediction difference between two same architecture classifiers to explore different task-specific knowledge, limiting the divergence of category decision boundaries captured across domains.

The cross-domain data distribution discrepancy, known as domain shift, is the main challenge of domain adaptation. Plenty of works exploits the potential of deep neural networks to capture explanatory attributes and domain-invariant features in recent years, which is conducive to mitigating domain shift while transferring underlying knowledge across domains in domain adaptation tasks [5, 22, 153]. Compared to traditional machine learning-based domain adaptation solutions, introducing deep architecture into domain adaptation promotes the generalization of frameworks dramatically [38, 95]. Some researchers integrate high-order statistical properties of different domains into a unified framework, such as maximum mean discrepancy (MMD), to align the data distribution across domains, which successfully eliminates domain shift and achieves promising classification performance on the target domain [74, 78]. By virtue of generative adversarial techniques, some works involve a domain discriminator in the game to distinguish which domain the sample belongs to while optimizing the generator and discriminator in an adversarial manner [31, 129, 67]. Moreover, the latest works rethink the domain adaptation problem from various perspectives and propose dual-classifiers-based frameworks that seek to align not only domain-wise data distributions but also classifier-class-specific boundaries [113, 62, 163].

0.3.2 Partial / Open-set Domain Adaptation (PDA/OSDA)

Different from UDA, Partial domain adaptation (PDA) is a special case of closed-set domain adaptation assuming that the target domain only covers a subset of the source domain label space, and re-weighing source instances to eliminate the distraction caused by the source classes not shared with the target domain is a typical strategy [46, 160, 9]. Moreover, Open-set Domain Adaptation (OSDA) manages a more realistic situation when the target domain contains samples from classes never seen in the source domain [98, 57, 104]. Many OSDA efforts aim to mitigate the negative impact of unknown classes on target domain alignment by finding and rejecting the target domain unknown categories data and assigning a single unknown label to them, and then only aligning the shared categories samples across domains.

Selective Adversarial Network (SAN) explores multiple adversarial networks to weigh and select out the outlier categories source samples and down their transferring weights [8]. Partial Adversarial Domain Adaptation (PADA) extends SAN and pays more attention to class-level transferability weighting on the source classifier [9]. Similarly, Importance Weighted Adversarial Nets (IWAN) consider the sigmoid output of an auxiliary domain classifier as the indicator to measure the probability of each source sample coming from the target domain [160]. Example Transfer Network (ETN) further explains the discriminative information as the transferability quantification of the source domain samples, through which the irrelevant examples from outlier categories are down-weighted for both the task-specific classifier and domain discriminator [10]. All the pioneering efforts achieve impressive performance improvements over conventional domain adaptation approaches on PDA tasks.

Although most existing PDA solutions seek to mitigate the negative transfer caused by outlier source classes by re-weighting samples' importance to reduce the distraction, they still train and predict the entire source domain label space, which dilutes the contribution of discriminative information within the shared categories across domains. Besides, some of them regard the prediction of the target samples as pseudo labels to align cross-domain conditional distribution, which would involve severe classification errors and mislead the optimizing direction of the model, especially at the initial stage of training when the classifier cannot handle the differently distributed unlabeled target domain samples.

Compared to classic closed set domain adaptation [152, 143, 142, 12, 127, 162, 126, 46, 44, 138], Open-set Domain Adaptation (OSDA) manages a more realistic task when the target domain contains data from classes never present in the source domain [6, 104, 80, 98, 57, 27, 125, 3, 114]. Busto *et al.* [99] attempts to study the realistic scenario when the source and target domain both include exclusive classes from each other. Later on, Saito *et al.* [114] focus on the situation when the source domain only covers a subset of the target domain label space and utilizes an adversarial framework to generate features and recognizes samples deviating from the pre-defined threshold as "unknown". Instead of relying on the manually pre-defined threshold, [27] takes advantage of the semantic categorical alignment and contrastive mapping to encourage the target data from known classes to move close to the corresponding centroid while staying away from unknown classes. STA adopts a coarse-to-fine weighting mechanism to progressively separate the target data into known and unknown classes [72]. Most recently, SE-CC augments the Self-Ensembling technique to with category-agnostic clustering in the target domain [98].

0.3.3 Tranfer Learning with Mismatched Label Spaces

The demand for leveraging annotated data to recognize novel classes unseen in the training set motivates a boom thread of research known as Zero-Shot Learning (ZSL) [42, 134, 14, 145, 60, 59, 1, 156, 25, 21]. Most existing ZSL works explore the projection from the visual representation to class-level semantic attributes or Word2vec as intermediates for searching novel categories. For instance, SOC [96] maps the image features into the semantic space and then searches the nearest class embedding vector. Differently, SSE [167] and JLSE [168] seek to embed both image and semantic features into another common intermediate space. ZSL has been criticized for the restriction that test data must come from classes that have never been seen in the training data. This can result in a bias towards seen classes in inference, and the learned model cannot guarantee discrimination between the seen and unseen categories when both exist in the test data. Generalized zero-shot learning (GZSL) seeks to address a more realistic scenario when the target data to evaluate are drawn from both seen and unseen categories [42, 116, 40, 40]. Specifically, [159] leverages label embeddings to learn latent representations for images. On the other hand, [4] proposes a model to detect and estimate the probability of an input being from an unknown class. In practice, however, most existing generalized zero-shot learning solutions require semantic knowledge of both seen and unseen classes in order to construct discriminative relationships between all classes.

0.3.4 Interpretable Artificial Intelligence (XAI)

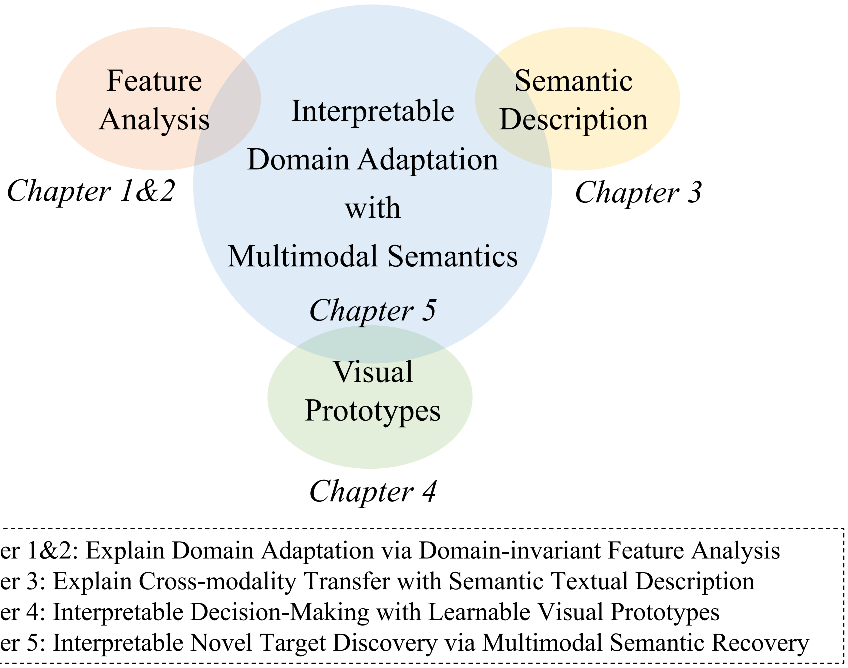
There are two general branches to interpreting deep neural networks from different perspectives: post hoc techniques and interpret-by-design. Specifically, post hoc techniques seek to explain trained neural networks, such as part-based methods [170, 173], saliency visualization [2, 28, 172], activation maximization to visualize neurons [90, 92], and deconvolution/up-convolution to explain layers [24, 158]. However, such post hoc explanations are not used by the original networks during training and prediction, thus the interpretation may not be faithful to what the network computes [110]. Differently, the interpret-by-design strategy aims to build inherently interpretable networks enhancing the transparency and interpretability of the model. Prototypical part networks (ProtPNet) and following extension work use case-based reasoning with prototypes to explain the prediction in the form of “this looks like that” via the similarity scores between an input image and learned prototypes [11, 111, 110, 51, 89].

Building a transparent and interpretable model is crucial for safety-critical problems, such as autonomous driving and medical diagnosis [93], [165]. Many efforts have been made to interpret deep neural networks from different perspectives. Typically, researches include part-based methods [170],[173], attributes-based methods [47],[50], saliency maps [2], [28], [172], activation maximization [90], [92], deconvolution or upconvolution to explain layers [24], [158] and have achieved inspiring progress to create human-interpretable black box models. However, such post-hoc solutions have limited capability in enhancing transparency and interpretability. Alternatively, prototype-based frameworks are proposed to build an inherently interpretable architecture [11], [111], [110], [51], [89].

This dissertation focuses on addressing the unsupervised domain adaptation (UDA) problem and introduces innovative solutions to handle imbalanced distributed source data, the partial domain adaptation (PDA), and a new problem semantic-recovery open-set domain adaptation (SR-OSDA). By analyzing feature representation distribution, the study supports the resolution of UDA with imbalanced source data and PDA. Additionally, high-level semantic attributes are leveraged to improve domain adaptation and discover novel categories in the target domain, inspired by the concept of zero-shot learning. The dissertation further proposes a concept-based interpretable model that simultaneously facilitates visual image to semantic attribute recovery and domain alignment, providing a transparent pipeline for effective SR-OSDA.

0.4 Dissertation Organization

In Chapters 1 & 2, we analyze the distribution of learned feature representations in both the source and target domains for conventional domain adaptation tasks with only image data available, in order to provide insights and explanations regarding the transfer of knowledge across domains. Specifically, We propose the Adversarial Dual Distinct Classifiers Network (AD^{2}CN) to address the Unsupervised Domain Adaptation (UDA) problem. By visualizing the learned features of both domains in the domain-invariant feature space, we can observe the alignment of task-specific category bound-



Chapter 1&2: Explain Domain Adaptation via Domain-invariant Feature Analysis
Chapter 3: Explain Cross-modality Transfer with Semantic Textual Description
Chapter 4: Interpretable Decision-Making with Learnable Visual Prototypes
Chapter 5: Interpretable Novel Target Discovery via Multimodal Semantic Recovery

Figure 1: Organization of the dissertation and discussed problems.

aries across domains. Additionally, for the Partial Domain Adaptation (PDA) problem, where the target domain contains a subset of the source domain label space, we introduce the Adaptively-Accumulated Knowledge Transfer framework (A^{2}KT) to address the mismatched label spaces and data distribution disparities between domains by measuring the similarities of features from both domains to explicitly align samples from categories shared across domains while filtering out samples from outlier categories only present in the source domain. Furthermore, leveraging the valuable insights derived from the analysis of feature representations across different domains, we propose several feature generation strategies to effectively tackle the fairness issue within the context of imbalanced domain adaptation in Chapter 2. This problem becomes even more challenging when faced with limited availability of training data from either the source or target domain.

In Chapter 3, we seek to leverage multimodal knowledge beyond only visual images to advance the interpretation of the domain adaptation process. Specifically, we first present the novel Semantic-Recovery Open-Set Domain Adaptation (SR-OSDA) problem, which brings the challenges of recovering semantic descriptions for unseen categories exclusively present in the target domain, while accurately identifying seen categories simultaneously. By involving the textual semantic descriptions of the categories in addition to the visual images in the source domain, we get to efficiently discover novel target domain classes never observed in the source domain, and provide human-friendly explanations with explicit semantic attributes prediction.

In Chapter 4, in order to enhance the interpretability of predicting semantic descriptions from visual images, our objective is to elucidate the inner workings of convolutional networks for visual feature extraction and establish a correspondence between the learned visual semantics and human annotations. To achieve this, we introduce the Interpretable Action Decision-Making (InAction) model. This model employs learnable concept-based visual prototypes to identify the regions and objects in ego-view images that influence driving actions. Additionally, it aligns the learned semantic prototypes with human annotations, thereby enabling interpretable decision-making in the context of driving actions.

Finally, in Chapter 5, we combine all the interpretation strategies explored and present an Interpretable Novel Target Discovery model with multimodal semantic knowledge to address the SR-OSDA problem. This model aligns the source and target domain data distributions while discovering novel target categories never observed in the source domain. The interpretation of domain alignment and novel category discovery is achieved using human-friendly multimodal semantic concept-based visual prototypes and analysis of extracted feature representations. The insights gained from this research have the potential to transform the integration of AI systems across various domains by ensuring transparency, interpretability, and trustworthiness in decision-making processes. Ultimately, this dissertation contributes to the development of interpretable transfer learning techniques, improving the understanding and practical application of deep learning models, and fostering more transparent and collaborative human-AI interactions.

The dissertation concludes with discussions on remaining challenges and future research directions, inviting further contributions to this field of study.

[44] Jing, Taotao, and Zhengming Ding. "Adversarial dual distinct classifiers for unsupervised domain adaptation." In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 605-614. 2021.

[46] Jing, Taotao, Haifeng Xia, and Zhengming Ding. "Adaptively-accumulated knowledge transfer for partial domain adaptation." In Proceedings of the ACM International Conference on Multimedia, pp. 1606-1614. 2020.

1

Cross-Domain Adaptation via Domain-invariant Feature Analysis

This chapter investigates the problem of visual domain adaptation and aims to provide insights into explaining this phenomenon through domain-invariant feature analysis. The chapter introduces two novel approaches: the Adversarial Dual Distinct Classifiers Network (AD^{2}CN) and the Adaptively-Accumulated Knowledge Transfer scheme (A^{2}KT). In AD^{2}CN, dual different-architecture classifiers are employed to align domain distributions and category decision boundaries. On the other hand, A^{2}KT addresses partial domain adaptation challenges by promoting positive transfer and mitigating negative transfer, which is achieved through an adaptively-accumulated knowledge transfer strategy. It is noteworthy that task-specific decision boundary is the boundary that separates different classes in a classification problem. It helps determine the class of a new data point based on its features. In contrast, the conditional distribution is a statistical concept describing the probability distribution of one variable given another. While not the same, the decision boundary is crucial for approximating the conditional distribution in classification tasks. By presenting these approaches, the chapter contributes to the understanding and advancement of visual domain adaptation with identical and mismatched label spaces across domains.

1.1 Unsupervised Domain Adaptation (UDA)

Unsupervised Domain adaptation (UDA) attempts to recognize the unlabeled target samples by building a learning model from a differently-distributed labeled source domain. Conventional UDA concentrates on extracting domain-invariant features through deep adversarial networks. However, most of them seek to match the different domain feature distributions, without considering the task-specific decision boundaries across various classes. In this work, we propose a novel Adversarial Dual Distinct Classifiers Network (AD^{2}CN) to align the source and target domain data distribution simultaneously with matching task-specific category boundaries. To be specific, a domain-invariant feature generator is exploited to embed the source and target data into a latent common space with the guidance of discriminative cross-domain alignment. Moreover, we naturally design two different structure classifiers to identify the unlabeled target samples over the supervision of the labeled source domain data. Such dual distinct classifiers with various architectures can capture diverse knowledge of the target data structure from different perspectives. Extensive experimental results on several cross-domain visual benchmarks prove the model's effectiveness by comparing it with other state-of-the-art UDA.

1.1.1 Summary of Contribution

In this work, we propose a novel Adversarial Dual Distinct Classifiers Network (AD^{2}CN) with two different-architecture classifiers, e.g., Neural Networks Classifier and Prototypical Classifier, to facilitate the alignment of both domain distributions and category decision boundaries (Fig. 1.1). To our best knowledge, it is a pioneering work to explore dual different structure classifiers in domain

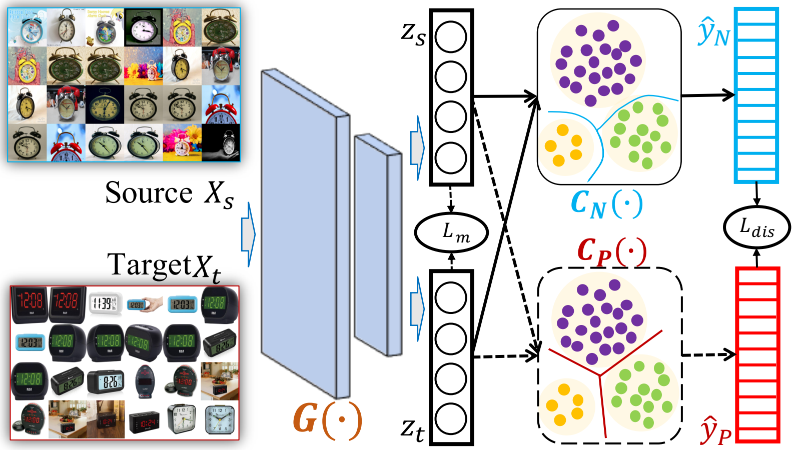


Figure 1.1: Framework overview of our proposed model, where $G(\cdot)$ is the domain-invariant embedding features generator, $C_N(\cdot)$ denotes the fully-connected neural networks classifier (solid line) and $C_P(\cdot)$ means the prototypical classifier (dash line). $\mathcal{L}_m$ and $\mathcal{L}_{dis}$ are explored to align the feature and prediction distribution differences across two domains and dual classifiers, respectively.

adaptation. The general idea is to explore adversarial training over two different architecture classifiers on the output of one domain-invariant feature generator. To sum up, we highlight the three-fold contributions of this work as follows:

- We exploit dual different architecture task-specific classifiers over source supervision to exploit the task-specific decision boundaries on the target domain. With different properties of dual classifiers in prediction, we have a better chance of capturing ground-truth classifier decision boundaries for the target domain.

- We propose a novel discriminative cross-domain alignment loss and *Importance Guided Optimization* strategy to mitigate the cross-domain mismatching. This will facilitate the process of aligning the domain-invariant embedding features distribution across domains, and eliminate the distraction of misestimated target samples at the beginning of optimizing.

- We adopt a discrepancy loss to maximally improve the prediction performance of dual classifiers in coupling the cross-domain label distributions, which is trained in an adversarial way with domain-invariant feature generator and dual classifiers. Thus, they can benefit from each other to boost the target learning task.

1.1.2 The Proposed Method

1.1.2.1 Preliminaries and Motivation

In unsupervised domain adaptation (UDA), the source domain $\mathcal{D}_s$ contains n_s labeled data, and the target domain $\mathcal{D}_t$ contains n_t unlabeled data. The source and target domain data distributions are different $P_s(\mathbf{x}_s) \neq P_t(\mathbf{x}_t)$, and the number of total categories are the same, *i.e.*, $C_s = C_t$. We denote $C = C_s = C_t$ for simplicity in this section.

Recent domain adaptation works apply adversarial networks to generate domain invariant features of the source and target domain samples, which will make the classifiers trained only on the source domain data available on the target domain[30, 33, 161]. Most of them aim to match the distribution of source and target domain completely, without considering the task-specific decision boundaries between different categories. Most recently, the idea of dual adversarial classifiers [113, 62, 163, 67] has been explored to replace the original adversarial domain adaptation with a binary domain discriminator. However, they obtain two same-type classifiers from scratch over labeled source data. This would limit the discriminative ability in target prediction since the same-type classifiers would tend to have similar properties. Traditional neural networks classifier aims to fit the training data by achieving optimal objective value, thus the learned classifier boundaries would capture the global structure of the data to maximally separate different classes. Such a decision boundary over source supervision cannot be well adapted to target samples in different distributions. Therefore, two same-architecture neural network classifiers over source supervision are challenging to diversify the decision boundaries.

This motivates us to explore two different architecture classifiers, and thus we propose a novel adversarial dual classifiers network with two different structure classifiers to capture various data distribution pattern and more diverse task-specific category boundaries from different perspectives, and also promote the out of source support target samples detection process. Interestingly, the prototypical classifier explores the local structure of the data since prototypes are used to assign labels based on the similarity between samples and each prototype. The competition between two different structure classifiers is more likely to diversify the decision boundaries to benefit from adversarial training with domain-invariant generators.

1.1.2.2 Adversarial Dual Distinct Classifiers Network

We first present the overall framework of our proposed adversarial dual classifier network in Fig. 1.1. Given the labeled source and unlabeled target domain data, the domain invariant embedding features are generated and aligned by the discriminative cross-domain alignment, then the dual classifiers, which consist of two classifiers with different architectures, will promote the task-specific decision boundaries further. $G(\cdot)$ is a feature extractor neural network used to take source and target domain data as input and project into a shared embedding feature space, in which the target samples are close to the support of the source domain data. The following two different structure classifiers, fully-connected neural network classifier $C_N(\cdot)$ and prototypical classifier $C_P(\cdot)$, will capture diverse and various task-specific categories knowledge on target domain from different perspectives.

1.1.2.2.1 Dual Classifiers Over Source Supervision Since $\mathbf{X}_s$ and $\mathbf{X}_t$ have different distributions, a domain-invariant feature generator $G(\cdot)$ is deployed to capture more enriched information across source and target through hierarchical structures, followed by our dual classifiers, $C_N(\cdot)$ (fully-connected neural network classifier) and $C_P(\cdot)$ (prototypical classifier). With the extracted feature $\mathbf{z}^i_{s/t} = G(\mathbf{x}^i_{s/t})$ as input, we can calculate the corresponding probability prediction with two classifiers $C_N(\cdot)$ and $C_P(\cdot)$ as $\hat{\mathbf{y}}^i_{N/P,s/t} = C_{N/P}(\mathbf{z}^i_{s/t})$.

Specifically, $C_N(\cdot)$ represents a multi-layer non-linear classifier, and $C_P(\cdot)$ calculates the similarity, such as cosine similarity, between the feature $\mathbf{z}^i_t$ of the target sample and each category prototype μc (i.e., class center). Subsequently, the output of the classifier $C_{N/P}(\cdot)$ is normalized using the $Softmax(\cdot)$ function across all categories, yielding the probability prediction. For each class, the prototype is calculated by $\mu_c = \frac{1}{n^c_t}\sum_{i=1}^{n^c_t}\mathbf{z}^{i(c)}_t$, where n^c_t and $\mathbf{z}^{i(c)}_t$ denote the number of target samples and extracted domain invariant feature belonging to class c, respectively. During initialization, the source domain class centers in the feature space are taken as prototypes. Then we apply the $C_P(\cdot)$ prediction $\hat{\mathbf{y}}^i_{P,t}$ as the predicted pseudo label to target sample $\mathbf{x}^i_t$ to get the category prototypes μ_c. Subsequently, these prototypes are iteratively refined using the predictions made by $C_P(\cdot)$ on the target domain data. This refinement process helps the model adapt and improve its representation of the target domain.

In order to obtain task-specific discriminative features from generator $G(\cdot)$, while keeping classification performance on the source domain, we add the supervision from the source to learn the parameters of $C_N(\cdot)$ and $G(\cdot)$. Since $C_P(\cdot)$ does not contain any trainable parameters, the supervision over $C_P(\cdot)$ prediction on the source domain will optimize the parameters in the feature extractor $G(\cdot)$. To this end, we aim to minimize the cross-entropy loss over $\mathbf{Y}_s$ and predicted labels from $C_N(\cdot)$ and $C_P(\cdot)$, defined as follows:

$$\mathcal{L}_s = \frac{1}{n_s}\sum_{i=1}^{n_s}\mathcal{L}(\hat{\mathbf{y}}^i_{N,s}, y^i_s) + \frac{1}{n_s}\sum_{i=1}^{n_s}\mathcal{L}(\hat{\mathbf{y}}^i_{P,s}, y^i_s), \tag{1.1}$$

where $\mathcal{L}$ is the cross-entropy loss. $\hat{\mathbf{y}}^i_{N,s}$ and $\hat{\mathbf{y}}^i_{P,s}$ are the probability outputs of classifier $C_N(\cdot)$ and $C_P(\cdot)$, while y^i_s is the ground-truth label of source sample $\mathbf{x}^i_s$, respectively.

1.1.2.2.2 Adversarial Dual Classifiers The dual classifiers are capable of recognizing target domain samples close to the support of the source domain. For those target domain samples which are far from the source domain support, the two classifiers would tend to obtain different probability outputs. To detect target samples outside of the support from source supervision, we propose to measure the disagreement of the classifiers prediction results with distribution discrepancy measurement [62, 113].

Existing works exploit varying the dual classifiers by maximizing the divergence between the predictions. However, the same classifier structure with slightly different random initializations [113, 62] will weaken the ability to capture diverse task-specific knowledge and decision boundaries from different perspectives. In our model, we build two different architecture classifiers, which are more likely to capture the inconsistent information from various perspectives. Thus, adversarial training would further enhance the target prediction performance, and the classifier discrepancy is defined as:

$$\mathcal{L}_{dis} = \mathcal{F}(\hat{\mathbf{y}}^i_{N,t}, \hat{\mathbf{y}}^i_{P,t}), \tag{1.2}$$

where $\hat{\mathbf{y}}^i_{N/P,t}$ represent the probability prediction obtained from the two classifiers for the sample $\mathbf{x}^i_t$ respectively. $\mathcal{F}(\cdot,\cdot)$ denotes the discrepancy measurement function, e.g., SWD [62], which is able to capture distribution geometric information to calculate the discrepancy between the probability prediction distributions, and solve gradient vanishing problems occurred in adversarial learning methods. The loss function $\mathcal{L}_{dis}$ is used to train the framework in an adversarial training manner. Specifically, the feature extractor $G(\cdot)$ is optimized to minimize the prediction difference between $C_P(\cdot)$ and $C_N(\cdot)$, while the classifier $C_N(\cdot)$ is trained to maximize $\mathcal{L}_{dis}$.

1.1.2.2.3 Discriminative Cross-Domain Alignment So far, our model only aligns cross-domain distributions in terms of label space, we further exploit feature distribution alignment to boost the domain-invariant feature learning. Maximum Mean Discrepancy (MMD) has been sufficiently explored as a promising strategy to reduce the domain-wise distance between the mean of source and target domain features, or class-wise distances between each class source and target features with the pseudo labels for target samples [77]. The domain-wise MMD to measure marginal distribution across domains is defined as $\mathcal{H}(\mathbb{E}_{\mathbf{x}_s^i \sim \mathcal{D}_s}[\mathbf{z}_s^i] - \mathbb{E}_{\mathbf{x}_t^j \sim \mathcal{D}_t}[\mathbf{z}_t^j])$ [77], where $\mathcal{H}(\cdot)$ is the function used to evaluate the distribution difference, which is L-2 norm in this work. Furthermore, existing works [20] also seek to explore the class-wise MMD to align conditional distribution disparity across domains:

$$\mathcal{L}_c = \frac{1}{C}\sum_{c=1}^{C}\mathcal{H}\Big(\mathbb{E}_{\mathbf{x}_s^i \sim \mathcal{D}_s^c}[\mathbf{z}_s^i] - \mathbb{E}_{\mathbf{x}_t^j \sim \mathcal{D}_t^c}[\mathbf{z}_t^j]\Big), \tag{1.3}$$

where C denotes the total number of categories, $\mathbf{z}_{s/t}^{i/j}$ denote the generated embedding representations of source sample $\mathbf{x}_s^i$ and target sample $\mathbf{x}_t^j$ belonging to class c.

However, conventional DA algorithms only seek to minimize the distribution difference between source and target domains when samples are from the same class. We further propose to explicitly take the information of different categories into account and measure the *diff-class* divergence across domains defined as:

$$\mathcal{L}_d = \frac{1}{C}\frac{1}{C-1}\sum_{c=1}^{C}\sum_{\substack{c'=1,\\ c'\neq c}}^{C}\mathcal{H}\Big(\mathbb{E}_{\mathbf{x}_s^i \sim \mathcal{D}_s^c}[\mathbf{z}_s^i] - \mathbb{E}_{\mathbf{x}_t^j \sim \mathcal{D}_t^{c'}}[\mathbf{z}_t^j]\Big), \tag{1.4}$$

where the *diff-class* divergence $\mathcal{L}_d$ calculates the average distances of all different class center pairs across domains. To sum up, our discriminative cross-domain alignment is defined as $\mathcal{L}_m = \mathcal{L}_c - \mathcal{L}_d$.

Due to the lack of target domain labels, we explicitly assign $\hat{\mathbf{y}}_{P,t}^i$, the prediction of $C_P(\cdot)$, as pseudo labels to the target samples $\mathbf{x}_t^i$. Despite domain shift affecting prediction reliability on the target domain data, leveraging pseudo labels has proven effective in enhancing model training. It is crucial that the target domain data shares some similarity in marginal and conditional distribution with the source domain. To achieve more effective knowledge transfer, we propose an *Importance Guided Optimization* strategy. This strategy focuses on high-confidence predictions for target samples during cross-domain alignment, disregarding lower-confident samples which could lead to misleading optimizations. That is, only samples with $\{(\mathbf{x}_t^i, \hat{\mathbf{y}}_{P,t}^{i(c)}) \mid \hat{y}_{P,t}^{i(c)} > \sigma, \mathbf{x}_t^i \in \mathcal{D}_t\}$ are accepted to construct the cross-domain alignment $\mathcal{L}_m$, where $\hat{\mathbf{y}}_{P,t}^{i(c)}$ is the $C_P(\cdot)$ probability prediction of $\mathbf{x}_t^i$ belonging to class c, and $\sigma \in [0, 1]$ is a constant threshold. It is noteworthy that we do not impose always covering the whole label space during training since only considering those classes with high-confident samples is prone to result in effective cross-domain alignment by avoiding too many misclassified target samples, especially in the early training stage.

1.1.2.3 Overall Objective and Optimization

To eliminate the side effect of uncertainty on unlabeled target prediction, we also explore the entropy minimization regularization [163, 75, 78]:

$$\mathcal{L}_{em} = -\frac{1}{n_t}\sum_{i=1}^{n_t}\sum_{c=1}^{C}(\hat{\mathbf{y}}_{N,c}^i \log \hat{\mathbf{y}}_{N,c}^i + \hat{\mathbf{y}}_{P,c}^i \log \hat{\mathbf{y}}_{P,c}^i), \tag{1.5}$$

where $\hat{\mathbf{y}}_{N,c}^i$ and $\hat{\mathbf{y}}_{P,c}^i$ denote the prediction of $\mathbf{x}_t^i$ belonging to class c obtained by $C_N(\cdot)$ and $C_P(\cdot)$, respectively.

To sum up, we integrate adversarial dual classifiers training and cross-domain discriminative alignment together and propose our overall objective function as:

$$\begin{gathered}\min_{G} \mathcal{L}_s + \mathcal{L}_{em} + \lambda_1 \mathcal{L}_{dis} + \lambda_2 \mathcal{L}_m, \\ \min_{C_N} \mathcal{L}_s - \lambda_1 \mathcal{L}_{dis},\end{gathered} \tag{1.6}$$

where λ_1 and λ_2 are hyper-parameters to balance the contribution of loss terms $\mathcal{L}_{dis}$, $\mathcal{L}_m$, respectively.

Similar to the existing adversarial networks training strategy, we freeze the generator $G(\cdot)$ to train classifiers, then freeze the parameters of the classifiers to update the generator $G(\cdot)$. It is noteworthy that only $C_N(\cdot)$ contains trainable parameters because $C_P(\cdot)$ only relies on the embedding features produced by the generator $G(\cdot)$. Meanwhile, inspired by [113], in order to keep the performance of the networks on the source domain and detect target samples far from source domain support, we train our framework in three steps:

Step A. We train the feature generator $G(\cdot)$ and classifier $C_N(\cdot)$ only on source domain $\mathcal{D}_s$ which is the same as supervised learning tasks. Due to $C_P(\cdot)$ does not have any trainable parameters, only parameters in $G(\cdot)$ and $C_N(\cdot)$ would be updated. Our model aims to detect target samples which are outside of source support from those which are close to support of source domain, keeping good ability and performance on classifying the source domain samples correctly is crucial and necessary. The optimization objective is defined as $\min\limits_{G,C_N} \mathcal{L}_s$.

Step B. We need to assign unlabeled target domain samples pseudo labels by classifiers we already have. In our experiments, we explore the prediction results of $C_P(\cdot)$ to obtain pseudo labels of the target samples, which are experimentally proven to achieve better performance, and we will discuss it in the ablation analysis section. We fix the feature generator $G(\cdot)$ and update the classifier $C_N(\cdot)$ to maximize the distribution discrepancy between the classification results of $C_N(\cdot)$ and $C_P(\cdot)$ on the target domain, which can detect the target samples excluded by the source domain data support, and we obtain the training objective function as $\min\limits_{C_N} \mathcal{L}_s - \lambda_1 \mathcal{L}_{dis}$.

Step C. We freeze the parameters of the classifier $C_N(\cdot)$ and update generator $G(\cdot)$ to minimize the distribution discrepancy between the predictions of $C_N(\cdot)$ and $C_P(\cdot)$ on the target domain, through which both $C_N(\cdot)$ and $C_P(\cdot)$ classifiers will have more similar and correct prediction on target domain samples. Furthermore, together with the discriminative cross-domain alignment, the generator $G(\cdot)$ tends to couple the source and target domain closer but discriminative in the embedding feature space. The optimization objective is $\min\limits_{G} \mathcal{L}_s + \mathcal{L}_{em} + \lambda_1 \mathcal{L}_{dis} + \lambda_2 \mathcal{L}_m$.

These three steps repeat once in each iteration of our experiments. The generator $G(\cdot)$ and classifier $C_N(\cdot)$ are initialized and pre-trained on source domain data.

Table 1.1: Comparisons of Recognition Rates (%) of Unsupervised Domain Adaptation on Office+Home Dataset (ResNet-50).

Method	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg.
Res-50 [35]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	51.2	59.9	46.1
DAN [74]	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
RevGrad [30]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
JAN [79]	45.9	61.2	68.9	50.4	59.7	60.0	45.8	43.4	70.3	63.9	52.4	76.8	58.3
SE [29]	48.8	61.8	72.8	54.1	63.2	65.1	50.6	49.2	72.3	66.1	55.9	78.7	61.5
DSR [7]	53.4	71.6	77.4	57.1	66.8	69.3	56.7	49.2	75.7	68.0	54.0	79.5	64.9
DWT-MEC [109]	50.3	72.1	77.0	59.6	69.3	70.2	58.3	48.1	77.3	69.3	53.6	82.0	65.6
CDAN+E [75]	50.7	70.6	76.0	57.6	70.0	70.0	57.4	50.9	77.3	70.9	56.7	81.6	65.8
MCS [71]	55.9	73.8	79.0	57.5	69.9	71.3	58.4	50.3	78.2	65.9	53.2	82.2	66.3
AFN [149]	52.0	71.7	76.3	64.2	69.9	71.9	63.7	51.4	77.1	70.9	57.1	81.5	67.3
SymNets [163]	47.7	72.9	78.5	64.2	71.3	74.2	64.2	48.8	79.5	74.5	52.6	81.6	67.6
BDG [152]	51.5	73.4	78.7	**65.3**	71.5	73.7	**65.1**	49.7	**81.1**	**74.6**	55.1	**84.8**	68.7
Ours	**57.4**	**77.3**	**80.0**	63.4	**76.4**	**76.4**	64.2	**52.4**	80.7	69.6	**57.2**	83.9	**69.9**

Table 1.2: Comparisons of Recognition Rates (%) of Unsupervised Domain Adaptation on Office-31 Dataset (ResNet-50).

Method	Res-50 [35]	DAN [74]	RevGrad [30]	JAN [79]	MADA [100]	CDAN+E [75]	AFN [149]	SymNets [163]	BDG [152]	Ours
A→W	68.4±0.2	80.5±0.4	82.0±0.4	86.0±0.4	90.0±0.1	**94.1**±0.1	90.1±0.1	90.8±0.1	93.6±0.4	93.6±0.3
D→W	96.7±0.1	97.1±0.2	96.9±0.2	96.7±0.3	97.4±0.1	98.6±0.1	98.6±0.2	98.8±0.3	**99.0**±0.1	98.9±0.2
W→D	99.3±0.1	99.6±0.1	99.1±0.1	99.7±0.1	99.6±0.1	**100.0**±0.0	99.8±0.0	**100.0**±0.0	**100.0**±0.0	99.8±0.0
A→D	68.9±0.2	78.6±0.2	79.7±0.4	85.1±0.4	87.8±0.2	92.9±0.2	90.7±0.5	93.9±0.5	93.6±0.3	**95.4**±0.3
D→A	62.5±0.3	63.6±0.3	68.2±0.4	69.2±0.3	70.3±0.3	71.0±0.3	73.0±0.2	74.6±0.6	73.2±0.2	**74.9**±0.3
W→A	60.7±0.3	62.8±0.2	67.4±0.5	70.70.5	66.4±0.3	69.3±0.3	70.2±0.3	72.5±0.5	72.0±0.1	**75.0**±0.5
Avg.	76.1	80.4	82.2	84.6	85.2	87.7	87.1	88.4	88.5	**89.6**

1.1.3 Experimental Results

1.1.3.1 Datasets & Experimental Setup

Office-Home [132] consists of 15,500 images from 65 categories in 4 different domains: Artistic images (Ar), Clip Art (Cl), Product (Pr), and Real-World images (Rw). In total, by choosing any two domains as one task, we can build 12 cross-domain tasks to evaluate our proposed model.

Office-31 contains 4,110 images of 3 domains: Amazon (A), Webcam (W), and DSLR (D) and each domain consists of 31 categories. We evaluate our method on 6 cross-domain tasks to testify to the validation of our model.

Comparisons. We compare our proposed method with several state-of-the-art unsupervised domain adaptation models: Deep Adaptation Networks (DAN) [74], Reverse Gradient (RevGrad) [30], Joint Adaptation Networks (JAN) [79], Self-Ensembling (SE) [29], Multi-adversarial Domain Adaptation (MADA) [100], Conditional Adversarial Domain Adaptation Networks (CDAN) [75], Disentangled

Semantic Representation (DSR) [7], Domain-specific Whitening Transform & Min-Entropy Consensus (DWT-MEC) [109], Minimum Centroid Shift (MCS) [71], Adaptive Feature Norm Approach (AFN) [149], Domain Symmetric Networks (SymNets) [163], Bi-Directional Generation (BDG) [152]. All our experiments follow standard unsupervised domain adaptation protocols: all labeled source domain data and labels, as well as unlabeled target domain data are used for training. All comparisons are back-boned with ResNet-50 or using ResNet-50 features [35].

Implementation Details. We implement our model with PyTorch and adopt ResNet-50[35] as the backbone. Specifically, a ResNet-50 network is pre-trained on ImageNet [16] and fine-tuned on the source domain, then applied to both source and target domain data to obtain the feature representation with dimension 2,048 without the last fully connected layer. $G(\cdot)$ is a two-layer fully-connected neural network, with hidden layer output as 1,024 followed by ReLU activation function, and the dropout probability retaining is 0.5. The output embedding features $\mathbf{z}_{s/t}$ dimension is 512. $C_N(\cdot)$ is a two-layer fully-connected neural network with 512 as the input and hidden layer dimension, the output dimension is the same as the number of categories in the whole label space C. Cosine similarity is accepted as the measurement metric $\langle\cdot,\cdot\rangle$ in $C_P(\cdot)$, and $Softmax(\cdot)$ function is applied to the output of $C_N(\cdot)$ and $C_P(\cdot)$ to get the probability prediction of the input sample. All parameters are updated with Adam optimizer [55] and the learning rate is set as 0.001 on Office-Home and Office-31 datasets. $G(\cdot)$ and $C_N(\cdot)$ are pre-trained and initialized on source domain data only with the learning rate as 0.1 for 2,000 iterations. We deploy SWD distance [62] as the discrepancy measurement function $\mathcal{F}(\cdot,\cdot)$, and accept L-2 norm as $\mathcal{H}(\cdot)$ to evaluate the distribution divergence. λ_1 and λ_2 are fixed as 0.1 for all tasks. σ is set to be 0.03. For the prototypical classifier $C_P(\cdot)$, we initialize the class prototypes with the source domain features class centers $\mu_c^s = \frac{1}{n_s^c}\sum_{i=1}^{n_s^c}\mathbf{z}_s^i$, then update the prototypes with target domain category centroids representation $\mu_c^t = \frac{1}{n_t^c}\sum_{j=1}^{n_t^c}\mathbf{z}_t^j$ after obtaining the target domain samples pseudo labels $\hat{\mathbf{y}}_{P,t}$ iteratively till reaching convergence or the max step (which is set as 3), and return the last step $C_P(\cdot)$ prediction. All results reported in Tables 1.1 and 1.2 are the average of three random experimental results obtained by classifier $C_P(\cdot)$, and we will discuss the performances of $C_N(\cdot)$ and $C_P(\cdot)$ in the ablation study section.

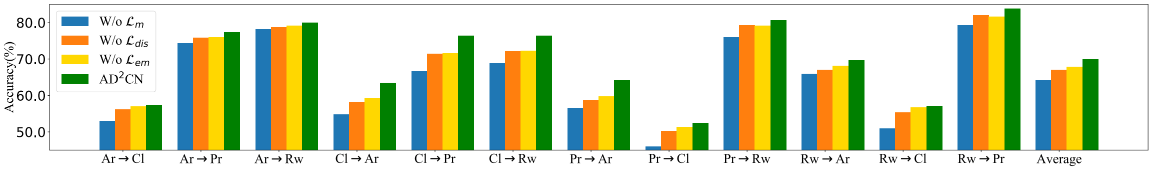


Figure 1.2: Ablation experiments about various loss terms contribution on Office+Home Dataset (ResNet-50).

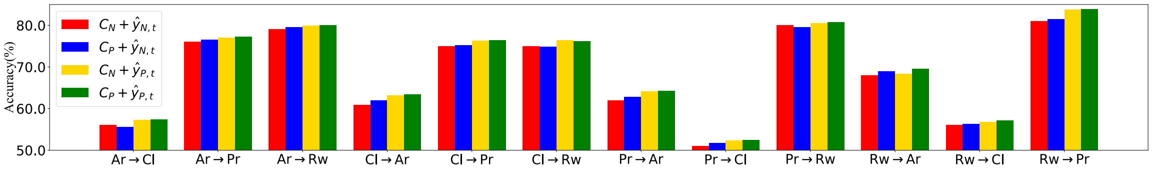


Figure 1.3: Accuracies of C_N and C_P on Office+Home. red and blue results are obtained with $\hat{\mathbf{y}}_{N,t}$ as target pseudo labels for $\mathcal{L}_m$, the others are based on $\hat{\mathbf{y}}_{P,t}$ as pseudo labels.

1.1.3.2 Comparison Results

Table 1.1 and Table 1.2 report the classification results on target domain data of our proposed model and other comparative methods on Office-Home and Office-31 datasets respectively. All comparison results are from their original paper or quoted from [58, 163, 152], as we adopt exactly the same settings. It is noteworthy that our proposed model outperforms state-of-the-art methods on all benchmark datasets in terms of average accuracy, and obtains the best or comparable performances to the state-of-the-art domain adaptation methods in most cases. Although the Office-Home dataset is more challenging than Office-31 due to more categories and samples, as well as significant distribution dissimilarity, our proposed model still improves the performance on most tasks, which demonstrates the efficiency and effectiveness of our proposed framework.

DAN and JAN are both MMD-based methods, which seek to eliminate the cross-domain distribution disparity and match the whole source and target domain to a shared domain-invariant feature space. DAN attempts to align feature representations from multiple layers through a multi-kernel variant of MMD. JAN aims to transfer joint distributions of multi-layers activation of the networks across domains. With the help of additional domain adaptation terms (e.g., MMD), DAN and JAN lead to a significant performance boost over the source-only-trained model (i.e., ResNet-50) on most adaptation tasks.

RevGrad implements adversarial networks and applies gradient reversal layer to train a domain

Table 1.3: C_N v.s. C_P accuracies (%) on Office+Home Ar → Cl

	Balanced			*Imbalanced*				
Y	Clock	Helmet	Knives	Bed	Couch	Folder	Marker	Pen
n_s	74	79	72	39	40	20	20	20
n_t	60	69	53	98	64	99	71	99
C_N	**75.0**	**71.0**	**52.8**	53.1	67.2	25.3	18.3	51.5
C_P	73.3	69.6	49.1	**55.1**	**68.8**	**28.3**	**21.1**	**53.5**

Table 1.4: Comparisons of Dual Classifiers Structure Influence to Recognition Rates (%) of Unsupervised Domain Adaptation on Office-31 Dataset (ResNet-50).

Method	A→W	D→W	W→D	A→D	D→A	W→A	Avg.
MCD [113]	88.6	98.5	100.0	92.2	69.5	69.7	86.5
SWD [62]	90.4	98.7	100.0	94.7	70.3	70.5	87.4
Ours (same)	93.3	98.8	**100**	94.7	72.4	73.6	88.8
Ours	**93.6**	**98.9**	99.8	**95.4**	**74.9**	**75.0**	**89.6**

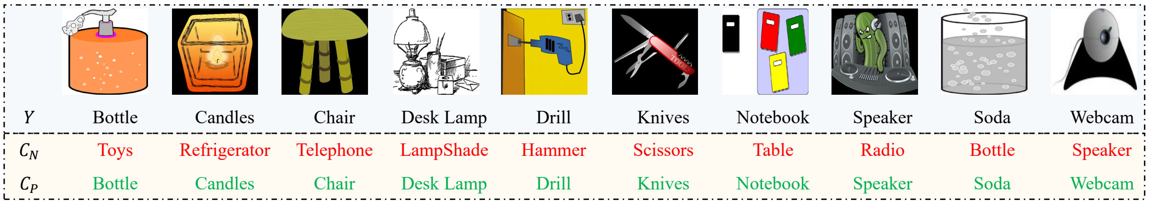
discriminator. CDAN and MADA both exploit the multiplicative combination of feature embeddings and task-specific predictions as high-order representations to promote adversarial optimization. SE studies the self-ensembling to boost the visual domain adaptation performance. DSR assumes that the data generation process is controlled by the semantic latent variables and domain latent variables independently, so employs a variational auto-encoder in order to reconstruct them. MCS designs a unified framework without accessing the source data and iteratively assigns pseudo labels to the target samples by an alternating minimization scheme.

DWT-MEC proposes domain alignment layers with feature whitening to match source and target domain distributions and applies the Min-Entropy Consensus loss to unlabeled target data. AFN proposes a novel Adaptive Feature Norm approach to adapting the source and target domain feature norms to a specific range of values progressively. SymNets exploits novel adversarial classifiers networks and a two-level domain confusion scheme driving the learning of categories invariant intermediate features across domains. BDG bridges the source and target domain through consistent classifiers interpolating two intermediate domains.

1.1.3.3 Ablation Analysis

In this section, we analyze the contribution and influence of several important terms and hyperparameters sensitivity in our proposed model.

First, we discuss the influence of each component in our framework. By removing one of $\mathcal{L}_{dis}$,



Y	Bottle	Candles	Chair	Desk Lamp	Drill	Knives	Notebook	Speaker	Soda	Webcam
C_N	Toys	Refrigerator	Telephone	LampShade	Hammer	Scissors	Table	Radio	Bottle	Speaker
C_P	Bottle	Candles	Chair	Desk Lamp	Drill	Knives	Notebook	Speaker	Soda	Webcam

Figure 1.4: Ten Samples from Office-Home Ar→Cl. Y row denotes the ground-truth labels, C_N row shows the mis-classified labels, while C_P means the correctly prediction.

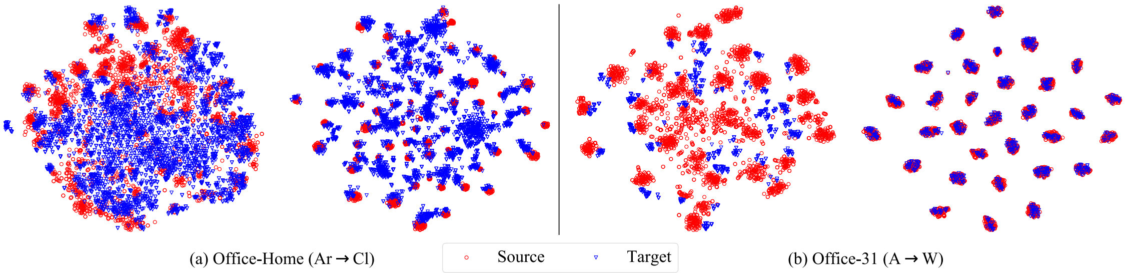


Figure 1.5: t-SNE visualization of source and target samples features before (left column) and after (right column) domain adaptation through our proposed model. (a) shows the task of Ar→Cl from Office-Home and (b) reports the task of A→W from Office-31.

$\mathcal{L}_m$, and $\mathcal{L}_{em}$, while keeping other terms same as original AD^{2}CN, we obtain three variants AD^{2}CN w/o $\mathcal{L}_{dis}$, AD^{2}CN w/o $\mathcal{L}_m$, and AD^{2}CN w/o $\mathcal{L}_{em}$. From Fig. 1.2, we notice that all three components contribute to improving the domain adaptation performance, while our proposed discriminative cross-domain alignment $\mathcal{L}_m$ plays a more crucial role than others, i.e., discrepancy and entropy minimization loss.

Secondly, we compare the performances of $C_N(\cdot)$ and $C_P(\cdot)$ while accepting $\hat{\mathbf{y}}_{N,t}$ or $\hat{\mathbf{y}}_{P,t}$ as target domain pseudo labels for $\mathcal{L}_m$. From the results in Fig. 1.3, we observe that results with $\hat{\mathbf{y}}_{P,t}$ as pseudo labels are better than the results with $\hat{\mathbf{y}}_{N,t}$ in most cases. Compared to $C_N(\cdot)$, which is trained on the source domain, $C_P(\cdot)$ is based on the target prototypes and keeps better performance even in the early training stage. Fig. 1.4 shows several test samples that $C_P(\cdot)$ classifies correctly while $C_N(\cdot)$ cannot handle, which emphasizes the superiority of $C_P(\cdot)$.

Thirdly, we discuss the necessity and effectiveness of two different types of classifiers in our framework. Table 1.3 shows the selective target domain class-wise recognition accuracy on Office-Home Ar → Cl case produced by the two classifiers C_N and C_P in our proposed model, as well as the number of samples in each class from the source and target domains. From the results, we notice that for the categories having sufficient well-labeled source samples as well as balanced target domain

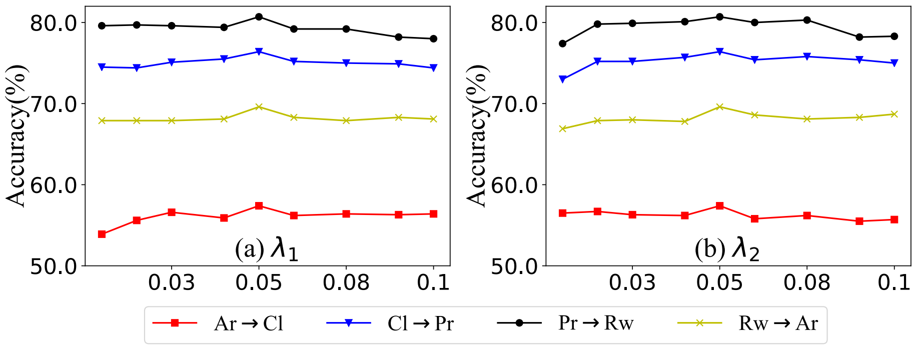


Figure 1.6: Parameters sensitivity analysis on 4 different tasks from Office-Home dataset of (a) λ_1 and (b) λ_2

samples for training, C_N have better performance than C_P, while for other categories with imbalanced distribution across domains and insufficient labeled source samples for training, C_P always performs better than C_N. The observation proves that for the imbalanced datasets, C_N and C_P have different specialties for different categories with various cross-domain distributions. Moreover, we show the comparison results of MCD [113], SWD[62], and our proposed model on the Office-31 dataset in Table 1.4. MCD and SWD are two dual classifier adversarial frameworks for domain adaptation, but using two completely same structure neural networks classifiers. We also replace the C_N and C_P in our proposed model with two same-structure neural network classifiers and report the results as Ours(same). It is noteworthy that our proposed model achieves the best performance in most cases as well as the average accuracy compared to other same classifier structure methods, which proves the effectiveness and necessity of applying two distinct architecture classifiers.

Fourthly, we visualize the t-SNE embeddings (Fig. 1.5) of feature representations generated by $G(\cdot)$ before and after the domain adaptation through our proposed model, in which each category is represented as a cluster and different colors denote the different domains. Before adaptation, the source and target domains are totally mismatched, while our method shows the promising ability to make inter-class separated and intra-class clustered tightly.

Finally, we analyze the sensitivity of λ_1 (Fig. 1.6 (a))and λ_2 (Fig. 1.6 (b)) by listing four tasks from Office-Home dataset (Ar → Cl, Cl → Pr, Pr → Rw, Rw → Ar). Specifically, we set the ranges of λ_1 and λ_2 from 0.001 to 0.2 and evaluate one by fixing the other one as 0.1. From the results, we notice the accuracy curves are almost flat and stable, which indicates our proposed model is not sensitive to the values of λ_1 nor λ_2.

1.1.4 Discussion and Limitation

This work introduces AD^{2}CN, an unsupervised domain adaptation method aligning marginal and conditional distributions across domains. It outperforms state-of-the-art techniques on cross-domain visual benchmarks. However, a key limitation lies in the Importance Guided Optimization process, where the threshold σ is a hyper-parameter. Finding the optimal value or selecting correctly predicted target samples during training is crucial for real-life applications of this framework. Further research in this area is needed to enhance its practical utility.

1.2 Partial Domain Adaptation (PDA)

Partial domain adaptation (PDA) attracts appealing attention as it deals with a realistic and challenging problem when the source domain label space substitutes the target domain. Most conventional domain adaptation (DA) efforts concentrate on learning domain-invariant features to mitigate the distribution disparity across domains. However, it is crucial to alleviate the negative influence caused by the irrelevant source domain categories explicitly for PDA. In this work, we propose an Adaptively-Accumulated Knowledge Transfer framework (A^{2}KT) to align the relevant categories across two domains for effective domain adaptation. Specifically, an adaptively-accumulated mechanism is explored to gradually filter out the most confident target samples and their corresponding source categories, promoting positive transfer with more knowledge across two domains. Moreover, a dual distinct classifier architecture consisting of a prototype classifier and a multilayer perceptron classifier is built to capture intrinsic data distribution knowledge across domains from various perspectives. By maximizing the inter-class center-wise discrepancy and minimizing the intra-class sample-wise compactness, the proposed model is able to obtain more domain-invariant and task-specific discriminative representations of the shared categories data. Comprehensive experiments on several partial domain adaptation benchmarks demonstrate the effectiveness of our proposed model, compared with the state-of-the-art PDA methods.

1.2.1 Summary of Contribution

In this work, we propose an Adaptively-Accumulated Knowledge Transfer scheme (A^{2}KT) to manage partial domain adaptation challenges by simultaneously promoting positive transfer in the shared label space while alleviating negative transfer caused by the outlier source categories. The general idea is to gradually filter out confident task-relevant target samples and corresponding categories to optimize both domain-wise distribution adaptation and class-wise distribution alignment. To sum up, the contributions of this work are highlighted as follows:

- First of all, we propose an adaptively-accumulated knowledge transfer strategy to iteratively weigh and filter out confident task-relevant target samples and corresponding categories under the guidance of the source domain data for effective cross-domain alignment.
- Secondly, we explore two different types of task-specific classifiers to capture and transfer intrinsic distribution knowledge across domains from various perspectives.
- Thirdly, we propose a cross-domain alignment loss function that is able to align the class-level discrimination across domains and compact the sample-level distribution within the same class.

1.2.2 The Proposed Method

1.2.2.1 Preliminaries and Motivation

In partial domain adaptation, the source domain label space $\mathcal{Y}_s$ subsumes the target domain label space $\mathcal{Y}_t$, i.e., $\mathcal{Y}_s \supset \mathcal{Y}_t$, partial domain adaptation attempts to predict unlabeled target samples with the relevant source knowledge out of the entire well-labeled source domain.

To eliminate the influence of irrelevant source categories, existing partial domain adaptation models mainly design a weighting strategy to select the relevant source categories for effective cross-domain alignment with discrepancy loss [160] or adversarial loss [8]. To mitigate the conditional distribution mismatch across two domains, most of them rely on the pseudo labels of target samples assigned from a source-supervised neural network classifier. Due to the cross-domain distribution

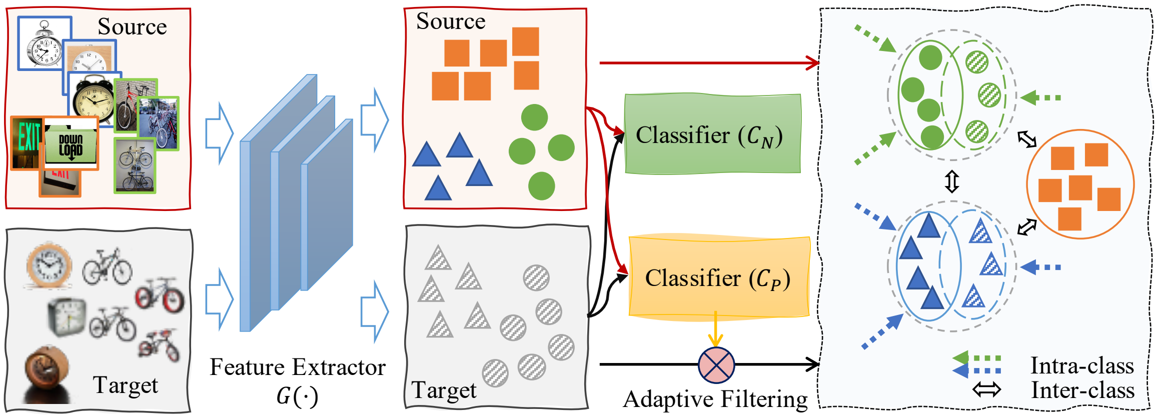


Figure 1.7: Illustration of the proposed model for partial domain adaptation, where the source contains more categories than the target. Both source and target data are input to the feature extractor $G(\cdot)$, then classified by multilayer perceptron classifier $C_N(\cdot)$ and prototype classifier $C_P(\cdot)$. The prediction results of $C_P(\cdot)$ are exploited to filter out confident target samples for further alignment across domains. Each shape denotes one category, colored and grey shapes mean the source and target samples respectively, while the colored but shaded shapes denote the filtered-out target samples with assigned pseudo labels.

gap, such pseudo labels are not reliable, which would further hurt the cross-domain alignment, since the neural network classifier fits perfectly for the source distribution while not for target distribution.

To address these issues, we consider not only detecting the irrelevant source categories to eliminate the negative influence but also selecting the most confident target samples during cross-domain alignment. Thus, our proposed model can adaptively select a subset of the target domain samples that are highly affiliated with the source domain and corresponding categories to align across domains. Moreover, the prototype classifier [121] is adopted to annotate the target samples via source prototypes, since it can capture the intrinsic structure and semantic knowledge across source and target domain. Exploring the dual classifier architecture consisting of two different types of classifiers, prototype classifier, and multilayer perceptron classifier, extends the ability of the proposed model to reveal the task-specific knowledge from various perspectives.

1.2.2.2 Adaptively-Accumulated Knowledge Transfer

1.2.2.2.1 Building Diverse Source-Supervised Classifiers As shown in Figure 1.7, we follow the architecture introduced in Section 1.1.2.2 to build the framework containing the feature extractor $G(\cdot)$, neural network classifier $C_N(\cdot)$ and teh prototype classifier $C_P(\cdot)$. To maintain performance on the source domain, we preserve the supervision from the source by minimizing the cross-entropy loss between the ground truth labels $\mathbf{Y}_s$ and the predicted labels $\hat{\mathbf{Y}}_{N,s}$ generated by the classifier $C_N(\cdot)$, as given by Eq. 1.1 as:

$$\mathcal{L}_y = \frac{1}{n_s}\sum\nolimits_{i=1}^{n_s} \mathcal{L}(\hat{\mathbf{y}}^i_{N,s}, y^i_s) \tag{1.7}$$

Upon observation, we have noticed that the supervision provided to the model $C_P(\cdot)$ using the source domain data has only a marginal impact on the performance of the partial task. Consequently, we have decided to focus solely on the objective loss associated with the model $C_N(\cdot)$, as it yields more significant improvements in the partial task's performance.

1.2.2.2.2 Adaptively Accumulating Cross-Domain Knowledge Empirical Maximum Mean Discrepancy (MMD) has been verified as a promising technique to minimize the cross-domain marginal distribution difference [79]. Some very recent works also adopt pseudo labels for target domain data in order to match the conditional distribution across-domain, by minimizing the distance between the source and target domain class-wise embeddings from the same category [130]. However, aligning all the target categories with the predicted label information is not effective since pseudo labels are not reliable, especially under the PDA settings.

To alleviate the negative impact of misclassified pseudo labels to target domain samples, as well as the outlier categories from source domain label space, we propose the *Adaptively-accumulated Knowledge Transfer* strategy to discard those target samples with low prediction confidence. That is, only samples with confidently predicted probability labels in

$$\widetilde{\mathcal{D}}_t = \{\mathbf{x}^i_t \in \mathcal{D}_t \mid \hat{\mathbf{y}}^{i,c}_{P,t} > p_0\}, \tag{1.8}$$

are accepted to update the cross-domain alignment, where c is the pseudo label of x^i_t, and $\hat{\mathbf{y}}^{i,c}_{P,t}$ is the probability confidence from prototype classifier $C_P(\cdot)$ of sample $\mathbf{x^i_t}$ belonging to class c. $p_0 \in [0, 1]$ is the threshold. It is noteworthy that we do not need to add another hyper-parameter to tune the model, as the probability confidence measures the similarity between the target sample to the source domain, we can let the model learn p_0 adaptively by setting it as the average of initial probability prediction produced by prototype classifier $C_P(\cdot)$ of source domain samples belonging to ground-truth class, which is $p_0 = \frac{1}{n_s} \sum_{\mathbf{x}_s^j \in \mathcal{D}_s} \hat{\mathbf{y}}_{P,s}^{j,c}$, where c is the ground-truth label of source sample $\mathbf{x}_s^j$. We only explore highly-confident target samples into the cross-domain alignment. In other words, the selected target samples may not cover the whole label space, which is reasonable and acceptable.

1.2.2.2.3 Preserving Inter-class Discrimination We treat the class-wise embeddings in a different way. Instead of matching the source and target domain mean embeddings from the same category, we seek to enlarge the distance between the source and target domain mean embeddings but from different classes. Specifically, we accept the L_2 distance to measure the distribution difference between two embeddings from two classes (c_i, c_j) and two domains (d_k, d_l):

$$\begin{aligned} \mathcal{F}_{c_i,c_j,d_k,d_l} &= \|\mu_{d_k,c_i} - \mu_{d_l,c_j}\|^2 \\ &= \left\| \frac{1}{N_{d_k,c_i}} \sum_{u=1}^{N_{d_k,c_i}} \mathbf{z}_{d_k,c_i}^{u} - \frac{1}{N_{d_l,c_j}} \sum_{v=1}^{N_{d_l,c_j}} \mathbf{z}_{d_l,c_j}^{v} \right\|^2 \end{aligned} \tag{1.9}$$

where $\mathbf{Z} \in \mathbb{R}^{d \times (n_s + n_t)}$ denotes the embedding feature matrix composed of $\{\mathbf{z}_s^1, \cdots, \mathbf{z}_s^{n_s}\}$ and $\{\mathbf{z}_t^1, \cdots, \mathbf{z}_t^{n_t}\}$, and $\mu_{d_{k/l},c_{i/j}} \in \mathbb{R}^d$ denotes the class center of data from category $c_{i/j}$ domain $d_{k/l}$.

It is noteworthy that d_k and d_l could be the same because we also seek to maximize the class-wise distance between different categories within the same domain. On the contrary, c_i and c_j are always different. The integrated inter-class discriminative alignment loss term includes **TWO** parts: (1) Aligning within source/target domain (2) Aligning across domains, which is shown as Eq. (1.10):

$$\begin{aligned} \mathcal{L}_{inter} =& \lambda_1 \Big(\sum_{c=1}^{C} \sum_{\substack{c'=1, \\ c' \neq c}}^{C} \frac{\mathcal{F}_{c,c',s,s}}{C(C-1)} + \sum_{c=1}^{\hat{C}} \sum_{\substack{c'=1, \\ c' \neq c}}^{\hat{C}} \frac{\mathcal{F}_{c,c',t,t}}{\hat{C}(\hat{C}-1)} \Big) \\ &+ \frac{1}{\hat{C}} \frac{1}{\hat{C}-1} \textstyle\sum_{c=1}^{\hat{C}} \sum_{\substack{c'=1, \\ c' \neq c}}^{\hat{C}} \mathcal{F}_{c,c',s,t}, \end{aligned} \tag{1.10}$$

where λ_1 is a hyper-parameter to balance the contribution of within-domain and between-domain terms in $\mathcal{L}_{inter}$. It is noteworthy that here C is the number of categories in the whole domain label space only when we align the inter-class discriminative distribution within source domain ($\mathcal{F}_{c,c',s,s}$), i.e., $C = \mid \mathcal{Y}_s \mid$. In other situations ($\mathcal{F}_{c,c',s,t}$, $\mathcal{F}_{c,c',t,t}$), $\hat{C}$ is the number of categories of the filtered out target domain subset $\widetilde{\mathcal{D}}_t$, which may be smaller than the number of categories in the whole source

Table 1.5: Comparisons of Recognition Rates (%) of Partial Domain Adaptation on Office-31 Dataset (ResNet-50).

Method	A31→W10	A31→D10	W31→A10	W31→D10	D31→A10	D31→W10	Average
Source Only	75.59±1.09	83.44±1.12	84.97±0.86	98.09±0.74	83.92±0.95	96.27±0.85	87.05±0.94
DAN [74]	59.32±0.49	61.78±0.56	67.64±0.29	90.45±0.36	74.95±0.67	73.90±0.38	71.34±0.46
DANN [31]	73.56±0.15	81.53±0.23	86.12±0.15	98.73±0.20	82.78±0.18	96.27±0.26	86.50±0.20
ADDA [130]	75.67±0.17	83.41±0.17	84.25±0.13	$\underline{99.85}$±0.12	83.62±0.14	95.38±0.23	87.03±0.16
RTN [62]	78.98±0.55	77.07±0.49	89.46±0.37	85.35±0.47	89.25±0.39	93.22±0.52	85.56±0.47
IWAN [160]	89.15±0.37	90.45±0.36	94.26±0.25	99.36±0.24	95.62±0.29	99.32±0.32	94.69±0.31
SAN [8]	90.90±0.45	94.27±0.28	88.73±0.44	99.36±0.12	94.15±0.36	99.32±0.52	94.96±0.36
PADA [9]	$\underline{96.54}$±0.31	82.17±0.37	$\underline{95.41}$±0.33	**100.00**±.00	92.69±0.29	99.32±0.45	92.69±0.29
DRCN [66]	90.80	94.30	94.80	**100.00**	95.20	**100.00**	95.90
ETN [10]	94.52±0.20	$\underline{95.03}$±0.22	94.64±0.24	**100.00**±.00	**96.21**±0.27	**100.00**±.00	$\underline{96.73}$±0.16
Ours(C_N)	92.18±0.12	92.95±0.24	**96.14**±0.23	**100.00**±.00	95.92±0.32	**100.00**±.00	96.20±0.15
Ours(C_P)	**97.28**±0.33	**96.79**±0.15	**96.14**±0.21	**100.00**±.00	$\underline{96.13}$±0.17	**100.00**±.00	**97.72**±0.14

domain label space, due to the *Adaptively-Accumulated Knowledge Transfer* strategy we proposed to filter out target samples with high prediction confidence.

1.2.2.2.4 Pursuing Intra-class Compactness Except for maximizing the inter-class distribution distance within/across domains, we also seek to pursue more intra-class compactness. Specifically, we develop an effective loss term to reduce the intra-class variation by minimizing the distance between every two samples belonging to the same category from any domains, which is shown as:

$$\mathcal{S}_c = \frac{1}{N_c(N_c-1)} \sum_{i=1}^{N_c} \sum_{\substack{j=1 \\ j \neq i}}^{N_c} \|\mathbf{z}^i - \mathbf{z}^j\|^2, \tag{1.11}$$

where N_c is the total number of samples belonging to class c from the source domain and filtered out target samples. Thus, we further define the total loss of all intra-class sample-wise distance as:

$$\mathcal{L}_{intra} = \frac{\lambda_2}{C} \sum_{c=1}^{C} \mathcal{S}_c, \tag{1.12}$$

where C is the number of categories in the source domain label space. It is noteworthy that for the target domain, we still only align those samples filtered out with high confidence to reduce the distraction of misclassification, while for samples from the source domain are always aligned over the whole label space. λ_2 is a hyper-parameter to balance the contribution of $\mathcal{L}_{intra}$.

Table 1.6: Comparisons of Recognition Rates (%) of Partial Domain Adaptation on Office-31 Dataset (VGG).

Method	A31→W10	A31→D10	W31→A10	W31→D10	D31→A10	D31→W10	Average
Source Only	60.34±0.84	76.43±0.48	79.12±0.54	99.36±0.36	72.96±0.56	97.97±0.63	81.03±0.57
DAN [74]	58.78±0.43	54.76±0.44	67.29±0.20	92.78±0.28	55.42±0.56	85.86±0.32	69.15±0.37
DANN [31]	50.85±0.12	57.96±0.20	62.32±0.12	94.27±0.16	51.77±0.14	95.23±0.24	68.73±0.16
ADDA [130]	53.28±0.15	58.78±0.12	63.34±0.08	95.36±0.08	50.24±0.10	94.33±0.18	69.22±0.12
RTN [62]	69.35±0.42	75.43±0.38	82.98±0.36	99.59±0.32	81.45±0.32	98.42±0.48	84.54±0.38
IWAN [160]	82.90±0.31	**90.95**±0.33	93.36±0.22	88.53±0.16	89.57±0.24	79.75±0.26	87.51±0.25
SAN [8]	83.39±0.36	90.70±0.20	91.85±0.35	**100.00**±.00	87.16±0.23	99.32±0.45	92.07±0.27
PADA [9]	86.05±0.36	81.73±0.34	**95.26**±0.27	**100.00**±.00	93.00±0.24	99.42±0.24	92.54±0.24
ETN [10]	85.66±0.16	89.43±0.17	92.28±0.20	**100.00**±.00	**95.93**±0.23	**100.00**±.00	93.88±0.13
Ours(C_N)	88.44±0.24	86.54±0.15	94.98±0.38	**100.00**±.00	94.98±0.21	99.32±0.18	94.04±0.19
Ours(C_P)	**90.48**±0.23	90.38±0.38	95.19±0.16	**100.00**±.00	94.67±0.19	99.66±0.23	**95.06**±0.20

1.2.2.3 Overall Objective and Optimization

Entropy minimization regularization is adopted to eliminate the side effect caused by the uncertainty of classifiers, due to the large domain shift and samples which are hard to transfer. Especially during the early training stage, the target domain samples are easy to be assigned to wrong categories and may deteriorate the optimization procedures. We also explore the entropy minimization regularization as:

$$\mathcal{L}_{em} = -\frac{1}{n_t}\sum\nolimits_{i=1}^{n_t}\sum\nolimits_{c=1}^{C}\hat{\mathbf{y}}_{N,t}^{i,c}\log\hat{\mathbf{y}}_{N,t}^{i,c}, \tag{1.13}$$

where C is the number of categories in source domain label space, n_t is the number of samples from the target domain.

To sum up, we propose our overall objective function as:

$$\min_{G,C_N}\mathcal{L}_y + \mathcal{L}_{intra} - \mathcal{L}_{inter} + \mathcal{L}_{em}. \tag{1.14}$$

The whole framework consists of a feature generator $G(\cdot)$, a multilayer perceptron classifier $C_N(\cdot)$, and a prototype classifier $C_P(\cdot)$. As $C_P(\cdot)$ is non-parameter, so only $G(\cdot)$ and $C_N(\cdot)$ are optimized with the objective as Eq. (1.14). Specifically, $\mathcal{L}_y$ is calculated on the source domain data, while $\mathcal{L}_{em}$ is based on the whole target domain. However, $\mathcal{L}_{intra}$ and $\mathcal{L}_{inter}$ are only based on the filtered-out target data, as well as the corresponding source data from the same categories as the filtered target samples pseudo labels. It is important to note that the inter-class and intra-class losses used here differ from the objective introduced in Section 1.1.2.2.3. The losses, denoted as $\mathcal{L}_{inter}$

Table 1.7: Comparisons of Recognition Rates (%) of Partial Domain Adaptation on Office+Home Dataset (ResNet-50).

Method	Ar → Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg.
Source Only	46.33	67.51	75.87	59.14	59.94	62.73	58.22	41.79	74.88	67.40	48.18	74.17	61.35
DAN [74]	43.76	67.90	77.47	63.73	58.99	67.59	56.84	37.07	76.37	69.15	44.30	77.48	61.72
DANN [31]	45.23	68.79	79.21	64.56	60.01	68.29	57.56	38.89	77.45	70.28	45.23	78.32	62.82
ADDA [130]	45.23	68.79	79.21	64.56	60.01	68.29	57.56	38.89	77.45	70.28	45.23	78.32	62.82
RTN [62]	49.31	57.70	80.07	63.54	63.47	73.38	65.11	41.73	75.32	63.18	43.57	80.50	63.07
IWAN [160]	53.94	54.45	78.12	61.31	47.95	63.32	54.17	52.02	81.28	76.46	56.75	82.90	63.56
SAN [8]	44.42	68.68	74.60	**67.49**	64.99	**77.80**	59.78	44.72	80.07	72.18	50.21	78.66	65.30
PADA [9]	51.95	67.00	78.74	52.16	53.78	59.03	52.61	43.22	78.79	73.73	56.60	77.09	62.06
DRCN [66]	54.00	76.40	83.00	62.10	64.50	71.00	**70.80**	49.80	80.50	**77.50**	59.10	79.90	69.00
ETN [10]	59.24	77.03	79.54	62.92	65.73	75.01	68.29	55.37	**84.37**	75.72	57.66	**84.54**	70.45
Ours(C_N)	61.41	83.81	86.36	64.15	74.12	75.15	67.22	**55.44**	83.88	72.15	60.22	83.59	72.29
Ours(C_P)	**62.54**	**83.92**	**86.69**	65.44	**74.96**	75.04	67.40	55.14	**84.37**	73.25	**60.51**	84.09	**72.78**

and $\mathcal{L}_{intra}$, are designed to minimize the distances between features belonging to the same category while maximizing the distances between features from different categories, regardless of whether they originate from the same or different domains. This optimization objective aims to enhance the discrimination of features among different categories in the domain-invariant feature space.

1.2.3 Experiments

1.2.3.1 Implementation Details

Table 1.8: Comparisons of Recognition Rates (%) of Unsupervised Domain Adaptation on Office+Home Dataset (ResNet-50).

Method		Ar → Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg.
No Adaptive	C_N	51.79	70.42	79.40	56.16	62.97	70.40	60.42	48.15	76.75	66.08	**63.94**	76.58	65.26
	C_P	51.31	70.31	79.18	56.16	63.08	70.04	60.51	48.03	75.76	66.08	53.52	76.64	64.25
C_N Guide	C_N	62.09	81.01	83.60	60.75	64.48	65.27	65.20	53.52	**84.76**	71.23	56.39	80.06	69.03
	C_P	61.95	80.84	83.32	60.94	64.71	65.93	65.56	53.58	**84.76**	71.14	56.39	79.89	69.08
Same C_N&C_P	C_N	56.75	80.06	87.36	60.20	64.99	**76.97**	65.75	55.14	83.27	69.30	55.08	82.18	69.75
	C_P	56.81	80.00	**87.41**	60.29	64.93	**76.97**	65.75	55.08	83.27	69.30	55.02	82.18	69.75
Ours	C_N	61.41	83.81	86.36	64.15	74.12	75.15	67.22	**55.44**	83.88	72.15	60.22	83.59	72.29
	C_P	**62.54**	**83.92**	86.69	**65.44**	**74.96**	75.04	**67.40**	55.14	84.37	**73.25**	60.51	**84.09**	**72.78**

Comparisons: We compare the performance of our proposed method with several domain adaptation and the state-of-the-art partial DA methods such as: Deep Adaptation Network (DAN) [74], Adversarial Discriminative Domain Adaptation (ADDA) [130], Residual Transfer Network (RTN) [62], Importance Weighted Adversarial Nets (IWAN) [160], Selective Adversarial Network (SAN) [8], Partial Adversarial Domain Adaptation (PADA) [9], Example Transfer Network (ETN) [10], and Adaptive Feature Norm (AFN) [149]. Specifically, DAN applies multi-kernel MMD to match source and target domain distribution and learn transferable features across the domain. ADDA combines the adversarial training idea and united weights sharing to generate domain invariant features. RTN jointly adapts feature distribution as well as source and target classifiers via a deep residual learning framework. IWAN and SAN select or re-weight outlier categories in the source domain label space to alleviate the negative influence caused by those classes that are not in the target domain label space. PADA, ETN, and AFN are state-of-the-art partial domain adaptation models. Through down-weighting source domain data which is from outlier categories, PADA reduces the negative transfer influence caused by outlier classes. ETN proposes a progressive weighting scheme to quantify the transferability of source examples. AFN proposes a parameter-free approach to progressively adapt the source and target domain feature norms to a large range of values, which results in significant transfer gains.

Implementation Details: For each source-target pair case, we finetune the ImageNet pre-trained convolutional neural networks on the source domain and remove the last fully-connected layer as the backbone network. Then we input the backbone networks output of all source and target domain data into two dense layers with hidden layer output as 1,024 followed by ReLU activation and 0.1 dropout probability as the feature extractor $G(\cdot)$. We accept ResNet-50 network [35] as the backbone of Office-Home and Office-31, and also explore the performance of the VGG network as the backbone [120] on the Offce-Home dataset. The output dimension of the generator $G(\cdot)$, as known as the embedding features $\mathbf{z}_{x/t}$, is 512. The multilayer perceptron classifier $C_N(\cdot)$ is a two-layer fully-connected neural network where the hidden layer output dimension is 512, and the output size is the number of source domain categories. For prototype classifier $C_P(\cdot)$, we take cosine similarity as the measurement function in $(\cdot,\cdot)$, and we directly take the source domain class centers as the prototypes, because the feature generator update every epoch, so the prototypes are also updating along with training. All experiments are implemented via PyTorch. We train the model for 100 epochs by Adam optimizer with learning rate as 0.0001, and report the last epoch results. p_0 is rounded to two decimal places. $\lambda_1 = 0.1$ and $\lambda_2 = 0.5$ on Office31 dataset, while $\lambda_1 = 0.01, \lambda_2 = 2$ on Office-Home. We will analyze the parameter sensitivity in Section 1.2.3.3.

1.2.3.2 Comparison Results

In this section, we will comprehensively evaluate our proposed model with several baselines on Office-31 and Office-Home benchmarks in terms of the target samples labels prediction accuracy to manifest the effectiveness of our model.

Specifically, we observe that PDA methods (IWAN, SAN, PADA, DRCN, and ETN) achieve better performance than standard DA efforts such as DAN, DANN, ADDA, and RTN. ETN achieves much greater improvement because it introduces a method to quantify the source samples' transferability. Our proposed method can still outperform all compared baselines on most partial domain adaptation tasks and obtain the best average performance.

Table 1.5 reports the classification accuracy on the Office-31 dataset obtained by all baselines and our model with ResNet-50 as the backbone of the feature extractor. It is noteworthy that the prototype classifier $C_P(\cdot)$ always generates better performance than the conventional multilayer perceptron classifier $C_N(\cdot)$. From the results, the prototype classifier achieves the best performance on 5 out of 6 tasks, compared to all the other baselines. To be specific, the average classification accuracy reaches the best performance 97.72% and reaches 100% accuracy on W31 $\rightarrow$ D10 and D31 $\rightarrow$ W10.

Moreover, we also explore the VGG network as the feature extractor backbone on Office-31 dataset and report the results in Table 1.6. Our proposed model achieves the best average performance compared with other baselines. Specifically, compared to the best baseline performance on task A31→W10, PADA, $C_N(\cdot)$ and $C_P(\cdot)$ improve the accuracy over 2% to 88.44% and 4% to 90.48%, respectively. It is noteworthy that the improvements of performance with VGG networks as backbone is more significant than using ResNet-50, because the ResNet-50 is more advanced deep convolutional neural networks model, which can generate more task specific discriminate features than VGG networks.

Experiment results on the Office-Home dataset are stated in Table 1.7. Both $C_N(\cdot)$ and $C_P(\cdot)$ obtain better performance against other baselines with significant improvements on average classification accuracy (1.84% and 2.33%). Moreover, our proposed method achieves more than 5% accuracy increase compared to the state-of-the-art baseline, e.g., Ar $\rightarrow$ Pr, Cl $\rightarrow$ Pr, etc.

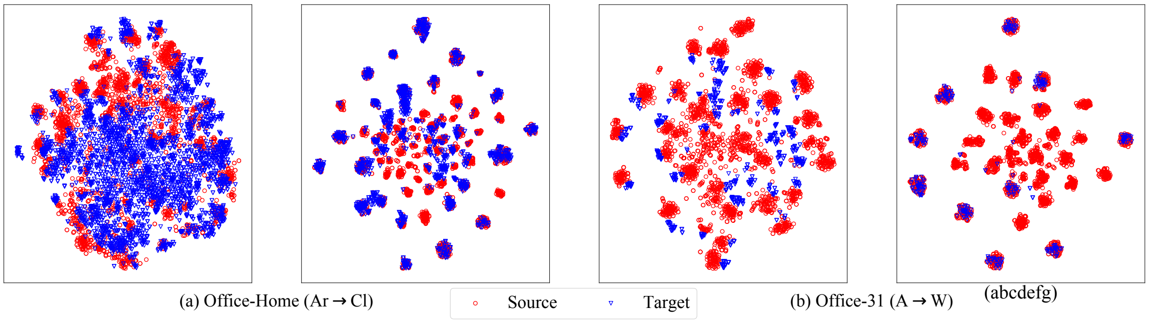


Figure 1.8: tSNE visualization of the original features and generator $G(\cdot)$ output embedding features after domain adaptation. (a) Office-Home dataset (Al → Cl) (b) Office-31 dataset (Amazon → Webcam).

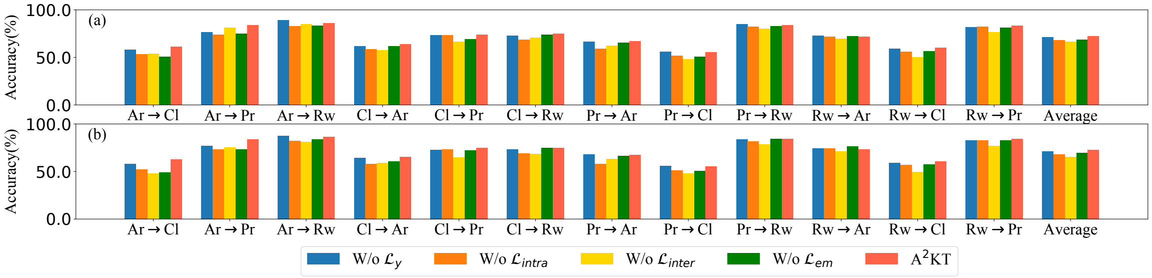


Figure 1.9: Evaluate each loss term contribution on the Office-Home dataset by removing each specific term while keeping other parts the same. (a) multilayer perceptron classifier $C_N(\cdot)$ (b) Prototype Classifier $C_P(\cdot)$.

1.2.3.3 Ablation Analysis

First, visualize the generator $G(\cdot)$ output features before and after the domain adaptation process on task Ar→Cl on Office-Home, and A→W on Office-31 dataset in the Fig. 1.8 (a) and (b). From the results, we observe that our proposed method aligns the source and target domain samples with respect to categories, and tights the compactness of the embedding features to each class center.

Secondly, we evaluate the contribution of every loss term in Eq. (1.14) by removing each specific term while keeping other terms as the original framework. The results are shown in Fig. 1.9. It is noteworthy that both $\mathcal{L}_{intra}$ and $\mathcal{L}_{inter}$ make crucial contributions to the PDA tasks because these two terms are aligning the data distribution inter-classes and intra-class. $\mathcal{L}_y$ keeps the model performance on the source domain stable, while it has limited contribution to the PDA process, but cannot be ignored. $\mathcal{L}_{em}$ helps to mitigate the negative transfer influence of the multilayer perceptron classifier C_N, especially at the beginning of the training stage.

Then, we monitor the training and optimization process of our model. Fig. 1.10 illustrates the

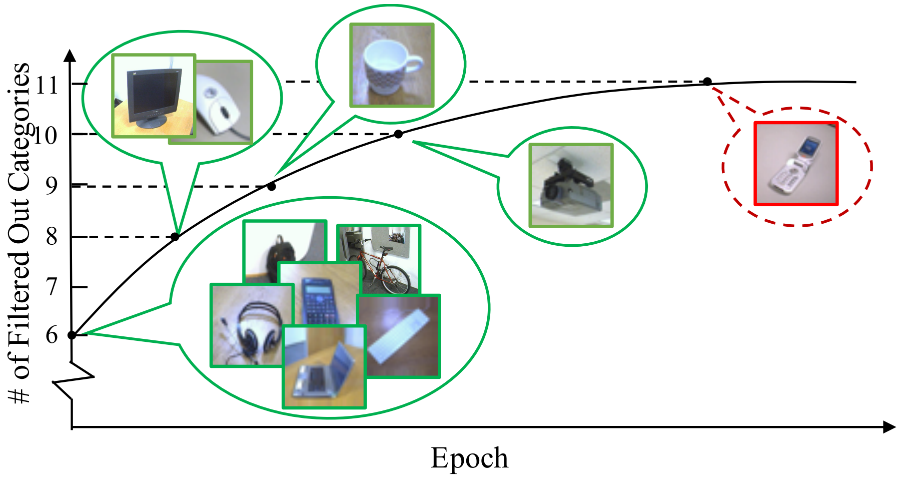


Figure 1.10: Filtered out shared categories of the target domain of task A31 → W10 on the Office-31 dataset.

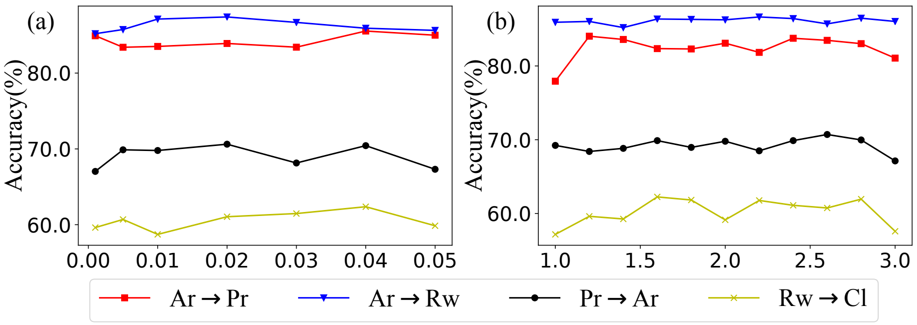
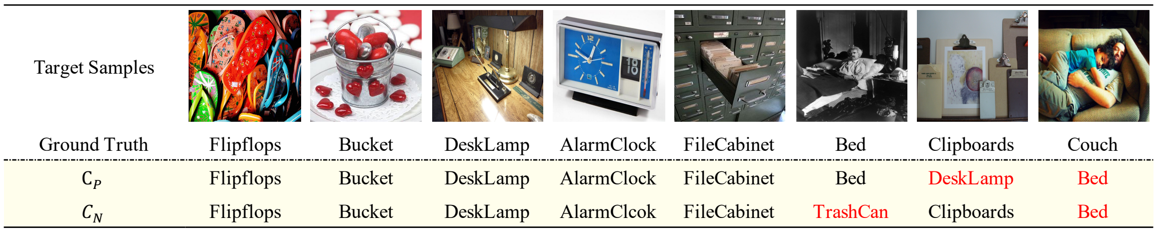


Figure 1.11: Parameters sensitivity analysis of (a) λ_1 (b) λ_2 on 4 different tasks from Office-Home dataset.

process of the *adaptively-accumulated knowledge transfer* process. We choose case A31 → W10 of the Office-31 dataset and show the changing of the filtered-out high prediction confidence categories used to align the data distribution across domains. In the beginning, high prediction target samples only spread into only 6 classes, but then more and more categories are involved, and the number finally reaches 11, while the total number of the target domain categories is 10. Although there is an incorrect outlier class involved, the adaptive optimization strategy still significantly narrows the range of the target domain label space.

Moreover, we implement several ablation experiments on the Office-Home dataset with different training details to explore the contribution of our proposed model and optimization strategy, the results are reported in Table 1.8. “No Adaptive” denotes the results without the adaptively accumulating knowledge transfer and target samples filtering out process. From the results, compared to



Target Samples								
Ground Truth	Flipflops	Bucket	DeskLamp	AlarmClock	FileCabinet	Bed	Clipboards	Couch
C_P	Flipflops	Bucket	DeskLamp	AlarmClock	FileCabinet	Bed	DeskLamp	Bed
C_N	Flipflops	Bucket	DeskLamp	AlarmClcok	FileCabinet	TrashCan	Clipboards	Bed

Figure 1.12: Prediction of $C_N(\cdot)$ and $C_P(\cdot)$ for selected target domain samples (Pr → Rw)

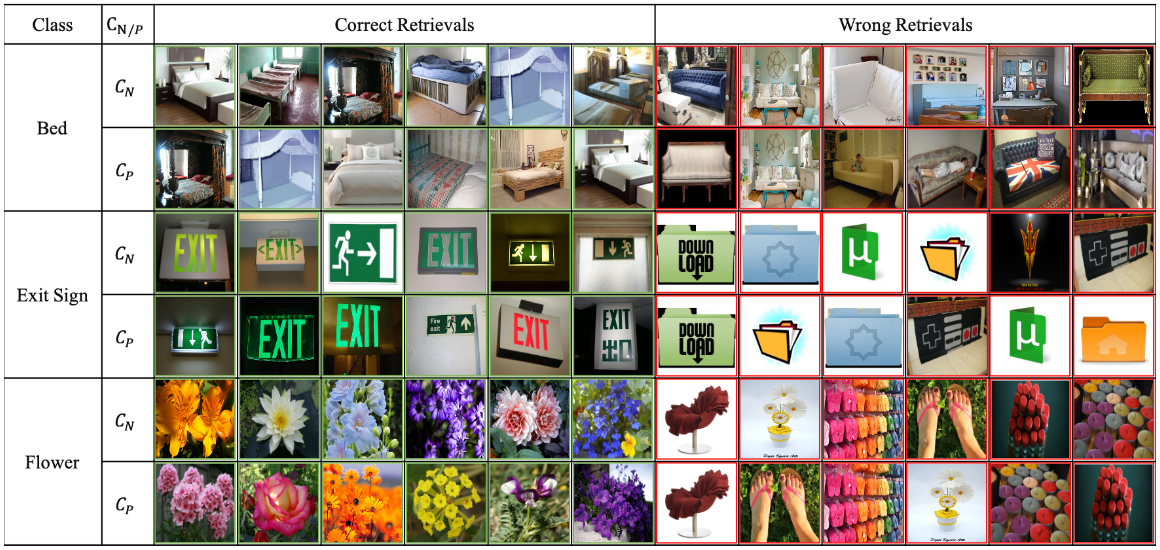


Figure 1.13: Retrieved target images with the highest 10 prediction confidence by $C_P(\cdot)$ (Pr → Rw).

our complete A^{2}KT model results, we notice how important the *adaptively accumulating knowledge* strategy is. "C_N Guide" are the results when we use the C_N probabilistic prediction to filter out high confidence target samples for domain alignment, instead of C_P. The way to decide the threshold is the same as when we use C_P. The results prove that the multilayer perceptron classifier C_N and the prototype classifier C_P have different classification philosophies, and using C_P probability prediction to accumulate can boost the performance significantly. Finally, we explore the motivation of adopting two different types of dual classifiers framework in our model by setting C_N and C_P both the same structure multilayer perceptron classifiers, all other settings and training strategies are the same as before, and the results are reported in "Same C_N&C_P." From the results, we observe that for some cases two same multilayer perceptron classifiers can get slightly better performance than our model, e.g., Ar → Rw and Cl → Rw. However, for most cases and the average performance, our model with different type classifiers outperforms much more. All the results with different training strategies in Table 1.8 demonstrate the effectiveness and motivation of our model and optimization strategies.

We present the parameter sensitivity analysis in Fig. 1.11. We vary λ_1 from 0.0001 to 0.05 and λ_2 from 1 to 3 on four cases on the Office-Home dataset (Ar $\rightarrow$ Pr, Ar $\rightarrow$ Rw, Pr $\rightarrow$ Ar, Rw $\rightarrow$ Cl) to analyze if the model is sensitive to the change of the hyper-parameters. The results in Fig. 1.11 show that our model has great stability across cases of the two parameters λ_1 and λ_2.

Finally, we select several representative target samples from task Pr$\rightarrow$Rw on Office-Home dataset and show the predictions of $C_N(\cdot)$ and $C_P(\cdot)$ in Fig. 1.12. We notice that some cases only $C_N(\cdot)$ or $C_P(\cdot)$ can handle, or even neither can predict correctly, which demonstrates the motivation of combining two different type classifiers $C_N(\cdot)$ and $C_P(\cdot)$ in our proposed model. Besides, we operate the image retrieval task by giving specific labels to retrieve the target samples. The 5 target images with the highest $C_P(\cdot)$ prediction confidence and 5 with the lowest in the retrieved images are shown in Fig. 1.13. The different samples retrieved by $C_N(\cdot)$ and $C_P(\cdot)$ demonstrate the motivation of integrating various classifiers.

1.2.4 Discussion and Limitation

This work presents a novel Domain-Invariant Feature learning framework for partial domain adaptation. The method uses the Adaptively-Accumulated Knowledge Transfer Optimization strategy to select relevant target domain samples, leading to improved results compared to prior approaches in extensive experiments on various benchmarks. However, determining an effective and robust threshold (σ) for confident prediction on the target domain remains a significant challenge, particularly when dealing with extremely imbalanced source domain data in the feature space. Consider a scenario where certain categories have only a few samples that are tightly clustered around the class center, while other classes have a larger number of samples but are sparsely distributed with significantly larger distances from their class center. In such cases, the threshold determination would be heavily influenced by the distribution of categories with a larger number of samples, leading to poor performance for the minority group categories.

Another limitation of the method is the expensive computational cost incurred by the objective loss functions $\mathcal{L}_{inter}$ and $\mathcal{L}_{intra}$, which involve all samples in the source and target domains. This computational burden becomes particularly pronounced when dealing with large-scale datasets. To make the approach more feasible for such datasets, it is crucial to devise an effective strategy to reduce the computing complexity. One potential solution is to sample the source samples during the training process, ensuring that each batch covers all source domain categories. By performing loss calculations on these sampled batches and optimizing the model accordingly, the computational overhead can be significantly mitigated.

1.3 Conclusion

In conclusion, this chapter has contributed to the understanding of visual domain adaptation by employing feature distribution analysis as a key interpretive tool. Through the proposed Adversarial Dual Distinct Classifiers Network (AD^{2}CN), the chapter has effectively aligned domain distributions and category decision boundaries, shedding light on the knowledge transfer process from the source to the target domain. The utilization of dual different-architecture classifiers has provided valuable insights into capturing ground-truth decision boundaries and improving prediction performance. Furthermore, the chapter's investigation into the Adaptively-Accumulated Knowledge Transfer scheme (A^{2}KT) has revealed important findings regarding the discovery of outlier classes when the source and target domains possess different label spaces. By analyzing domain-invariant feature distributions, this chapter has deepened our understanding of the underlying mechanisms involved in visual domain adaptation. The insights gained from this analysis offer valuable guidance for developing effective strategies and methodologies to address the challenges posed by domain shifts in various visual recognition tasks.

[50] Jing, Taotao, Bingrong Xu, and Zhengming Ding. "Towards fair knowledge transfer for imbalanced domain adaptation." IEEE Transactions on Image Processing 30 (2021): 8200-8211.

[48] Jing, Taotao, Haifeng Xia, Jihun Hamm, and Zhengming Ding. "Marginalized Augmented Few-Shot Domain Adaptation." IEEE Transactions on Neural Networks and Learning Systems (2023).

2

Domain Adaptation with Limited Training Data using Feature Generation

We have discussed that domain adaptation methods enable deep neural networks to overcome limitations due to scarce labeled data by leveraging knowledge from an external source domain. However, the insufficiency of unlabeled target domain data severely limits the adaptation ability and knowledge transfer effectiveness of existing domain adaptation models, giving rise to the problem of *few-shot domain adaptation* (FSDA) [84, 86, 150]. FSDA faces two primary challenges: domain shift, which refers to the difference in data distributions between domains and often results in negative transfer during adaptation, and data imbalance, which occurs when the source and target domains have disparate amounts of training data, leading to models that overfit to the abundant source data.

Besides, to address the data imbalance in domain adaptation, some researchers have focused specifically on imbalanced domain adaptation tasks. These methods aim to mitigate the negative effects of data imbalance by assigning importance weights to each sample in the target domain [33, 160, 62, 163, 26, 54]. However, these strategies rely on the source classifier to assign the importance weights, which can be unreliable in extreme cases where certain categories lack sufficient data for training a reliable classifier. Consequently, maintaining performance on specific categories with limited training data becomes crucial and challenging, giving rise to transfer fairness problems [117].

In this work, we focus on addressing the challenges of imbalanced domain adaptation and domain adaptation with limited training data. We aim to develop novel approaches that effectively adapt models to target domains with data imbalance and overcome the limitations imposed by limited labeled data in the target domain. By exploring the potential of class-wise adaptation and developing techniques that account for data scarcity and imbalance, our goal is to improve the performance and fairness of domain adaptation in practical settings.

2.1 Towards Fair Knowledge Transfer for Imbalanced Domain Adaptation

Domain adaptation (DA) becomes an up-and-coming technique to address the insufficient or no annotation issue by exploiting external source knowledge. Existing DA algorithms mainly focus on practical knowledge transfer through domain alignment. Unfortunately, they ignore the fairness issue when the auxiliary source is extremely imbalanced across different categories, which results in severely under-presented knowledge adaptation of the minority source set. In addressing the challenging imbalanced domain adaptation problem, we introduce the Towards Fair Knowledge Transfer (TFKT) framework. Our framework centers on a novel cross-domain knowledge propagation technique, guided by within-source and cross-domain structure graphs. This technique effectively generates additional data for the minority source set, addressing the data imbalance issue. Additionally, we employ a cross-domain mix-up augmentation strategy to facilitate domain adaptation. By combining these approaches, we aim to foster a more equitable and efficient knowledge transfer process. Moreover, hybrid distinct classifiers and cross-domain prototype alignment are adopted to seek a more robust classifier boundary and mitigate the domain shift. Such three strategies are formulated into a unified framework to address the fairness issue and domain shift challenge. Ex-

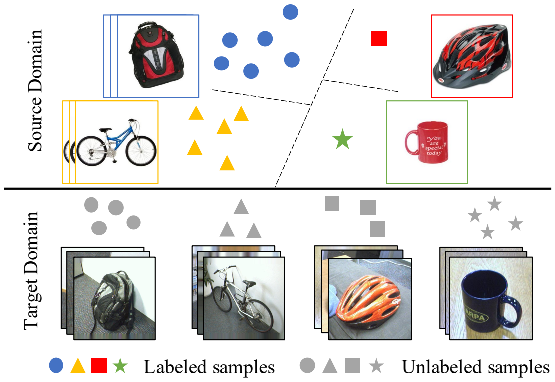


Figure 2.1: Illustration of the imbalanced domain adaptation task. The colored shapes denote labeled but extremely imbalanced source domain data, in which some categories only contain few samples, e.g., one-shot. The gray shapes are unlabeled target domain data.

tensive experiments over two popular benchmarks have verified the effectiveness of our proposed model by comparing it to existing state-of-the-art DA models, and especially our model significantly improves over 20% on two benchmarks in terms of overall accuracy.

2.1.1 Summary of Contribution

In this work, we consider the source fairness challenge in domain adaptation under the extreme condition when the available source domain is extremely imbalanced as illustrated in Fig. 2.1, i.e., some source domain categories only contain a few labeled samples for training. Consequently, we propose a novel Towards Fair Knowledge Transfer (TFKT) framework to guarantee faithful cross-domain adaptation. The contributions of this work are summarized in four folds as follows:

- First of all, we propose knowledge propagation within the source domain and across source and target domains with the weighted structure graph guidance to smooth the manifold and alleviate the distraction caused by the undesired random few-shot samples belonging to the source domain minority categories.

- Secondly, we exploit the cross-domain mix-up augmentation strategy based on the refined data through knowledge propagation to achieve cross-domain alignment and eliminate the domain shift.

- Thirdly, we enhance the faithful knowledge transfer by exploring dual-classifier mechanisms and cross-domain alignment, to seek more robust task-specific classification boundaries and domain-invariant feature representation.

- Finally, we extensively evaluate our proposed model under various challenging but realistic settings and compare the performance with state-of-the-art methods. The superior results, even in extreme situation with only one labeled source sample available for some classes, emphasize the effectiveness of our model.

2.1.2 The Proposed Method

2.1.2.1 Preliminaries and Motivation

In this work, we consider a challenging but realistic domain adaptation task involving a well-labeled but extremely-imbalanced source domain training data. The source domain, denoted as $\mathcal{D}_s$, consists of two subsets, a majority-set $\mathcal{D}_s^m$ and a minority-set $\mathcal{D}_s^f$, where the majority-set $\mathcal{D}_s^m$ have n_s^m samples available and each category consists of sufficient instances with annotations from the label space $\mathcal{Y}^m$, while the minority-set $\mathcal{D}_s^f$ only contains n_s^f data drawn from P categories from label space $\mathcal{Y}^f$, with limited Q samples per class, and we describe it as P way Q shot task. $\mathbf{Z}_{s/t} = E(\mathbf{X}_{s/t}), \mathbf{Z}_{s/t} \in \mathbb{R}^{n_{s/t} \times d}$ is the source/target embedding representations extracted from pre-trained backbone network $E(\cdot)$, where d is the embedding dimension. In the rest of this work, we choose ResNet-50 [35] pre-trained on ImageNet [16] as the convolutional backbone and accept the output before the last fully-connected layer as the embedding representation. Unlike the feature extractor $G(\cdot)$ introduced in Section 1.1, the parameters of $E(\cdot)$ are frozen without training in our experiments to reduce computing costs. The source and target domains are drawn from different distributions P_s and P_t, but lie in the same label space $\mathcal{Y} = \mathcal{Y}^m \cup \mathcal{Y}^f$, thus the number of categories in the source and target domains are identical.

Some source domain categories only contain a few samples for training that will fail the conventional unsupervised domain adaptation solutions, relying on plenty of well-labeled source domain data for supervised training. The extremely imbalanced distribution will distract the optimization direction and mislead the model overfitting to those classes with adequate training data while ignoring the few-shot categories. We address such challenges by refining the embedding representations of samples from few-shot categories through structure-guided knowledge propagation to eliminate the undesired noise distraction, then synthesizing those few-shot categories to expand the feature

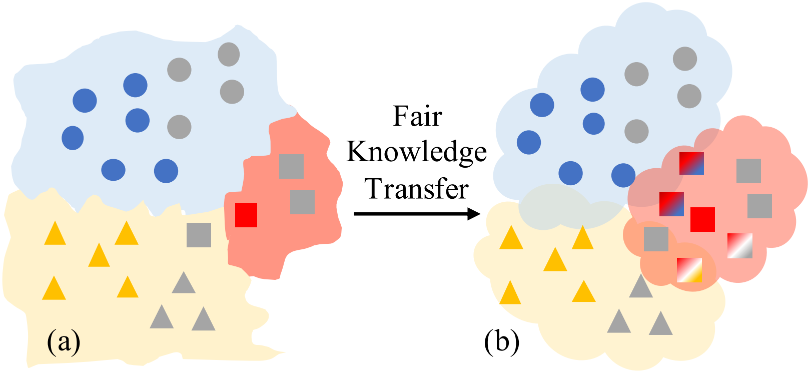


Figure 2.2: Illustration of the fair knowledge transfer process, which is able to expand the feature space of categories with a few labeled samples, and balance the decision boundaries. Different shapes represent different categories, while colored and gray instances denote labeled source and unlabeled target samples, respectively.

distribution space and avoid the imbalance issues, and finally exploring the dual classifier mechanism to align the source and target domain and alleviate domain shift. The fair knowledge transfer process can expand the feature space of minority set categories with few labeled training samples, and smooth the decision boundaries (Fig. 2.2).

2.1.2.2 Towards Fair Knowledge Transfer

We first present an overview of our proposed framework (Fig. 2.3). Firstly, the source domain few-shot category sample ($\mathbf{z}_s^f$) is augmented to obtain refined embeddings ($\tilde{\mathbf{z}}_{s/o}^f$) through the knowledge propagation within the source and across domains under the guidance of weighted structure graphs ($\mathbf{H}_{s/o}$). Besides, the refined synthesized samples ($\tilde{\mathbf{z}}_{s/o}^f$) are used to generate more synthesized instances ($\hat{\mathbf{z}}_m^f$) to expand the feature space and fulfill the distribution gap across domains through random combination. Finally, all the real and synthesized instances are mapped into a domain invariant space through the feature generator $F(\cdot)$, denoting the output features as $\mathbf{f} = F(\mathbf{z})$, guided by the discriminative cross-domain alignment and source domain supervision objectives obtained from the dual-classifier mechanism, consisting of a multi-layer neural network classifier $C_N(\cdot)$ and a prototypical classifier $C_P(\cdot)$. We will discuss details about each part in the following sections.

2.1.2.3 Cross-domain Feature Augmentation (CDA)

2.1.2.3.1 Data Augmentation through Embedding Propagation The most challenging difference between the extreme imbalanced domain adaptation and conventional domain adaptation tasks is

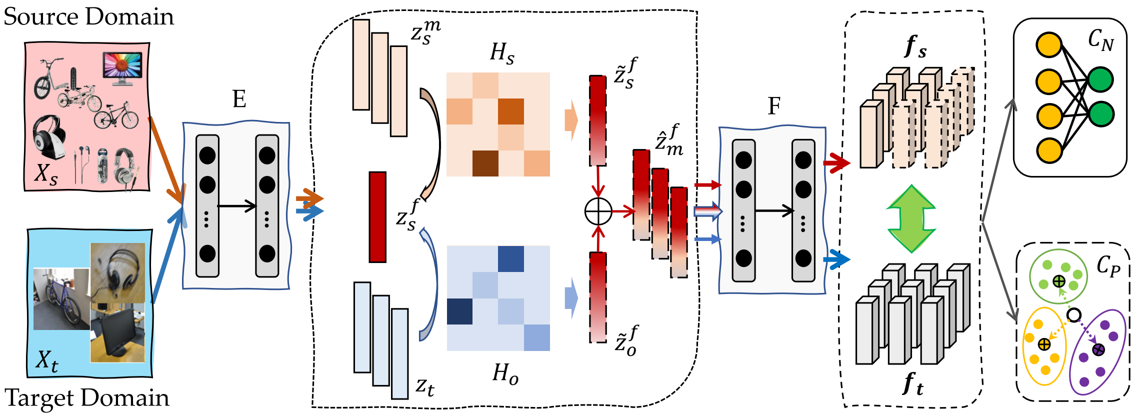


Figure 2.3: Overview of our proposed framework, where both source and target raw images ($\mathbf{x}_{s/t}$) are input to pre-trained deep convolutional neural networks $E(\cdot)$ to extract embedding representations ($\mathbf{z}_{s/t}$). $\mathbf{z}_s^m$ denotes the samples from the majority set categories, while $\mathbf{z}_s^f$ denotes the samples from the source domain minority categories, and $\mathbf{z}_t$ are the samples from the target domain without labels. $\mathbf{H}_s$ is the weighted graph illustrating the structure of the source domain data, while graph $\mathbf{H}_o$ is obtained based on the relationships between samples from the source domain few-shot set and the target domain samples. $\tilde{\mathbf{z}}_{s/o}^f$ are the augmented refined embeddings through knowledge propagation within the source domain and across domains, respectively, which are used to generate more synthesized instances ($\hat{\mathbf{z}}_m^f$) through random combination to expand the feature space and fulfill the distribution gap across domains. All real and synthesized embedding instances are mapped into a domain invariant space through the feature generator $F(\cdot)$, denoting the output features as $\mathbf{f}_{s/t}$. The dual-classifier scheme consists of a multi-layer neural network classifier $C_N(\cdot)$ and a prototypical classifier $C_P(\cdot)$, which aims to preserve source supervision.

that some categories from the source domain only contain a few labeled samples for training. The optimization process of previous domain adaptation solutions would be dominated and misled by the majority set classes having sufficient training data and fail on the minority set categories. Besides, the limited labeled samples from the minority set categories may lie far from the class center in the feature space, which cannot represent the corresponding categories' distribution characteristics. To address this challenge, we first explore the augmentation strategy *embedding propagation* within the source domain [107]. Specifically, for each source sample from the minority set $\mathcal{D}_s^f$, an interpolated embedding is constructed through the combination of its neighbors with the knowledge propagated under the guidance of a weighted graph. The goal of embedding propagation is to remove the noise from the features and smooth the embedding manifold, which will benefit the generalization and effectiveness of semi-supervised learning methods [122, 63, 107].

Firstly, we build a similarity adjacency matrix $A_s \in \mathbb{R}^{n_s \times n_s}$ for all samples in the source domain

$\mathcal{D}_s$, and each element in A_s is computed as:

$$A_s^{i,j} = \exp(-\frac{d^2_{ij(s)}}{\sigma_s^2}) = \exp(-\frac{\|\mathbf{z}_s^i - \mathbf{z}_s^j\|^2}{\sigma_s^2}), \tag{2.1}$$

in which $d_{ij(s)} = \|\mathbf{z}_s^i - \mathbf{z}_s^j\|_2$ is the distance between two samples features $\mathbf{z}_s^i$ and $\mathbf{z}_s^j$, both from the source domain $\mathcal{D}_s$, and σ^2 is the scaling factor which is set as the standard deviation of the distances, i.e., $\sigma^2 = Var(d^2_{ij(s)})$, and $A_s^{kk} = 0, \forall k$ [73]. Then based on the pair-wise similarity adjacency graph, the Laplacian matrix can be obtained as:

$$L_s = D_s^{-1/2} A_s D_s^{-1/2}, \tag{2.2}$$

where $D_s^{ii} = \sum_j A_s^{ij}$. Then based on the propagator proposed in [174], the weighted knowledge propagation graph within the source domain, denoted as H_s, can be calculated as:

$$H_s = (\mathrm{I} - \alpha L_s)^{-1}, \tag{2.3}$$

where $\alpha \in \mathbb{R}$ is a scaling factor which is fixed as 0.2 following [107], and I is the identity matrix.

Then for each sample $\mathbf{z}_s^i$ from the source minority set $\mathcal{D}_s^f$, an interpolated embedding $\tilde{\mathbf{z}}_s^i$ is constructed by the structure knowledge propagated from all its neighbors under the guidance of the weighted propagation graph:

$$\tilde{\mathbf{z}}_s^i = \sum\nolimits_{\mathbf{x}_s^j \in \mathcal{D}_s^f} H_s^{ij} \mathbf{z}_s^j, \tag{2.4}$$

in which $\tilde{\mathbf{z}}_s^i$ share the same label as $\mathbf{z}_s^i$, and the augmented interpolated embeddings for all source domain minority set samples make up the set $\tilde{\mathcal{D}}_s^f = \{\tilde{\mathbf{z}}_s^i | \mathbf{x}_s^i \in \mathcal{D}_s^f\}$ lying in the identical label space as $\mathcal{D}_s^f$. Since the constructed $\tilde{\mathbf{z}}_s^i$ involves the structure information from the whole source domain, so such knowledge propagation augmented samples can expand the corresponding category feature space and eliminate the undesired noise from outliers.

2.1.2.3.2 Cross-domain Knowledge Propagation Except labeled source domain data, the unlabeled target domain is also rich in the structure information corresponding to the source domain. However, due to the domain shift, which is one of the main challenges in domain adaptation tasks caused by the different data distribution across domains, we cannot directly put the source and target domain data together and apply the knowledge propagation globally. Because the knowledge propagation graph computed overall source and target domain data will be dominated by the relationship between samples from the same domain, while the structure knowledge between samples across domains is easy to be ignored compared to the close relationship within the source/target domain. Based on this, instead of directly combining the source and target domain together, we propose the cross-domain knowledge propagation from the target domain $\mathcal{D}_t$ to the few-shot source domain minority set $\mathcal{D}_s^f$.

Specifically, by putting the source domain minority set $\mathcal{D}_s^f$ and the target domain data $\mathcal{D}_t$ together making up a new dataset $\mathcal{D}_o = \mathcal{D}_s^f \cup \mathcal{D}_t$. The adjacency matrix $A_o \in \mathbb{R}^{(n_s^f+n_t)\times(n_s^f+n_t)}$ based on the dataset $\mathcal{D}_o$ is computed as:

$$A_o^{i,j} = \exp(-\frac{d_{ij(o)}^2}{\sigma_o^2}) = \exp(-\frac{\|\mathbf{z}_o^i - \mathbf{z}_o^j\|^2}{\sigma_o^2}), \tag{2.5}$$

where $d_{ij(o)}^2 = \|\mathbf{z}_o^i - \mathbf{z}_o^j\|^2$ is the distance between the samples $\mathbf{z}_o^i$ and $\mathbf{z}_o^j$, both from $\mathcal{D}_o$, the scaling factor $\sigma_o^2 = Var(d_{ij(o)}^2)$, and $A_o^{kk} = 0, \forall k$ [73]. It is noteworthy that due to the different distribution between the source and target domain, only considering the relationship between the given source domain minority-set sample and the target domain may mislead the structure knowledge in the adjacency matrix. So A_o keeps the structure knowledge about the corresponding relationships among the source domain samples from minority set in $\mathcal{D}_s^f \subset \mathcal{D}_o$. In other words, A_o contains both within-source and source-target structure information.

Similarly, the Laplacian matrix of A_o is computed as:

$$L_o = D_o^{-1/2} A_o D_o^{-1/2}, \tag{2.6}$$

where $D_o^{ii} = \sum_j A_o^{ij}$, and the weighted cross-domain knowledge propagation graph is calculated as:

$$H_o = (\mathrm{I} - \alpha L_o)^{-1}, \tag{2.7}$$

in which α is fixed as 0.2 following [107], same as Eq. 2.3.

Based on the cross-domain knowledge propagator H_o, for each sample $\mathbf{z}_s^i$ from the source domain minority set $\mathcal{D}_s^f$, a synthesized embedding $\tilde{\mathbf{x}}_o^i$ is constructed by the combination of its neighbors in the set $\mathcal{D}_o$ under the guidance of the weighted knowledge propagation graph H_o:

$$\tilde{\mathbf{z}}_o^i = \sum\nolimits_{\mathbf{x}_o^j \in \mathcal{O}} H_o^{ij} \mathbf{z}_o^j, \tag{2.8}$$

where $\tilde{\mathbf{z}}_o^i$ and $\mathbf{z}_s^i$ have the same label. The augmented embeddings through *Cross-domain Knowledge Propagation* raise the set $\tilde{\mathcal{D}}_o^f = \{\tilde{\mathbf{z}}_o^i | \mathbf{x}_s^i \in \mathcal{D}_s^f\}$ to augment $\mathcal{D}_s^f$.

2.1.2.3.3 Cross-domain Mix-up Augmentation Moreover, besides the extremely imbalanced distributed source domain data, the different data distribution across source and target domains is another crucial challenge in imbalanced domain adaptation problems. Existing image generation strategies designed for few-shot problems, such as F2GAN [39], train a generator with images from the seen categories mapping a few conditional images to synthetic samples belonging to the same category. Then the trained model translates the images from unseen categories to diverse images with random interpolation coefficients. Such an augmentation strategy does not take advantage of the discriminative knowledge in the annotated source domain data, and can not manage the cross-domain distribution difference. Recent image recognition works reveal that the features deep in networks are usually linearized, and various directions in the feature space correspond to some specific semantic translations [131]. Such intriguing observation motivates the thoughts that translating one sample along a specific feature direction results in new synthesized data with different semantics but still lying in the same class. Moreover, Xu *et al.* [147] notice that in DA tasks, only samples from the source and target domain alone are not sufficient to ensure domain-invariance at most parts of latent space, which inspires us to generate synthesized data involving cross-domain information to fulfill the gap between source and target domain, as well as guarantee domain-invariance in a more continuous latent space.

However, due to the extremely imbalanced distribution in source domain and lacking of data from some specific categories causes directly translating the information across domains is vulnerable to negative transfer, especially when the available few-shot categories samples cannot represent the specific class distribution characteristics because they may lie far from the class center in the feature space. Thus aiming to implement a moderate augmentation strategy without severely misleading under the imbalance domain adaptation situation, we seek to generate synthesized samples through the feature level mix-up involving the augmented embeddings refined with knowledge propagation within the source domain and across source and target domains, i.e., $\tilde{\mathcal{D}}_s^f$ and $\tilde{\mathcal{D}}_o^f$.

Specifically, for each source domain sample $\mathbf{x}_s^i \in \mathcal{D}_s^f$ drawn from the minority set categories, two refined augmentation embeddings $\tilde{\mathbf{z}}_s^i \in \tilde{\mathcal{D}}_s^f$ and $\tilde{\mathbf{z}}_o^i \in \tilde{\mathcal{D}}_o^f$ are synthesized through the within-source and cross-domain structure knowledge propagation, respectively. To explore the internal information across domains, these two synthesized embeddings are linearly interpolated to fulfill the feature space across domains and produce mix-up samples as:

$$\hat{\mathbf{z}}_m^i = (1-\gamma)\tilde{\mathbf{z}}_s^i + \gamma\tilde{\mathbf{z}}_o^i, \tag{2.9}$$

where $\gamma \sim \text{Beta}(a, b)$ is to control the interpolation between the two embeddings ($\gamma \in [0, 1], a, b > 0$). Because $\tilde{\mathbf{z}}_s^i$ and $\tilde{\mathbf{z}}_o^i$ have identical class labels, the mixup samples are also assigned the same class label. For each source domain minority set sample $\mathbf{x}_s^i$, k different mix-up samples with label $\mathbf{y}_s^i$, the same label as $\mathbf{x}_s^i$, are generated with different randomly selected factor γ. The synthesized samples created by the cross-domain mix-up augmentation constitute a new set denoted as $\hat{\mathcal{D}}_m^f = \{\hat{\mathbf{z}}_m^{i(k)} | \gamma^k \sim \text{Beta}(a, b)\}$, which is used to optimize the frameworks with corresponding class label.

It is noteworthy that our proposed *Cross-domain Mix-up Augmentation* is different from DM-ADA proposed in [147]. Due to the extremely imbalanced distribution in the source domain caused by the lack of labeled samples from the minority set categories, directly combining samples across domains could produce fake samples leaning towards the majority set feature space severely. Moreover, the risk that the given few-shot samples lying far from the class center in the corresponding feature space will mislead the augmentation and domain alignment process. With our proposed strategy, the interpolated samples are produced based on the refined embeddings obtained the within source and across source and target domain knowledge propagation under the guidance of the weighted graph, which can eliminate the undesired noise and distraction caused by outliers. Such a strategy can balance the contribution of the majority and minority set during the domain adaptation process.

2.1.2.3.4 Hybrid Distinct Classifiers In this study, we adopt the architecture introduced in Section 1.1, which comprises a dual classifiers structure consisting of a multi-layer neural network classifier $C_N(\cdot)$ and a prototype classifier $C_P(\cdot)$. The network classifier $C_N(\cdot)$ takes the output of the network $F(\cdot)$ as input and predicts the probabilities denoted by $\hat{\mathbf{y}}_N$, whereas the prototype classifier $C_P(\cdot)$ also takes the output of the network $F(\cdot)$ as input and predicts the probabilities denoted by $\hat{\mathbf{y}}_P$. However, the imbalance domain adaptation problems we face present extreme class-wise distribution imbalance, making it impractical to train a promising classifier solely based on the limited few-shot samples from the minority set. Despite leveraging knowledge propagation augmentation and cross-domain mix-up strategies proposed in our work, the synthesized samples still fail to match the efficiency of real labeled data from the majority set categories in the source domain. As a solution, our approach relies on the prototype classifier $C_P(\cdot)$ to recognize target samples based on their similarity to category prototypes (class centers) instead of relying on the vast training data used by $C_N(\cdot)$. This eliminates the drawbacks caused by the insufficiency of training data from the source domain's minority set. Conversely, sufficient source domain training data allows us to train a promising classifier for the majority set categories, enabling successful adaptation to the target domain.

Similar as the observation in Section 1.2, we only apply supervision optimization to update the parameters in generator $F(\cdot)$ and classifier $C_N(\cdot)$ by minimizing the cross-entropy loss, which is defined as:

$$\mathcal{M}_s = -\frac{1}{\tilde{n}_s} \sum\nolimits_{i=1}^{\tilde{n}_s} \sum\nolimits_{c=1}^{C} 1_{[c=y_s^i]} \log \hat{y}_{N,s}^{i,c}, \tag{2.10}$$

where $\tilde{n}_s = n_s + \tilde{n}_s^f + \tilde{n}_o^f + \hat{n}_m^f$, is the total number of samples including all real source domain data (n_s) as well as synthesized samples through the within source *Embedding Propagation*($\tilde{n}_s^f$), *Cross-domain Knowledge Propagation* ($\tilde{n}_o^f$), and *Cross-domain Mix-up Augmentation*($\hat{n}_m^f$).

2.1.2.4 Cross-domain Prototypes Alignment (CPA)

So far, we have sufficient well-labeled source domain majority-set samples, few-shot minority-set real samples, together with the synthesized samples to make up the minority-set categories. In order to simultaneously solve the domain distribution disparity problem and augment the minority set data, we adopt the discriminative cross-domain alignment learning objective intorduced in Section 1.1.2.2.3, and involving all real and synthesized samples in both domains and all categories.

Specifically, for class c in the source domain minority set label space $\mathcal{Y}^f$, the amended class prototype is calculated as:

$$\tilde{\mu}_s^c = \frac{\sum\limits_{\mathbf{z}_s^i \in \mathcal{D}_s^{f(c)}} \mathbf{f}_s^i + \sum\limits_{\tilde{\mathbf{z}}_s^i \in \tilde{\mathcal{D}}_s^{f(c)}} \tilde{\mathbf{f}}_s^i + \sum\limits_{\tilde{\mathbf{z}}_o^i \in \tilde{\mathcal{D}}_o^{f(c)}} \tilde{\mathbf{f}}_o^i + \sum\limits_{\hat{\mathbf{z}}_m^i \in \hat{\mathcal{D}}_m^{f(c)}} \hat{\mathbf{f}}_m^i}{n_s^{f(c)} + \tilde{n}_s^{f(c)} + \tilde{n}_o^{f(c)} + \hat{n}_m^{f(c)}}, \tag{2.11}$$

where $\mathbf{f}_s^i/\tilde{\mathbf{f}}_s^i/\tilde{\mathbf{f}}_o^i/\hat{\mathbf{f}}_m^i$ are the output of the network $F(\cdot)$ with $\mathbf{z}_s^i/\tilde{\mathbf{z}}_s^i/\tilde{\mathbf{z}}_o^i/\hat{\mathbf{z}}_m^i$ as input, $\mathcal{D}_s^{f(c)}/\tilde{\mathcal{D}}_s^{f(c)}/\tilde{\mathcal{D}}_o^{f(c)}/\hat{\mathcal{D}}_m^{f(c)}$ are the subset of samples belonging to class c drawn from $\mathcal{D}_s^f/\tilde{\mathcal{D}}_s^f/\tilde{\mathcal{D}}_o^f/\hat{\mathcal{D}}_m^f$, respectively, while $n_s^{f(c)}/\tilde{n}_s^{f(c)}/\tilde{n}_o^{f(c)}/\hat{n}_m^{f(c)}$ are the corresponding number of samples in each subset.

If all the source and target domain data are treated as a large batch during the augmentation process, global structure information will be propagated within and across domains. If so, $\tilde{n}_s^{f(c)} = \tilde{n}_o^{f(c)} = 1$ and $\hat{n}_m^{f(c)} = K$ in each training epoch. However, for large-scale benchmarks, it is inefficient to deal with all data together and calculate the global knowledge propagation graph. Thus to handle large-scale datasets and reduce the complexity, we can build *episodes* training strategy referring to few-shot learning tasks [73, 85, 121]. Each episode consists of e_s source domain instances and e_t instances from the target domain. For the source data in each episode, data from the majority set $\mathcal{Y}^m$ are randomly sampled without replacement (p examples per class), and data belonging to the minority set $\mathcal{Y}^f$ can be used multiple times in each training epoch due to the lack of data (q examples per class), i.e., $e_s = |\mathcal{Y}^m| * p + |\mathcal{Y}^f| * q$. The computing complexity will be negligible with the episodes training strategy since the size of each episode is small [73]. The source domain prototypes are updated adaptively during training.

Based on the revised source domain class prototypes, the class-wise MMD can be calculated as:

$$\mathcal{M}_c = \frac{1}{C} \sum\nolimits_{c=1}^{C} \|\tilde{\mu}_s^c - \mu_t^c\|_2^2, \tag{2.12}$$

where $\tilde{\mu}_s^c$ is the revised source domain class c prototype, calculated on $\tilde{\mathcal{D}}_s^c = \{\mathcal{D}_s^{f(c)}, \tilde{\mathcal{D}}_s^{f(c)}, \tilde{\mathcal{D}}_o^{f(c)} \hat{\mathcal{D}}_o^{f(c)}\}$ when class c belongs to the minority set categories. Due to the missing of the target domain samples labels, we accept the $C_P(\cdot)$ prediction $\hat{\mathbf{y}}_P^t$ as pseudo labels for target domain samples, and compute

Table 2.1: Comparisons of Recognition Rates (%) on Office-Home Dataset (5-shot).

Method	DAN			DM-ADA			MCD			SWD			SymNets			Ours		
Acc	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o
Ar→Cl	22.1	**57.1**	43.6	5.13	20.2	14.4	14.5	32.7	25.7	15.3	47.3	35.0	13.9	39.6	30.3	**47.7**	52.9	**50.9**
Ar→Pr	30.5	71.7	55.1	2.35	28.2	17.8	26.6	49.9	40.5	20.1	67.1	48.2	36.5	64.4	55.4	**71.3**	**74.0**	**72.9**
Ar→Rw	42.0	**77.5**	62.7	4.31	37.1	23.5	35.1	51.3	44.6	30.9	70.8	54.2	40.8	71.8	58.2	**72.0**	77.2	**75.0**
Cl→Ar	15.7	**60.6**	40.4	6.52	20.3	14.1	17.8	28.3	23.6	11.8	46.2	30.8	9.60	58.7	37.2	**57.3**	**60.6**	**59.1**
Cl→Pr	25.7	72.0	52.7	0.00	29.3	17.5	6.50	67.0	42.7	2.20	38.5	23.9	12.3	67.1	46.6	**65.1**	**74.8**	**70.9**
Cl→Rw	18.4	73.2	51.1	0.00	28.1	16.4	15.2	12.8	13.8	4.10	34.3	21.7	20.4	66.1	48.4	**69.3**	**74.7**	**72.5**
Pr→Ar	22.0	57.0	41.3	1.01	23.8	13.6	27.4	14.0	20.0	28.2	39.0	34.2	18.1	51.0	37.8	**60.9**	**58.8**	**59.8**
Pr→Cl	20.2	**52.3**	40.0	5.85	27.1	19.0	4.50	35.1	23.4	3.90	36.5	17.8	7.30	40.9	26.5	**46.0**	51.4	**49.3**
Pr→Rw	42.6	77.8	36.1	3.42	43.4	26.8	36.7	54.1	46.9	13.4	74.6	49.2	38.4	70.9	58.6	**72.6**	**80.5**	**77.2**
Rw→Ar	26.5	**68.2**	49.5	6.52	41.2	25.6	27.0	44.0	36.4	25.1	60.8	44.8	27.4	65.4	50.0	**63.0**	66.3	**64.9**
Rw→Cl	25.0	**60.9**	47.1	2.05	31.0	19.9	4.10	35.9	23.7	3.00	50.5	32.3	14.9	46.8	33.3	**49.9**	54.9	**53.0**
Rw→Pr	38.8	**84.9**	66.3	1.29	55.6	33.8	32.2	59.0	48.2	7.60	50.9	33.5	30.7	81.0	60.8	**76.8**	84.0	**81.1**
Avg.	30.9	64.3	51.1	3.20	32.1	20.2	20.6	40.3	32.4	13.8	50.5	35.5	22.5	60.3	45.3	**62.7**	**67.5**	**65.6**

the average of the features predicted belonging to the same category as the corresponding prototype.

Beyond that, we extend to explicitly consider the data distribution among different classes across domains, and maximize the inter-class divergence as:

$$\mathcal{M}_d = \frac{1}{C}\frac{1}{C-1}\sum\nolimits_{c=1}^{C}\sum\nolimits_{\substack{c'=1\\c'\neq c}}^{C}\|\tilde{\mu}_s^c - \mu_t^{c'}\|_2^2, \tag{2.13}$$

where the inter-class divergence $\mathcal{M}_d$ evaluates the distances of all different class prototype pairs across domains.

In contrast to the learning objective introduced in Section 1.1.2.2.3, where only real samples were considered, Eq.2.12 and Eq.2.13 incorporate both real and synthesized samples to calculate the prototypes/class-centers. Consequently, we refer to this approach as the Cross-Domain Prototype Alignment (CPA) to distinguish it from the loss function presented in Section 1.1.2.2.3. Overall, our discriminative Cross-Domain Prototype Alignment is proposed to minimize the cross-domain intra-class prototypes distances, while maximizing the inter-class distances.

2.1.2.5 Overall Objective and Optimization

To sum up, by exploring the source supervision over the real and augmented instances, as well as the discriminative cross-domain alignment, we have our overall objective function:

$$\min_{F,C_N} \mathcal{M}_s + \lambda(\mathcal{M}_c - \mathcal{M}_d), \tag{2.14}$$

Table 2.2: Comparisons of Recognition Rates (%) on Office-Home Dataset (1-shot).

Method	DAN			DM-ADA			MCD			SWD			SymNets			Ours		
Acc	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o
Ar→Cl	0.30	**56.7**	35.0	2.81	23.1	15.3	14.3	48.3	35.3	15.3	45.7	34.0	0.40	43.5	27.2	**29.4**	54.8	**45.1**
Ar→Pr	4.00	72.0	44.7	1.18	28.3	17.4	23.0	65.9	48.5	23.4	66.6	49.2	0.00	69.4	41.5	**46.2**	**76.3**	**64.2**
Ar→Rw	4.20	78.0	47.3	0.00	39.2	22.9	32.4	70.3	54.5	31.4	71.3	54.7	0.00	75.0	43.7	**53.9**	**78.7**	**68.4**
Cl→Ar	0.00	**60.6**	33.4	0.00	18.6	10.3	16.8	33.8	26.2	12.1	47.0	31.3	0.00	60.1	32.9	**41.2**	60.5	**51.8**
Cl→Pr	0.00	73.5	44.0	2.30	28.8	18.1	4.90	21.9	15.1	5.40	68.8	43.3	0.00	74.3	44.2	**43.8**	**76.0**	**63.0**
Cl→Rw	0.30	71.5	41.9	1.77	25.0	15.4	17.7	42.2	32.0	3.40	65.5	39.7	0.00	71.0	41.7	**50.7**	**76.0**	**65.5**
Pr→Ar	0.00	57.6	31.8	0.00	26.2	14.5	25.2	50.2	39.0	25.0	49.1	38.3	0.00	56.5	31.8	**47.6**	**60.1**	**54.5**
Pr→Cl	0.80	**53.2**	33.1	0.00	29.3	18.0	4.30	47.0	30.6	4.10	46.3	30.1	0.00	45.0	27.0	**34.8**	52.3	**45.6**
Pr→Rw	7.50	77.3	48.3	3.64	40.8	25.3	36.5	74.8	57.7	14.9	74.7	49.5	0.20	74.4	43.6	**63.8**	**80.5**	**73.6**
Rw→Ar	0.00	68.5	37.8	2.02	37.3	21.5	26.7	39.0	33.5	25.9	61.6	45.6	0.00	**71.5**	39.3	**56.9**	66.7	**62.3**
Rw→Cl	2.70	**61.5**	39.0	0.00	31.7	19.5	4.10	19.2	13.4	3.10	11.3	8.20	0.00	54.1	33.1	**40.2**	57.3	**50.7**
Rw→Pr	1.90	**84.6**	51.3	1.74	56.3	34.4	36.0	80.6	62.7	6.3	80.7	50.8	0.00	84.4	41.7	**59.8**	**84.6**	**74.6**
Avg.	1.80	67.9	40.6	1.29	32.0	19.4	20.2	49.2	37.4	14.2	57.3	39.6	0.10	64.9	37.3	**47.4**	**68.7**	**59.9**

where λ is a hyper-parameter to balance the contributions of different terms. We need to train the generator $F(\cdot)$ to map both source and target domain samples to a shared domain-invariant embedding feature space. It is noteworthy that the prototype classifier $C_P(\cdot)$ is free of parameters. Inspired by [113, 62], our training process consists of two steps:

Step A. We train the feature generator $F(\cdot)$ and neural networks classifier $C_N(\cdot)$ over the source supervision, including real and synthesized data. Keeping the performance on the source domain is crucial for obtaining discriminative whilst domain-invariant embedding features. Moreover, due to the lack of source domain minority-set samples, optimizing the whole model over the real samples as well as the synthesized samples will benefit the performance on the minority-set categories, avoiding the model overfitting to the majority-set classes. The optimization objective is listed as $\min_{F,C_N} \mathcal{M}_s$.

Step B. We freeze the parameters of classifier $C_N(\cdot)$ and update the generator $F(\cdot)$, which will map the source and target domain samples into a shared embedding feature space, where both source and target samples from the same class will be distributed close to each other, while far from samples belonging to other categories. The optimization objective is provided as $\min_F \mathcal{M}_s + \lambda(\mathcal{M}_c - \mathcal{M}_d)$.

2.1.3 Experiments

We evaluate our proposed model on two domain adaptation visual benchmarks, Office-31 and OfficeHome, and compare the evaluation performances with several state-of-the-art domain adaptation methods. Then we analyze the components of our framework in detail and explore the parameter sensitivity.

2.1.3.1 Datasets and Experimental Setting

Implementation Details: We implement our model based on PyTorch. ImageNet pre-trained ResNet-50 [35] without the last fully-connected layer is accepted as $E(\cdot)$ to obtain the embeddings $\mathbf{z}_{s/t}$. $F(\cdot)$ is a two-layer fully-connected neural network with a hidden layer output dimension of 1,024 and ReLU activation. The output of $F(\cdot)$ is the domain invariant features $\mathbf{f}_{s/t}$ with dimension 512. $C_N(\cdot)$ is a two-layer fully-connected neural network classifier with a hidden layer dimension of 512. Cosine similarity is accepted as the measurement function $\varphi(\cdot)$ in the prototype classifier $C_P(\cdot)$. We take the embedding features mean of the source domain samples belonging to each category as the initialized prototype μ_s^c which will be used by $C_P(\cdot)$ for classification, while for the Cross-Domain Prototype Alignment, we take the synthesized samples into account to update the prototype $\mu_s^c \rightarrow \tilde{\mu}_s^c$. All trainable parameters are optimized by Adam optimizer with a learning rate of 0.001 for both Office-31 and Office-Home datasets. $F(\cdot)$ and $C_N(\cdot)$ are initialized and pre-trained on the source domain with a learning rate of 0.0001 for 2,000 epochs, while $E(\cdot)$ is fixed. λ is fixed as 0.1 for Office-Home, while 0.01 for Office-31, $\gamma \sim \text{Beta}(2, 2)$, and k is fixed as 5, which will be discussed in the ablation study section. To optimize hyper-parameter values, we follow the approach proposed in [74]. We construct a validation set with labeled source and unlabeled target data and train a binary domain classifier to distinguish between them. By jointly assessing source classifier error and domain classifier error, we determine the parameters for each task. The assumption is that in the adapted domain-invariant feature space, high source accuracy implies high target accuracy, and the domain classifier should exhibit high error when recognizing domain labels. We also explore additional validation techniques for hyper-parameter optimization and checkpoint selection, but these are beyond the scope of this work [171, 112, 106].

We focus on two challenging extremely-imbalanced domain adaptation tasks as 1-shot and 5-shot in our experiments. Specifically, for each domain transfer task, we randomly select 1 or 5 samples of each source domain minority-set category, together with all the rest labeled majority-set source domain data as well as the unlabeled target domain samples for training. The first 10 and the first 25 alphabetical classes are treated as the minority set in the Office-31 and Office-Home datasets, respectively, the rest classes constitute the majority set. We randomly run the experiments 3 times

Table 2.3: Comparisons of Recognition Rates (%) on Office-31 Dataset (5-shot).

Method	DAN			DM-ADA			MCD			SWD			SymNets			Ours		
Acc	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o
A→W	41.2	77.1	67.1	9.65	51.9	38.8	56.6	6.30	21.9	44.3	7.30	18.7	17.0	**92.1**	69.9	**97.3**	90.3	**92.4**
D→W	99.6	96.3	97.2	3.70	73.9	52.2	63.4	57.0	61.1	54.9	22.5	32.5	76.2	**98.6**	92.1	**99.7**	**98.6**	**99.0**
W→D	92.9	99.7	97.6	0.00	85.7	60.3	31.2	37.5	35.6	51.3	79.1	70.9	79.9	**100**	93.8	**100**	99.3	**99.5**
A→D	47.4	85.5	73.7	6.73	49.6	36.9	56.5	14.5	26.9	58.4	72.4	68.3	13.6	88.7	65.5	**97.4**	**89.8**	**92.2**
D→A	64.6	62.1	62.8	0.52	35.9	25.6	63.6	43.0	49.0	53.5	64.3	61.1	53.8	71.4	65.9	**81.3**	**73.6**	**75.9**
W→A	53.3	63.7	60.7	3.68	43.1	31.5	68.2	24.9	37.6	58.5	42.7	47.3	49.8	69.7	63.1	**82.2**	**72.2**	**75.1**
Avg.	66.5	80.7	76.5	4.05	56.7	40.9	56.6	31.0	38.7	53.5	48.1	49.8	48.4	86.8	75.1	**93.0**	**87.3**	**89.0**

Table 2.4: Comparisons of Recognition Rates (%) on Office-31 Dataset (1-shot).

Method	DAN			DM-ADA			MCD			SWD			SymNets			Ours		
Acc	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o
A→W	0.00	76.3	53.7	4.11	50.3	36.0	55.7	4.30	20.2	41.3	72.3	62.7	0.00	**91.1**	64.2	**85.7**	88.2	**87.5**
D→W	48.9	96.2	82.2	0.00	71.8	49.6	63.0	24.5	36.4	45.1	96.8	80.8	0.90	98.4	69.7	**97.0**	**98.8**	**98.2**
W→D	37.7	**99.7**	80.5	0.00	76.3	53.7	35.7	99.4	80.6	38.3	**99.7**	81.6	0.00	**99.7**	68.8	**97.2**	99.5	**98.8**
A→D	0.00	83.7	57.8	0.00	49.3	34.7	57.1	14.5	27.1	52.6	75.9	69.0	0.00	**93.6**	64.7	**85.1**	89.2	**88.0**
D→A	26.8	63.4	52.7	0.00	40.8	28.9	67.4	18.0	32.4	53.0	46.3	48.3	0.00	68.3	47.9	**79.7**	**74.1**	**75.8**
W→A	12.7	64.5	49.4	0.00	38.4	27.1	65.4	32.4	42.1	56.8	61.9	60.4	0.00	67.5	47.5	**80.4**	**72.1**	**74.5**
Avg.	21.0	80.7	62.7	0.68	54.5	38.3	57.4	32.2	39.8	47.9	74.5	67.1	0.15	86.4	60.5	**87.5**	**87.0**	**87.1**

and report the average results of the 30th epoch. For each case, we report the results of $C_N(\cdot)$ on the target domain majority-set categories, while for the minority-set categories, we show the results of $C_P(\cdot)$, and the overall average performance is also based on these two results, as illustrated in the *Hybrid Distinct Classifiers*. We will discuss the different specialties of $C_N(\cdot)$ and $C_P(\cdot)$ in the ablation study section. All baselines are implemented with the official codes with hyper-parameters tuning as instructed by the original papers. For all experiments results, we mark the best performance on the minority set as **blue**, the best performance on the majority set as **red**, and the best overall performance as **bold**.

2.1.3.2 Results and Comparisons

The classification results on Office-Home and Office-31 under 5-shot and 1-shot settings are reported in Tables 2.1, 2.2, 2.3, and 2.4, respectively. A_f means minority-set accuracy, A_m represents majority-set accuracy, and A_o denotes the overall accuracy on the whole target domain.

From the results, it is obvious that our method significantly outperforms all the comparisons on both two benchmarks under 4 challenging settings in terms of overall accuracy. Especially for the performance of the minority set, our model achieves promising results while keeping reliable

Table 2.5: Comparisons of Recognition Rates (%) on Office-31 Dataset (1-shot).

Shot	1 - shot									5 - shot								
Method	Source-only			WDAN			Ours			Source-only			WDAN			Ours		
Acc	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o
A→W	68.51	78.04	75.09	13.19	81.07	61.01	**85.7**	**88.2**	**87.5**	73.19	77.68	76.29	40.00	80.71	68.68	**97.3**	**90.3**	**92.4**
D→W	91.49	97.14	95.39	52.34	97.32	84.03	**97.0**	**98.8**	**98.2**	93.19	97.32	96.04	87.23	98.04	94.84	**99.7**	**98.6**	**99.0**
W→D	95.45	99.71	98.45	46.10	100.0	83.33	**97.2**	99.5	**98.8**	96.75	99.71	98.84	89.61	99.42	96.39	**100.0**	**99.3**	**99.5**
A→D	77.92	79.65	79.14	18.83	76.74	58.84	**85.1**	**89.2**	**88.0**	79.22	76.54	77.33	51.30	82.85	73.09	**97.4**	**89.8**	**92.2**
D→A	65.66	66.63	66.35	27.31	71.15	58.32	**79.7**	**74.1**	**75.8**	67.60	66.73	66.98	60.92	70.65	67.80	**81.3**	**73.6**	**75.9**
W→A	66.50	63.92	64.67	11.29	66.68	50.51	**80.4**	**72.1**	**74.5**	69.66	62.82	64.82	62.50	65.73	64.79	**82.2**	**72.2**	**75.1**
Avg.	77.59	80.85	79.85	28.18	82.16	66.01	**87.5**	**87.0**	**87.1**	79.94	80.13	80.05	65.26	82.90	77.60	**93.0**	**87.3**	**89.0**

Table 2.6: Comparisons of Recognition Rates (%) on Office-Home Dataset (1-shot).

Shot	1 - shot									5 - shot								
Method	Source-only			WDAN			Ours			Source-only			WDAN			Ours		
Acc	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o	A_f	A_m	A_o
Ar→Cl	26.39	43.79	37.11	0.00	49.52	30.52	**29.4**	**54.8**	**45.1**	30.09	47.40	40.76	13.49	47.96	34.73	**47.7**	**52.9**	**50.9**
Ar→Pr	41.29	62.36	53.89	0.06	70.80	42.35	**46.2**	**76.3**	**64.2**	41.79	68.09	57.51	19.27	70.80	50.08	**71.3**	**74.0**	**72.9**
Ar→Rw	49.25	72.82	63.02	0.00	75.96	44.39	**53.9**	**78.7**	**68.4**	41.79	71.21	58.98	25.40	76.04	54.99	**72.0**	**77.2**	**75.0**
Cl→Ar	32.32	51.57	42.93	0.00	56.13	30.94	**41.2**	**60.5**	**51.8**	39.03	50.90	45.57	7.35	55.53	33.91	**57.3**	**60.6**	**59.1**
Cl→Pr	34.90	68.69	55.10	0.00	73.47	43.93	**43.8**	**76.0**	**63.0**	40.11	69.37	57.60	0.34	73.32	43.97	**65.1**	**74.8**	**70.9**
Cl→Rw	40.03	67.36	56.00	0.00	72.03	42.09	**50.7**	**76.0**	**65.5**	47.05	67.75	59.15	4.64	72.27	44.16	**69.3**	**74.7**	**72.5**
Pr→Ar	44.72	53.89	49.78	0.00	55.31	30.49	**47.6**	**60.1**	**54.5**	49.49	52.39	51.09	0.18	56.58	31.27	**60.9**	**58.8**	**59.8**
Pr→Cl	27.94	48.51	40.62	0.00	50.07	30.86	**34.8**	**52.3**	**45.6**	31.58	47.51	41.40	0.00	49.48	30.49	**46.0**	**51.4**	**49.3**
Pr→Rw	60.57	74.86	68.92	0.00	77.61	45.35	**63.8**	**80.5**	**73.6**	62.62	75.18	69.96	0.00	76.75	44.85	**72.6**	**80.5**	**77.5**
Rw→Ar	44.08	64.65	55.42	0.00	65.47	36.09	**56.9**	**66.7**	**62.3**	50.60	65.40	58.76	5.05	65.70	38.48	**63.0**	**66.3**	**64.9**
Rw→Cl	30.27	51.34	43.25	0.00	53.57	33.01	**40.2**	**57.3**	**50.7**	33.55	52.01	44.93	2.93	52.04	33.20	**49.9**	**54.9**	**53.0**
Rw→Pr	55.74	82.10	71.50	0.00	83.65	50.01	**59.8**	**84.6**	**74.6**	62.35	82.18	74.21	2.30	84.14	51.20	**76.8**	**84.0**	**81.1**
Avg.	40.63	61.83	51.13	0.01	65.30	38.34	**47.4**	**68.7**	**59.9**	44.17	62.45	54.99	6.75	65.05	40.94	**62.7**	**67.5**	**65.6**

performance on the majority set, which emphasizes the robustness of our model to manage the extremely imbalanced distribution challenges. On the contrary, conventional UDA solutions, e.g., DM-ADA [147] and SymNets [163], suffer from the source distribution imbalance problem, and fail to perform well in the minority-set categories, due to no consideration of imbalance distribution. For the Office-31 dataset 5-shot tasks, our model achieves 93.0% average accuracy on the minority-set, which beats DAN over 26% and maintains 87.3% on the majority-set, which is higher than SymNets. Furthermore, our model gets promising overall performance on the Office-31 dataset as 89.0%, which is even comparable to the state-of-the-art performance 88.4% achieved by SymNets, which include more source domain minority set labeled samples for training [163]. Moreover, in another extremely challenging case when only 1 sample is available for each category in the minority set, our model still gets reasonable results on the minority set as well as stable performance on the majority set. It highly affirms the effectiveness and robustness of our method in dealing with domain adaptation problems in which the source domain data is extremely imbalanced and insufficient for training. From the results reported in Tables 2.1 - 2.4, we observe that the classification accuracy of the proposed methods is much higher than the other compared baselines in most cases. This justifies the efficacy of Cross-Domain Augmentation and Cross-Domain Prototype Alignment in dealing with imbalanced domain adaptation challenges.

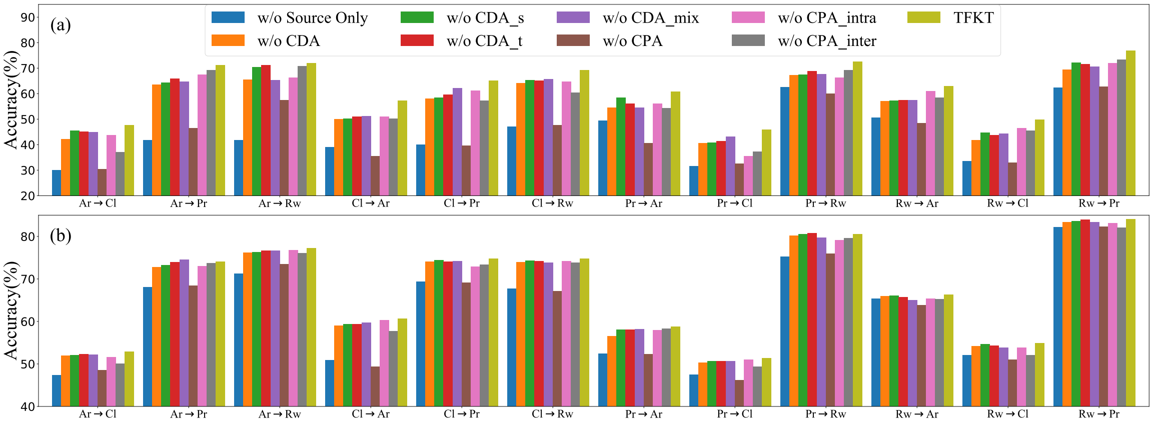


Figure 2.4: Contribution of the Cross-Domain Prototype Alignment (CPA) and the Cross-domain Augmentation (CDA) strategy on Office-Home 25-way 5-shot tasks (a) $C_P(\cdot)$ performance on Minority-set and (b) $C_N(\cdot)$ performance on Majority-set.

2.1.3.3 Comparison with Imbalanced DA Solution

To demonstrate the effectiveness of the proposed TFKT, we show more results of source-only hybrid classifiers and Weighted Maximum Mean Discrepancy (WDAN) [151]. Source-only hybrid classifiers consider $C_N(\cdot)$ on the target domain majority-set categories, while the results of $C_P(\cdot)$ on the minority-set. WDAN manages the domain adaptation with data distribution imbalance issues through reweighing the importance of each source sample during the domain alignment process. We re-implement WDAN with ResNet-50 [35] as the backbone, as the ResNets are the preferred base networks contemporaneously *.

From the results in Tables 2.5 & 2.6, we observe that our TFKT beats all compared baselines in most cases and achieves the best average results. WDAN obtains good performance in the original imbalance situation claimed in [151], but it cannot handle the extreme situations when only 1 or 5 samples per class are available for training. In addition, we notice that the Source-only results also

*The original WDAN is implemented with LeNet [61], AlexNet [56], GoogleLeNet [123], and VGG16 [120] as the backbone.

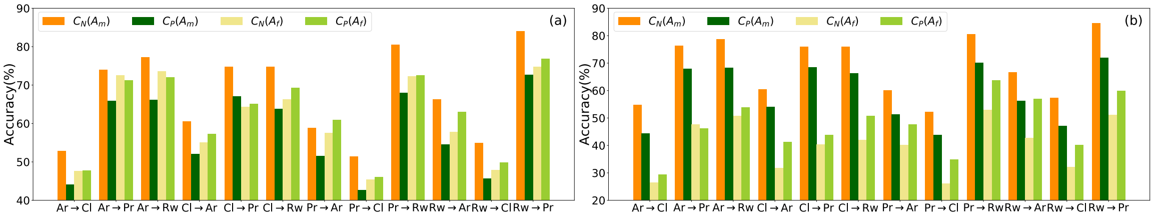
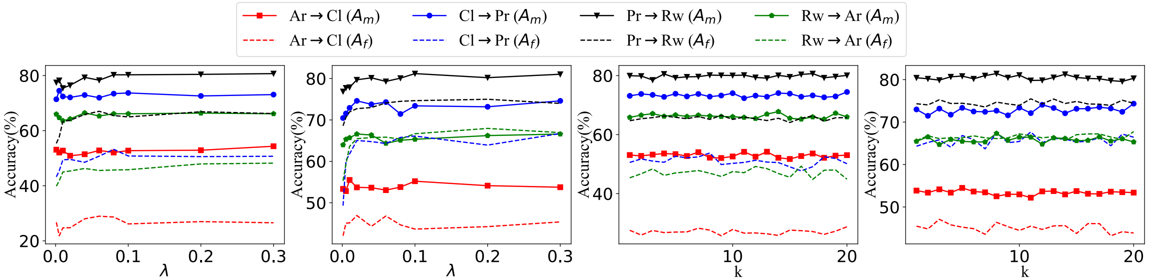


Figure 2.5: $C_N(\cdot)$ and $C_P(\cdot)$ performance comparison on Office-Home majority- and minority-set (a) 5-shot, (b) 1-shot.



(a) 1-shot (b) 5-shot (c) 1-shot (c) 5-shot

Figure 2.6: Parameters sensitivity analysis of λ and k on the Office-Home selected tasks (a)(c) 1-shot, (b)(d) 5-shot.

surpass some conventional domain adaptation solutions, especially in the minority set, emphasizing the contribution and benefits of involving the prototypical classifier $C_P(\cdot)$. The proposed *Hybrid Distinct Classifiers* framework can significantly counteract the negative effect caused by the training data insufficiency.

2.1.3.4 Ablation Analysis

First of all, we evaluate the contribution of the Cross-domain Prototypes Alignment (CPA) and the Cross-domain Augmentation (CDA) to our model by removing one of them and keeping all other architectures and training strategies. In Fig.2.4, we remove CPA, or the CDA strategy, or both of them and show the results on Office-Home as "w/o CPA", "w/o CDA", and "Source Only", respectively. Besides, the source domain minority set data is augmented by CDA to three kinds of synthetic data $\tilde{\mathcal{D}}_s^f$, $\tilde{\mathcal{D}}_o^f$, and $\tilde{\mathcal{D}}_m^f$, through *Data Augmentation through Embedding Propagation*, *Cross-domain Knowledge Propagation*, and *Cross-domain Mix-up Augmentation*, respectively. By removing each one kind of synthetic data while keeping others, the results are reported as "w/o CDA_s", "w/o CDA_t", "w/o CDA_mix", respectively. Moreover, CPA consists of two loss terms, minimizing class-wise MMD ($\mathcal{M}_c$) and maximizing inter-class divergence ($\mathcal{M}_d$). We remove each one term and the results are denoted as "w/o CPA_intra" and "w/o CPA_inter", respectively. Fig.2.4 (a)

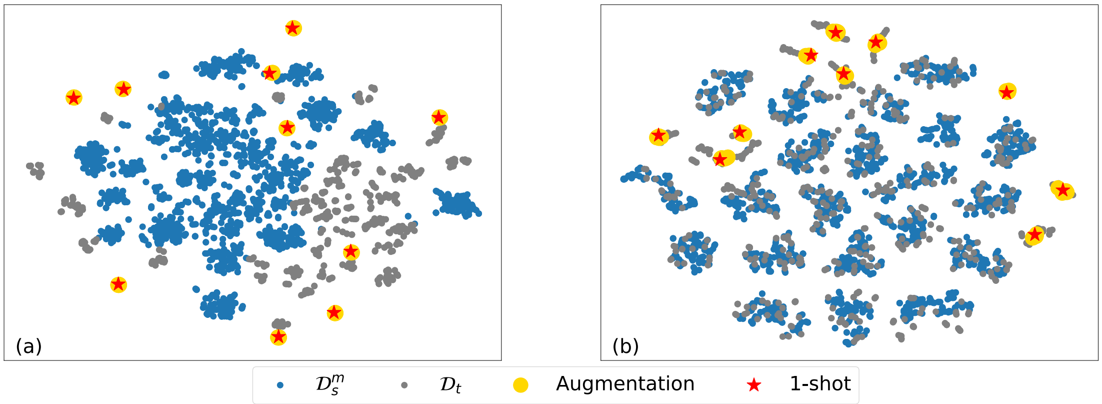


Figure 2.7: Visualization of embedding features of the Office-31 dataset A→W 10-way 1-shot task, including real and augmented fake samples. (a) Embeddings output from $E(\cdot)$ (b) Features output from $F(\cdot)$

claims the $C_P(\cdot)$ performance on the target domain minority-set classes, and (b) shows the $C_N(\cdot)$ performance on the majority-set. From the results, we observe that both CPA and CDA strategies benefit the fair cross-domain learning tasks, especially on the recognition performance on the minority set. It is reasonable that $C_N(\cdot)$ performance on the majority-set is not promoted significantly by CDA, because CDA only augments the minority-set categories. But the CPA strategy boosts the $C_P(\cdot)$ performance on the minority-set categories impressively.

Secondly, we compare the different classification specialties of $C_N(\cdot)$ and $C_P(\cdot)$ on different subsets. In Fig.2.5, we show $C_N(\cdot)$ and $C_P(\cdot)$ recognition rate of the target domain majority-set and minority-set categories on the Office-Home 25-way 5-shot tasks. The main difference between the roles of neural network classifier $C_N(\cdot)$ and prototypical classifier $C_P(\cdot)$ is their classification ability on the minority-set categories. The insufficiency of training data from the majority-set classes makes the trained neural network classifier dominated by the majority set and fail on the minority set. On the contrary, the prototype classifier is based on the estimated prototype from given samples per category, which is decided by the quality of available data instead of the number of samples available. From the results, we notice that for categories with sufficient well-labeled source samples for training in the majority-set, $C_N(\cdot)$ always obtains better performance than the prototype classifier $C_P(\cdot)$, e.g., Pr→Cl. However, for those classes lacking training samples in the minority set, $C_P(\cdot)$ can handle it much better and achieve promising performance in most cases, e.g., Pr→Cl and Rw→Cl. The generated samples contribute to refining the prototypes during training and benefit the classification performance of $C_P(\cdot)$. From the results, we notice that the improvement of $C_P(\cdot)$ compared to $C_N(\cdot)$

for the minority set is more significant in the 1-shot setting than the 5-shot. So the fewer source domain minority-set data available for training, choosing $C_P(\cdot)$ to recognize the minority set is more reasonable and superior.

Thirdly, we analyze the parameter sensitivity of our model. Four hard-to-transfer tasks of the Office-Home dataset are used for evaluation. The results are reported in Fig. 2.6. We can see that transfer performance is not sensitive to the variance of hyper-parameter λ from 0.1 to 0.3, in both 5-shot and 1-shot settings, which demonstrate the importance of the *Cross-domain Prototype Alignment*. Moreover, we change the number k of generated fake samples in each class from 1 to 20. Fig. 2.6(c) and (d) show that the results are not sensitive to the number of fake samples generated by the *Cross-domain Mix-up Augmentation* after $k = 5$.

Finally, we evaluate the quality of the generated synthesized samples belonging to the minority-set categories by drawing the t-SNE embeddings of all the real source and target samples, together with the generated fake samples. The results are visualized in Fig. 2.7. The red star points are the 1-shot set samples and the blue dots are the source domain majority set samples, gray dots are the target domain data. The augmented fake data are represented as yellow dots. (a) shows the output embeddings of network $E(\cdot)$, and (b) shows the output features of $F(\cdot)$. It is obvious that the generated samples are very similar to the available source domain minority-set samples, and the comparison between the chaos in (a) and organized data distribution in (b) demonstrate the effectiveness of the proposed cross-domain augmentation and prototypes alignment strategies.

2.1.4 Discussion and Limitation

In this work, we introduced the Towards Fair Knowledge Transfer (TFKT) model to tackle fairness challenges in highly imbalanced cross-domain learning scenarios. The proposed model employs cross-domain feature augmentation, knowledge propagation, and prototype alignment to improve classification performance on minority-set categories during domain adaptation, demonstrating significant improvements over existing approaches in various experiments. One of the primary limitations of the proposed solution is its applicability to large-scale datasets, which poses challenges during implementation. A potential approach to address this limitation involves sampling the source data in the training process to ensure that each batch contains all categories. Additionally, the framework may encounter difficulties in handling tasks with significant domain shifts, leading to ineffective knowledge propagation across domains and generating irrelevant noisy samples, thereby hindering optimization.

2.2 Marginalized Augmented Few-shot Domain Adaptation

Domain adaptation has recently drawn a lot of attention as it facilitates unlabeled target learning by borrowing knowledge from an external source domain. Most existing domain adaptation solutions seek to align feature representations between the labeled source and unlabeled target data. However, the scarcity of target data easily results in negative transfer as it misleads the cross-domain adaptation to the dominance of the source. To address the challenging few-shot domain adaptation (FSDA) problem, in this work, we propose a novel Marginalized Augmented Few-shot domain adaptation (**MAF**) approach to address the cross-domain distribution disparity and insufficiency of target data simultaneously. On one hand, *Cross-domain Continuity Augmentation* synthesizes abundant intermediate patterns across domains leading to a continuous domain-invariant latent space. On the other hand, sufficient *Source-supervised Semantic Augmentation* is explored to progressively diversify the conditional distribution within and across domains. Moreover, the proposed augmentation strategies are implemented efficiently via an expected transferable cross-entropy loss over the augmented distribution instead of explicit data synthesis, and minimizing the upper bound of the expected loss introduces negligible extra computing cost. Experimentally, our method outperforms the state-of-the-art in various few-shot domain adaptation benchmarks, which demonstrates the effectiveness and contribution of our work. Our source code is provided at https://github.com/scottjingtt/MAF.git.

2.2.1 Summary of Contribution

In this work, we propose a Marginalized Augmented Few-shot domain adaptation (**MAF**) model to alleviate the aforementioned issues, domain shift, and lack of target data, simultaneously. Specifically, MAF is inspired by the fact that there exist many different semantic transformation directions in the deep feature space, and translating the deep feature of one sample along a specific direction can be represented as meaningful semantic altering in the original input space [140]. Thus, the limited target samples can be augmented with the guidance of the source domain conditional semantics, estimating the target domain distribution and enhancing the generalizability of the target classifier. Moreover, to alleviate the domain distribution disparity, we first estimate the features mean of each class in the source/target domain as a domain-specific class-wise prototype equipped with the integration of various semantics of each corresponding category. Then, the source domain intra-class feature covariances are transferred as semantic variations to the target domain progressively along with the process of cross-domain continuity augmentation, diversifying the within and across domain features distribution. Our contributions are summarized as:

- Firstly, we propose a novel Marginalized Augmented Few-shot domain adaptation (MAF) approach for the few-shot domain adaptation problem. Specifically, the source domain class-wise semantics are progressively transferred to the target domain, diversifying the target distribution and enhancing the adaptation of the target classifier.
- Secondly, we derive the upper bound of the expected cross-entropy loss over the augmented distribution. Through minimizing the upper bound of the expected loss, MAF is performed efficiently as a lightweight module easily plugged into most existing domain adaptation models without noticeable extra computational cost.
- Finally, we demonstrate the effectiveness of the proposed model on various few-shot cross-domain visual benchmarks. Our model outperforms the state-of-the-art by 2-3% in most 1-/3-shot domain adaptation tasks on average.

2.2.2 The Proposed Method

2.2.2.1 Preliminaries and Motivation

In few-shot domain adaptation (FSDA), The source domain $\mathcal{D}_s = \{\mathbf{X}_s, \mathbf{Y}_s\} = \{(\mathbf{x}_s^i, y_s^i)\}_{i=1}^{n_s}$ contains n_s labeled samples, and the target domain $\mathcal{D}_t = \{\mathbf{X}_t, \mathbf{Y}_t\} = \{(\mathbf{x}_t^j, y_t^j)\}_{j=1}^{n_t}$ consists of a limited number of annotated data, i.e., $n_t \ll n_s$. The source and target data are drawn from different distributions but identical label spaces with C categories. In test stage, the framework is evaluated on additional unlabeled target domain data.

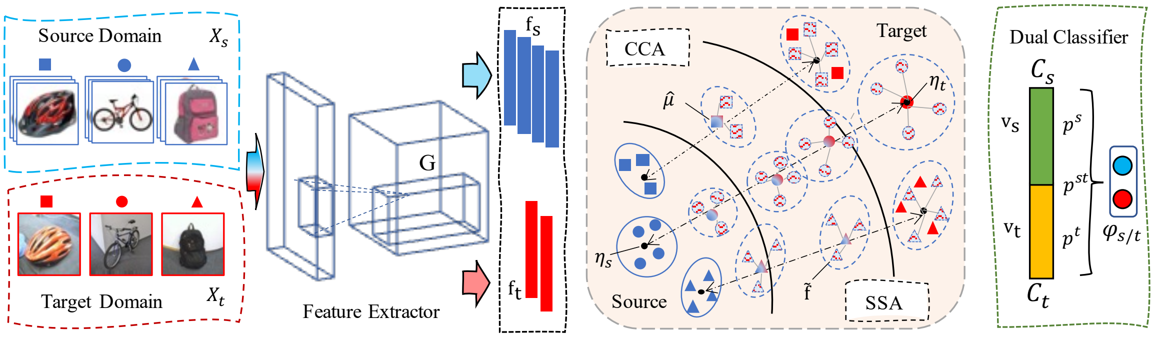


Figure 2.8: Illustration of MAF framework, where $G(\cdot)$ is a shared convolutional feature extractor, and $C_{s/t}(\cdot)$ is the source/target classifier.

The scarcity of tremendous target data for training fails conventional domain adaptation methods due to overfitting to the sufficient source domain. To address the challenges of domain shift and lack of target data, we propose a Marginalized Augmented Few-shot domain adaptation (MAF) approach to progressively transfer the source domain semantic knowledge to the target domain. Specifically, we first estimate the features mean of each class in the source/target domain as domain-specific class-wise prototypes, integrating overall semantic characteristics of the corresponding category in the specific domain. Then, *Cross-domain Continuity Augmentation* (CCA) is developed to enrich the inter-domain feature patterns with random linear interpolations between prototypes from the same class across domains. Moreover, we propose *Source-supervised Semantic Augmentation* (SSA) to progressively transfer the semantic knowledge of the source domain to the inter-domain feature space around the synthesized intermediate prototypical patterns, and enrich the target domain distribution. Finally, instead of explicitly generating a massive number of fake data, we derive the upper bound of the expected cross-entropy loss on the augmented distribution. By minimizing the upper bound of the expected loss, the proposed MAF can be implemented with negligible extra computational cost involved, and applicable to be plugged into most existing domain adaptation models.

Different from prior few-shot learning and conventional domain adaptation methods, our proposed MAF can simultaneously enrich the target domain distribution and eliminate the cross-domain distribution disparity. Through progressively transferring the source domain distribution knowledge to the target domain via synthesized intermediate sub-domains, the FSDA problem can be addressed. Moreover, instead of explicitly generating an infinite number of fake samples, optimizing the framework by minimizing the upper bound of the expected cross-entropy loss of the synthetic data saves tremendous computing resources needed.

2.2.2.2 Framework Overview

The overall framework of our proposed model is shown in Figure 2.8. Similar as the architecture introduced in the prior works, we accept a dual-classifier adversarial network for domain adaptation, which consists of a shared deep convolutional feature extractor $G(\cdot)$, and two classifiers $C_s(\cdot)$ and $C_t(\cdot)$. The difference is that $C_s(\cdot)$ and $C_t(\cdot)$ are two same architecture neural networks classifiers. For the purpose of mathematical simplicity, we use $\mathbf{f}_{s/t} = G(\mathbf{x}_{s/t})$, $\mathbf{f}_{s/t} \in \mathbb{R}^d$ to represent the output of $G(\cdot)$ in this section. $\eta^c_{s/t} \in \mathbb{R}^d$ denotes the feature prototype of class c in the source/target domain and $\Sigma^c_s \in \mathbb{R}^{d\times d}$ is the class-specific conditional covariance matrix computed from the source domain class c features. In addition, classifiers $C_s(\cdot)$ and $C_t(\cdot)$ are trained with both labeled source and target domain data. For simplicity, one feature $\mathbf{f} = G(\mathbf{x})$ input to $C_{s/t}(\cdot)$, where $\mathbf{x} \in \mathcal{D}_s \cup \mathcal{D}_t$ is the input image from source/target domain, the output logit vector of the classifier $C_{s/t}(\cdot)$ before the softmax operation is denoted as $\mathbf{v}_{s/t} \in \mathbb{R}^C$, and $\mathbf{p}_{s/t} \in [0,1]^C$ denotes the probability prediction after softmax function, i.e., $\mathbf{p}_{s/t} = C_{s/t}(\mathbf{f}) = \mathbf{softmax}(\mathbf{v}_{s/t})$, and $p^i_{s/t}, i \in \{1, \cdots, C\}$ is the i^{th} element of the probability prediction vector $\mathbf{p}_{s/t}$.

Firstly, we apply supervised optimization on the labeled source and target data for both $C_s(\cdot)$ and $C_t(\cdot)$ by minimizing the cross-entropy loss defined as:

$$\mathcal{L}^C_c = \mathbf{E}_{(\mathbf{x},y)\sim\mathcal{D}_s\cup\mathcal{D}_t}[-\log(p^y_s) - \log(p^y_t)], \tag{2.15}$$

where $p^y_{s/t}$ is the y^{th} probability prediction produced by $C_{s/t}(\cdot)$ for sample $\mathbf{x} \in \mathcal{D}_s \cup \mathcal{D}_t$, y is the ground-truth label.

Furthermore, to eliminate the domain shift across domains, we borrow the idea of dual adversarial classifiers and apply the domain confusion loss to train the model in an adversarial manner, where the outputs from $C_s(\cdot)$ and $C_t(\cdot)$ are used to discriminate domain class, without any additional domain discriminator network [164]. Specifically, the concatenated logits vector $\mathbf{v}_{st} = [\mathbf{v}_s; \mathbf{v}_t] \in \mathbb{R}^{2C}$ is input to a *Softmax* layer to obtain the normalized probability output $\mathbf{p}_{st} \in [0,1]^{2C}$. Then we calculate $\varphi_s = \sum_{i=1}^C p^i_{st}$ and $\varphi_t = \sum_{i=1}^C p^{i+C}_{st}$ as the probabilities of classifying an input sample $\mathbf{x}$

belonging to the source and target domains, respectively. The domain confusion loss is defined as:

$$\mathcal{L}_d^C = \mathbf{E}_{(\mathbf{x},y)\sim\mathcal{D}_s}[-\log(\varphi_s)] + \mathbf{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}[-\log(\varphi_t)], \tag{2.16}$$

through which the *discriminator* constructed by $C_s(\cdot)$ and $C_t(\cdot)$ is optimized to recognize the domain class of the input samples.

On the contrary, in order to adversarially adapt the source and target domains, the generator $G(\cdot)$ is optimized to fool the *discriminator* and map all data into a domain-invariant latent space, by minimizing:

$$\mathcal{L}_d^G = \mathbf{E}_{(\mathbf{x},y)\sim\mathcal{D}_s\cup\mathcal{D}_t}[-\log(\varphi_s) - \log(\varphi_t)]. \tag{2.17}$$

Moreover, we also leverage the normalized dual-classifier prediction $\mathbf{p}^{st}$ to train the framework with category-level confusion loss defined as:

$$\mathcal{L}_c^G = \mathbf{E}_{(\mathbf{x},y)\sim\mathcal{D}_s\cup\mathcal{D}_t}[-\log(p_{st}^y) - \log(p_{st}^{y+C})], \tag{2.18}$$

which can align the source and target domains with task-specific decision boundaries [113, 164]. It is noteworthy that classifiers $C_s(\cdot)$ and $C_t(\cdot)$ are optimized to identify the domain label (e.g., source or target) via minimizing Eq. (2.16), while the feature generator $G(\cdot)$ is optimized to fool the "domain discriminator" constructed by $C_s(\cdot)$ and $C_t(\cdot)$ through minimizing Eq. (2.17) and Eq. (2.18). Thus, $C_s(\cdot)$ is trained with the source domain supervision while $C_t(\cdot)$ is adapted to the target domain thanks to the category-level adversarial domain adaptation training process.

So far, we build our cross-domain framework with dual adversarial classifiers. However, due to the scarcity of sufficient unlabeled target data for training, the classifiers do not generalize well to novel target samples for inference due to overfitting to the source domain. Next, we propose two augmentation strategies to progressively transfer the source-supervised semantic knowledge to the target domain, which will expand the target distribution and enhance the generalizability of the framework.

2.2.2.2.1 Cross-domain Continuity Augmentation (CCA) Recent domain adaptation explored the cross-domain interaction and intermediate feature patterns across domains [148, 141]. However, randomly combining the source and target samples may result in confusing synthetic data and distract the task-specific boundaries. Thus, we propose the *Cross-domain Continuity Augmentation* (CCA) with category-wise guidance. Specifically, we first estimate the features mean of each class for the source/target domain as the prototype integrating overall class-specific semantic knowledge. Then, random linear interpolations between the same class source and target domain prototypes are exploited to generate abundant intermediate patterns between the two separate domains:

$$\hat{\mu}^c = (1 - \lambda\beta)\eta_s^c + \lambda\beta\eta_t^c, \tag{2.19}$$

where $\eta_{s/t}^c$ is the c-class conditional prototype of the source/target domain in the latent feature space, $\lambda \sim \text{Beta}(a, b)$ with $a, b > 0$ is a random positive coefficient drawn from Beta distribution, and $\beta = (t/T) \times \beta_0$ is a function of the current iteration t. As β is close to 0 at the early stage of adaptation, the augmented samples are closely around the source prototypes, reducing the impact of significant domain shift at the beginning of training. Along with the training progress, β gradually increases towards β_0, so the target domain prototypes play a more and more crucial role in the augmentation. For each class c in the t^{th} epoch, K augmented intermediate patterns are generated with random combining coefficient λ, making up synthesized features $\hat{\mathcal{D}}_{\mathbf{CCA}} = \{(\hat{\mu}^{c(1)}, c), \cdots, (\hat{\mu}^{c(K)}, c)\}_{c=1}^{C}$ of size CK, where $\hat{\mu}^{c(k)}$ is the k^{th} augmented feature for class c.

However, the augmented features in $\hat{\mathcal{D}}_{\mathbf{CCA}}$ are not directly used to optimize the framework, because only the linear interpolations between prototypes across domains are infeasible to explore all possible feature transformation directions and meaningful semantic variations. The cross-domain intermediate patterns are synthesized from the category-wise prototypes progressively and effectively bridge the overall semantic bias across domains, thus we exploit the intermediate features in $\hat{\mathcal{D}}_{\mathbf{CCA}}$ as *anchors* for the progressive source-supervised class-wise semantics augmentation as described below.

2.2.2.2.2 Source-supervised Semantic Augmentation (SSA) As aforementioned, certain translating directions in deep feature space represent meaningful semantic transformations in the original input space [131, 140]. Thus, we explore to approximate the procedure and facilitate meaningful cross-domain semantic knowledge transferring from the source domain to the target domain conditioned on the synthesized cross-domain intermediate *anchors* in $\hat{\mathcal{D}}_{\mathbf{CCA}}$.

Technically, we randomly sample vectors σ_s^c from a zero-mean multivariate normal distribution $\mathcal{N}(0, \Sigma_s^c)$ as the semantic transformation directions for the synthesized cross-domain intermediate *anchor*, $\hat{\mu}^{c(k)} \in \hat{\mathcal{D}}_{\mathbf{CCA}}$, to obtain augmented features $\tilde{\mathbf{f}}_k^c = \hat{\mu}^{c(k)} + \alpha\sigma_s^c$, where α is a positive coefficient to control the strength of semantic data augmentation. As the covariances are computed dynamically during training, the estimation in the first few epochs is not quite informative when the network is not well-trained. To address this issue, $\alpha = (t/T) \times \alpha_0$ is a function of the current epoch t where T is the total number of epochs. Thus, α can reduce the impact of the incorrectly-estimated covariances in the early training stage. Equivalently, we will have augmented samples $\tilde{\mathbf{f}}_k^c \sim \mathcal{N}(\hat{\mu}^{c(k)}, \alpha\Sigma_s^c)$, following a Gaussian distribution.

If each synthesized intermediate *anchor* $\hat{\mu}^{c(k)}$ is augmented for M times, an augmented set can be formed as $\tilde{\mathcal{D}}_{\mathbf{SSA}} = \{\{(\tilde{\mathbf{f}}_k^{c(1)}, c), \cdots, (\tilde{\mathbf{f}}_k^{c(M)}, c)\}_{k=1}^{K}\}_{c=1}^{C}$ of size MKC, where $\tilde{\mathbf{f}}_k^{c(m)}$ is the m^{th} augmented feature given the synthesized intermediate *anchor* $\hat{\mu}^{c(k)}$. Then, the augmented features are passed to the framework, which is optimized by minimizing the cross-entropy (CE) loss as:

$$\mathcal{L}_{aug} = \frac{-1}{CKM}\sum_{c=1}^{C}\sum_{k=1}^{K}\sum_{m=1}^{M}\log\Big(\frac{e^{\mathbf{w}_c^{\top}\tilde{\mathbf{f}}_k^{c(m)}+b_c}}{\sum_{j=1}^{C} e^{\mathbf{w}_j^{\top}\tilde{\mathbf{f}}_k^{c(m)}+b_j}}\Big), \tag{2.20}$$

where $\mathbf{W} = [\mathbf{w}_1, \cdots, \mathbf{w}_C]^{\top} \in \mathbb{R}^{C\times d}$ and $\mathbf{b} = [b_1, \cdots, b_C]^{\top} \in \mathbb{R}^{C}$ can be the weight matrix and biases for classifier $C_{s/t}(\cdot)$. Ideally, we would like $M \to \infty$ and $K \to \infty$, synthesizing infinite augmented samples with different semantic covariances to train the framework.

Proposition I. *Given synthesized samples $\tilde{\mathcal{D}}_{SSA} \in \mathbb{R}^{MKC}$, as $M/K \to \infty$, the expected cross-entropy loss $\mathcal{L}_{aug}^{\infty}$ is upper-bounded as $\overline{\mathcal{L}}_{aug}^{\infty}$, which can be calculated as follows:*

$$\begin{aligned}\mathcal{L}_{aug}^{\infty} &= \mathbb{E}_c\mathbb{E}_{\hat{\mu}^c}\mathbb{E}_{\tilde{\mathbf{f}}_k^c}\Big[-\log\big(\frac{e^{\mathbf{w}_c^{\top}\tilde{\mathbf{f}}_k^{c}+b_c}}{\sum_{j=1}^{C} e^{\mathbf{w}_j^{\top}\tilde{\mathbf{f}}_k^{c}+b_j}}\big)\Big] \\ &\leq \mathbb{E}_c\Big[-\log\Big(\frac{e^{\mathbf{w}_c^{\top}((1-\beta)\eta_s^c+\beta\eta_t^c)+b_c}}{\sum_{j=1}^{C} e^{\mathbf{w}_j^{\top}((1-\beta)\eta_s^c+\beta\eta_t^c)+b_j+\mathcal{A}}}\Big)\Big],\end{aligned} \tag{2.21}$$

where $\mathcal{A} = \frac{\alpha}{2}(\mathbf{w}_j^{\top} - \mathbf{w}_c^{\top})\Sigma_s^c(\mathbf{w}_j - \mathbf{w}_c)$.

***Proof*.** For k^{th} *anchor* $\hat{\mu}^{c(k)}$ of class c, the augmented samples by SSA are $\tilde{\mathcal{D}}_{\mathbf{SSA}}^{c(k)} = (\tilde{\mathbf{f}}_k^{c(1)}, c), \cdots, (\tilde{\mathbf{f}}_k^{c(M)}, c)\}$ of size M, where $\tilde{\mathbf{f}}_k^{c(m)}$ is the m^{th} augmented feature given the synthesized intermediate *anchor* $\hat{\mu}^{c(k)}$. Then the expected cross-entropy loss is defined as:

$$\begin{aligned}
\lim_{M\to\infty} \mathcal{L}_{aug}^{c(k)} &= \frac{1}{M}\sum_{m=1}^{M} -\log\Big(\frac{e^{\mathbf{w}_c^\top \tilde{\mathbf{f}}_k^{c(m)}+b_c}}{\sum_{j=1}^{C} e^{\mathbf{w}_j^\top \tilde{\mathbf{f}}_k^{c(m)}+b_j}}\Big)\\
&= \mathbb{E}_{\tilde{\mathbf{f}}_k^c}\Big[-\log\Big(\frac{e^{\mathbf{w}_c^\top \tilde{\mathbf{f}}_k^{c}+b_c}}{\sum_{j=1}^{C} e^{\mathbf{w}_j^\top \tilde{\mathbf{f}}_k^{c}+b_j}}\Big)\Big]\\
&= \mathbb{E}_{\tilde{\mathbf{f}}_k^c}\Big[\log(\sum_{j=1}^{C} e^{(\mathbf{w}_j^\top-\mathbf{w}_c^\top)\tilde{\mathbf{f}}_k^c+(b_j-b_c)})\Big]\\
&\leq \log\Big(\sum_{j=1}^{C}\mathbb{E}_{\tilde{\mathbf{f}}_k^c}\Big[e^{(\mathbf{w}_j^\top-\mathbf{w}_c^\top)\tilde{\mathbf{f}}_k^c+(b_j-b_c)}\Big]\Big),
\end{aligned} \tag{2.22}$$

where inequality follows the Jensen's inequality $\mathbf{E}[\log(X)] \leq \log(\mathbf{E}[X])$ [41], as the logarithmic function $\log(\cdot)$ is concave. The upper-bound of $\lim_{M\to\infty}\mathcal{L}_{aug}^{c(k)}$ is obtained by leveraging the moment-generating function $M_X(t) = \mathbf{E}(e^{tX}), t \in \mathbb{R}$. Specifically, for $\tilde{\mathbf{f}}_k^{c(m)} \sim \mathcal{N}(\hat{\mu}^{c(k)}, \alpha\Sigma_s^c)$ which is drawn from a Gaussian distribution, it is provable that $(\mathbf{w}_j^\top - \mathbf{w}_c^\top)\tilde{\mathbf{f}}_k^{c(m)} + (b_j - b_c)$ follows Gaussian distribution, i.e., $(\mathbf{w}_j^\top - \mathbf{w}_c^\top)\tilde{\mathbf{f}}_k^{c(m)} + (b_j - b_c) \sim \mathcal{N}((\mathbf{w}_j^\top - \mathbf{w}_c^\top)\hat{\mu}^{c(k)} + (b_j - b_c), \alpha(\mathbf{w}_j^\top - \mathbf{w}_c^\top)\Sigma_s^c((\mathbf{w}_j - \mathbf{w}_c)))$. Referring to the moment-generating function of Gaussian distribution: $\mathbf{E}[e^{tX}] = e^{t\mu+\frac{1}{2}\sigma^2 t^2}, X \sim \mathcal{N}(\mu, \sigma^2)$, we have the upper bound $\lim_{M\to\infty}\mathcal{L}_{aug}^{c(k)}$ as:

$$\lim_{M\to\infty} \mathcal{L}_{aug}^{c(k)} \leq -\log\Big(\frac{e^{\mathbf{w}_c^\top \hat{\mu}^{c(k)}+b_c}}{\sum_{j=1}^{C} e^{\mathbf{w}_j^\top \hat{\mu}^{c(k)}+b_j+\mathcal{A}}}\Big), \tag{2.23}$$

where $\mathcal{A} = \frac{\alpha}{2}(\mathbf{w}_j^\top - \mathbf{w}_c^\top)\Sigma_s^c(\mathbf{w}_j - \mathbf{w}_c)$. Moreover, as there are K synthesized intermediate *anchor* $\hat{\mu}^{c(k)}$ generated by CCA, the overall expected cross-entropy loss for all augmented samples based on all possible *anchors* are:

$$\begin{aligned}\mathcal{L}_{aug}^{\infty} &= \lim_{\substack{M\to\infty\\K\to\infty}} \mathbb{E}_c\Big[\frac{1}{K}\sum_{k=1}^{K}\mathcal{L}_{aug}^{c(k)}\Big] \\ &= \mathbb{E}_c\mathbb{E}_{\hat{\mu}^c}\Big[-\log\Big(\frac{e^{\mathbf{w}_c^\top\hat{\mu}^{c(k)}+b_c}}{\sum_{j=1}^{C}e^{\mathbf{w}_j^\top\hat{\mu}^{c(k)}+b_j+\mathcal{A}}}\Big)\Big] \\ &\leq \mathbb{E}_c\Big[\log\Big(\sum_{j=1}^{C}\mathbb{E}_{\hat{\mu}^c}[e^{(\mathbf{w}_j^\top-\mathbf{w}_c^\top)\hat{\mu}^c+(b_j-b_c)+\mathcal{A}}]\Big)\Big].\end{aligned} \tag{2.24}$$

As we know that $\hat{\mu}^c = (1-\lambda\beta)\eta_s^c + \lambda\beta\eta_t^c = \beta(\eta_t^c - \eta_s^c)\lambda + \eta_s^c$, and $\lambda \sim Beta(a,b)$ follows Beta distribution. Thus,

$$\begin{aligned}\mathcal{L}_{aug}^{\infty} &\leq \mathbb{E}_c\Big[\log\Big(\sum_{j=1}^{C}\mathbb{E}_{\hat{\mu}^c}[e^{(\mathbf{w}_j^\top-\mathbf{w}_c^\top)\hat{\mu}^c+(b_j-b_c)+\mathcal{A}}]\Big)\Big] \\ &= \mathbb{E}_c\Big[\log\Big(\sum_{j=1}^{C}\mathbb{E}_{\lambda}[e^{\beta(\mathbf{w}_j^\top-\mathbf{w}_c^\top)(\eta_t^c-\eta_s^c)\lambda}]e^{\mathcal{A}+\mathcal{B}}\Big)\Big],\end{aligned} \tag{2.25}$$

where $\mathcal{A} = \frac{\alpha}{2}(\mathbf{w}_j^\top - \mathbf{w}_c^\top)\Sigma_s^c(\mathbf{w}_j - \mathbf{w}_c)$, $\mathcal{B} = (\mathbf{w}_j^\top - \mathbf{w}_c^\top)\eta_s^c + (b_j - b_c)$.

As the moment-generating function of Beta distribution is defined as: $\mathbf{E}[e^{tX}] = 1+\sum_{k=1}^{\infty}(\prod_{r=0}^{k-1}\frac{a+r}{a+b+r})\frac{t^k}{k!}, X \frown$ Beta(a,b). and $a,b > 0$, such that $\prod_{r=0}^{k-1}\frac{a+r}{a+b+r} < 1$, then we obtain $\mathbf{E}[e^{tX}] \leq 1 + \sum_{k=1}^{\infty}\frac{t^k}{k!} = e^t$, thus the upper bound of $\mathcal{L}_{aug}^{\infty}$ is obtained as:

$$\mathcal{L}_{aug}^{\infty} \leq \mathbb{E}_c\Big[-\log\Big(\frac{e^{\mathbf{w}_c^\top((1-\beta)\eta_s^c+\beta\eta_t^c)+b_c}}{\sum_{j=1}^{C}e^{\mathbf{w}_j^\top((1-\beta)\eta_s^c+\beta\eta_t^c)+b_j+\mathcal{A}}}\Big)\Big], \tag{2.26}$$

□

2.2.2.2.3 Overall Objective By integrating the supervised cross-entropy loss of all labeled data, the domain-level confusion loss, the category-level adversarial loss, and the upper-bound of the expectation cross-entropy loss on the augmented distribution, we propose our overall objective as:

$$\min_{C^s,C^t} \ \mathcal{L}_c^C + \mathcal{L}_d^C + \gamma\overline{\mathcal{L}}_{aug}^{\infty}, \tag{2.27}$$

$$\min_{G} \ \mathcal{L}_c^G + \mathcal{L}_d^G, \tag{2.28}$$

Table 2.7: Accuracy (%) of Few-shot Domain Adaptation on Office-31

	UDA		**1-shot**					**3-shot**				
Method	$\mathcal{S}$-only	SymNets	$\mathcal{T}$-only	SymNets	CDAN	*d*-SNE	Ours	$\mathcal{T}$-only	CDAN	*d*-SNE	SymNets	Ours
$\mathcal{D}_{tr}$	$\mathcal{S}$	$\mathcal{U}$	$\mathcal{T}$	$\mathcal{ST}$	$\mathcal{ST}$	$\mathcal{ST}$	$\mathcal{ST}$	$\mathcal{T}$	$\mathcal{ST}$	$\mathcal{ST}$	$\mathcal{ST}$	$\mathcal{ST}$
A→W	68.40	90.80	69.72	80.92	80.77	83.22	**86.29**	85.59	82.40	90.87	90.44	**94.21**
D→W	96.70	98.80	67.23	97.23	97.49	93.71	**99.24**	88.16	98.36	95.29	98.57	**99.75**
W→D	99.30	(100.0)	69.74	99.73	99.77	99.36	**100.0**	88.37	**100.0**	98.27	99.53	**100.0**
A→D	68.90	93.90	75.97	**91.12**	85.23	83.69	89.95	86.63	86.91	88.61	91.77	**95.18**
D→A	62.50	(74.60)	50.09	62.39	67.18	67.36	**69.12**	66.84	68.35	73.26	**74.04**	73.80
W→A	60.70	72.50	52.46	60.88	65.45	**69.16**	66.49	69.56	66.45	72.46	73.45	**74.97**
Avg.	76.08	88.40	64.20	80.38	82.65	82.75	**85.18**	80.86	83.75	86.46	87.97	**89.65**

Table 2.8: Accuracy (%) of Few-shot Domain Adaptation on Office-Home

	Method	$\mathcal{D}_{tr}$	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg.
UDA	$\mathcal{S}$-only	$\mathcal{S}$	34.90	50.00	58.00	37.40	41.90	46.20	38.50	31.20	60.40	53.90	41.20	59.90	46.13
	SymNets	$\mathcal{U}$	47.70	72.90	(78.50)	64.20	71.30	(74.20)	64.20	48.80	(79.50)	(74.50)	52.60	(82.70)	67.60
1-shot	$\mathcal{T}$-only	$\mathcal{T}$	21.47	52.05	47.12	28.72	50.42	47.05	26.64	23.87	49.87	35.20	26.19	50.45	38.25
	SymNets	$\mathcal{ST}$	41.47	65.01	69.88	55.25	60.80	62.63	52.16	38.97	68.56	64.48	45.21	74.83	58.27
	CDAN	$\mathcal{ST}$	42.18	65.24	73.43	51.94	58.45	63.25	53.98	39.00	72.71	67.73	45.76	77.80	59.28
	d-SNE	$\mathcal{ST}$	**47.66**	**70.23**	72.15	55.87	**69.11**	67.75	55.02	43.92	72.50	61.58	47.55	75.17	61.54
	Ours	$\mathcal{ST}$	45.86	68.71	**73.97**	**58.26**	67.08	**69.17**	**58.63**	**44.10**	**73.61**	**68.48**	**50.95**	**78.49**	**63.11**
3-shot	$\mathcal{T}$-only	$\mathcal{T}$	37.37	67.24	57.77	46.39	65.64	59.36	44.46	36.44	58.86	46.48	35.69	68.68	52.03
	CDAN	$\mathcal{ST}$	43.45	67.64	74.43	55.78	61.80	64.29	57.07	42.49	73.21	68.36	46.86	79.29	61.22
	SymNets	$\mathcal{ST}$	49.84	72.21	74.08	62.30	70.04	68.36	61.38	49.13	72.12	69.88	53.39	79.15	65.16
	d-SNE	$\mathcal{ST}$	53.59	75.94	75.99	58.72	**76.01**	72.58	60.02	50.52	75.61	66.02	54.14	80.60	66.65
	Ours	$\mathcal{ST}$	**54.25**	**77.88**	**77.67**	**67.53**	75.21	**73.86**	**66.83**	**54.07**	**77.04**	**73.30**	**58.30**	**82.65**	**69.88**

where γ determines the relative importance of our marginalized augmentation. $C_{s/t}(\cdot)$ and $G(\cdot)$ are optimized alternatively until model converges.

Remark: Inspired by ISDA [140] with the semantic augmentation in the feature space, our MAF aims to address the FSDA problem with different motivation and progressive semantic transformation strategy, as there exists a large domain shift in FSDA and limited unlabeled target data are available. The proposed CCA and SSA can successfully bridge the domain gap by transferring the source semantic knowledge to the target domain progressively along with the process of domain adaptation. Besides, MAF is different from TSA [69], which focuses on transforming the source samples towards the target data semantic directions and relies on sufficient unlabeled target data during training, while lack of such target data is one of the main challenges in FSDA.

Table 2.9: Comparisons of Accuracy (%) for FSDA on Digits

Method	$\mathcal{D}_{tr}$	MNIST→SVHN	MNIST → MNIST-M
CCSA	$\mathcal{ST}$	37.63 ± 3.62	78.29 ± 2.00
d-SNE	$\mathcal{ST}$	61.73 ± 0.47	87.80 ± 0.16
Ours	$\mathcal{ST}$	$\mathbf{63.53} \pm 0.28$	$\mathbf{88.52} \pm 0.18$

2.2.3 Experiments

2.2.3.1 Datasets and Experimental Setting

Office-31 is a popular domain adaptation benchmark with 31 categories from 3 different domains. We follow the same protocol [150], and randomly select 20 samples per class from A, 8 samples per class from D/W making up the source domain. For the target, we formulate 1- and 3-shot experimental settings by randomly sampling 1 and 3 labeled samples per class, together with the labeled source data to train the model, while evaluating the rest of the target domain samples.

Office-Home consists of more than 15,500 images drawing from 4 different domains belonging to 65 categories, constituting a much larger and challenging benchmark. We also randomly select 1- and 3-shot target samples per class for the training process while keeping the rest of the target data for test, and finally obtain 12 cases based on 12 source-target pairs under 1-shot and 3-shot settings, respectively.

Implementation: We adopt ImageNet [16] pre-trained ResNet-50 [36] by removing the last fully-connected layer as the feature generator $G(\cdot)$, and plug-in two parallel one-layer fully-connected neural network as classifier $C_s(\cdot)$ and $C_t(\cdot)$, respectively. Due to the mini-batch strategy, the prototypes and covariance matrix are computed in an online fashion by aggregating statistics from all mini-batches [140]. We follow the annealing strategy of learning rate l as [164, 30]: $l_p = \frac{l_0}{(1+\delta p)^q}$, where p is the progress of training epochs linearly changing from 0 to 1, $l_0 = 0.0001$, $\delta = 10$ and $q = 0.75$, which is optimized to promote convergence and low error during training. Similarly, γ progressively changes as $\gamma_p = \gamma_0 \cdot (\frac{2}{1+e^{(-\xi \cdot p)}} - 1)$, in which $\xi = 10$. We follow [148] and accept $\lambda \sim \text{Beta}(2.0, 2.0)$. $\alpha_0 = 0.5, \beta_0 = 1.0$, and $\gamma_0 = 0.1$ are selected via deeply embedded validation and fixed for all experiments [154]. We define the max number epoch as 100, and observe that the training loss is stable around 30^{th} epoch for most tasks, so the results of the 30^{th} epoch are reported.

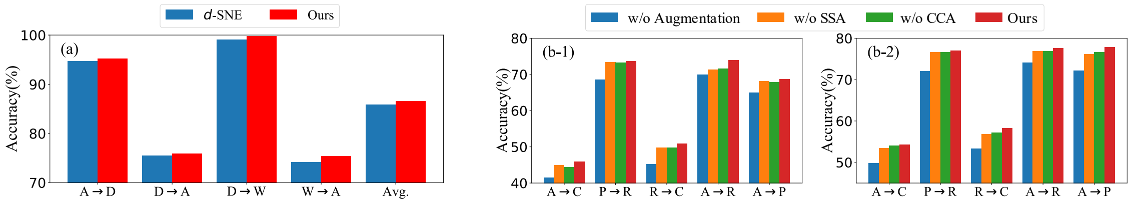


Figure 2.9: (a) Selected results on Office-31 produced by *d*-SNE and MAF with ResNet-101 as backbone.(b) Variants evaluation of MAF by removing either one or both marginalized augmentation strategies. (b-1) and (b-2) show the 1- and 3-shot results on Office-Home, respectively.

Moreover, we notice that $C_s(\cdot)$ and $C_t(\cdot)$ achieve close results after the training is stable, and all results reported in the paper are produced by classifier $C_t(\cdot)$.

Online Estimation: For the implementation details of calculating the source and target domain class prototypes and source data covariance, because we cannot have all training data available at once due to the limit of memory, during the mini-batch training, we estimate the source and target domain prototypes $\eta^c_{s/t}$, as well as the source domain covariance matrix Σ^c_s of each category c in an online fashion as:

$$\begin{aligned}
\eta^{c(t)}_{s/t} &= \frac{n^{c(t-1)}_{s/t}\eta^{c(t-1)}_{s/t} + m^{c(t)}_{s/t}\mu^{c(t)}_{s/t}}{n^{c(t-1)}_{s/t} + m^{c(t)}_{s/t}}, \\
\Sigma^{c(t)}_s &= \frac{n^{c(t-1)}_s\Sigma^{c(t-1)}_s + m^{c(t)}_s\Sigma'^{c(t)}_s}{n^{c(t-1)}_s + m^{c(t)}_s} \\
&\quad + \frac{n^{c(t-1)}_s m^{c(t)}_s(\eta^{c(t-1)}_s - \mu^{c(t)}_s)(\eta^{c(t-1)}_s - \mu^{c(t)}_s)^\top}{(n^{c(t-1)}_s + m^{c(t)}_s)^2},
\end{aligned}$$

where we accept the mean of each category features as the class-specific prototype, resulting in $\eta^{c(t)}_{s/t}$ and $\Sigma^{c(t)}_s$ as the estimates of features prototype and covariance matrix of the class c after the t-th mini-batch, $\mu^{c(t)}_{s/t}$ and $\Sigma'^{c(t)}_s$ are the mean and covariance of class c features in the t-th mini-batch, and $n^{c(t)}_{s/t} = n^{c(t-1)}_{s/t} + m^{c(t)}_{s/t}$ and $m^t_{s/t}$ denote the number of samples involved in all t mini-bathces and specific t-th mini-batch, respectively.

It is noteworthy that the target domain data are limited, even only one sample per class available in extreme cases, thus estimating the target domain covariance is infeasible.

Baselines: We compare our method with the source-only softmax classifier ($\mathcal{S}$-only), target-only nearest neighbor classifier ($\mathcal{T}$-only), two *unsupervised domain adaptation* (UDA) methods (CDAN [76] and SymNets [164]), and one state-of-the-art few-shot domain adaptation method (*d*-SNE [150]). Specifically, for $\mathcal{S}$-only, we fine-tune the ImageNet pre-trained ResNet-50 [36] only on the labeled

Table 2.10: Accuracy (%) of Few-Shot Domain Adaptation with various backbones on Office-31 (VGG16/ResNet-101), where * denotes the ResNet-101 backbone.

Method	$\mathcal{D}_{tr}$	A→D	D→A	D→W	W→A	Avg.
DRCN [33]	$\mathcal{U}$	67.10±0.30	56.00±0.50	96.40±0.30	54.09±0.50	68.40
KNN-Ad [118]	$\mathcal{U}$	84.10	58.30	96.40	63.80	75.65
I2I [87]	$\mathcal{U}$	71.10	50.10	96.50	52.10	67.45
G2A [115]	$\mathcal{U}$	87.70±0.50	72.80±0.30	97.90±0.30	71.40±0.40	82.45
SDA [129]	$\mathcal{ST}$	86.10±1.20	66.20±0.30	95.70±0.50	65.00±0.50	78.25
FADA [84]	$\mathcal{ST}$	88.20±1.00	68.10±0.60	96.40±0.80	71.10±0.90	80.95
CCSA [86]	$\mathcal{ST}$	89.00±1.20	71.80±0.50	96.40±0.80	72.10±1.00	80.95
d-SNE [150]	$\mathcal{ST}$	91.44±0.23	71.06±0.18	97.10±0.07	71.74±0.42	82.84
Ours	$\mathcal{ST}$	**92.12**±0.14	**71.26**±0.23	**97.21**±0.21	**72.15**±0.27	**83.19**
d-SNE* [150]	$\mathcal{ST}$	94.65 ± 0.38	75.51 ± 0.44	99.10 ± 0.24	74.20 ± 0.24	85.87
Ours*	$\mathcal{ST}$	**95.22**± 0.28	**75.88** ±0.19	**99.85** ±0.12	**74.56** ± 0.23	**86.38**

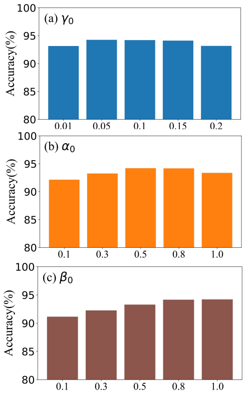


Figure 2.10: Parameter analysis

source domain, and evaluate it on the target domain. $\mathcal{T}$-only is based on the features produced by ImageNet pre-trained ResNet-50, and 1-/3-shot labeled target samples per class are used to infer the rest of the target test data by Euclidean distance. For UDA, we follow the original UDA experimental pipeline with a labeled source and unlabeled target for training. The idea is to evaluate that FSDA can outperform UDA by giving how many shots, which becomes very practical in privacy-related applications. For *d*-SNE, we re-implement it with ResNet-50 following the experimental instructions described in the original paper. All experiments are repeated 5 times with randomly selected labeled training data, then the average results are reported.

2.2.3.2 Comparison Results

All experimental results of our method and other baselines are reported in Table 2.7 to Table 2.8. The best performances of 1-shot and 3-shot settings are marked as **bold**. The UDA results are highlighted with (parentheses) when it is higher than the best 3-shot FSDA ones. However, the UDA results are obtained with a massive number of unlabeled target data for training, which is not a one-to-one comparison with other results. $\mathcal{D}_{tr}$ denotes the data used for training, $\mathcal{S}$ is *source-only*, $\mathcal{T}$ is *target-only*, $\mathcal{U}$ is the original *unsupervised domain adaptation* setting with all unlabeled target samples together with labeled source domain for training, and $\mathcal{ST}$ is *few-shot domain adaptation* tasks training on the whole source domain and a few target samples with annotations.

From Table 2.7, we observe that for the domain pairs with small distribution differences, e.g.,

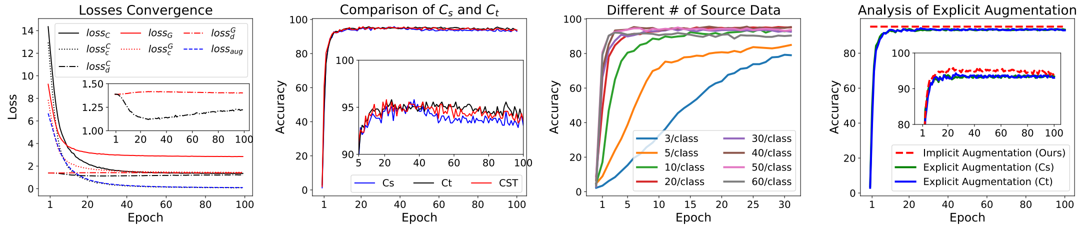


Figure 2.11: Ablation study of task A→D with 3-shot on Office-31 dataset, where (a) analysis of losses changing during training, (b) evaluation performance comparison of $C_s(\cdot)$ and $C_t(\cdot)$, and the fused performance, denoted as *CST*, (c) analysis of the performance with different number of source domain training data available, (d) comparison of optimizing the model with implicit and explicit augmentation.

D→W and W→D, the source-only softmax classifier also achieves good performance. However, for other domains with large domain distribution gaps, e.g., A→W, the source-only classifier cannot handle the target domain evaluation anymore. Besides, the target-only nearest neighbor classifier achieves poor performance on the target data evaluation when the labeled target data is limited, which indicates that lack of training data cripples the capability of the model, especially suffering from the unreliability of randomly selected training shots. Moreover, without sufficient unlabeled target domain data for training, conventional unsupervised domain adaptation approaches fail to manage the extreme scenario, especially when given only 1-shot target sample for training, SymNets even gets similar results as source-only, e.g., W→A and D→A. *d*-SNE achieves better performance under the 1-shot setting than SymNets, while SymNets achieves significant performance improvement with only two more labeled target data per class given for training. Our proposed model improves the average test performance by 2.43% and 1.68% under 1-shot and 3-shot settings, respectively, compared to the second-best method.

Since Office-Home has much more data in each domain compared to Office-31 data, which means under the few-shot domain adaptation setting given only 1- or 3-shot target data per class for training, most existing solutions will be distracted and overfitting to the source domain distribution. In Table 2.8, our proposed method achieves 1.57% and 3.23% average performance improvement on 1-shot and 3-shot settings, respectively, compared to the best compared baseline.

We follow the settings of *d*-SNE [150] to apply our proposed model on the Digits dataset, and report the results of the most two challenging tasks in Table 2.9. From the results, we observe the superiority of our proposed model compared to other baselines.

Furthermore, we report the results of our proposed model with ResNet-101 as the architecture

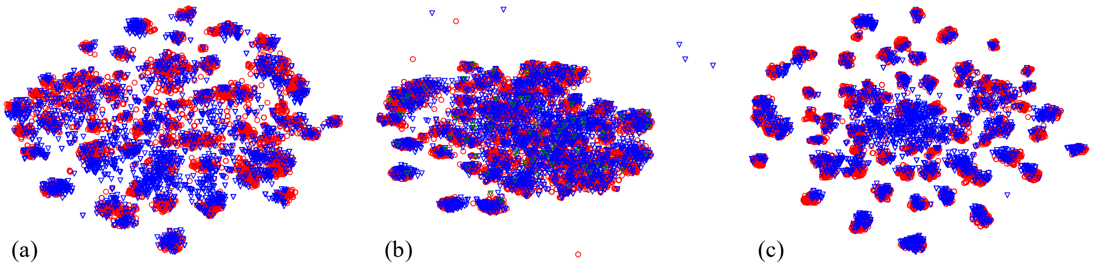


Figure 2.12: *t*-SNE visualization of features generated by different models of R→A 1-shot task from Office-Home. (a) Source-only (b) *d*-SNE (c) Ours. Red circles represent source samples, while blue triangles denote target domain test data.

on the Office-31 dataset and compare the results with *d*-SNE. Figure 2.9(a) lists four cross-domain learning tasks including two challenging source-target pairs with extremely large domain shift, i.e., D → A and W→A. It can be observed that our model outperforms all the state-of-the-art benchmarks including *d*-SNE with ResNet-101 as base network with 0.68% average improvement. Since ResNet-101 is a more complex feature extractor, the performance improvements compared to other baselines are relatively smaller than the results with ResNet-50 as the feature extractor. However, the increase in performance still shows the effectiveness of our model.

It is worth mentioning that only with 3-shot target labeled samples per class, our proposed model beats SymNets under the unsupervised domain adaptation settings requiring a large number of unlabeled target domain samples for training. On the other hand, we notice that *d*-SNE achieves promising performance with only 1-shot per class target data available for training, while the UDA solution (e.g., SymNets) achieves a significant performance boost with only two more target training data per class under 3-shot settings. These observations demonstrate the importance of a few labeled target data to address the challenges of domain adaptation.

2.2.3.3 Quantitative Analysis

Comparison of Different Augmentations: Firstly, to understand the contributions of the two different augmentation strategies, we report the performances of several variants on Office-Home in Figure 2.9 (b-1) 1-shot (b-2) 3-shot. Specifically, we remove either one of the two augmentation strategies (CCA and SSA), or both, while keeping other terms and training processes the same as the original model. From the results, we notice that CCA and SSA both are crucial and contribute to improving the evaluation performance from different perspectives.

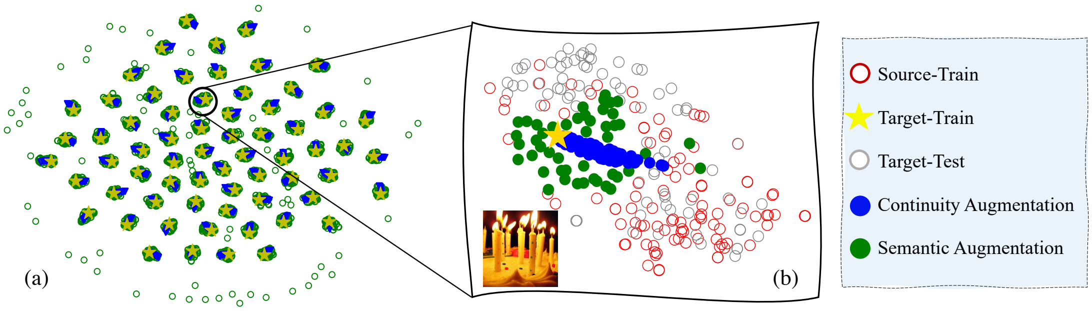
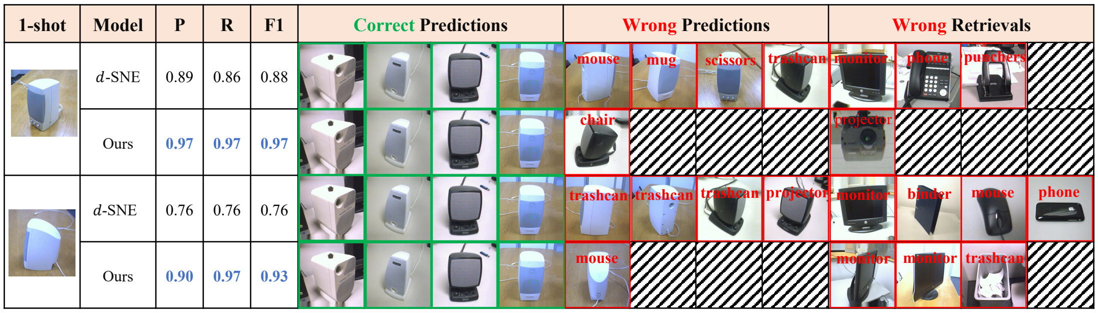


Figure 2.13: (a) Detailed visualization of synthesized samples generated by two different augmentation strategies on Office-Home R→A 1-shot task. (b) The zoomed-in part shows the category “candle”. Samples augmented by SSA (green dots) spread further and cover a larger range around the cross-domain intermediate anchors produced by CCA (blue dots).



1-shot	Model	P	R	F1	Correct Predictions	Wrong Predictions	Wrong Retrievals
	d-SNE	0.89	0.86	0.88		mouse mug scissors trashcan	monitor phone punchers
	Ours	**0.97**	**0.97**	**0.97**		chair	projector
	d-SNE	0.76	0.76	0.76		trashcan trashcan trashcan projector	monitor binder mouse phone
	Ours	**0.90**	**0.97**	**0.93**		mouse	monitor monitor trashcan

Figure 2.14: Selected samples from class “speaker” on Office-31 A→W 1-shot with results produced by *d*-SNE and our model.

Comparison of Various Backbones: We show more results on Office-31 3-shot tasks produced by our method and other compared baselines with different backbones in Table 2.10, and the best results with the same backbone are highlighted as **bold**. Specifically, the top part shows the results with VGG-16 as backbone [120], and the bottom two rows are results based on ResNet-101 [37], denoted as *d*-SNE* and Ours*. All compared results are directly copied from [150]. SDA introduces a shared feature extractor for both source and target domains to improve the discriminative capabilities of feature representations [129]. CCSA and FADA further involve the contrastive loss to create a unified framework for supervised domain adaptation and generalization [103, 84]. *d*-SNE uses stochastic neighborhood embedding techniques and a novel modified-Hausdorff distance to address the supervised domain adaptation [150]. We observe that our method outperforms all compared baselines with various backbones.

Parameter Analysis: There are three key parameters in the proposed MAF mode, α_0 controls the contribution of the source semantic knowledge during SSA augmentation, β_0 balances the importance of overall semantic prototypes across the domain during CCA intermediate patterns synthesis,

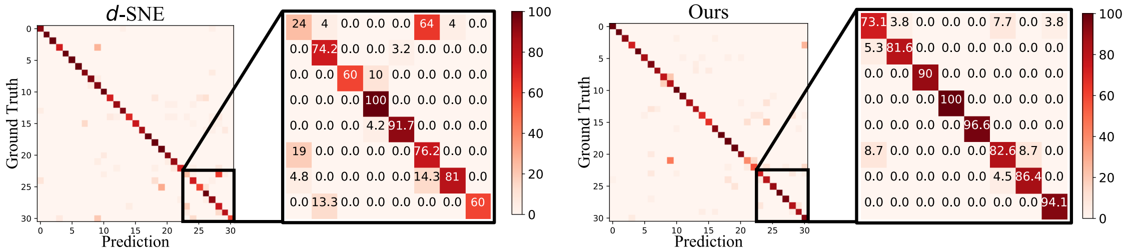


Figure 2.15: Confusion matrices on Office-31 A→W of *d*-SNE (left) and that of Ours (right). Some classes are zoomed in for better visualization.

and γ_0 decide the contribution of the synthesized distribution. Here we select different values of each parameter for task A $\rightarrow$ W with 3-shot on the Office-31 dataset and report the results as Fig. 2.10. From Fig. 2.10, we observe that the results are stable regarding the values of the parameters in a reasonable range.

Ablation Study: We deploy various ablation discussions about training convergence, comparison of the dual classifiers, the influence of accessible training data, and the comparison of optimizing the model with implicit and explicit data augmentation. And the results are shown in Fig. 2.11. From Fig. 2.11(a), we observe that the training losses converge around 30^{th} epoch, and the adversarial optimization process of the domain discrimination losses $\mathcal{L}_d^C$ and $\mathcal{L}_d^G$. In Fig. 2.11(b), we compare the performance of $C_s(\cdot)$ and $C_t(\cdot)$ on the test data during the model training, and we notice that they do not converge to the exactly same ones, although their performances are quite close along with the training progress. Moreover, we compare the performance of our model given different sizes of source samples for training, and the results are reported in Fig. 2.11(c). From the results, we observe that more source data will benefit the cross-domain adaptation and improve the performance on the target domain, but when the number is large enough, the improvement is not significant anymore. Finally, we compare the performance of explicitly generating synthesized data to train the model and implicitly optimizing the model with the upper bound of the expected cross-entropy loss of the augmented data in Fig. 2.11(d). From the results, we can see that the performance improves along with the increase of the number of synthesized data, which could reach a similar performance as what is achieved by implicit augmentation. However, it is noteworthy that although the performances of the two augmentation strategies become stable around a similar number of epochs, the time cost of each epoch during training with explicit augmentation is massively more than implicit augmentation (i.e., 3 hours/epoch V.S. 30 minutes/epoch), which is mainly due to the random sampling operations from a high-dimension multi-variant Gaussian distribution to produce synthesized samples.

2.2.3.4 Qualitative Analysis

Firstly, we show the t-SNE visualization of features of target data extracted from different models of 1-shot task R→A on Office-Home (Figure 2.12), to demonstrate the generalization ability of the learned models. We observe that the embedding from our proposed model is more discriminative compared to the features produced by source-only ResNet-50 and *d*-SNE, and the within-class samples across domains are more compact. Moreover, to visualize the effect of two augmentation strategies, we also visualize the explicitly generated samples by CCA and SSA, and the embeddings are shown in Figure 2.13 (a)(b). It is noteworthy that the augmented synthesized samples generated by the two mechanisms extend the range of the given 1-shot target sample in different and complementary directions.

In addition, we visualize the confusion matrix of *d*-SNE and our model of task A→W on Office-31 (Figure 2.15). We notice that our method improves on *d*-SNE results more than 30% for certain classes, which supports the superiority of our proposed model for few-shot domain adaptation. Qualitatively, we further show selected samples from class "speaker" on the Office-31 dataset A→W task in Figure 2.14 with different 1-shot target training samples to demonstrate the effectiveness of our model compared to *d*-SNE. From the results, we observe that our model achieves better performance on Precision (P), Recall (R), and F1-score (F1). *Wrong Predictions* show the "speaker" samples are wrongly predicted as other classes, and *Wrong Retrievals* denote the instances from other categories are wrongly retrieved as "speaker". Our model produces fewer wrong predictions and wrong retrievals with different annotated samples available for training, which attests to the robustness and generalizability of our proposed framework.

2.2.4 Discussion and Limitation

This study proposes the Marginalized Augmented Few-shot (MAF) domain adaptation model, incorporating Cross-domain Continuity Augmentation (CCA) and Source-supervised Semantic Augmentation (SSA). The MAF model outperforms state-of-the-art methods on various few-shot domain adaptation benchmarks. However, two main limitations hinder its application in large-scale real-life scenarios: the computationally expensive process of generating synthesized samples during training and the insufficient diversity of the generated samples. Addressing these limitations is crucial for future work, necessitating the design of more diverse and efficient augmentation strategies to enhance the few-shot domain adaptation process.

2.3 Conclusion

In conclusion, domain adaptation plays a crucial role in bridging the gap between labeled source domains and unlabeled target domains, allowing the transfer of knowledge across different data distributions. However, the challenges of imbalanced domain adaptation and domain adaptation with limited training data pose significant obstacles in achieving effective adaptation and maintaining fairness. Existing approaches have made strides in mitigating the impact of data imbalance and addressing few-shot domain adaptation, but more innovative solutions are needed. By developing novel techniques that consider class-wise adaptation, data scarcity, and imbalance, we can enhance the performance and fairness of domain adaptation methods in practical applications. Overcoming these challenges will pave the way for more robust and reliable domain adaptation models that can effectively handle real-world scenarios with limited data availability and imbalanced class distributions.

[45] Jing, Taotao, Hongfu Liu, and Zhengming Ding. "Towards novel target discovery through open-set domain adaptation." In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9322-9331. 2021.

3

Interpretable Multi-modal Transfer with Semantic Textual Description

Open-set domain adaptation (OSDA) considers that the target domain contains samples from novel categories unobserved in the external source domain. Unfortunately, existing OSDA methods always ignore the demand for information on unseen categories and simply recognize them as "unknown" set without further explanation. This motivates us to understand the unknown categories more specifically by exploring the underlying structures and recovering their interpretable semantic attributes. In this work, we propose a novel framework to accurately identify the seen categories in the target domain, and effectively recover the semantic attributes for unseen categories. Specifically, structure-preserving partial alignment is developed to recognize the seen categories through domain-invariant feature learning. Attribute propagation over a visual graph is designed to smoothly transit attributes from seen to unseen categories via visual-semantic mapping. Moreover, two new cross-domain benchmarks are constructed to evaluate the proposed framework in the novel and practical challenge. Experimental results on open-set recognition and semantic recovery demonstrate the superiority of the proposed method over other compared baselines.

3.1 Summary of Contribution

In recent years, domain adaptation (DA) attracts great interest to address the label insufficiency or unavailability issues, which is the bottleneck to the success of deep learning models [35]. DA casts light by transferring existing knowledge from a relevant source domain to the target domain of interest via eliminating the distribution gap across domains [23, 99]. Most DA efforts focus on the *closed-set domain adaptation* (CSDA) [23, 20], assuming the source and target domain share identical label space, which is not always satisfied in real-world scenarios, since the target domain may contain more than we know from the source domain. Following this, *open-set domain adaptation* (OSDA) has been widely studied given the source domain only covers a subset of the target domain label space[114, 99, 72, 57]. Unfortunately, these pioneering OSDA attempts simply identify the known categories while leaving the remaining unobserved samples as an "unknown" outlier set. Without any further steps, OSDA fails to discover what the unknown categories really are. Interestingly, the target domain may contain some exactly-new categories human beings never see before. This motivates us to further analyze the unknown set more specifically and discover novel categories.

In this work, we define such a problem as *Semantic Recovery Open-Set Domain Adaptation* (**SR-OSDA**), where source domain is annotated with both class labels and semantic attribute annotation, while the target domain only contains the unlabeled and unannotated data samples from more categories. The goal of SR-OSDA is to identify the seen categories and also recover the missing semantic information for unseen categories to interpret the new categories in the target domain. To our best knowledge, this is a completely new problem in literature with no exploration. The challenges now become two folds: (1) how to accurately identify seen and unseen categories in the target domain with well-labeled source knowledge; (2) how to effectively recover the missing attributes of

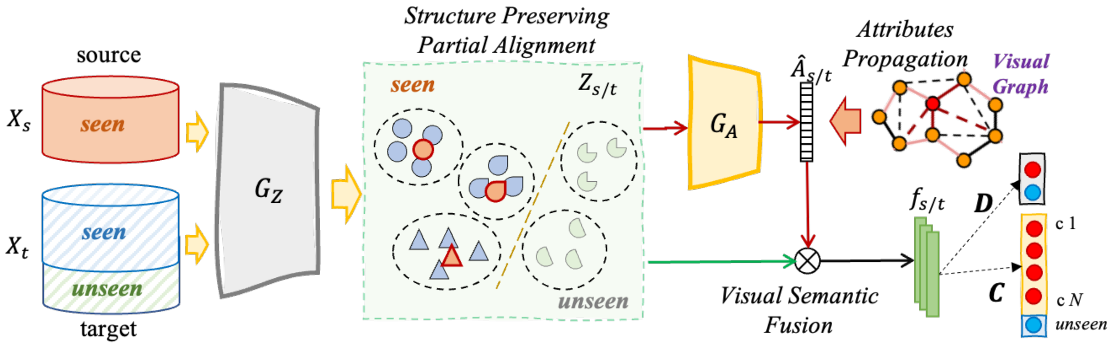


Figure 3.1: Illustration of our proposed framework, where X_t contains some unseen categories from X_s. Convolutional neural networks (e.g., ResNet [35]) are used as the backbone to extract visual features $X_{s/t}$, which are further input to $G_Z(\cdot)$ to learn domain-invariant features $Z_{s/t}$ through partial alignment. $G_A(\cdot)$ then maps $Z_{s/t}$ to semantic attributes A_s. Visual-semantic features are fused for the final classification tasks, one is $D(\cdot)$ to identify seen/unseen from target data, and the other $C(\cdot)$ to recognize all cross-domain data into C_s+1 classes (i.e., C_s seen + one unseen large category).

unseen categories.

To this end, we propose a novel framework to simultaneously recognize the known categories and discover new categories from the target domain as well as interpret them at the semantic level. The general idea of our model is to learn domain-invariant visual features by mitigating the cross-domain shift, and consequently build visual-semantic projection to recover the missing attributes of unknown target categories. Our contributions are highlighted as follows:

- We are the first to address the SR-OSDA problem and propose a novel and effective solution to identify seen categories and discover unseen ones.
- We propose structure preserving partial alignment to mitigate the domain shift when the target covers larger label space than the source, and attributes propagation over a visual graph to seek the visual-semantic mapping for better missing attribute recovery.
- Two new benchmarks are built for SR-OSDA evaluation. Our proposed method achieves promising performance in both target sample recognition and semantic attribute recovery.

Table 3.1: Notations and Descriptions

Notation	Description
$\mathcal{D}_t^s, \mathcal{D}_t^u$	seen/unseen target set
n_t^s, n_t^u	seen / unseen set samples number
$\mathbf{Y}_s, \mathbf{A}_s$	source domain labels / attributes
$\hat{\mathbf{a}}_s^i, \hat{\mathbf{a}}_t^j$	predicted source / target attributes
$\mathcal{R}_{\mathbf{x}}, \mathcal{R}_{\mathbf{z}},$	visual / embedding features prototypes
$\mathcal{F}_s^i, \mathcal{F}_t^j$	source / target joint representations

3.2 Motivations and Problem Definition

In this section, we illustrate our motivations and provide the problem definition of the semantic recovery open-set domain adaptation.

Open-set domain adaptation tasks [99] focus on the scenario when the target domain contains data from classes never observed in the source domain, which is more practical than the conventional closed-set domain adaptation [23]. However, existing open-set domain adaptation efforts simply identify those unseen target samples as one large unknown category and give up exploring the discriminative and semantic knowledge inside the unknown set. The demand for further understanding the novel classes that only exist in the target domain motivates us to study how to recover missing semantic attributes to explain the target data and discover novel classes, which leads to the problem *Semantic Recovery Open-Set Domain Adaptation* (**SR-OSDA**) addressed in this work. The main challenges of SR-OSDA lie in not only identifying the target samples in the unseen classes but also providing the partitional structures of these samples with recovered semantic attributes for further interpretation.

For better understanding, we clarify the problem with mathematical notations. The target domain is defined as $\mathcal{D}_t = \{\mathbf{X}_t\}$ containing n_t samples with visual features from C_t categories. The auxiliary source domain $\mathcal{D}_s = \{\mathbf{X}_s, \mathbf{Y}_s, \mathbf{A}_\mathbf{s}\}$ consists of n_s samples from C_s classes with visual features $\mathbf{X}_s$, labels $\mathbf{Y}_s$, and semantic attributes $\mathbf{A}_s$. For each source sample, the semantic attributes $\mathbf{a}_s^i = \mathcal{A}^{y_s^i}, \mathbf{a}_s^i \in \mathbb{R}^{d_\mathbf{a}}$ are obtained from $\mathcal{A}$, which consists of class-wise attributes of the source domain. SR-OSDA aims to recover the missing semantic attributes for the target data based on the visual features and uncover novel categories never present in the source domain. Table 3.1 shows several key notations and descriptions in the SR-OSDA setting in addition to Table 1.

It is noteworthy that the source and target domains are drawn from different distributions. Besides, the target data set covers all classes in the source domain, as well as K exclusive categories only exist in the target domain, where $K = C_t - C_s > 0$. SR-OSDA is different from open-set domain adaptation, which ignores recovering interpretable knowledge and discovering new classes in the target domain. Moreover, the defined problem is different from generalized zero-shot learning [116], as we have no access to the semantic knowledge of the target domain unseen categories.

To our best knowledge, SR-OSDA is the first time proposed, aiming to discover novel target classes via recovering semantic attributes from the auxiliary source data. In the following, we illustrate our solution to learn the relationship between the visual features and semantic attributes with the guidance of the source data, which can be transferred to the target data and interpretably discover unseen classes.

3.3 The Proposed Method

3.3.1 Framework Overview

To tackle the above SR-OSDA problem, we propose a novel target discovery framework (Figure 3.1) to simultaneously recognize the target domain data from categories already observed in the source domain, and recover the interpretable semantic attributes for the unknown target classes from the source. To achieve this, three modules are consequently designed to address the cross-domain shift, semantic attributes prediction, and task-driven open-set classification. Specifically, the source data are adapted to the target domain feature space through partial alignment while preserving the target structure. A projector $G_A(\cdot)$ bridging the domain invariant feature space $\mathbf{z}^i_{s/t}$ and the semantic attributes space $\mathbf{a}^i_{s/t}$ is trained by the source data as well as the target data with confident pseudo attributes. Moreover, the visual features will guide the attributes propagated from seen categories to unseen ones, and the semantic attributes will also promote the visual features discrimination through joint visual-semantic representation recognition for $C(\cdot)$ and $D(\cdot)$, where $D(\cdot)$ is a binary classifier to identify seen and unseen target samples, and $C(\cdot)$ is an extended multi-class classifier with $C_s + 1$ outputs.

Since the target data are totally unlabeled and all three modules rely on the label information in the target domain, we first discuss how to obtain the pseudo labels of target samples through our design progressive seen-unseen separation stage. That is, we will assign target samples into C_s observed categories and K unobserved categories. In the following, we introduce the progressive seen-unseen separation and three key modules in our proposed framework.

3.3.2 Modules and Objective Function

Progressive Seen-Unseen Separation. Here we describe the initialization strategy to separate the target domain data into seen and unseen sets based on the visual features space. Intuitively, part of source-style target samples is promisingly identified by the well-trained source model, which is actually belonging to seen categories more probably. On the other hand, those target samples assigned with even and mixed prediction probabilities across multiple classes tend to be unseen categories, as no source classifier can easily recognize them. To achieve this, we apply the prototypical classifier to measure the similarities between each target sample to all source class prototypes [121]. For each target sample $\mathbf{x}_t^i$ and the source C_s prototypes $\{\mu^c|_{c=1}^{C_s}\}$, the probability prediction is defined as:

$$p(y_t^i = c|\mathbf{x}_t^i) = \frac{\exp\big(-d(\mathbf{x}_t^i,\ \mu^c)\big)}{\sum_{c'} \exp\big(-d(\mathbf{x}_t^i,\ \mu^c)\big)}, \tag{3.1}$$

where $d(\cdot)$ is the distance function. The highest probability prediction p_t^i is adopted as the pseudo label $\tilde{y}_t^i$ for $\mathbf{x}_t^i$. Next, we adopt a threshold τ to progressively separate all target samples into seen $\mathcal{D}_t^s$ and unseen sets $\mathcal{D}_t^u$. The number of samples in $\mathcal{D}_t^s$ and $\mathcal{D}_t^u$ are denoted as n_t^s and n_t^u, respectively. Specifically, we define τ the mean of the highest probability prediction of all target samples, i.e., $\tau = \frac{1}{n_t}\sum_{\mathbf{x}_t^i \in \mathcal{D}_t} p_t^i$. Based on that, we can build two sets:

$$\begin{cases} \mathbf{x}_t^i \in \mathcal{D}_t^s, & p_t^i \geq \tau \\ \mathbf{x}_t^i \in \mathcal{D}_t^u, & p_t^i < \tau \end{cases}. \tag{3.2}$$

Since we only have the source prototypes in the beginning, they are not accurate to identify seen and unseen sets due to the domain shift. Thus, we can gradually update the seen prototypes by involving newly-labeled target samples from $\mathcal{D}_t^s$ as $\mu^c = (1-\alpha)\mu^c + \alpha\frac{1}{n_t^{s(c)}}\sum_{\mathbf{x}_t^i \in \mathcal{D}_t^{s(c)}} \mathbf{x}_t^i$, where

$\mathcal{D}_t^{s(c)}$ denotes a set of $n_t^{s(c)}$ target samples predicted as $\tilde{y}_t^i = c$ confidently, and α is the small value to control the mixture of cross-domain prototypes.

After obtaining all pseudo labels in the seen set $\mathcal{D}_t^s$, we also need to explore more specific knowledge in $\mathcal{D}_t^u$ instead of treating it as a whole like OSDA [114]. To this end, we apply K-means clustering algorithm to group $\mathcal{D}_t^u$ into K clusters with the cluster center as $\{\eta^{k_1}, \cdots, \eta^K\}$. In this way, we can obtain all prototypes of *seen* and *unseen* categories as $\mathcal{R}_{\mathbf{x}} = \{\mu^1, \cdots, \mu^{C_s}, \eta^{k_1}, \cdots, \eta^K\}$. In order to refine the pseudo labels of target samples, we adopt a K-means clustering algorithm with centers initialized as $\mathcal{R}_{\mathbf{x}}$ over X_t until the results are converged.

To this end, we obtain all pseudo labels for target samples. We also assign semantic attributes to seen target samples based on their pseudo label belonging to which source category. Next, we explore structure preserving partial alignment, attribute propagation, and task-driven classification to solve SR-OSDA.

Structure Preserving Partial Alignment. Due to the disparity between the source and target domains' label spaces, directly matching the feature distribution across domains is destructive. Considering our goal of uncovering the *unseen* categories in the target domain, preserving the structural knowledge of the target domain data becomes even more crucial. Thus, instead of mapping the source and target domains into a new domain-invariant feature space, we seek to align the source data to the target domain distribution through partial alignment.

Specifically, with the help of the target domain pseudo labels $\tilde{\mathbf{Y}}_t$, for each class c in the pseudo label space, which contains $C_s + K$ categories, the prototype can be calculated as the class center in the space of feature $\mathbf{z}$ can be calculated as $\mathcal{R}_{\mathbf{z}}^c = \mathbb{E}_{\mathbf{x}_t^i \in \mathcal{D}_t} \mathbf{z}_t^i \cdot 1_{\tilde{y}_t^i = c}$. The prototypes $\mathcal{R}_{\mathbf{z}}$ describe the class-wise structural knowledge in the target domain in the $\mathbf{z}$ feature space. To solve the domain disparity, we align each source sample to its specific target center and also keep away from other target centers as:

$$\mathcal{L}_s^R = \frac{1}{n_s} \sum_{i=1}^{n_s} \sum_{c=1}^{|\mathcal{R}_{\mathbf{z}}|} \Big(1_{y_s^i = c} d(\mathbf{z}_s^i, \mathcal{R}_{\mathbf{z}}^c) - \frac{1_{y_s^i \neq c}}{|\mathcal{R}_{\mathbf{z}}| - 1} d(\mathbf{z}_s^i, \mathcal{R}_{\mathbf{z}}^c) \Big), \tag{3.3}$$

where $C_s + K = |\mathcal{R}_{\mathbf{z}}|$ is the total number of prototypes in $\mathcal{R}_{\mathbf{z}}$. Moreover, we deploy a similar loss to make within-class target samples more compact while keeping between-class target samples more discriminative as:

$$\mathcal{L}_t^R = \frac{1}{n_t} \sum_{i=1}^{n_t} \sum_{c=1}^{|\mathcal{R}_\mathbf{z}|} \Big(1_{\tilde{y}_t^i = c} d(\mathbf{z}_t^i, \mathcal{R}_\mathbf{z}^c) - \frac{1_{\tilde{y}_t^i \neq c}}{|\mathcal{R}_\mathbf{z}| - 1} d(\mathbf{z}_t^i, \mathcal{R}_\mathbf{z}^c)). \tag{3.4}$$

Such a loss function will make within-class target samples more compact while pushing away from others.

These two loss functions help align source and target to obtain domain-invariant visual features and also seek more discriminative knowledge over target samples. Then we obtain the objective of structure preserving partial domain adaptation as $\mathcal{L}^R = \mathcal{L}_s^R + \mathcal{L}_t^R$.

Attributes Propagation with Visual Structure. Since unseen target samples are totally without any annotations either class label or semantic attributes, our goal is to recover their semantic attributes via visual-semantic projector $G_A(\cdot)$. However, only attributes knowledge of the classes seen in the source domain is available for training, while the target samples from unseen categories have no way to optimize the $G_A(\cdot)$, which might lead the projector $G_A(\cdot)$ towards bias to the seen categories when dealing with unseen target class samples. To this end, we propose the mechanism of attribute propagation to aggregate the visual graph knowledge into the semantic description projection, which is beneficial to the attributes propagated from seen classes to unseen classes.

Specifically, for features $\mathbf{z}^i = G_Z(\mathbf{x}^i)$ of a training batch, the adjacency matrix A is calculated as $A_{ij} = \exp(-d_{ij}^2/\sigma^2)$, where $A_{ii} = 0, \forall i$, and $d_{ij} = \|\mathbf{z}^i - \mathbf{z}^j\|_2$ is the distance of $(\mathbf{z}^i, \mathbf{z}^j)$. σ is a scaling factor set as $\sigma^2 = \mathrm{Var}(d_{ij}^2)$ as [107] to stabilize training. The attributes projected from visual features are reconstructed as:

$$\hat{\mathbf{a}}^i = \sum\nolimits_j W_{ij} G_A\Big(G_Z(\mathbf{x}^j)\Big), \tag{3.5}$$

where $L = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$, $D_{ii} = \sum_j A_{ij}$ and $W = (\mathbf{I} - \beta L)^{-1}$, in which $\beta \in \mathbb{R}$ is a scaling factor fixed as suggested by [107], and $\mathbf{I}$ is the identity matrix. After the semantic attributes propagation, $\hat{\mathbf{a}}_{s/t}^i$ is refined as a weighted combination of its neighbors guided by the visual graph. This benefit attributes projector from overfitting to the *seen* categories while removing undesired noise [107].

After the projected attributes refinement via attribution propagation, we optimize the attributes projector $G_A(\cdot)$ on the seen categories across two domains:

$$\mathcal{L}^A = \frac{1}{N_s + N_t^s} \sum_{\mathbf{x}^i \in \mathcal{D}_s \cup \mathcal{D}_t^s} L_{bce}(\hat{\mathbf{a}}^i,\ \ \mathbf{a}^i), \tag{3.6}$$

where $L_{bce}(\cdot)$ is the binary cross-entropy loss, and N_t^s is the number of samples in $\mathcal{D}_t^s$. Each dimension of the semantic attributes $\mathbf{a}^i \in \mathbb{R}^{d_\mathbf{a}}$ represents one specific semantic characteristic, and $\hat{\mathbf{a}}^i$ describes the predicted probability that the input sample has specific characteristics.

Visual-Semantic Fused Recognition. Since visual features and semantic attributes describe the data distribution from different perspectives. To simultaneously leverage the multi-modality benefits of visual and semantic descriptions, we explore the joint visual and semantic representation by conveying the semantic discriminative information $\mathbf{a}^i$ into the visual feature $\mathbf{z}^i$ as $\mathbf{f}^i = \mathbf{z}^i \oplus \mathbf{a}^i$, where $\oplus$ is concatenating $\mathbf{z}^i$ and $\mathbf{a}^i$ as joint feature $\mathbf{f}^i$.

It is noteworthy that during the training, several different semantic attributes are available in different stages, e.g., ground-truth ($\mathbf{a}^i$), pseudo attributes ($\tilde{\mathbf{a}}^i$), and predicted attributes ($\hat{\mathbf{a}}^i$). We take them all into account and will obtain various joint representations as:

$$\begin{cases} \mathcal{F}_s^i = \{\mathbf{f}_s^i, \hat{\mathbf{f}}_s^i\}, & \mathbf{x}_s^i \in \mathcal{D}_s \\ \mathcal{F}_t^i = \{\tilde{\mathbf{f}}_t^i, \hat{\mathbf{f}}_t^i\}, & \mathbf{x}_t^i \in \mathcal{D}_t^s \\ \mathcal{F}_t^i = \{\hat{\mathbf{f}}_t^i\}, & \mathbf{x}_t^i \in \mathcal{D}_t^u \end{cases}, \tag{3.7}$$

where $\mathbf{f}_s^i = \mathbf{z}_s^i \oplus \mathbf{a}_s^i$, $\tilde{\mathbf{f}}_t^i = \mathbf{z}_t^i \oplus \tilde{\mathbf{a}}_t^i$, and $\hat{\mathbf{f}}_{s/t}^i = \mathbf{z}_{s/t}^i \oplus \hat{\mathbf{a}}_{s/t}^i$. All joint features in $\mathcal{F}_s$ and $\mathcal{F}_t$ are input into the classifier $C(\cdot)$ and $D(\cdot)$ to optimize the framework.

To maintain the performance of classifier $C(\cdot)$ over supervision from source and target domains, we construct the cross-entropy classification loss as:

$$\mathcal{L}^C = \frac{1}{N_s + N_t} \sum_{\mathbf{f}^i \in \mathcal{D}_s \cup \mathcal{D}_t} L_{ce}(C(\mathbf{f}^i), y^i), \tag{3.8}$$

where $L_{ce}(\cdot)$ is the cross-entropy loss and y^i denotes the C_s source labels and $C_s + 1$ target labels. Moreover, we train a binary classifier $D(\cdot)$ to separate the target domain into *seen* and *unseen* subsets,

Table 3.2: Statistical characteristics on D2AwA and I2AwA dataset

Dataset	*D2AwA*						*I2AwA*	
Domain	**A**		**P**		**R**		**I**	**Aw**
Role	source	target	source	target	source	target	source	target
#Images	9,343	16,306	3,441	5,760	5,251	10,047	2,970	37,322
#Attributes	85	85	85	85	85	85	85	85
#Classes	10	17	10	17	10	17	40	50

Table 3.3: Open-set domain adaptation accuracy (%) on D2AwA and I2AwA

Dataset	*D2AwA*																		*I2AwA*		
Task	**A→P**			**A→R**			**P→A**			**P→R**			**R→A**			**R→P**			**I→Aw**		
Method	OS*	OS$^\diamond$	OS	OS*	OS$^\diamond$	OS	OS*	OS$^\diamond$	OS	OS*	OS$^\diamond$	OS	OS*	OS$^\diamond$	OS	OS*	OS$^\diamond$	OS	OS*	OS$^\diamond$	OS
OSBP [114]	49.6	10.8	46.0	74.2	13.6	68.7	76.0	9.1	69.9	63.3	6.9	58.2	90.1	13.7	83.2	55.9	10.6	51.7	67.6	7.5	66.2
STA [72]	60.1	33.0	57.6	85.5	10.8	78.7	**90.2**	5.7	**82.5**	**82.8**	7.4	76.0	88.5	7.2	81.1	**66.9**	13.5	62.0	51.5	45.5	51.4
AOD [27]	50.7	9.5	46.9	78.4	12.7	72.4	80.3	5.1	73.5	79.7	5.3	73.0	92.0	12.8	84.8	61.2	9.6	56.5	75.2	6.3	73.5
Ours(Init)	53.1	45.1	52.3	78.8	**72.3**	78.2	75.3	94.8	77.1	67.3	82.0	68.6	86.2	87.7	86.4	52.0	77.8	54.4	82.2	6.3	73.5
Ours(Vis)	54.1	**76.1**	56.1	75.4	70.3	75.0	69.5	**98.5**	72.1	57.4	83.1	59.7	88.3	**98.8**	89.2	58.7	**91.2**	61.6	48.2	**70.3**	48.7
Ours	**62.8**	47.2	**61.4**	**90.9**	71.4	**89.1**	79.2	**98.5**	81.0	78.3	**83.7**	**78.8**	**94.9**	90.5	**94.5**	61.2	80.4	**63.0**	**83.2**	70.2	**82.8**

which can be optimized by:

$$\mathcal{L}_t^D = \frac{1}{n_t} \sum_{\mathbf{x}_t^i \in \mathcal{D}_t} \sum_{\mathbf{f} \in \mathcal{F}_t^i} L_{bce}(D(\mathbf{f}),\ \psi(\tilde{y}_t^i)), \tag{3.9}$$

in which $\psi(\tilde{y}_t^i)$ indicates if the target sample $\mathbf{x}_t^i$ is from the *seen* categories ($\psi(\tilde{y}_t^i) = 0,\ \mathbf{x}_t^i \in \mathcal{D}_t^s$), or from the *unseen* categories ($\psi(\tilde{y}_t^i) = 1,\ \mathbf{x}_t^i \in \mathcal{D}_t^u$).

Then we have our classification supervision objective on both source and target domain with joint visual and semantic representations as $\mathcal{L}^T = \mathcal{L}^C + \mathcal{L}_t^D$.

Overall Objective Function. To sum up, we can obtain the overall objective function by integrating the structure-preserving partial adaptation, semantic attributes propagation and prediction, and joint visual-semantic representation recognition as:

$$\min_{G_Z, G_A, C, D} \mathcal{L}^T + \lambda_1 \mathcal{L}^R + \lambda_2 \mathcal{L}^A, \tag{3.10}$$

where λ_1 and λ_2 are two trade-off parameters. Through minimizing the proposed objective, the semantic descriptive knowledge is aggregated from the source data into the unlabeled target domain through joint visual-semantic representation supervision and attribute propagation. Meanwhile, the discriminative visual structure in the target domain is promoted by the cross-domain partial adaptation.

Table 3.4: Semantic Recovery Accuracy (%) on D2AwA and I2AwA

Dataset	*D2AwA*																		*I2AwA*		
Task	**A→P**			**A→R**			**P→A**			**P→R**			**R→A**			**R→P**			**I→Aw**		
Method	S	U	H	S	U	H	S	U	H	S	U	H	S	U	H	S	U	H	S	U	H
Source-only	67.6	0.0	0.0	87.6	0.0	0.0	91.3	0.0	0.0	**85.3**	0.0	0.0	94.1	0.0	0.0	71.1	0.0	0.0	77.2	0.3	0.7
ABP [175]	68.1	0.0	0.0	87.9	0.0	0.0	**91.7**	0.0	0.0	83.6	0.0	0.0	94.4	0.0	0.0	70.0	0.0	0.0	79.8	0.0	0.0
TF-VAE [88]	**70.4**	0.0	0.0	88.4	0.0	0.0	85.1	0.0	0.0	79.6	0.0	0.0	**96.4**	0.0	0.0	**72.5**	0.0	0.0	62.8	0.0	0.0
ABP* [175]	64.5	6.4	11.7	86.0	5.9	11.1	84.0	24.4	37.8	81.3	12.7	21.9	93.8	16.2	27.6	67.6	7.9	14.1	78.0	13.4	22.9
TF-VAE* [88]	59.7	12.8	21.0	77.9	16.4	27.1	35.1	35.6	35.3	34.8	**32.7**	**33.7**	68.5	36.1	47.3	50.7	**21.0**	29.7	37.7	20.0	26.2
Ours	62.5	**27.0**	**37.7**	**90.7**	**30.0**	**45.1**	79.2	**36.7**	**50.2**	78.0	15.7	26.1	95.2	**37.8**	**54.1**	59.0	20.8	**30.8**	**83.1**	**22.0**	**34.8**

3.4 Experiments

3.4.1 Experimental Settings

Datasets. We construct two datasets for the novel SR-OSDA setting. (1) *D2AwA* is constructed from the DomainNet dataset [102] and AwA2[144]. Specifically, we choose the shared 17 classes between the DomainNet and AwA2, and select the alphabetically first 10 classes as the seen categories, leaving the rest 7 classes as unseen. The corresponding attribute features in AwA2 are used as the semantic description. It is noteworthy that DomainNet contains 6 different domains, while some of them barely share the semantic characteristics described by the attributes of AwA2, e.g., quick draw. Thus, we only take the "real image" (R) and "painting" (P) domains into account, together with the AwA2 (A) data for model evaluation. (2) *I2AwA* is collected by [176] consisting of 50 animal classes, and split into 40 seen categories and 10 unseen categories as [144]. The source domain (I), includes 2,970 images from seen categories collected via the Google image search engine, while the target domain comes from AwA2 (Aw) dataset for zero-shot learning with 37,322 images in all 50 classes [144]. We use the binary attributes of AwA2 as the semantic description, and only the seen categories attributes of source data are available for training. Only one task I→Aw is evaluated on *I2AwA*. Table 3.2 shows several statistical characteristics of *D2AwA* and *I2AwA*.

Evaluation Metrics. We evaluate our method in two aspects: (1) target sample recognition under the open-set domain adaptation and (2) generalized semantic attribute recovery. For the first one, we follow the conventional open-set domain adaptation studies [99, 114], recognizing the whole target domain data into one of the seen categories or the "unknown" category. The standard open-set domain adaptation average accuracy calculated on all the classes are reported as OS. Besides, we report the average accuracy calculated on the target domain seen classes as OS*, while for the target domain unseen categories, the accuracy is reported as OS$^\diamond$. For semantic attribute recovery, we compare the predicted semantic description with the ground-truth semantic attributes. Specifically, we adopt a TWO-stage test: (a) identifying a test sample from *seen* or *unseen* set, (b) applying prototypical classification with corresponding *seen*/*unseen* ground-truth attributes. We report the performances on the seen categories and unseen categories as S and U, respectively, and calculate the harmonic mean H [116], defined as $H = 2 \times S \times U/(S+U)$. Note that all results we reported are the average of class-wise top-1 accuracy, to eliminate the influence caused by the imbalanced class.

Implementation. We use the pre-trained ResNet-50 [35] on ImageNet as the backbone and take the second last fully connected layer as the features $\mathbf{X}_{s/t}$ [16, 35]. $G_Z(\cdot)$ is a two-layer fully connected neural network with a hidden layer dimension of 1,024, and the output feature dimension is 512. $C(\cdot)$ and $D(\cdot)$ are both two-layer fully connected neural networks classifiers with hidden layer dimension of 256, and the output dimension of $C(\cdot)$ is $C_s + 1$, while the output of $D(\cdot)$ is just two dimensions indicating seen or unseen classes. $G_A(\cdot)$ is a two-layer neural network with a hidden layer dimension of 256 followed, and the final output dimension is the same as the semantic attributes dimension followed by the Sigmoid function. We employ the cosine distance for the prototypical classification, while all other distances used in the paper are Euclidean distances. For simplicity, we adopt the ground-truth novel classes number as K, and we notice that the results are not sensitive to the value of K within a range. There are many cluster number estimation methods but out of scope in this work. For parameters, we fix $\alpha = 0.001$, $\beta = 0.2$, $\lambda_1 = 10^{-4}$, $\lambda_2 = 0.1$, and the learning rate is fixed as 10^{-3} for all experiments, and report the 100-th epoch results for all the experiments. The source code of this work is available online*.

Competitive Methods. Since the problem we address in this work is in a novel and practical setting, we mainly compare two distinctive branches of baselines in terms of open-set domain adaptation and zero-shot learning.

For open-set domain adaptation, we compare our method with OSBP [114], AOD [27], and STA [72]. OSBP utilizes the adversarial training strategy to extract features for the target data, which is recognized into seen/unseen classes by a pre-defined threshold [114]. AOD exploits the semantic structure of open set data from categorical alignment and contrastive mapping to push the unknown

*https://github.com/scottjingtt/SROSDA.git

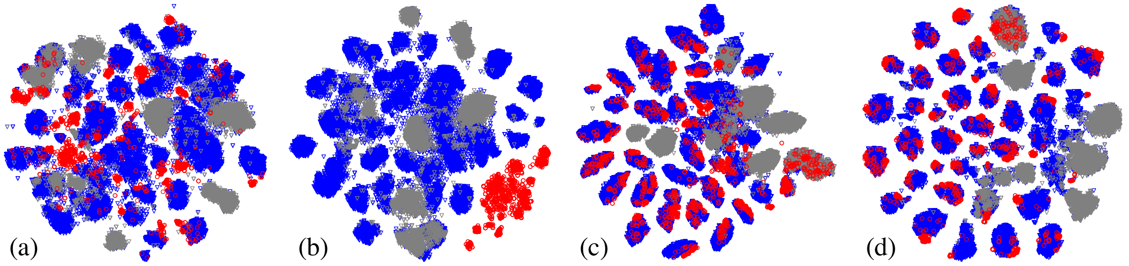


Figure 3.2: tSNE visualization of representations generated by (a) ResNet, (b) STA, and (c) Ours on I2AwA. (d) shows the joint visual-semantic features proposed in our paper. Red circles denote source data. Blue and gray triangles denote target domain seen and unseen classes.

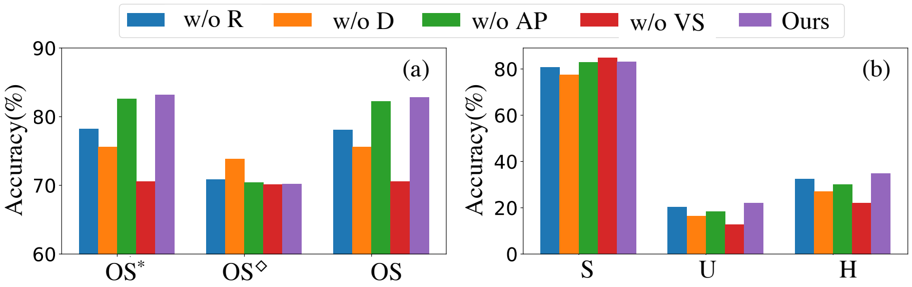


Figure 3.3: Ablation study of our proposed model on I2AwA by removing specific one of structure-preserving partial alignment (w/o $\mathcal{L}^R$), binary classifier (w/o $\mathcal{L}^D$,) attributes propagation (w/o AP), or joint visual-semantic representation (w/o VS).

classes away from the decision boundary [27]. Differently, STA adopts a coarse-to-fine mechanism to progressively separate the known and unknown data without any manually set threshold [72].

For the semantic recovery tasks, we implement a source-only trained neural network, and two zero-shot learning methods, ABP [175] and TF-VAE [88] under our setting, as baselines. The source-only model is a fully-connected neural network trained with only source domain ResNet-50 [35] features available, which learns a projector mapping the visual features to semantic attributes. ABP trains a conditional generator mapping the class-level semantic features and Gaussian noise to visual features [175]. TF-VAE proposes to enforce semantic consistency at all training, feature synthesis, and classification stages [88]. Besides, both ABP and TF-VAE are able to handle generalized zero-shot learning problems given the semantic attributes from the whole target label space. We also report ABP* and TF-VAE*, which take extra the semantics of unseen target categories as inputs.

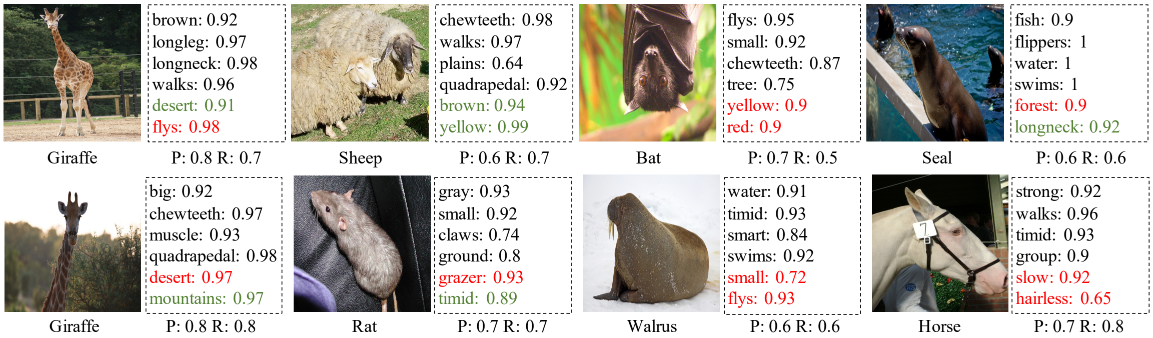
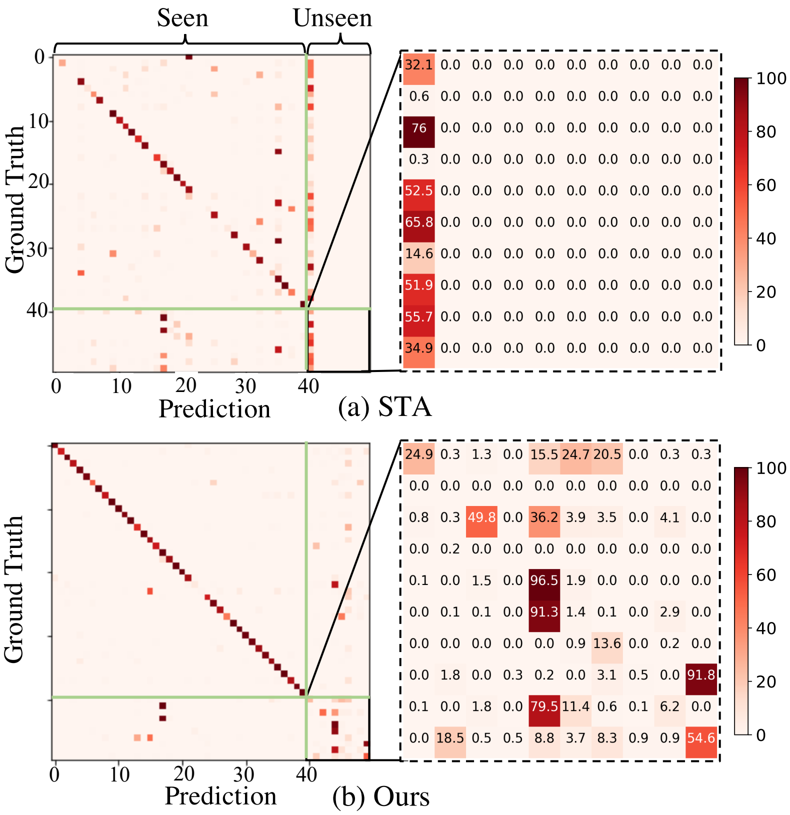


Figure 3.4: Selected samples from AwA2 dataset and attributes predicted by our method. The black ones are correctly predicted attributes, red ones are wrong predictions, and the green ones are wrong predictions but reasonable for the specific instance. "P" and "R" denote precision and recall of the attributes prediction for each sample, respectively.

3.4.2 Algorithmic Performance

Table 3.3 shows the open-set domain adaptation accuracy on *D2AwA* and *I2AwA*. From the results, we observe that our proposed method outperforms all compared baselines in terms of overall accuracy on most tasks. Especially on task A→R, our model improves 10.4% over the second-best compared method. The significant improvements come from our effective framework and the extra source semantic information. Note that in the classical open-set domain adaptation, none of the semantic attributes are leveraged. For fair comparisons, we provide the initialized results based on the visual features reported as "Ours(Init)" and further implement another variant of our method with only visual features available for training, denoted as "Ours(Vis)". The performance decrease of "Ours(Vis)" proves the contribution and effectiveness of the semantic attributes for the open-set domain adaptation. Moreover, our proposed method reaches promising results on the unseen classes while keeping performance on the seen classes for all tasks. For example, STA achieves the best overall accuracy on task P → A, but completely fails on the unseen categories and overfitting to the seen classes. Such an observation emphasizes the superiority of our method in exploring target domain unseen categories.

Table 3.4 show the semantic recovery accuracy on *D2AwA* and *I2AwA*, respectively. Within the expectation, all ZSL methods fail to recognize the data from unseen categories and overfit to the seen classes due to a lack of capacity on tackling the open-set setting. Our proposed method achieves promising results in recognizing both seen and unseen categories, e.g., our method achieves 37.8%



32.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
76	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
52.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
65.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
51.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
55.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
34.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

(a) STA

24.9	0.3	1.3	0.0	15.5	24.7	20.5	0.0	0.3	0.3
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.8	0.3	49.8	0.0	36.2	3.9	3.5	0.0	4.1	0.0
0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.1	0.0	1.5	0.0	96.5	1.9	0.0	0.0	0.0	0.0
0.0	0.1	0.1	0.0	91.3	1.4	0.1	0.0	2.9	0.0
0.0	0.0	0.0	0.0	0.0	0.9	13.6	0.0	0.2	0.0
0.0	1.8	0.0	0.3	0.2	0.0	3.1	0.5	0.0	91.8
0.1	0.0	1.8	0.0	79.5	11.4	0.6	0.1	6.2	0.0
0.0	18.5	0.5	0.5	8.8	3.7	8.3	0.9	0.9	54.6

(b) Ours

Figure 3.5: Confusion matrix of target samples from I2AwA. (a) shows the results of STA and (b) lists ours. The unseen classes are zoomed in for better visualization.

accuracy for unseen class data while keeping 95.2% performance on seen classes for task R→A. Moreover, our proposed method even outperforms the ABP* and TF-VAE*. They have access to both the seen and unseen categorical attributes in the source and target domains, while our method only employs the seen category attributes information in the source domain.

3.4.3 In-Depth Factor Exploration

In this subsection, we first visualize the representation from our model, explore the ablation study of the proposed method, showcase several representative samples with the predicted attributes and finally provide more details on the seen and unseen target categories by confusion matrix.

Representation Visualization. We show the t-SNE embeddings of *I2AwA* from different models in Figure 3.2, where red circles denote source data, and blue and gray triangles denote target domain seen and unseen classes, respectively. The embedding of our method shows that the same class samples across domains are more compact while discriminative inter classes than the representation produced by source only ResNet-50 [35] and STA [72]. Moreover, our embedding shows the joint visual-semantic representations with more discriminative distribution and separates the unseen categories from seen classes more clearly. Such an observation demonstrates the effectiveness of the semantic attributes, which is not only beneficial to the unseen categories but also promotes the quality of features of the seen classes.

Ablation Study. We dive into our complete method and several variants for open-set domain adaptation and semantic recovery tasks to understand the contribution of each specific design in our framework. As shown in Figure 3.3, we have the following observations. (1) Compared to w/o R which removes the structure-preserving partial alignment term $\mathcal{L}^R$, our method achieves significant performance gains on the open-set domain task, especially for the seen categories. This demonstrates the effectiveness of aligning the source data to the target domain while preserving the target data's structural characteristics. (2) Our method improves the performance D on both tasks compared to w/o, which removes the binary classifier $D(\cdot)$ and only uses classifier $C(\cdot)$ to recognize seen/unseen categories. We conclude that the binary classifier can refine the separation of seen and unseen classes. (3) By removing the attributes propagation mechanism, the performance w/o decreases significantly on the semantic recovery tasks, especially for the unseen categories, proving the contribution of attributes propagation for semantic recovery tasks and uncovering unseen classes. (4) Our method outperforms the variant without constructing visual-semantic fusion w/o VS, which only uses visual features for prediction. For both open-set domain adaptation seen classes and semantic recovery unseen classes, validating the effectiveness of semantic knowledge to the visual features in both preserving performance on seen classes and exploring unseen categories.

Qualitative Demonstration. To qualitatively illustrate the effectiveness of our method in discovering novel classes and recovering missing semantic information, we further show several representative samples from the target domain unseen categories on *I2AwA* in Figure 3.4. For each sample, we show some of the correct and wrong predicted attributes with corresponding prediction probabilities. "P" and "R" indicate the precision and recall score of predicting attributes of each sample. Moreover, some predicted attributes are wrong for the corresponding category but reasonable for the specific image. From the results, we demonstrate the ability of our model in transferring semantic knowledge from the source domain into the target data and discovering novel classes through missing semantic information recovery.

Confusion Matrix. We visualize the confusion matrix of STA and our method on *I2AwA* in Figure 3.5. STA only recognizes those target samples from unseen categories as unknown. On the contrary, our proposed method can discover novel categories in the target domain. Surprisingly, the accuracy of our method for the category "Giraffe" achieves 96.5%. Moreover, we also notice that not just benefiting uncover unseen categories, but our method also enhances the accuracy of the seen classes compared to STA.

3.4.4 Discussion and Limitation

In our study, we delved into a novel problem called "Semantic Recovery Open-Set Domain Adaptation" (SR-OSDA) and presented an efficient solution that tackles both domain adaptation and the discovery of novel categories in the target domain. Nonetheless, we acknowledge some limitations and challenges in our approach. Specifically, we observed that the proposed solution heavily relies on the initialization of clustering results from the target domain data, which plays a crucial role in both domain adaptation and semantic recovery. Additionally, we recognize the need for more effective and promising evaluation metrics to better measure the classification performance and the quality of the recovered attributes in this new problem.

3.5 Conclusion

We addressed a novel and practical *Semantic Recovery Open-set Domain Adaptation* problem, which aimed to discover target samples from classes unobserved in the source domain and interpreted based on recovered semantic attributes. To this end, we proposed a novel framework consisting of structure preserving partial alignment, attributes propagation via visual graph, and task-driven classification over joint visual-semantic representations. Finally, two semantic open-set domain adaptation benchmarks were constructed to evaluate our model in terms of open-set recognition and semantic attribute recovery.

[49] Jing, Taotao, Haifeng Xia, Renran Tian, Haoran Ding, Xiao Luo, Joshua Domeyer, Rini Sherony, and Zhengming Ding. "InAction: Interpretable Action Decision Making for Autonomous Driving." In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVIII, pp. 370-387. 2022.

4

Interpretable Decision-Making with Learnable Visual Prototypes

Autonomous driving has attracted interest in interpretable action decision models that mimic human cognition. Existing interpretable autonomous driving models explore static human explanations, ignoring the implicit visual semantics that are not explicitly annotated or consistent across annotators. In this work, we propose a novel Interpretable Action decision-making (**InAction**) model to provide an enriched explanation from both explicit human annotation and implicit visual semantics. First, a proposed visual-semantic module captures the region-based action-inducing components from the visual inputs, which learns the implicit visual semantics to provide a human-understandable explanation in action decision-making. Second, an explicit reasoning module is developed by incorporating global visual features and action-inducing visual semantics, which aims to jointly align the human-annotated explanation and action decision-making. Experimental results on two autonomous driving benchmarks demonstrate the effectiveness of our **InAction** model for explaining both implicitly and explicitly by comparing it to existing interpretable autonomous driving models. The source code is available at https://github.com/scottjingtt/InAction.git.

4.1 Background and Summary of Contribution

Deep learning has recently accelerated the progress of autonomous through remarkable success in computer vision tasks. Existing driving action decision systems can primarily be recognized to be in two major groups, one is the *pipelined* framework [157] and the other is *end-to-end* system [52], [146], [53], [135], [136], [124]. Specifically, pipelined systems decompose the problem into a series of smaller tasks, such as pedestrian trajectory planning and object detection. The final driving action decision is made by relying on the performance of all the modules designed for the sub-tasks. However, pipelined systems are vulnerable to inaccuracies in each sub-task module, which may cause the entire system to perform unreliably if the interactions between modules are ignored. On the contrary, end-to-end systems take advantage of the entire visual scene to directly predict driving action, avoiding the loss of information caused by the intermediate decisions adopted in pipelined systems.

Unfortunately, most end-to-end systems are complex deep neural network models, performing as a black box with opaque reasoning for human interpretation. In safety-critical domains, such as autonomous driving and medical diagnosis, building a transparent and interpretable learning model has recently attracted attention beyond the performance alone [110]. Various interpretation strategies have been explored to explain learning models, e.g., part-based methods [170], [173], saliency maps [2], [28], [172], activation maximization to visualize neurons [90], [92], deconvolution/upconvolution to explain layers [24], [158]. However, such post-hoc methods give a superficial understanding of the black box models, rather than being a comprehensive interpretable system [110]. Alternatively, prototypical visual explanations are incorporated in deep network architecture for intrinsic interpretation and case-based reasoning [11], [111], [89], [83]. Most prior prototype-based work explicitly explores the presence of prototypical parts, which are utilized to recognize objects. However, such strategies ignore the notion of spatial relationships, which is crucial for tasks like driving decision making with complicated context and multiple objects.

For interpretable autonomous driving decision-making, Xu *et al.* [**?**] proposed a new paradigm to predict driving action based on finite action-inducing objects and generated a set of potential explanations in a multi-task fashion. Unfortunately, there are four major limitations of this work from an interpretability perspective. First, although the multi-task framework is supervised by both driving action and a human-defined explanation, the proposed model does not interpret the reasoning process of the prediction for black-box model. Second, the proposed BDD-OIA dataset annotates the reasons for action into 21 explanations; however, it is impractical that the human-defined finite explanation set can cover all possible scenarios considering the complex scene context and objects input for autonomous driving action prediction tasks. For example, the explanation set in the BDD-OIA dataset recognizes "obstacles on the right lane" as a reason for "cannot turn right", which is not accurate since different distances and locations of the obstacles could lead to different decisions for drivers. Moreover, the logical reasoning process from the explanation to the driving action decision is ambiguous, especially under a multi-label setting where all possible actions are annotated. For instance, we notice that the proposed model predicts two explanations "traffic light is green" and "obstacle: car", but still predicts the action as "forward", without any reasoning about how the predicted explanation results in the action prediction. Last but not least, OIA estimates the driving decision only based on the last frame of the observed sequence, ignoring the temporal information.

In this work, we propose a novel Interpretable Action decision-making (**InAction**) to provide reasoning of action prediction from both explicit human annotation and implicit visual semantics (Figure 4.1). Generally, we consider the explanation for action decisions from two perspectives to compensate for the limitations of each method: existing human-annotated interpretation and AI-based implicit visual hints. To sum up, our contributions are in three areas:

- First, we propose an inherently interpretable reasoning framework for autonomous driving action prediction from both implicit visual semantics and explicit human annotation perspectives.

- Second, the proposed *Implicit Visual-Semantic Interpretation* module interacts with the *Ex-*

plicit Human-like Reasoning module by revealing action-inducing concepts, and the learned implicit and explicit explanations compensate for the limitations of each other in predicting the action decision.

- Finally, experimental results on two interpretable autonomous driving benchmarks demonstrate the effectiveness of the proposed model by comparing it with existing models showing enriched interpretation and reasoning.

4.2 The Proposed Framework

4.2.1 Motivation

For autonomous driving, beyond pursuing high performance, interpretability is needed for safety-critical domains [93], [165]. This aims to imbue autonomous vehicles with reasoning abilities similar to human drivers. Existing efforts mainly adopt human-annotated explanations to guide system learning and generate human-understandable reasoning given the video inputs [**?**], which skews the model towards human annotation.

Unfortunately, human annotation has some drawbacks like insufficient explanation and inconsistent reasoning. Insufficient explanation means there are always implicit visual semantics not annotated by finite human-defined explanation sets, which cannot be easily tracked through an end-to-end system with visual inputs and explanation outputs. Inconsistent reasoning is particularly challenging since different people have different explanations, especially for complicated scenarios, leading to biases and insufficiency of the ground-truth annotation.

Motivated by this, we explore both the implicit visual-semantic interpretation and explicit human annotation jointly and propose the Interpretable Action decision-making model (**InAction**), whose goal is to enhance transparency and interpretability for autonomous driving action decision-making.

4.2.2 Framework Architecture

An overview of the proposed InAction framework is shown in Figure 4.1. The model consists of a convolutional backbone $G(\cdot)$, and two interpretable action prediction modules—an implicit visual-semantic module and an explicit human-annotated reasoning module—to predict driving action and

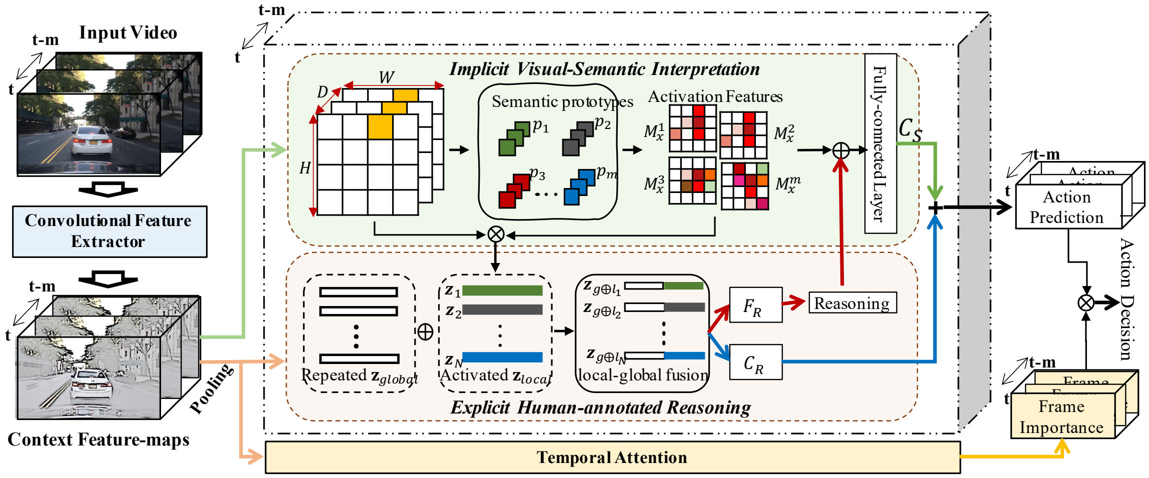


Figure 4.1: Illustration of the proposed framework.

reasoning of the decision from different perspectives. Specifically, the implicit visual-semantic module is denoted as $G_S(\cdot)$, which takes the feature map per frame extracted by convolutional backbone as input to discover action-inducing concepts and the presence of learned semantic prototypes as visual cues for following prediction. For the explicit reasoning module, global visual features and the discovered action-inducing local regions are fused and input to two multi-task classifiers, predicting the driving action and human-annotated explanations, denoted as $C_R(\cdot)$ and $F_R(\cdot)$, respectively. Finally, the learned prototypical visual cues and predicted human-annotated explanations are fused and input to a fully-connected layer without bias as the action predictor, denoted as $C_S(\cdot)$. For the input video sequence, such prediction is applied to each frame, with a temporal attention layer employed to explore the contribution of each frame.

Mathematically, given an input video with m frames, $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{m}$, whose action label as $\mathbf{y}_a \in \mathbf{A}$ and human annotated explanation $\mathbf{y}_e \in \mathbf{E}$, where $C_{\text{act}} = |\mathbf{A}|$ and $C_{\text{exp}} = |\mathbf{E}|$ are the numbers of categories of actions and human-annotated explanations, respectively. For each frame $\mathbf{x}$, the convolutional backbone extracts the feature map $\mathbf{f} = G(\mathbf{x})$ with shape $H \times W \times D$, where W and H denote the width and height, respectively, and D is the number of channels. For the clarity of description, denoting all the patches in the feature map as $\mathbf{Z}_{\mathbf{x}} = \{\mathbf{z}_i \in \mathbf{f}\}_{i=1}^{HW}$, and the shape of each patch $\mathbf{z}_i$ is $\mathbb{R}^{D\times1\times1}$. The implicit visual-semantic module will slide over the whole feature map and calculate the activation scores for all patches in the feature map with respect to the presence of learned semantic prototypes. On the one hand, those regions primarily activated corresponding to specific prototypes are selected as action-inducing semantic regions and are fused with the global features to predict the action and explicit human-annotated explanation. On the other hand, the limitations of the activation map will be compensated by the predicted human-annotated explanations for the action prediction.

Implicit Visual Semantic Interpretation

To explore the action-inducing local regions in the visual input, we assign m_k semantic prototypes for each action class k, resulting in $m = m_k \times C_{\text{act}}$ prototypes in total, making up the visual-semantic layer $\mathbf{P} = \{\mathbf{P}_k\}|_{k=1}^{C_{\text{act}}}$, in which $\mathbf{P}_k = \{\mathbf{p}_j\}|_{j=1}^{m_k}$, and $\mathbf{p}_j$ denotes the semantic visual prototypes to be learned for predicting action class k. Given the convolutional output feature map $\mathbf{Z_x}$ and prototype $\mathbf{p}_j$, the visual-semantic layer will go though all patches $\mathbf{z}_i \in \mathbf{Z_x}$ of the feature map to compute the activation score between them:

$$s_{ij} = \log\Big(\frac{\|\mathbf{z}_i - \mathbf{p}_j\|^2 + 1}{\|\mathbf{z}_i - \mathbf{p}_j\|^2 + \varepsilon}\Big), \tag{4.1}$$

where ε is a small positive value, and the activation score s_{ij} represents how strongly a semantic prototype is presented in the specific region of the input frame. The activation scores of all the patches in the feature map produce an activation heat map $\mathbf{M}^j_\mathbf{x}$ with shape $H \times W$, identifying how similar each part of the input frame is to one specific prototype $\mathbf{p}_j$. Calculating activation maps for all prototypes results in an activation feature set $\mathbf{M_x} = \{\mathbf{M}^j_\mathbf{x}\}_{j=1}^m, \mathbf{M}^j_\mathbf{x} \in \mathbb{R}^{H\times W}$.

Intuitively, the most important patches for making action decisions should be clustered around semantically similar prototypes of each specific action category, and the clusters centered on prototypes from different action categories are well separated. Thus, we also adopt a discriminative prototype learning loss as:

$$\mathcal{L}_d = \lambda_1 \mathbb{E}_{\mathbf{x}\in\mathbf{X}} \min_{\mathbf{p}_j \in \mathbf{P}_{\mathbf{y}_a}} \min_{\mathbf{z}\in\mathbf{Z_x}} \|\mathbf{z} - \mathbf{p}_j\|^2 - \lambda_2 \mathbb{E}_{\mathbf{x}\in\mathbf{X}} \min_{\mathbf{p}_j \notin \mathbf{P}_{\mathbf{y}_a}} \min_{\mathbf{z}\in\mathbf{Z_x}} \|\mathbf{z} - \mathbf{p}_j\|^2, \tag{4.2}$$

where λ_1 and λ_2 are two hyper-parameters determining the contributions of the two loss terms. Minimizing $\mathcal{L}_d$ encourages that every input frame at least has one prototype from its own action strongly activated in one of its latent feature map patches while maximizing the distances between the patches and the prototypes from different classes. Such an optimization objective shapes the latent space into a semantically meaningful clustering structure.

Explicit Human-annotated Reasoning

Compared to implicit region-based action-inducing prototype searching, human-annotated reasoning explains the driving decision in a more intuitive and abstract way. Normally natural language annotation involves temporal and spatial knowledge from visual inputs, which provides a more high-level explanation of the decision-making. Intuitively, such an explanation includes the global scene understanding and corresponding action-inducing objects.

Inspired by OIA [**?**], we propose an Explicit Human-annotated Reasoning module in a multi-task fashion to jointly generate human-annotated explanations and predict action. Specifically, for all the patches in the extracted feature map, we select top-N patches that activate any one of the prototypes assigned to the same action class as the action-inducing local components, denoted as $\mathbf{Z}_{local} = \{\mathbf{z}_l\}_{l=1}^{N}$, where $\mathbf{z}_l \in \mathbf{Z}_{\mathbf{x}}$. The activation scores denote the importance of such patches contributing to the action decision-making. It is noteworthy that the action-inducing local components $\mathbf{Z}_{local}$ are the presence of specific learned semantic prototypes, thus are not limited to be objects detected by the pre-trained object detection backbone, which is one of the limitations of OIA [**?**]. The selected top-N most activated patches can represent various scene contexts, and environmental information, in addition to human-defined objects. Furthermore, we consider that the global feature map provides an overall understanding of the visual input and the information like environmental status, e.g., “Road is clear”, and agent relationship, e.g., “There is a vehicle parking on the right”. In this sense, the local action-inducing components are concatenated with the global features, then input into the action predictor $C_R(\cdot)$ and human-annotated explanation predictor $F_R(\cdot)$.

Specifically, the global feature map $\mathbf{Z}_{\mathbf{x}}$ is processed with global average pooling and represented as a feature vector with the same dimension as each local patch $\mathbf{z}_l$, denoted as $\mathbf{z}_{global}$. Every local patch $\mathbf{z}_l$ is concatenated with the global feature $\mathbf{z}_{global}$ producing the local-global fused feature $\mathbf{Z}_{g \oplus l} = \{\mathbf{z}_l \oplus \mathbf{z}_{global}\}_{l=1}^{N}$, where $\mathbf{z}_l \in \mathbf{Z}_{local}$, and $\oplus$ is concatenation operation. The local-global feature is further vectorized and then input to the following action and explanation prediction networks, optimizing the important local components that are highly associated with both action and explanation prediction. Eventually the predicted action and explanation are denoted as $\hat{\mathbf{y}}_a^R$ and $\hat{\mathbf{y}}_e^R$, respectively.

Considering the possible action decisions, we can explore making a prediction with only one action or more than one action. If more than one action can be made, which is for a multi-label prediction task, the prediction logits are normalized by sigmoid function to the range between 0 and 1. If only one action can be made, which is a multi-class single-label task, the prediction logits are normalized by softmax function. Therefore, we formulate the multi-task learning objective of the explicit reasoning module as:

$$\mathcal{L}_r = L(\mathbf{y}_a, \hat{\mathbf{y}}_a^R) + L(\mathbf{y}_e, \hat{\mathbf{y}}_e^R), \tag{4.3}$$

where $L(\cdot,\cdot)$ denotes the cross-entropy loss and binary cross-entropy loss for single-label and multi-label prediction tasks, respectively.

Interpretable Decision Prediction So far, we design two kinds of explanations, i.e., $\mathbf{M}_\mathbf{x}$ and $\hat{\mathbf{y}}_e^R$, for the decision making from two different perspectives. In order for these two explanations to interact and compensate for one another, the concatenated explanation vector $\hat{\mathbf{y}}_e = [\mathbf{M}_\mathbf{x}, \hat{\mathbf{y}}_e^R]$ is exploited to a fully-connected layer $C_S(\cdot)$ to predict the action decision $\hat{\mathbf{y}}_a^S = C_S(\hat{\mathbf{y}}_e)$.

It is noteworthy that driver action decision-making has more complicated scene contexts with many different agents, which is different from other prototype-based interpretable object recognition only considering the presence of some specific prototypical parts [11], [89], [111], [83]. Thus, the learned semantically meaningful prototypes that contribute to the final decision could be a part of or a complete object, even a set of objects or an environment region, in the input frame. Moreover, the location of a specific prototype, and the relationships between it with other objects and the environment, play crucial roles in determining the final action. Thus, rather than only choosing the maximum activation score for each prototype in the corresponding activation heat map, the whole activation feature set is considered for the fully-connected layer $C_S(\cdot)$ to integrate the spatial and relationship knowledge for predicting the action decision.

Similarly, we consider single-label and multi-label tasks with different activation functions, and the learning objective of action prediction is defined as:

$$\mathcal{L}_s = L(\mathbf{y}_a, \hat{\mathbf{y}}_a^S), \tag{4.4}$$

where $L(\cdot,\cdot)$ represents cross-entropy loss for multi-class single-label tasks, while it is the binary cross-entropy loss for multi-label prediction tasks.

Cross-module Fusion and Temporal Aggregation Two action decision predictions $\hat{\mathbf{y}}_a^R$ and $\hat{\mathbf{y}}_a^S$ are obtained with different input knowledge. The former is based on visual features, while the latter is based on explored explicit-and-implicit explanations. Thus, we accept two prediction logits followed by the specific activation function for multi-label or single-label problems, making the final aggregated action prediction, which is denoted as $\hat{\mathbf{y}}_a = \hat{\mathbf{y}}_a^R + \hat{\mathbf{y}}_a^S$.

Moreover, for the video input $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^m$ with m frames, we make the decision prediction for each frame $\mathbf{x}_i \in \mathbf{X}$, resulting in a sequence of predictions $\{\hat{\mathbf{y}}_a^1, \ldots, \hat{\mathbf{y}}_a^m\}$. To find the most relevant information (key frames) in the observed sequence, a temporal attention layer is developed with a fully-connected layer followed by the Softmax activation function, generating the importance δ_i for each frame $\mathbf{x}_i$. The objective with a temporal attention layer is defined as:

$$\mathcal{L}_t = L(\mathbf{y}_a, \sum\nolimits_{i=1}^m \delta_i \hat{\mathbf{y}}_a^i), \tag{4.5}$$

where $L(\cdot,\cdot)$ is cross-entropy loss or binary cross-entropy loss for single-label and multi-label prediction tasks, respectively.

Overall Objective. To sum up, we integrate two explanation modules into our unified framework and formulate the overall optimization objective as follows:

$$\mathcal{L} = \mathcal{L}_d + \mathcal{L}_r + \mathcal{L}_s + \mathcal{L}_t, \tag{4.6}$$

which includes two action decision classifiers and one explicit explanation predictor, and these two action decision classifiers will compensate for each other as they are based on different knowledge. In the test stage, we fuse the two predictions of action decisions to obtain a more robust output.

4.3 Experiments

4.3.1 Experimental Setup

Pedestrian Situated Intent (PSI) dataset [13] contains 110 about 15 seconds long videos with 30 fps, and each is annotated with one of three speed change actions (“maintain speed”, “slow

Table 4.1: Statistics of BDD-OIA and PSI dataset

Dataset	Action	# Frame	# Reasoning
BDD-OIA [?]	Forward	12,491	21 [Human-defined]
	Stop/Slow Down	10,432	
	Turn Left	5,902	
	Turn Right	6,541	
PSI [13]	Maintain Speed	5,800	29 [*k*-means clustered]
	Slow Down	4,925	
	Stop	1,177	

down", and "stop") on frame level. The reasoning of the action decision is described in natural language, which will be used as explanation knowledge in our experiments. We split all videos into train/validation/test set with a ratio of 75%/5%/20%. We sample the tracks with a length of 15 frames, and the overlap ratio is 0.8 while predicting the 16^{th} frame's action and explanation. Samples in the PSI dataset are assigned one single label out of three actions, so we evaluate the model by overall prediction accuracy and class-wise average accuracy for action prediction.

The original explanations are sentence-based, and each sentence contains descriptions of environmental context and human behaviors. We first split the original sentences into segments reflecting the environmental context or human behaviors. A syntactic dependency tree is applied to generate the dependency tagging of words, and then a set of heuristic rules are adopted to group each sentence into segments. Afterward, the pre-trained BERT [19] is used to generate embeddings for all segments. The embedding of each segment is generated by averaging the embeddings of the words within the sentence segment. Consequently, we apply k-means clustering to obtain k semantic categories ($k = 29$ in our experiment). Given an explanation, since it is split into multiple segments and each might belong to different semantic categories, we generate k binary labels for each explanation to represent its semantics. For the human-annotated explanation, we report the overall F1 score and class-wise mean F1 score.

BDD-OIA dataset [?] is a subset of BDD100K [155] consisting of 22,924 5-second video clips, which were annotated with 4 action decisions ("move forward", "stop/slow down", "left turn", and "right turn") and 21 human-defined explanations. Specifically, each video contains at least 5 pedestrians or bicycle riders and more than 5 vehicles. The videos are collected with complex driving scenes to increase the scene diversity. Following the setting of [?], only the final frame of each video clip is used thus the temporal attention layer is neglected. As there are multiple possible action

Table 4.2: Single-label action and multi-label explanation prediction on PSI dataset

Method	Maintain	Slow	Stop	act. Acc$_{all}$	act. mAcc	exp. F1$_{all}$	exp. mF1
OIA-global[**?**]	0.540	0.774	0.537	0.635	0.617	0.178	0.119
OIA [**?**]	0.693	0.622	0.463	0.643	0.593	0.189	0.110
Ours-f	0.703	0.771	0.641	0.719	0.704	0.277	0.203
Ours-v	**0.717**	**0.776**	**0.672**	**0.734**	**0.722**	**0.285**	**0.223**

choices for each sample, we evaluate the performance by F1 score for each specific action, overall F1 score, and the class-wise average F1 score for both action and explanation prediction.

More statistics of the benchmarks are shown in Table 4.1.

Implementation Details. The Faster R-CNN [105] is pre-trained on the annotated images from BDD100K [155] and set as the backbone, which is followed by two 3×3 convolutional layers generating the global feature map with shape $7 \times 7 \times 256$ for each input frame. For the implicit visual semantic interpretation module, we assign $m_k = 6$ prototypes with dimension 128 for each action class, resulting in $m = 24$ prototypes for the BDD-OIA dataset, and $m = 18$ prototypes in total for the PSI dataset. For our InAction model, we set $N = 10$ thus the $top - 10$ patches from the input feature map with the smallest distances compared to all semantic prototypes are selected to be fused with the global features for explicit human-annotated explanation and action prediction. The feature map is input to two additional 1×1 convolutional layers to reduce the channel dimension to be the same as the prototype dimension and normalized by sigmoid function following [11] before calculating the activation scores. The action predictor $C_S(\cdot)$ based on the fused explanation vector is one fully-connected layer without bias. We follow the same strategy of [11] to initialize and train the model. For the explicit human-annotated reasoning module, the action decision predictor $C_R(\cdot)$ is a three-layer fully-connected neural network, and the explanation predictor $F_R(\cdot)$ is a two-layer fully-connected neural network. ReLU activation is used for all hidden layers. The model is optimized by Adam optimizer with learning rate initialized as 10^{-3}, and decayed by 0.1 every 10 epochs. For simplicity, we set $\lambda_1 = 0.1$ and $\lambda_2 = 0.01$ by default for all experiments. We empirically fix $m_k = 6$, and we observe the results are not sensitive to it if $m_k > 3$ on the validation set.

Table 4.3: Multi-label action and explanation prediction on BDD-OIA dataset

Method	F	S	L	R	act. $F1_{all}$	act. mF1	exp. $F1_{all}$	exp. mF1
Res-101[?]	0.755	0.607	0.098	0.108	0.601	0.392	0.331	0.180
OIA[?]	**0.829**	**0.781**	**0.630**	**0.634**	**0.734**	**0.718**	0.422	0.208
OIA*[?]	0.792	0.742	0.594	0.627	0.705	0.689	0.501	0.293
Ours(proposals)	0.795	0.743	0.597	0.613	0.706	0.687	0.558	0.332
Ours(global)	0.800	0.747	0.612	0.619	0.714	0.694	**0.565**	**0.347**

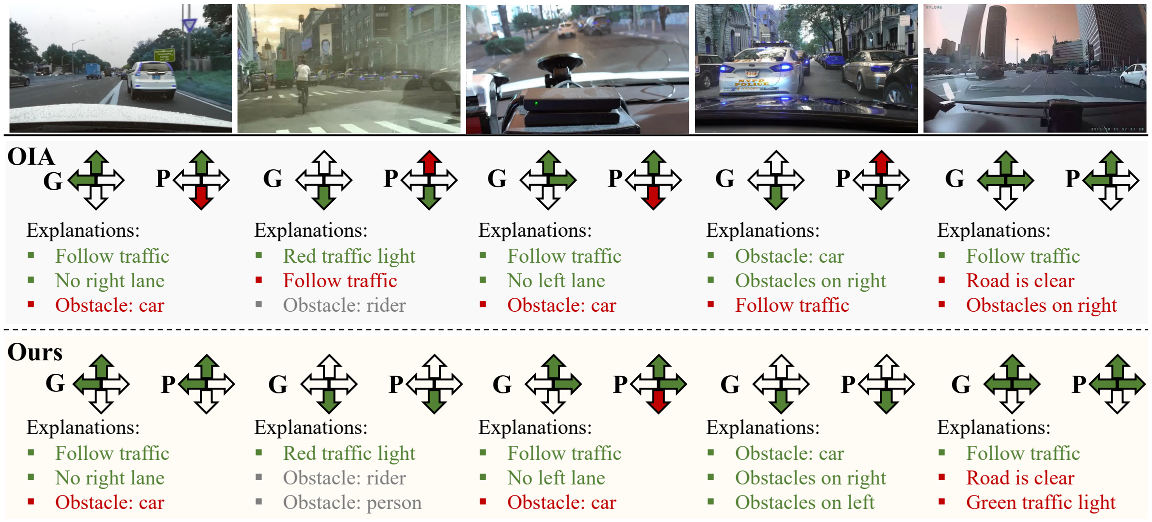


Figure 4.2: Selected comparison examples of action and explicit explanation prediction between OIA and InAction on BDD-OIA dataset. G denotes the ground-truth annotation, and P shows the predicted result from OIA/Ours. green predictions are True Positive, red are False Positive, and gray are False Negative.

4.3.2 Comparison Results

We compare our proposed InAction model with the OIA method [?] on the PSI and BDD-OIA datasets, and the results are reported in Table 4.2 and Table 4.3. OIA model only adopts the last frame of a sequence as input, thus we report two results produced by our model with only the last frame or the whole observed video sequence as input, denoted as Ours-f and Ours-v in Table 4.2, respectively. For experiments on BDD-OIA in Table 4.3, we reproduce the OIA model based on the official implementation released by the author, denoted as OIA*, in addition to the results reported by OIA [?]. The reproduced results of OIA on BDD-OIA are lower in action decision while better in explanation in terms of F-1 score, compared with the reported OIA. Note that OIA adopts the detected proposals generated by the backbone as local features. We utilize the implicit visual-semantic prototypes learned from the global feature map and from the detected proposals, and report the re-

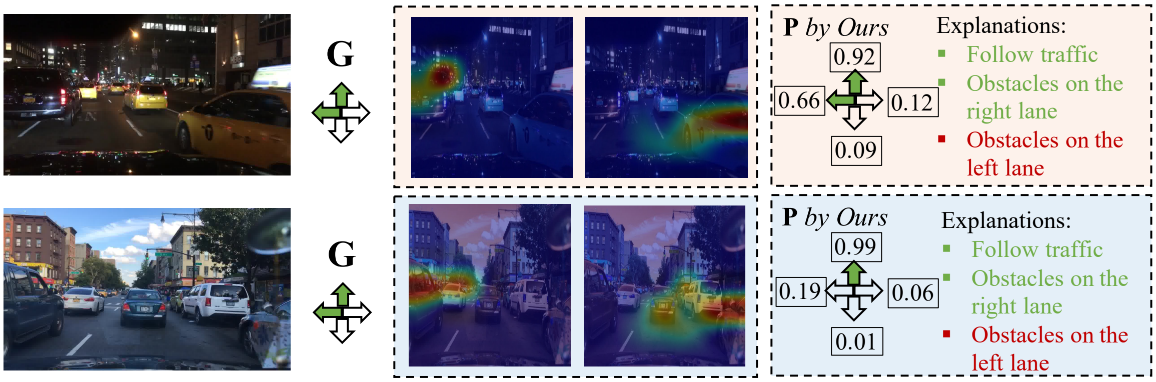

G
G
P by Ours
0.92
0.66
0.12
0.09
Explanations:
Follow traffic
Obstacles on the right lane
Obstacles on the left lane
P by Ours
0.99
0.19
0.06
0.01
Explanations:
Follow traffic
Obstacles on the right lane
Obstacles on the left lane

Figure 4.3: Comparison of explanations produced by the implicit visual-semantic module and the explicit human-annotated reasoning module for examples on BDD-OIA.

sults as Ours(global) and Ours(proposals), respectively. Specifically, to obtain Ours(proposals), we extract the top-100 detected proposals features after the average pooling process into the same size as the learned prototypes, then follow the same fusing strategy as aforementioned.

For the PSI results (Table 4.2), we notice that our proposed InAction model with only the last frame as input outperforms OIA around **0.07** and **0.01** for the overall and class-wise mean action prediction accuracy, respectively. When the whole video sequence is input to our model, the performance is improved further by 0.015 and 0.018, respectively, demonstrating our model can benefit from the temporal knowledge from the input sequence. The PSI dataset has an imbalanced distribution and there are much fewer samples belonging to the category “Stop”, thus both OIA-global and OIA* obtain worse performance in this category compared to “Main speed” and “Slow down”. Surprisingly, our model is able to achieve better performance on this decision. Moreover, as OIA adopts both global and local detection proposals as input for prediction, while InAction only uses the global feature map we compare our model with another baseline OIA-global, which has the same architecture with OIA excluding the local proposal branch. From the results, we observe that OIA-global obtains worse overall performance compared to OIA and InAction.

From the BDD-OIA results (Table 4.3), we observe InAction can improve the action prediction performance compared to the reproduced OIA. For the reason prediction, we notice that the reproduced results outperform the numbers reported in the OIA paper by around 0.08, and our proposed method can further improve the overall F1 and class-wise mean F1 both over **0.5**. This demonstrates that our model works well in both action prediction and explanation reasoning. Moreover, we ob-

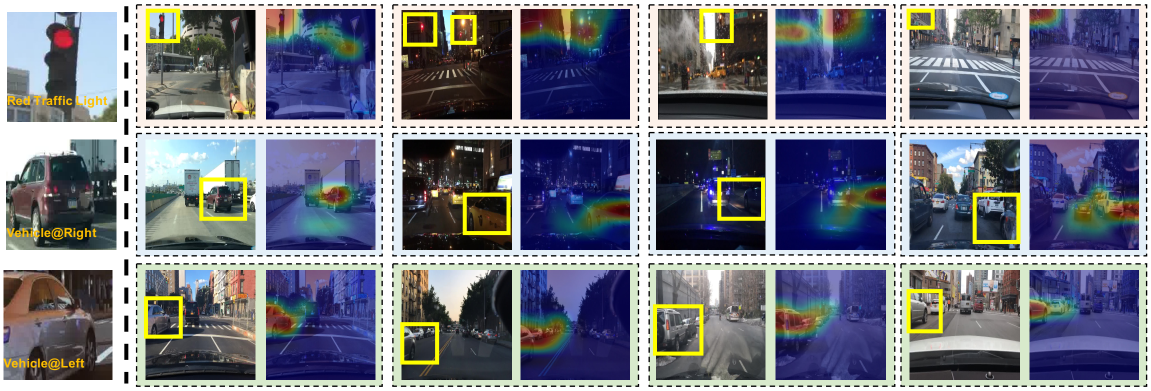


Figure 4.4: Visualizing prototypes by selecting the most similar patches from the training samples, where each row shows one explanation.

serve that the results produced with prototypes learned on the global feature maps are better than those based on the detected proposals. We argue that relying on the detected proposals will make the model fail, and constrain the representative capabilities of learned semantic prototypes, compared to exploring the implicit visual-semantic knowledge based on the whole input image.

4.3.3 Interpretability Analysis

Comparison with OIA. We present qualitative results in Figure 4.2 to demonstrate the interpretability and transparency of the propose InAction model. For the same visual input, we compare both the action and explanation prediction of OIA and our InAction. From the selected examples, we notice that OIA made wrong action predictions while InAction can achieve correct results in some cases. The only wrong prediction in the 3^{rd} example is that both OIA and the explicit human-annotated reasoning module in InAction recognize the white vehicle in front and predict the explanation as “Obstacle: car”, then make the “Stop/Slow down” decision. However, the ground-truth action annotation does not contain this label. Such an observation demonstrates that insufficient explanation and inconsistency reasoning always exists in the human-defined annotations, especially on single-frame-based prediction tasks.

Compensation between Implicit and Explicit Interpretation. In Figure 4.3, we compare the generated explanations from the implicit and explicit modules for the same task. We notice that some human-annotated reasoning is also captured by the implicit semantic prototypes, e.g., “Obstacles on the right lane”. However, some explanations discovered by the implicit visual prototypes compensate for the lack of human annotation. For example, the vehicle on the left lane in the second-row example is quite close but not annotated, and the ground-truth label is “forward” and “turn left”, while fortunately, our model notices the obstacle on the left lane and predict “forward” only.

Implicit Visual Semantics Analysis.

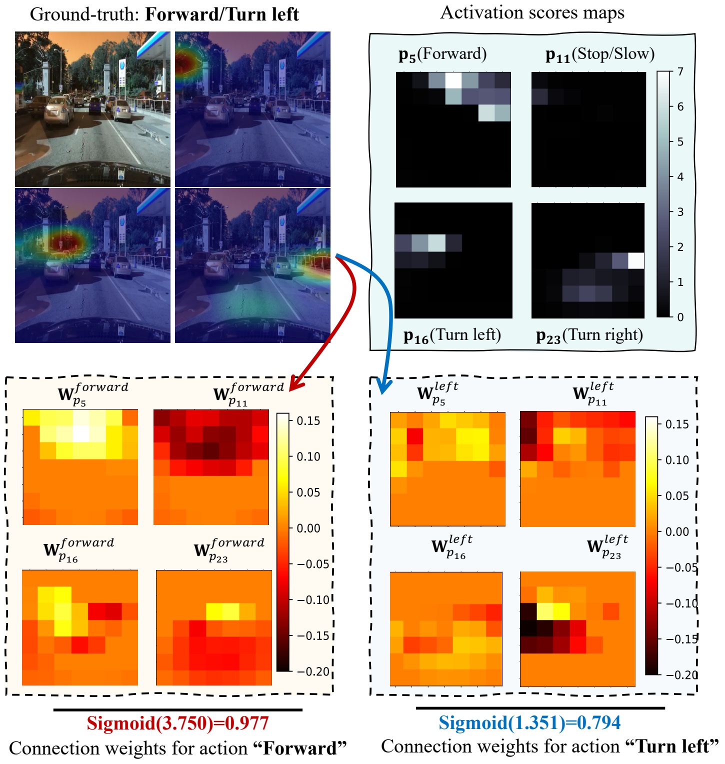

Ground-truth: **Forward/Turn left**
Activation scores maps
$\mathbf{p_5}$(Forward)
$\mathbf{p_{11}}$(Stop/Slow)
$\mathbf{p_{16}}$(Turn left)
$\mathbf{p_{23}}$(Turn right)
0
1
2
3
4
5
6
7
$\mathbf{W}_{p_5}^{forward}$
$\mathbf{W}_{p_{11}}^{forward}$
$\mathbf{W}_{p_{16}}^{forward}$
$\mathbf{W}_{p_{23}}^{forward}$
$\mathbf{W}_{p_5}^{left}$
$\mathbf{W}_{p_{11}}^{left}$
$\mathbf{W}_{p_{16}}^{left}$
$\mathbf{W}_{p_{23}}^{left}$
0.15
0.10
0.05
0.00
−0.05
−0.10
−0.15
−0.20
Sigmoid(3.750)=0.977
Connection weights for action **“Forward”**
Sigmoid(1.351)=0.794
Connection weights for action **“Turn left”**

Figure 4.5: Visualization of reasoning of the selected instance.

To illustrate the learned implicit visual semantic prototypes in an intuitive way, we visualize the prototypes via the most similar patches of images in the BDD-OIA dataset [11]. Figure 4.4 shows the selected examples with patches highly activated by specific semantic prototypes from action decisions “Stop”, “Turn left”, and “Turn right”. The most activated patch of the given input for selected prototypes is marked by bounding boxes in the original input, which represent the image patches that InAction considers focusing on corresponding to specific prototypes. From the results, we observe that when the implicit visual-semantic reasoning module slides over the whole input to obtain an activation map, these three prototypes are represented as “Red traffic light”, “Vehicle at right”, and “Vehicle at left”, respectively. Any region is strongly activated by one of the specific prototypes, or, in other words, one of the prototypes presents strongly in the input frame, will play a crucial role in the final prediction.

Reasoning Process of InAction. Prior prototype-based models only observe the most strongly activated region. However, driving action prediction has much more complicated scene context and multiple objects involved as hints, so the spatial location of each prototype presence and the relationships among different components make a crucial influence on the final decision prediction.

Figure 4.5 shows the reasoning process of our InAction predicting the action decision for a test sample, which is annotated as “Forward/Turn left”. Given the input frame, the implicit visual semantic interpretation module compares every patch in the feature map against the learned prototypes, producing the activation score maps. The activation maps that are most strongly activated by prototypes are shown as the top-right heatmaps in Figure 4.5, where $\mathbf{p}_5, \mathbf{p}_{11}, \mathbf{p}_{16}, \mathbf{p}_{23}$ are assigned to action classes “Forward”, “Stop/Slow down”, “Turn left”, and “Turn right”, respectively. Although m prototypes are assigned to C action decision resulting in m_k prototypes per class during training, all activation scores produced by m prototypes over the feature map of the input frame are multiplied by the weight matrix in the last fully-connected layer $C_S(\cdot)$ to generate the output prediction. The weights in the fully-connected layer represent the connections between prototypes and the predicted classes. In Figure 4.5, we select the weights ($\mathbf{W}$) for class “Forward” and “Turn left” corresponding to the selected prototypes, and show them after reshaping into the same shape as the activation map. From the weights over different regions of the feature map/activation map, we observe that the same prototype plays different roles for different action decisions. For example, components similar to prototype $\mathbf{p}_{11}$ appearing in the top area of the view will make a negative contribution to the prediction of “Forward”, while for the prediction of class “Turn left”, it will reduce the probability of “Turn left” only when it appears at the top-left corner, otherwise, this prototype is comparably neutral. Interestingly, the prototype shown in the first row of Figure 4.4 is prototype $\mathbf{p}_{11}$, which represents “Red traffic light”.

4.3.4 Discussion and Limitation

We presented the **InAction** model, combining explicit human annotations and implicit visual semantics to provide enriched explanations for action decision-making. Experimental results on autonomous driving benchmarks validate its effectiveness. Despite its strengths, this work also reveals certain limitations and challenges that require further investigation. For instance, the prototype-based interpretation module and the use of all activation maps for prediction result in high computational costs during training. Additionally, there were observations of redundant prototypes with minimal activation or learning during experiments. To address these issues, determining appropriate semantic prototype initialization and relationships could enhance the learning of meaningful and promising interpretations.

4.4 Conclusion

In this work, we developed a novel Interpretable Action (**InAction**) decision-making model to provide enriched explanations from both explicit human annotation and implicit visual semantics perspectives. To implement this, two interpretable modules were proposed including a visual semantic module and an explicit reasoning module. Specifically, the first module aimed to capture the region-based action-inducing semantic concepts from the visual inputs, so that our model could automatically learn the implicit visual cues to provide a human-understandable explanation. The second module attempted to benefit from the human-annotated reasoning for action decision-making so that our model was able to provide a more high-level interpretation by aligning visual inputs to human annotations. Experimental results on two autonomous driving benchmarks demonstrated the effectiveness of our **InAction** model.

Jing, Taotao, Haifeng Xia, Hongfu Liu, and Zhengming Ding. "Interpretable Novel Target Discovery in Open-set Domain Adaptation." under review.

5

Interpretable Novel Target Discovery with Multimodal Semantic Knowledge

Open-set domain adaptation (OSDA) considers a special domain adaptation problem in which the target domain contains novel categories never appear in the well-labeled source domain. Unfortunately, prior efforts on OSDA simply detect and recognize all novel categories as one "unknown" group without further exploration. The demand for exploring these novel categories prompts us to consider the underlying multi-class structure and semantic description of those unknown categories in more detail. In this work, we propose a novel interpretable framework to accurately identify the seen categories in the target domain and effectively recover the semantic knowledge of the unseen categories with attributes and visual interpretations, which is referred to as Semantic Recovery

Open-Set Domain Adaptation (SR-OSDA). Specifically, the proposed framework includes an explicit attribute interpretable module and an implicit semantic interpretable module, which provide insight into the process of domain adaptation and the discovery of new categories. Furthermore, structure-preserving partial alignment is developed as a method of recognizing and aligning the visible categories across domains with the aid of domain-invariant feature learning. The visual-structural semantic attribute propagation is designed to provide smooth transitions from seen categories to unseen categories via visual-semantic mapping. Three new cross-domain SR-OSDA benchmarks are constructed in order to evaluate the proposed framework in novel and practical challenges. Experimental results and empirical analysis of our proposed solution to open-set recognition and semantic recovery demonstrate its superiority over other state-of-the-art solutions. Our source code is available at https://github.com/scottjingtt/XSROSDA.

5.1 Summary of Contribution

In this work, we explore a novel problem named Semantic Recovery Open-Set Domain Adaptation (SR-OSDA), where the source domain is annotated with both class labels and semantic attributes descriptions, while the target domain consists of unlabeled data from categories seen and unseen in the source domain. SR-OSDA aims to recognize the seen categories as well as recover the missing semantic information of the samples from unseen categories while interpreting the domain adaptation and discovering novel categories in the target domain. To our best knowledge, this is a completely unexplored problem in literature with challenges including (1) how to effectively eliminate the domain shift across domains; (2) how to accurately identify seen and unseen categories in the target domain; (3) how to explicitly recover the missing attributes of unseen data; (4) how to explain the domain adaptation and novel categories discovery.

Overall, we present an interpretable framework that identifies and discovers novel categories from a target domain simultaneously, while revealing domain adaptation along with the discovery of novel categories, both at a semantic level and prototype level. Our main contributions are summarized below:

- We are the first to study the SR-OSDA problem with an effective framework to identify seen categories and discover novel categories simultaneously.

- We present structure-preserving partial alignment and visual-based attribute propagation designs to eliminate the domain shift and recover missing semantic attributes across domains.
- We propose an interpretable attributes prediction module to reveal the domain adaptation and novel categories discovery process.
- Our proposed method achieves promising performance on both target recognition and semantic recovery on three new protocols built for SR-OSDA evaluation.

This is an extension of our conference work [50] with the following improvements. First, we enhance the model by proposing a prototype-based interpretable module to reveal the domain adaptation and novel categories discovery process. In this sense, we are able to interpret the learning mechanism from two perspectives. Second, considering there are no existing benchmarks for the new problem, we construct more evaluation benchmarks for the SR-OSDA problem to demonstrate the effectiveness of the proposed model. Third, we provide more quantitative and qualitative analyses to study knowledge transfer across visual and semantic spaces, which provides new insight to understand the interpretation scheme.

The goal of SR-OSDA includes three parts: 1) Recognize the target data into C_s seen categories and one unknown class, which is similar to the conventional OSDA problem. 2) Recover the semantic descriptions $\mathbf{A}_t$ of the target data both seen and unseen classes. 3) Infer the class label of target data from $C_t = C_s + K$ categories by searching for the class with the most similar semantic embedding. It is noteworthy that the semantic knowledge of target domain unseen categories is only available during the test phase, while the training phase only has access to the semantic description of classes seen in the source domain.

5.2 The Proposed Solution

In this section, we first illustrate the framework overview of the proposed methods addressing SR-OSDA problem. Then we introduce the training strategy and learning objectives including progressive target annotation initialization, structure-preserving partial alignment, visual-structural seman-

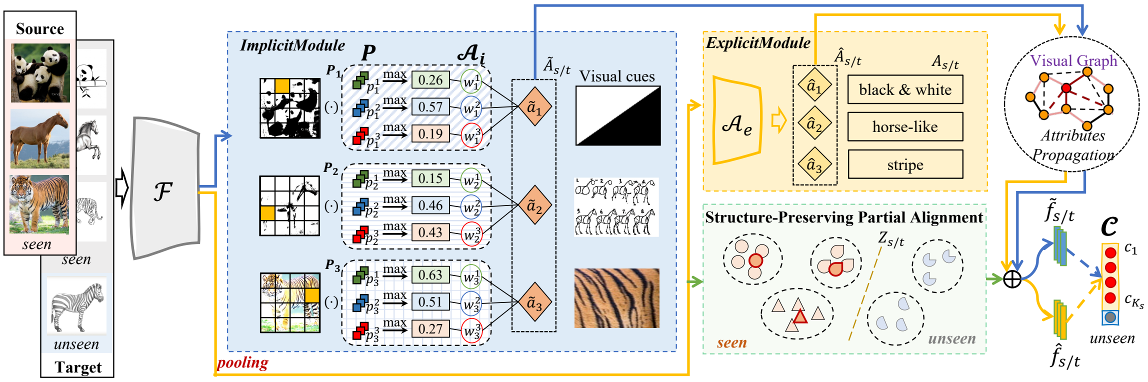


Figure 5.1: Illustration of the proposed framework. Given the raw source and target images as input, convolutional neural networks $\mathcal{F}(\cdot)$ act as the backbone to extract the visual features. The explicit semantic interpretable module (ExplicitModule) predicts the attributes by $\mathcal{A}_e(\cdot)$ based on the visual features. The implicit semantic interpretable module (ImplicitModule) explores visual cues through learned prototypes $\mathbf{P}$ to justify the attributes predicted by $\mathcal{A}_i(\cdot)$. The source and target domain data from shared classes are aligned in the visual feature space through the structure-preserving partial alignment module, while the target domain unseen categories data discriminative structure is preserved. With the visual-structural attributes propagation mechanism, visual features are used to promote predicted attributes from seen to unseen categories. In the end, visual and semantic features are fused for the open-set task classifier $\mathcal{C}(\cdot)$ to be recognized as belonging to one of $C_s + 1$ categories.

tic attributes recovery, interpretable visual to semantic projection, and cross-modality representations fusion.

5.2.1 Framework Overview

We propose a framework illustrated in Figure 5.1 to jointly recognize the target domain data based on categories already observed in the source domain, while also recovering the semantic attributes for the unknown target classes through an interpretable prototype-based mechanism to tackle the SR-OSDA problem. Specifically, there is one *explicit attribute interpretable module* (***ExplicitModule***) and one *implicit semantic interpretable module* (***ImplicitModule***) to predict the semantic attributes based on visual input from different perspectives. Meanwhile, the source and target domain data from shared categories are aligned in the target domain feature space through the structure-preserving partial alignment to preserve the target domain discriminative structure. Moreover, the visual features will guide the predicted attributes propagation from seen categories to unseen ones with the visual-structural semantic attributes propagation, and the semantic attributes will promote the visual features discrimination through the visual-semantic fused representation for open-set classification supervision.

Mathematically, given one image $\mathbf{x} \in \mathcal{D}_{s/t}$, $\mathcal{F}(\cdot)$ extracts the feature map $\mathbf{F}$ and feature vector $\mathbf{z}$ as:

$$\mathbf{F} = \mathcal{F}(\mathbf{x}), \qquad \mathbf{z} = \varphi(\mathbf{F}), \tag{5.1}$$

where $\varphi(\cdot)$ is the *pooling* operation, $\mathbf{z} \in \mathbb{R}^{D}$, and $\mathbf{F} \in \mathbb{R}^{W \times H \times D}$. Then $\mathbf{z}$ and $\mathbf{F}$ are input to the *explicit attribute interpretable module* and *implicit semantic interpretable module* to predict the semantic attributes, respectively.

For the ***ExplicitModule***, the extracted feature $\mathbf{z}$ for one of the source or target domain data is input to the semantic attributes predictor $\mathcal{A}_e(\cdot)$ to predict the attributes as:

$$\hat{\mathbf{a}} = \mathcal{A}_e(\mathbf{z}), \tag{5.2}$$

which is followed by a *Sigmoid* function to obtain the probabilities that the input sample has each specific attribute characteristic.

For the ***ImplicitModule***, instead of bridging the visual and semantic feature space via black-box neural networks, we further explore a prototype-based interpretable projector from the visual to semantic space. Specifically, with m learned prototypes $\mathbf{P} = \{\mathbf{p}_i\}_{i=1}^{m}$, the activation scores $\mathbf{s} \in \mathbb{R}^{m}$ on each extracted feature map $\mathbf{F}$ and the predicted attributes $\tilde{\mathbf{a}}$ are obtained as:

$$\mathbf{s} = \mathcal{S}(\mathbf{F},\ \mathbf{P}), \qquad \tilde{\mathbf{a}} = \mathcal{A}_i(\mathbf{s}), \tag{5.3}$$

where $\mathcal{S}(\cdot, \cdot)$ is the function calculating the activation scores of the learned prototypes on the feature map of the input image, which will be described in Section 5.2.4. Then the activation scores are input to a linear projector $\mathcal{A}_i(\cdot)$ to predict the attributes $\tilde{\mathbf{a}}$.

Finally, to leverage the multimodality benefits of visual and semantic descriptions, we explore the joint visual-semantic representation by conveying the semantic discriminative information into the visual feature as $\hat{\mathbf{h}} = \hat{\mathbf{a}} \oplus \mathbf{z}$ and $\tilde{\mathbf{h}} = \tilde{\mathbf{a}} \oplus \mathbf{z}$, where $\oplus$ is the concatenating operation. Then both joint features are input to the open-set classifier $\mathcal{C}(\cdot)$ and the output class labels are:

$$\hat{y} = \mathcal{C}(\hat{\mathbf{h}}), \qquad \tilde{y} = \mathcal{C}(\tilde{\mathbf{h}}), \tag{5.4}$$

where $\hat{y}/\tilde{y}$ denotes that the input sample is recognized from one of the C_s seen categories or the unknown class.

5.2.2 Progressive Target Annotation Initialization

One of the key challenges of SR-OSDA is the lack of annotations of the target domain data, especially the novel categories, and the discriminative distribution of the target data in the visual feature space is the only information accessible for training. To leverage the target domain class-wise structural knowledge and inspired by the impressive performance of exploring pseudo-labels in unsupervised domain adaptation recently [12, 127, 162], we initialize the annotations for the target data only based on the visual features. Intuitively, some target domain samples are distributed close to the source domain, which can be confidently recognized by a well-trained source model. On the contrary, those samples from unknown categories never present in the source domain tend to obtain even or mixed prediction probabilities by the source model with low confidence, because no classifiers can recognize them easily.

To achieve this, we first apply an adaptive nearest neighbor classification strategy to recognize all target samples into one of C_s seen categories plus one "unknown" class in the latent embedding space. Specifically, given C_s class centroids of the seen categories, denoted as $\{\mu^c = \frac{1}{n_s^c}\sum_{\mathbf{z}_s \in \mathbf{Z}_s^c} \mathbf{z}_s|_{c=1}^{C_s}\}$, where $\mathbf{Z}_s^c$ denotes n_s^c features of source data from class c. Then probability prediction of one target data $\mathbf{z}_t \in \mathbf{Z}_t$ from class c is defined as:

$$p(c|\mathbf{x}_t) = \frac{\exp(-d(\mathbf{z}_t, \mu^c))}{\sum_{c'} \exp(-d(\mathbf{z}_t, \mu^{c'}))}, \quad \mathbf{z}_t \in \mathbf{Z}_t, \tag{5.5}$$

where $d(\cdot,\cdot)$ is the distance function measuring the similarities in the visual feature space, *i.e.*, $d(\mathbf{z}_t, \mu^c) = 1 - \frac{\mathbf{z}_t \cdot \mu^c}{\|\mathbf{z}_t\|\|\mu^c\|}$. For each sample, we adopt category c with the highest probability prediction as the pseudo label for $\mathbf{z}_t$, which is denoted as $\bar{y}_t = \underset{c}{\operatorname{argmax}}\ p(c|\mathbf{x}_t)$. Then, we are able to separate the target domain data $\mathcal{D}_t$ in to *seen* subset $\bar{\mathcal{D}}_t^s$ and *unseen* subset $\bar{\mathcal{D}}_t^u$ as:

$$\begin{cases} \mathbf{x}_t \in \bar{\mathcal{D}}_t^s, \quad p(\bar{y}_t|\mathbf{x}_t) \geq \tau \\ \mathbf{x}_t \in \bar{\mathcal{D}}_t^u, \quad p(\bar{y}_t|\mathbf{x}_t) < \tau \end{cases}, \tag{5.6}$$

where the threshold τ is defined as the mean of the pseudo labels probabilities of all target samples, i.e., $\tau = \frac{1}{n_t} \sum_{\mathbf{x}_t \in \mathcal{D}_t} p(\bar{y}_t|\mathbf{x}_t)$.

Since we only have access to the source data and ground-truth annotations, which are not sufficient to identify the target data due to domain shift, we adaptively update the classes centroids by involving newly-labeled target samples from $\bar{\mathcal{D}}_t^s$ as:

$$\mu^c = (1-\lambda)\mu^c + \lambda \frac{1}{n_t^{s(c)}} \sum_{\mathbf{x}_t \in \bar{\mathcal{D}}_t^{s(c)}} \mathbf{z}_t, \tag{5.7}$$

where $\bar{\mathcal{D}}_t^{s(c)} = \{\mathbf{x}_t|\mathbf{x}_t \in \mathcal{D}_t,\ \bar{y}_t = c\}$ contains $n_t^{s(c)}$ target samples predicted as category $\bar{y}_t = c$ with high confidence, and λ is a small value controlling the cross-domain mixture.

In addition to obtaining confident pseudo labels for target samples in the seen set $\bar{\mathcal{D}}_t^s$, we also need to explore the discriminative structure of data in $\bar{\mathcal{D}}_t^u$. Thus, we apply K-means clustering algorithm to group $\bar{\mathcal{D}}_t^u$ resulting in K clusters with cluster centers denoted as $\{\eta^{k_1}, ..., \eta^K\}$. Based on the results, we obtain all classes and clustering centroids of both seen and unseen subsets as $\mathcal{R}_\mathbf{Z} = \{\mu^1, ..., \mu^{C_s}, \eta^{k_1}, ..., \eta^K\}$. In order to refine the pseudo labels of the target data, we adopt K-means clustering algorithm with centers initialized as $\mathcal{R}_\mathbf{Z}$ over $\mathbf{Z}_t$ to obtain the target domain pseudo labels as $\bar{\mathbf{Y}}_t = \{\bar{y}_t|\mathbf{x}_t \in \mathcal{D}_t,\ \mathcal{R}_\mathbf{Z}\}$ covers both C_s seen categories and K clusters.

With the obtained pseudo labels for all target samples, we also assign the corresponding semantic attributes $\bar{\mathbf{a}}_t$ to each sample $\mathbf{x}_t$ in the seen subsets $\bar{\mathcal{D}}_t^s$ based on the pseudo labels $\bar{y}_t$. The semantic knowledge of the target data bridges the visual to semantic space while contributing to the structure-preserving cross-domain alignment as described in Section 5.2.3.

5.2.3 Explicit Attribute interpretable Module

The *ExplicitModule* shares similarities with the framework introduced in Section 3.3.2. Consequently, we adopt similar optimization objectives, namely, Structure Preserving Partial Alignment

($\mathcal{L}_R$) and Attributes Propagation with Visual Structure ($\mathcal{L}_A^{ex} = \mathcal{L}^A$). These objectives are utilized to optimize the explicit attribute prediction branch, setting it apart from the implicit branch.

5.2.4 Implicit Semantic Interpretable Module

Moreover, the estimated attributes for both seen and unseen categories in the target domain by $\mathcal{A}_e(\cdot)$ contribute to novel categories discovery with attribute-based explanation, however, the semantic-to-visual projection is a black box lacking transparency and interpretation. Thus, we propose the implicit semantic interpretable module (*ImplicitModule*), which is a prototype-based interpretable module, to reveal representative visual cues for each specific attribute via learning corresponding semantic prototypes based on the visual input. For each input image, the implicit visual-semantic module will slide over all patches in the extracted feature map $\mathbf{F}$ and calculate the activation scores with respect to the presence of learned semantic prototypes $\mathbf{P}$. The regions most activated by the learned prototypes are selected as semantic-inducing regions for predicting the attributes, while the corresponding prototypes act as visual interpretations for the projection from visual to semantic space.

Specifically, for each sample $\mathbf{x} \in \mathcal{D}_{s/t}$ with semantic attributes $\mathbf{a} \in \mathbb{R}^{d_a}$, we assign m_j prototypes for each attribute element $a_j \in \mathbf{a}$, resulting in $m = m_j \times d_a$ prototypes in total, denoted as $\mathbf{P} = \{\mathbf{P}_j\}_{j=1}^{d_a}$, in which $\mathbf{P}_j = \{\mathbf{p}_j^l\}_{l=1}^{m_j}$, and $\mathbf{p}_j^l \in \mathbb{R}^{D\times 1\times 1}$ is the l^{th} learned semantic prototype for attribute element j. Intuitively, prototypes in $\mathbf{P}_j$ should capture the most relevant parts for identifying images of attribute a_j.

For one image input to the feature extractor $\mathcal{F}(\cdot)$, the patches in the extracted feature map $\mathbf{F}_{s/t} \in \mathbb{R}^{W\times H\times D}$ are denoted as $\mathbf{F}_{s/t} = \{\mathbf{F}_{s/t}^k\}_{k=1}^{HW}$. The shape of each patch is $\mathbf{f}_{s/t}^k \in \mathbb{R}^{D\times 1\times 1}$. For one feature map $\mathbf{F}$ and one prototype $\mathbf{p}_j^l$, $\mathbf{p}_j^l$ will go through all patches $\mathbf{f}^k \in \mathbf{F}$ and compute the activation score as:

$$s_j^l = \max_{\mathbf{f}^k \in \mathbf{F}} \frac{\mathbf{f}^k \cdot \mathbf{p}_j^l}{\|\mathbf{f}^k\|\|\mathbf{p}_j^l\|}, \tag{5.8}$$

where the activation score s_j^l is monotonically increasing with respect to the similarity between $\mathbf{f}^k$ and $\mathbf{p}_j^l$. If the activation score s_j^l is large, a patch in the latent feature map $\mathbf{F}$ is similar to the prototype $\mathbf{p}_j^l$, denoting that the corresponding region in the input image contains a similar concept as what prototype $\mathbf{p}_j^l$ represents.

The obtained activation scores produced by all prototypes $\mathbf{P}$ with respect to the input feature map $\mathbf{F}$ is denoted as $\mathbf{s} = \mathcal{S}(\mathbf{F}, \mathbf{P}) = \{s_j^1, ..., s_j^{m_j} | \mathbf{F}, \mathbf{P}_j\}_{j=1}^{d_a}$, which is then input to the attributes projector $\mathcal{A}_i(\cdot)$ with output as:

$$\tilde{\mathbf{a}} = \mathcal{A}_i(\mathbf{s}), \tag{5.9}$$

where $\mathcal{A}_i(\cdot)$ is a fully connected layer without bias with d_a output followed by *Sigmoid* activation, predicting if each specific attribute characteristic exists in the input image or not.

Similar to Eq. (**??**) in Section 5.2.3, the visual-structural semantic attributes propagation strategy is also applied to $\tilde{\mathbf{a}}_i$ as:

$$\tilde{\mathbf{a}}_i = \sum\nolimits_j W_{ij} \mathcal{A}_i(\mathbf{s}_i), \tag{5.10}$$

where $\mathbf{s}_i$ is the prototype activation scores vector of sample $\mathbf{x}_i$ in a training batch. Then the *ImplicitModule* is optimized with the supervision of both source and target domain data as:

$$\mathcal{L}_A^{im} = \mathop{\mathbb{E}}_{\mathbf{x}_s \in \mathcal{D}_s} \Big(L_{bce}(\tilde{\mathbf{a}}_s, \ \mathbf{a}_s) \Big) + \mathop{\mathbb{E}}_{\mathbf{x}_t \in \bar{\mathcal{D}}_t^s} \Big(L_{bce}(\tilde{\mathbf{a}}_t, \ \bar{\mathbf{a}}_t) \Big), \tag{5.11}$$

where $\mathbf{a}_s$ is the source data ground-truth attributes, and $\bar{\mathbf{a}}_t$ is the target data attributes corresponding to the pseudo-label.

Furthermore, if attribute $a_j \in \mathbf{a}$ is true, at least one of the m_j prototypes in $\mathbf{P}_j$ should be activated significantly. In contrast, none of $\mathbf{P}_j$ should be activated if a_j is False. Thus, the prototypes $\mathbf{P}$ are learned with discriminative constraint as:

$$\begin{aligned} \mathcal{L}_{clst}^{im} &= \mathop{\mathbb{E}}_{\mathbf{x} \in \mathcal{D}_s \cup \bar{\mathcal{D}}_t^s} \ \mathop{\mathbb{E}}_{\{\mathbf{P}_j \in \mathbf{P} | \mathbf{a}_j = 1\}} \ \min_{\mathbf{p}_j^l \in \mathbf{P}_j} \ \min_{\mathbf{f}^k \in \mathbf{F}} \quad d(\mathbf{f}^k, \mathbf{p}_j^l), \\ \mathcal{L}_{sep}^{im} &= \mathop{\mathbb{E}}_{\mathbf{x} \in \mathcal{D}_s \cup \bar{\mathcal{D}}_t^s} \ \mathop{\mathbb{E}}_{\{\mathbf{P}_j \in \mathbf{P} | \mathbf{a}_j = 0\}} \ \min_{\mathbf{p}_j^l \in \mathbf{P}_j} \ \min_{\mathbf{f}^k \in \mathbf{F}} \quad d(\mathbf{f}^k, \mathbf{p}_j^l), \end{aligned} \tag{5.12}$$

where $d(\mathbf{f}^k, \mathbf{p}_j^l) = 1 - \frac{\mathbf{f}^k \cdot \mathbf{p}_j^l}{\|\mathbf{f}^k\| \|\mathbf{p}_j^l\|}$ measures the distance between $\mathbf{f}^k$ and $\mathbf{p}_j^l$. Intuitively, $\mathcal{L}_{clst}^{im}$ minimizes the closes distance of the feature patch and prototype pair for every true attribute element of each input sample, and $\mathcal{L}_{sep}^{im}$ maximize the distance of the patch and prototype pair for every attribute element that does not exist in the input sample class.

Clustering structures derived from such optimization objectives are semantically meaningful. The overall learning objective for optimizing prototypes is as follows:

$$\mathcal{L}_P = \mathcal{L}_{clst}^{im} - \alpha_1 \mathcal{L}_{sep}^{im} + \alpha_2 \sum_{j=1}^{d_a} \sum_{\mathbf{p}_j^l \notin \mathbf{P}_j} |w_E^{(j,l)}|, \tag{5.13}$$

where α_1 and α_2 are two hyper-parameters determining the contributions of the two loss terms, respectively, and $w_E^{(j,l)}$ denotes the connection weights of $\mathcal{A}_i(\cdot)$ between the l^{th} prototype ($\mathbf{p}_j^l$) and the j^{th} attribute (a_j). We optimize the weights of the last attributes predictor layer $\mathcal{A}_i(\cdot)$ to obtain sparse property which makes our model relies less on a *negative* reasoning process of the form "this attribute element does not exist since it contains a patch that is *not* prototypical of the specific attribute element" [11].

5.2.5 Overall Objective

To incorporate classification supervision into the fused visual-semantic features, we propose optimizing the classifier via explicit and implicit branches in conjunction with the following open-set classification objective:

$$\begin{aligned}
\mathcal{L}_C^{ex} &= \mathop{\mathbb{E}}_{\substack{\mathbf{x}_s \in \mathcal{D}_s \\ y_s \in \mathbf{Y}_s}} L_{ce}(\hat{y}_s, \psi(y_s)) + \mathop{\mathbb{E}}_{\substack{\mathbf{x}_t \in \bar{\mathcal{D}}_t^s \\ \bar{y}_t \in \bar{\mathbf{Y}}_t}} L_{ce}(\hat{y}_t, \psi(\bar{y}_t)), \\
\mathcal{L}_C^{im} &= \mathop{\mathbb{E}}_{\substack{\mathbf{x}_s \in \mathcal{D}_s \\ y_s \in \mathbf{Y}_s}} L_{ce}(\tilde{y}_s, \psi(y_s)) + \mathop{\mathbb{E}}_{\substack{\mathbf{x}_t \in \bar{\mathcal{D}}_t^s \\ \bar{y}_t \in \bar{\mathbf{Y}}_t}} L_{ce}(\tilde{y}_t, \psi(\bar{y}_t)),
\end{aligned} \tag{5.14}$$

where $L_{ce}(\cdot,\cdot)$ is the cross-entropy loss, and $\psi(y)$ indicates if y is from one of C_s seen categories or the "unknwon" class in target domain, and $\hat{y} = C(\hat{\mathbf{h}})$ and $\tilde{y} = C(\tilde{\mathbf{h}})$.

To sum up, we have our overall objective by integrating the joint visual-semantic representation recognition supervision for both explicit and implicit branches($\mathcal{L}_C^{ex/im}$), explicit and implicit visual-semantic recovery ($\mathcal{L}_A^{ex/im}$), structure-preserving partial alignment ($\mathcal{L}_R$), and discriminative

Table 5.1: Statistics of evaluation benchmarks. (S: Source. T: Target)

Dataset	Domain	Role	# Images	# Attributes	# Classes
DomainNet → AwA	AwA	S / T	9,343 / 16,306	85	10 / 17
	Paint	S / T	3,441 / 5,760	85	10 / 17
	Real	S / T	5,251 / 10,047	85	10 / 17
I → AwA	I / AwA	S / T	2,970 / 37,322	85	40 / 50
DomainNet → LAD	LAD	S / T	13,322 / 19,744	253	40 / 56
	Paint	S / T	11,714 / 15,311	253	40 / 56
	Real	S / T	22,395 / 31,066	253	40 / 56

prototypes constraint ($\mathcal{L}_P$) to train the whole framework alternatively as:

$$\begin{array}{ll} \min\limits_{\mathcal{F},\mathcal{A}_e,\mathcal{C}} & \mathcal{L}_C^{ex} + \mathcal{L}_A^{ex} + \beta_1 \mathcal{L}_R \\ \min\limits_{\mathcal{F},\mathbf{P},\mathcal{A}_i,\mathcal{C}} & \mathcal{L}_C^{im} + \mathcal{L}_A^{im} + \beta_2 \mathcal{L}_P \end{array}, \tag{5.15}$$

where β_1 and β_2 are two trade-off parameters. Through minimizing the loss illustrated in the first row, the feature generator $\mathcal{F}(\cdot)$, explicit visual-semantic projector $\mathcal{A}_e(\cdot)$, open-set classifier $\mathcal{C}(\cdot)$ are optimized to aggregate the source data semantic descriptive knowledge into the unlabeled target domain in the latent embedding space via the joint visual-semantic representation supervision, attributes prediction, and the cross-domain visual structure-preserving partial alignment. The parameters of prototypes $\mathbf{P}$ and the implicit attributes predictor $\mathcal{A}_i(\cdot)$ are trained in addition.

Inference Stage. Given a target raw input first passing to the feature extractor $\mathcal{F}(\cdot)$, the open-set classifier $\mathcal{C}(\cdot)$ then recognizes if the input sample is from one of the C_s seen categories or it is unknown for OSDA. For the SR-OSDA problem, the classifier $\mathcal{C}(\cdot)$ first recognizes if the input sample is from one of the seen or unseen categories, then infer the class label from $C_t = C_s + K$ classes by searching for the class with the most similar ground-truth attributes.

5.3 Experiments

5.3.1 Benchmark Datasets

We construct three evaluation protocols for the SR-OSDA problem: (1) **DomainNet → AwA** is constructed from the DomainNet dataset [102] and AwA2 dataset [144]. Specifically, there are 17 categories shared between these two datasets, from which the alphabetically first 10 categories are

Table 5.2: Open-set domain adaptation (OSDA) accuracy (%) on DomainNet → AwA

Task	**AwA → Paint**				**AwA → Real**				**Paint → AwA**			
Method	OS*	OS$^{\diamond}$	OS	OS^{H}	OS*	OS$^{\diamond}$	OS	OS^{H}	OS*	OS$^{\diamond}$	OS	OS^{H}
OSBP [114]	49.6	10.8	46.0	17.7	74.2	13.6	68.7	23.0	76.0	9.1	69.9	16.2
STA [72]	60.1	33.0	57.6	42.6	85.5	10.8	78.7	19.2	**90.2**	5.7	82.5	10.7
AOD [27]	50.7	9.5	46.9	16.0	78.4	12.7	72.4	21.9	80.3	5.1	73.5	9.6
Ours (*conf.*) [50]	**62.8**	47.2	**61.4**	53.9	**90.9**	71.4	**89.1**	80.0	79.2	**98.5**	81.0	**87.8**
Ours (*Expl.*)	45.0	**79.4**	48.1	57.4	81.4	**81.1**	81.4	**81.3**	83.4	90.0	**84.0**	86.6
Ours (*Impl.*)	48.8	71.9	50.9	**58.1**	82.6	75.9	82.0	79.1	83.1	86.4	83.4	84.7

Task	**Paint → Real**				**Real → AwA**				**Real → Paint**			
Method	OS*	OS$^{\diamond}$	OS	OS^{H}	OS*	OS$^{\diamond}$	OS	OS^{H}	OS*	OS$^{\diamond}$	OS	OS^{H}
OSBP [114]	63.3	6.9	58.2	12.4	90.1	13.7	83.2	23.8	55.9	10.6	51.7	17.8
STA [72]	**82.8**	7.4	76.0	13.6	88.5	7.2	81.1	13.3	**66.9**	13.5	62.0	22.5
AOD [27]	79.7	5.3	73.0	9.9	92.0	12.8	84.8	22.5	61.2	9.6	56.5	16.6
Ours (*conf.*) [50]	78.3	**83.7**	**78.8**	**80.9**	**94.9**	90.5	**94.5**	**92.7**	61.2	**80.4**	**63.0**	**69.5**
Ours (*Expl.*)	78.7	78.3	78.7	78.5	91.7	**93.1**	91.8	92.4	52.8	72.7	54.6	61.2
Ours (*Impl.*)	77.7	78.1	77.8	77.9	92.6	89.3	92.3	90.9	58.7	59.0	58.7	58.9

Table 5.3: Semantic recovery open-set DA (SR-OSDA) accuracy (%) on DomainNet → AwA

Task	**AwA → Paint**			**AwA → Real**			**Paint → AwA**			**Paint → Real**			**Real → AwA**			**Real → Paint**		
Method	S	U	H	S	U	H	S	U	H	S	U	H	S	U	H	S	U	H
ABP [175]	68.1	0.0	0.0	87.9	0.0	0.0	**91.7**	0.0	0.0	**83.6**	0.0	0.0	94.4	0.0	0.0	70.0	0.0	0.0
TF-VAE [88]	**70.4**	0.0	0.0	88.4	0.0	0.0	85.1	0.0	0.0	79.6	0.0	0.0	**96.4**	0.0	0.0	**72.5**	0.0	0.0
ABP* [175]	64.5	6.4	11.7	86.0	5.9	11.1	84.0	24.4	37.8	81.3	12.7	21.9	93.8	16.2	27.6	67.6	7.9	14.1
TF-VAE* [88]	59.7	12.8	21.0	77.9	16.4	27.1	35.1	35.6	35.3	34.8	32.7	33.7	68.5	36.1	47.3	50.7	21.0	29.7
Ours (*conf.*) [50]	62.5	27.0	37.7	**90.7**	30.0	45.1	79.2	36.7	50.2	78.0	15.7	26.1	95.2	37.8	54.1	59.0	20.8	30.8
Ours (*Expl.*)	42.4	36.4	39.2	81.3	38.5	52.2	73.8	**68.0**	**70.8**	75.9	57.6	**65.5**	91.4	54.2	68.1	50.2	**36.1**	42.0
Ours (*Impl.*)	44.2	**38.2**	**41.0**	81.0	**45.9**	**58.6**	73.7	59.6	65.9	65.9	**59.8**	62.7	91.2	**54.9**	**68.5**	53.5	35.5	**42.7**

selected as the *seen* classes across domains, while the rest 7 categories are *unseen* categories that only exist in the target domain. The corresponding attribute features about the shared 17 categories from the AwA2 dataset are used as semantic descriptions. In view of the fact that some domains in the DomainNet dataset barely share common semantic characteristics as images in the AwA2 dataset, such as quick draw, we only take the "real image" (Real) and "Painting" (Paint) into account, together with the AwA2 data (AwA) for evaluation. (2) **I → AwA** is collected by [176] consisting of 50 animal classes, and split into 40 seen categories and 10 unseen categories as [144]. The source domain (I), includes 2,970 images from seen categories collected via the Google image search engine, while the target domain comes from the AwA2 (AwA) dataset for zero-shot learning with 37,322 images in all 50 classes [144]. We use the binary attributes of AwA2 as the semantic description, and only one task I → AwA is evaluated. (3) **DomainNet → LAD** is based on the data from Domainnet [102] and LAD [169]. LAD is a large-scale attribute dataset consisting of 78,017 images from 230 classes. We

Table 5.4: Open-set domain adaptation (OSDA) accuracy (%) and semantic recovery open-set domain adaptation (SR-OSDA) accuracy(%) on I → AwA

OSDA					SR-OSDA			
Method	OS*	OS$^\diamond$	OS	OS^H	Method	S	U	H
OSBP [114]	67.6	7.5	66.2	13.5	ABP [175]	79.8	0.0	0.0
STA [72]	51.5	45.5	51.4	48.3	ABP* [175]	78.0	13.4	22.9
AOD [27]	75.2	6.3	73.5	11.6	TF-VAE* [88]	37.7	20.0	26.2
Ours (*conf.*) [50]	**83.2**	70.2	**82.8**	76.1	Ours (*conf.*) [50]	**83.1**	22.0	34.8
Ours (*Expl.*)	81.7	70.8	81.4	75.9	Ours (*Expl.*)	78.7	32.1	45.6
Ours (*Impl.*)	82.6	**71.8**	82.4	**76.9**	Ours (*Impl.*)	79.2	**33.9**	**47.5**

Table 5.5: Open-set domain adaptation (OSDA) accuracy (%) on DomainNet → LAD

Task	**LAD → Paint**				**LAD → Real**				**Paint → LAD**			
Method	OS*	OS$^\diamond$	OS	OS^H	OS*	OS$^\diamond$	OS	OS^H	OS*	OS$^\diamond$	OS	OS^H
OSBP [114]	31.5	**89.5**	32.9	46.6	49.3	**84.3**	50.2	62.2	25.2	**86.1**	26.7	39.0
STA [72]	45.3	67.9	45.9	54.4	75.1	31.4	74.1	44.3	73.7	27.8	72.5	40.3
AOD [27]	35.2	85.5	36.4	49.8	51.4	79.6	52.1	62.5	34.3	74.0	35.3	46.9
Ours (*conf.*) [50]	41.1	79.8	42.1	54.3	77.7	79.8	77.8	**78.7**	54.8	85.6	55.5	66.8
Ours (*Expl.*)	49.5	87.4	50.5	63.2	82.3	68.9	82.0	75.0	81.6	67.8	81.3	74.1
Ours (*Impl.*)	**53.1**	80.7	**53.8**	**64.1**	**83.3**	64.2	**82.8**	72.5	**82.0**	70.3	**81.7**	**75.7**

Task	**Paint → Real**				**Real → LAD**				**Real → Paint**			
Method	OS*	OS$^\diamond$	OS	OS^H	OS*	OS$^\diamond$	OS	OS^H	OS*	OS$^\diamond$	OS	OS^H
OSBP [114]	23.5	80.0	24.9	36.3	55.4	**84.7**	56.1	67.0	39.9	**82.2**	40.9	53.7
STA [72]	69.8	19.4	68.5	30.3	87.6	19.8	85.9	32.2	54.7	52.2	54.7	53.5
AOD [27]	25.0	75.2	26.2	37.5	49.9	80.0	50.6	61.5	40.1	79.7	41.0	53.3
Ours (*conf.*) [50]	53.8	**83.9**	54.5	**65.6**	89.7	79.4	89.5	84.3	47.6	75.9	48.3	58.5
Ours (*Expl.*)	**82.2**	50.2	**81.4**	62.4	**91.0**	83.2	**90.8**	86.9	59.8	75.9	60.2	66.9
Ours (*Impl.*)	82.1	50.3	81.3	62.4	**91.0**	83.6	**90.8**	**87.1**	**62.4**	76.1	**62.8**	**68.6**

select the 56 categories shared between DomainNet and LAD to evaluate the model. In the same way, the first 40 alphabetically listed categories are *seen*, while the rest 16 remain as *unseen*. LAD also provides more diverse semantic attribute descriptions, and we adopt 253 out of 359 binary attributes found in the seen categories as the semantic description, whereas the remaining 106 attributes are ignored. It is noteworthy that only the attributes of the *seen* categories are accessible during the training phase, the semantic information about *unseen* categories is only available for testing.

5.3.2 Implementation Details

In this work, ResNet-50 [35] without the last fully-connected layer pre-trained on ImageNet is adopted as the convolutional backbone $\mathcal{F}(\cdot)$. Before input to the *ExplicitModule*, the output of the convolutional backbone is input to a *pooling* layer followed by two fully-connected layers to

Table 5.6: Semantic recovery open-set DA (SR-OSDA) accuracy (%) on DomainNet → LAD

Task	**LAD → Paint**			**LAD → Real**			**Paint → LAD**			**Paint → Real**			**Real → LAD**			**Real → Paint**		
Method	S	U	H	S	U	H	S	U	H	S	U	H	S	U	H	S	U	H
ABP [175]	62.2	0.0	0.0	**85.2**	0.0	0.0	**83.4**	0.0	0.0	**80.9**	0.0	0.0	92.0	0.0	0.0	65.8	0.0	0.0
TF-VAE [88]	**62.9**	0.0	0.0	82.3	0.0	0.0	68.8	0.0	0.0	68.5	0.0	0.0	**92.4**	0.0	0.0	**67.7**	0.0	0.0
ABP* [175]	57.2	11.7	19.5	83.3	12.2	21.3	81.4	12.0	20.9	78.8	14.2	24.1	91.5	3.2	6.2	64.7	1.2	2.3
TF-VAE* [88]	49.0	13.5	21.2	67.4	18.7	29.2	54.4	13.7	21.8	54.2	16.2	25.0	89.0	4.3	8.3	65.2	2.9	5.5
Ours (*conf.*) [50]	48.0	**32.8**	**39.0**	72.3	**40.0**	**51.5**	61.3	32.0	42.1	55.2	**28.8**	37.9	69.9	**35.1**	46.7	41.0	17.0	24.1
Ours (*Expl.*)	44.6	22.6	30.0	78.9	26.6	39.8	76.9	36.0	**49.1**	80.8	23.3	36.1	87.9	33.8	**48.8**	56.0	**20.9**	**30.4**
Ours (*Impl.*)	51.7	22.2	31.1	79.7	25.3	38.4	72.2	**36.3**	48.4	78.9	27.0	**40.2**	88.7	32.2	47.3	59.7	18.9	28.7

reduce the dimension to 512, while before input to the *ImplicitModule*, the backbone output is input to two add-on convolutional layers with filter size as 1×1 to reduce the number of channels of the feature map to 512.

For the *ExplicitModule*, $\mathcal{A}_e(\cdot)$ and $\mathcal{C}(\cdot)$ are both two-layer fully-connected layer neural networks with the output dimension as the number of attributes (85 for DomainNet → AwA and I → AwA, while 253 for DomainNet → LAD) and the number of seen categories plus one unknown class ($C_s + 1$), respectively. The hidden layer output dimensions of both $\mathcal{A}_e(\cdot)$ and $\mathcal{C}(\cdot)$ are 256 and the activation function is ReLU, and the last layer output of $\mathcal{A}_e(\cdot)$ and $\mathcal{C}(\cdot)$ are followed by *Sigmoid* and *Softmax*, respectively. For *ImplicitModule*, the prototype layer $\mathbf{P}$ consists of d_a prototypes group $\mathbf{P}_j = \{\mathbf{p}_j^l\}_{l=1}^{m_j}$ where $m_j = 3$ for each attribute element $a_j \in \mathbf{a}$, and the shape of $\mathbf{p}_j^l$ is $1 \times 1 \times 512$. In addition, the activation scores of prototypes observed in the input image are input to one-layer fully-connected layer without bias as $\mathcal{A}_i(\cdot)$ followed by a *Sigmoid* function to predict the attributes probabilities. We employ cosine distances for all distance measurement operations $d(\cdot, \cdot)$. The framework is optimized by SGD optimizer, and the learning rate for parameters except the backbone is initialized as $l_0 = 10^{-3}$ with annealing strategy $l_p = \frac{l_0}{(1+\delta p)^q}$, where p is the progress of training epochs linearly changing from 0 to 1, $\delta = 10$ and $q = 0.75$, which is optimized to promote convergence and low error during training [163], while the learning rate for the convolutional backbone is one-tenth of other layers. We construct a validation set consisting of a subset of the rest target data to apply early-stop during training, and the hyper-parameters are empirically set as $\lambda = 0.001, \alpha_1 = 0.1, \alpha_2 = 0.001, \beta_1 = 0.1, \beta_2 = 0.1$ for all tasks.

5.3.3 Evaluation Metrics

To evaluate the capability of the proposed model on the SR-OSDA task recovering semantic descriptions of the target domain both seen and novel categories, we construct two evaluation metrics to quantitatively measure the performance.

Open-set Domain Adaptation. The conventional open-set domain adaptation protocol is followed, with the target domain data being broken down into C_s seen categories plus one "unknown" category [114, 99, 72, 57]. The class-wise average accuracy on the target domain seen categories are reported as OS*, while the class-wise average accuracy for the target domain "unknown" group samples is denoted as OS$^\diamond$ to alleviate the influence of the test data imbalance. Besides, the average accuracy over $C_s + 1$ seen plus the "unknown" class is reported as OS. Moreover, we observe that the overall accuracy is dominated by the performance on the seen classes, thus the harmonic mean is calculated as $\mathrm{OS}^H = \frac{2\times\mathrm{OS}^*\times\mathrm{OS}^\diamond}{\mathrm{OS}^*+\mathrm{OS}^\diamond}$ to fairly evaluate the overall performance of the model on the whole label space.

Semantic Recovery OSDA. To evaluate the quality of recovered missing semantic descriptions to the target data, we infer the predicted class label of the target data from the whole label space of $C_t = C_s + K$ categories by searching the category with the most similar ground-truth attributes as recovered semantic attributes. The class-wise average classification accuracy on the seen and unseen categories are denoted as S and U, respectively. Moreover, the harmonic mean $\mathrm{H} = \frac{2\times\mathrm{S}\times\mathrm{U}}{\mathrm{S}+\mathrm{U}}$ is also reported as the overall performance on the whole label space.

5.3.4 Competitive Methods and Results

We compare our model with different baselines on the three datasets. Specifically, "Ours (*conf.*)" denotes the our conference version [50], while "Ours (*Expl.*)" and "Ours (*Impl.*)" report the results calculated based on the attributes predicted by "*ExplicitModule*" and "*ImplicitModule*", respectively.

5.3.4.1 Open-set Domain Adaptation

For conventional open-set domain adaptation (OSDA) problem, we compare our model with several state-of-the-art open-set domain adaptation methods: OSBP [114], AOD [27], and STA [72]. OSBP

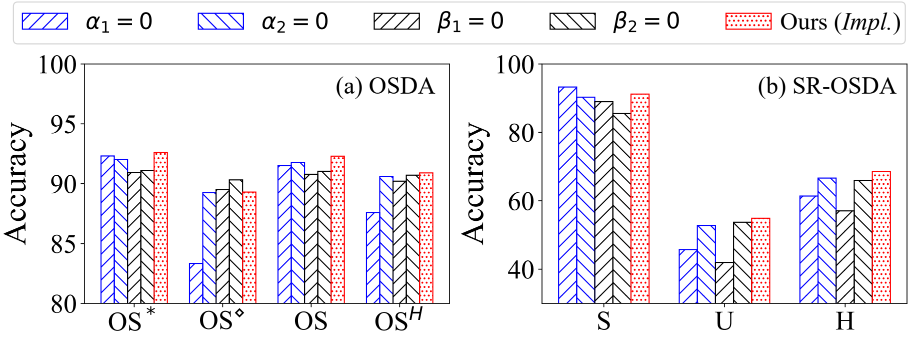


Figure 5.2: Analyses about the contribution of each loss term to our model (ImplicitModule) for task Real → AwA on DomainNet → AwA.

utilizes the adversarial training strategy to extract features from the target data, which is recognized into seen/unseen classes by a pre-defined threshold [114]. AOD exploits the semantic structure of open set data from categorical alignment and contrastive mapping to push the unknown classes away from the decision boundary [27]. Differently, STA adopts a coarse-to-fine mechanism to progressively separate the known and unknown data without any manually determined threshold [72]. The results are reported in Table 5.2, Table 5.4, and Table 5.5 for DomainNet → AwA, I → AwA, and DomainNet → LAD, respectively.

From the results, we observe that the proposed interpretable framework achieves comparable performance to our conference work, outperforming all compared baselines in terms of overall and harmonic mean accuracy on most tasks. Especially for task Real → LAD on DomainNet → LAD in Table 5.5, our interpretable model further improves the performance achieved by the conference version and outperforms the best-compared baseline over 4.9% and 20.1% in terms of OS and OS^H, respectively. The significant improvements come from our effective framework and the additional source of semantic information. Moreover, our proposed method reaches promising results on the unseen classes while keeping performance on the seen classes for all tasks. For example, OSBP achieves the best accuracy on unknown classes for task Paint → LAD, but fails on the seen categories classification, resulting in unsatisfactory results OS and OS^H. Such an observation emphasizes the superiority of our method in exploring target domain seen and unseen categories simultaneously.

5.3.4.2 Semantic Recovery Open-set Domain Adaptation

For the novel semantic recovery open-set domain adaptation (SR-OSDA) problem, we compare our model with the latest zero-shot learning (ZSL) and generalized zero-shot learning (GZSL) methods, ABP [175] and TF-VAE [88], under our setting. ABP trains a conditional generator mapping the class-level semantic features and Gaussian noise to visual features [175]. TF-VAE proposes to enforce semantic consistency at all training, feature synthesis, and classification stages [88]. Besides, both ABP and TF-VAE are able to handle generalized zero-shot learning problems given the semantic attributes from the whole target label space. We also report ABP* and TF-VAE*, which take the extra semantics of unseen target categories as inputs. It is noteworthy that for ZSL models, only the data and corresponding class labels, as well as category attributes, are available for training, while for GZSL models, class labels and corresponding semantic attributes of both seen and unseen categories are known in the training stage. The results are reported in Table 5.3, Table 5.4, and Table 5.6 for DomainNet → AwA, I → AwA, and DomainNet → LAD, respectively.

Within the expectation, all ZSL methods fail to recognize data from unseen categories and overfit the seen classes as a result of a lack of ability to handle an open-set setting.

Our proposed method achieves promising results in recognizing both seen and unseen categories. Specifically, our method achieves the best overall accuracy of 68.5% with improved unseen classes data accuracy to 54.9% while keeping 91.2% performance on seen classes for task Real → AwA. Moreover, our proposed method even outperforms ABP* and TF-VAE*, although they have access to both the seen and unseen categorical attributes from two domains, while our method only employs the seen categories attribute information in the source domain.

5.3.5 Quantitative Analysis

Ablation Study. We compare our complete model with several variants for open-set domain adaptation and semantic recovery open-set domain adaptation tasks to analyze the contribution of each design in our framework, the sensitivity to the hyper-parameters, and the influence of the number of trainable prototypes. (1) First, to explore the contribution of $\mathcal{L}_P$, $\mathcal{L}_R$, $\mathcal{L}_{sep}^{im}$, and the L1 regularization of the classifier $\mathcal{A}_i(\cdot)$ weights, we set the weights of these loss terms as 0 and evaluate the trained

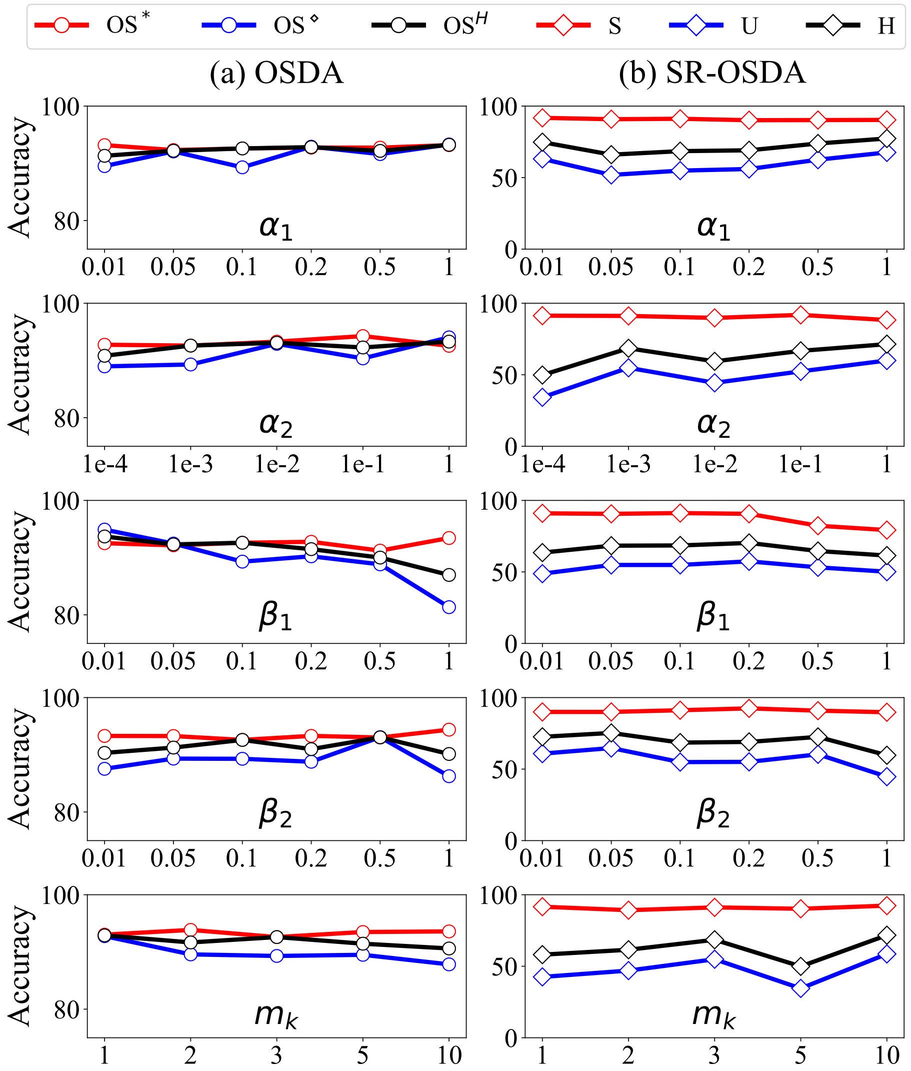


Figure 5.3: Parameters sensitivity analyses of our model (ImplicitModule) for task Real → AwA on DomainNet → AwA dataset.

model for task Real → AwA on DomainNet → AwA dataset, and the results generated by *ImplicitModule* are shown in Fig. 5.2. From the results, we notice that the structure-preserving partial alignment module plays a crucial role in both seen and unseen categories' performance. (2) Second, to analyze the sensitivity of the proposed model to selected hyper-parameters, we vary one of α_1, β_1, and β_2 from 0.01 to 1, or α_2 from $1e^{-4}$ to 1, while keeping the others as default values in our complete model, and report the results as the first 4 rows in Fig. 5.3. It is noteworthy that the performance is not significantly sensitive to the hyper-parameters values in a reasonable range. (3) Moreover, to study the influence of the number of trainable prototypes m_j assigned to the *ImplicitModule*, we vary

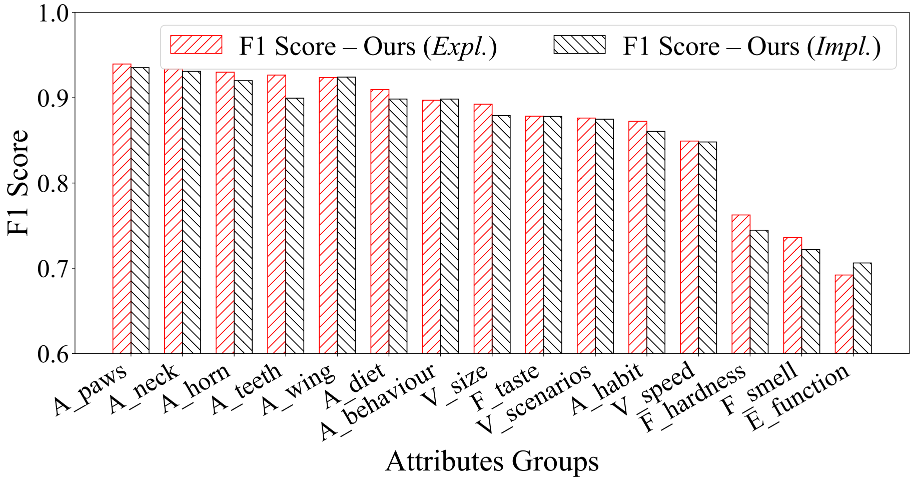
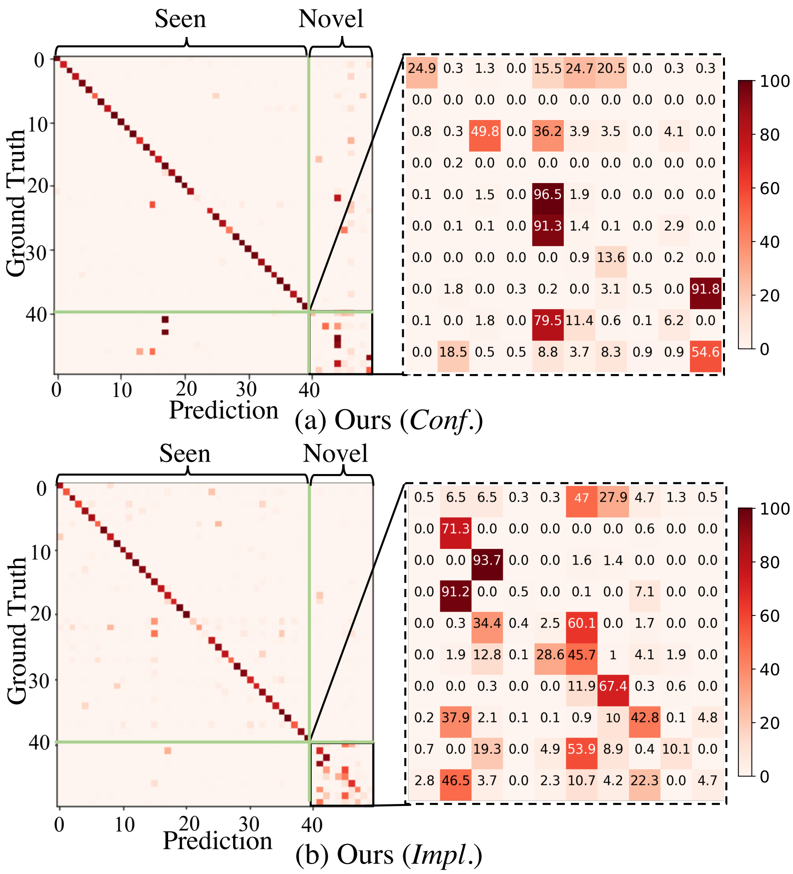


Figure 5.4: Analyses of F1 scores of attributes prediction on different attribute groups for task LAD → Real on DomainNet → LAD dataset.

m_j in $[1, 2, 3, 6, 10]$, and report the results as the bottom row in Fig. 5.3. From the results, we notice that only assigning one prototype for each attribute ($m_j = 1$), the model can perform similarly as more prototypes are assigned for task OSDA. However, for the SR-OSDA results, we observe that more prototypes lead to better results, especially for the novel (U) categories as well as the overall performance (H). It is noteworthy that techniques such as prototype selection, pruning, and sharing strategies are applicable to the proposed model [11, 51, 111].

Attributes Predictability. In Fig. 5.4, we recognize the 253 semantic attributes into several groups describing different types of semantic characteristics of the corresponding category following LAD [169], then report the average F1 scores of predicted attributes from selected groups. From the results, we notice that the model performs better on some attributes groups describing visual characteristics, e.g., "wing." However, for some non-visual patterns, the prediction becomes harder only based on images.

Confusion Matrix. We compare the confusion matrices obtained by the results generated by the *ImplicitModule* in this work, denoted as Ours(*Impl.*), and the conference version, denoted as Ours(*Conf.*). From the results in Figure 5.5, we notice that the learnable semantic prototypes of the implicit interpretable module benefit the attributes recovery and novel categories discovery. For example, Ours(*Impl.*) recognizes 5 out of 10 novel categories (50 classes in total), while Ours(*Conf.*) only recognizes 3 novel categories with accuracies over 30%.



(a) Ours (*Conf.*)

(b) Ours (*Impl.*)

Figure 5.5: Confusion matrix comparison between Ours (Conf.) and Ours (Impl.) on I → AwA dataset.

5.3.6 Qualitative Visualization

Interpretation of Learned Prototypes. To conceptually visualize the learned prototypes in the *ImplicitModule*, we collect samples from both source and target domain with patches most strongly activated by prototype $\mathbf{p}_j^l \in \mathbf{P}_j$ assigned for the j^{th} attribute, and display selected results with activation maps in Fig. 5.6. From the results, we observe the learned prototypes can discover corresponding semantic characteristics from different classes, e.g., the prototype learned for the attribute "Furry" discovers such information from cats and dogs. However, we also observe some cases where the prototypes are not learned as expected, e.g., the bottom row in Fig. 5.6 shows the prototype owe to represent the "Two-wheel" attribute, but focuses on the top tube of the bicycle frame, although the F1 score of the corresponding attribute prediction is 0.93. Thanks to the interpretable *ImplicitModule*, we are able to reveal the black box of the attributes predictor, improving the transparency and

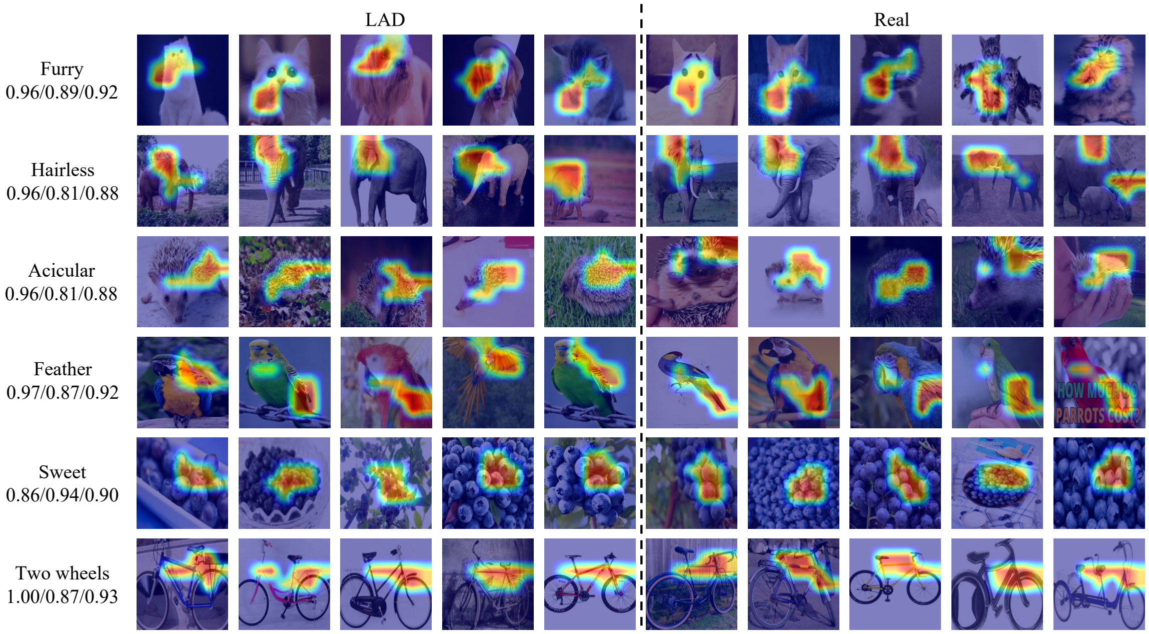


Figure 5.6: Visualization of selected learned prototypes for specific attribute via the nearest samples in LAD and Real domains on DomainNet → LAD dataset. The first column lists the selected attributes and the prediction Precision / Recall/ F1 are reported below the attribute names.

trustworthiness of the framework.

Semantic Recovery for Novel Categories. In this study, we seek to recover semantic attributes from seen categories for novel categories with the same characteristics. In Fig. 5.7, the selected attributes occur in both seen and novel classes in the target domain, and we display some examples with the attention map generated by the most activated prototypes corresponding to specific attributes. For example, in the first column, the learned prototype successfully recognizes the "stripes" on the butterfly and tiger, although the class "tiger" is not known in the training stage.

5.3.7 Discussion and Limitation

This study focuses on exploring a visual prototype-based module to understand the convolutional layers better and establish an interpretable projection connecting visual images with semantic attributes. However, certain limitations and challenges have been identified for future improvements. Specifically, we need to consider the hierarchical structure of high-level semantic descriptions for different category characteristics to enhance representative prototype learning during domain adaptation. Additionally, effectively leveraging implicit high-level semantic information from visual

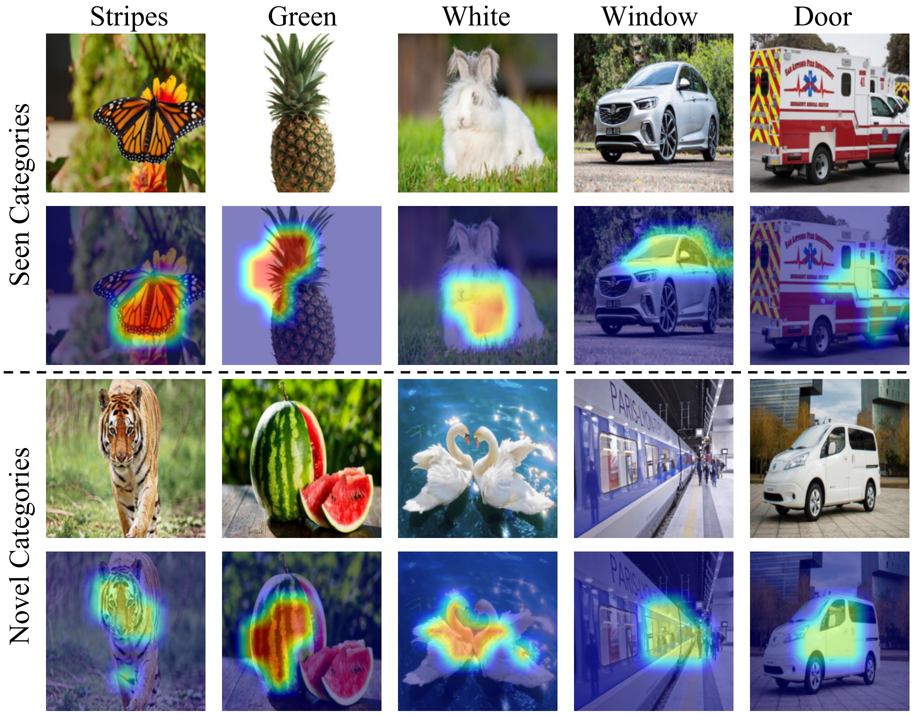


Figure 5.7: Comparison of the recovered attributes and the corresponding activation maps produced by the same learned prototypes between seen and novel categories. All samples are selected from the target domain of task LAD → Real on DomainNet → LAD dataset.

appearances is crucial for improving classification in the target domain. Furthermore, incorporating the spatial relationships of different prototypes' appearances is important for accurate classification based on multiple parts and local patterns.

5.4 Conclusion

In this work, we explored a novel and practical Semantic Recovery Open-Set Domain Adaptation (SR-OSDA) problem identifying target domain data from categories seen in the source domain and recovering the semantic information of the classes unobserved in the source domain with explicit attribute and implicit semantic interpretation. To this end, we proposed a novel framework consisting of an explicit attribute interpretable module and an implicit prototype-based semantic interpretable module optimized by structure-preserving partial alignment, visual-structural semantic attributes propagation, task-driving classification over joint visual-semantic representations, and discriminative prototypes regularization. Finally, three semantic recovery open-set domain adaptation benchmarks were newly constructed to evaluate our model in terms of open-set recognition and semantic attribute recovery.

6

Conclusion and Future Directions

In this dissertation, we have made substantial contributions towards addressing the challenges posed by domain shift and lack of interpretability in visual domain adaptation. Our objective was to improve the accuracy, transparency, and interpretability of transfer learning models. To accomplish this, we explored the interpretation of visual domain adaptation from feature representation analysis to the utilization of multimodal semantic knowledge.

Throughout this study, we presented innovative solutions and frameworks to tackle various transfer learning challenges and explain the domain adaptation process by analyzing the learned domain-invariant feature representations in the latent space. In the context of unsupervised domain adaptation, we proposed the Adversarial Dual Distinct Classifiers Network (AD^{2}CN). This network effectively aligned the data distributions of the source and target domains while preserving category boundaries. Building upon this, we introduced the Adaptively-Accumulated Knowledge Transfer framework (A^{2}KT) to address the label space mismatch problem in partial domain adaptation (PDA). A^{2}KT aligns relevant categories across domains while eliminating the cross-domain data distribution differences. Furthermore, we developed the Augmented Multi-modality Fusion (AMF) framework to transfer knowledge across different modalities and from seen to unseen categories in the context of generalized zero-shot sketch-based image retrieval (GZS-SBIR). This framework efficiently generalized seen concepts to unobserved ones, enhancing the applicability of transfer learning. Additionally, we proposed the Interpretable Action Decision-Making (InAction) model to improve the interpretability of autonomous systems, particularly in the context of action decision-making in autonomous vehicles. InAction aligns human-annotated explanations with the decision-making process, promoting transparency.

Finally, for the first time, we addressed the Semantic-Recovery Open-Set Domain Adaptation (SR-OSDA) problem by presenting a novel framework that accurately identified seen categories in the target domain and recovered semantic attributes for unseen categories. By unraveling the black-box nature of the domain adaptation, this framework provided valuable insights into the knowledge transfer between source and target data with different label spaces. Moreover, we proposed an interpretable framework that employed semantic concept-based visual prototypes to uncover the knowledge transferred across domains.

In conclusion, this dissertation contributes to the development of comprehensive and transparent transfer learning techniques that tackle the challenges of domain shift and lack of interpretability. We have presented innovative solutions and frameworks to enhance the accuracy, transparency, and interpretability of transfer learning models. By fostering collaboration between humans and AI, our work paves the way for the development of responsible and trustworthy AI systems, contributing to advancements in various domains. The insights gained from this research endeavor have the potential to transform the field, ensuring a more transparent and reliable approach to transfer learning.

While this dissertation has made significant progress, several challenges and avenues for future research remain. Further exploration is necessary to align the source and target domains with significant domain shift while preserving performance and equipping the interpretation to the knowledge transfer process. Additionally, integrating ethical considerations and fairness into transfer learning frameworks is an important direction for future investigation. Moreover, Large Language Models (LLMs) have made remarkable strides in recent years, showing exciting possibilities in the integration of visual input and intelligence acquired from vast amounts of language data. These advancements hold immense potential for enhancing knowledge transfer in an open-vocabulary world, all while ensuring that human-friendly explanations are provided simultaneously.

References

[1] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7):1425–1438, 2015.

[2] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

[3] Mahsa Baktashmotlagh, Masoud Faraki, Tom Drummond, and Mathieu Salzmann. Learning factorized representations for open-set domain adaptation. In *Proceedings of the International Conference on Learning Representations*, 2019.

[4] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1563–1572, 2016.

[5] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[6] Silvia Bucci, Mohammad Reza Loghmani, and Tatiana Tommasi. On the effectiveness of image rotation for open set domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 422–438. Springer, 2020.

[7] Ruichu Cai, Zijian Li, Pengfei Wei, Jie Qiao, Kun Zhang, and Zhifeng Hao. Learning disentangled semantic representation for domain adaptation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 2060–2066, 7 2019.

[8] Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Partial transfer learning with selective adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2724–2732, 2018.

[9] Zhangjie Cao, Lijia Ma, Mingsheng Long, and Jianmin Wang. Partial adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 135–150, 2018.

[10] Zhangjie Cao, Kaichao You, Mingsheng Long, Jianmin Wang, and Qiang Yang. Learning to transfer examples for partial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2985–2994, 2019.

[11] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan Su. This looks like that: Deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems*, pages 8928–8939, 2019.

[12] Minghao Chen, Shuai Zhao, Haifeng Liu, and Deng Cai. Adversarial-learned loss for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3521–3528, 2020.

[13] Tina Chen, Renran Tian, Yaobin Chen, Joshua Domeyer, Heishiro Toyoda, Rini Sherony, Taotao Jing, and Zhengming Ding. Psi: A pedestrian behavior dataset for socially intelligent autonomous car. *arXiv preprint arXiv:2112.02604*, 2021.

[14] Xingyu Chen, Xuguang Lan, Fuchun Sun, and Nanning Zheng. A boundary based out-of-distribution classifier for generalized zero-shot learning. In *Proceedings of the European Conference on Computer Vision*, pages 572–588. Springer, 2020.

[15] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *Proceedings of the International Conference on Machine Learning*, pages 1081–1090, 2019.

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

[17] Jinhong Deng, Wen Li, Yuhua Chen, and Lixin Duan. Unbiased mean teacher for cross-domain object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4091–4101, 2021.

[18] Zhijie Deng, Yucen Luo, and Jun Zhu. Cluster alignment with a teacher for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, October 2019.

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics, 2019.

[20] Zhengming Ding, Sheng Li, Ming Shao, and Yun Fu. Graph adaptive knowledge transfer for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 37–52, 2018.

[21] Zhengming Ding and Hongfu Liu. Marginalized latent semantic encoder for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6191–6199, 2019.

[22] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *Proceedings of the International Conference on Machine Learning*, pages 647–655, 2014.

[23] Jiahua Dong, Yang Cong, Gan Sun, Yuyang Liu, and Xiaowei Xu. Cscl: Critical semantic-consistent learning for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 745–762. Springer, 2020.

[24] Alexey Dosovitskiy and Thomas Brox. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4829–4837, 2016.

[25] Mohamed Elhoseiny and Mohamed Elfeki. Creativity inspired zero-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5784–5793, 2019.

[26] H. Feng, M. Chen, J. Hu, D. Shen, H. Liu, and D. Cai. Complementary pseudo labels for unsupervised domain adaptation on person re-identification. *IEEE Transactions on Image Processing*, pages 1–1, 2021.

[27] Qianyu Feng, Guoliang Kang, Hehe Fan, and Yi Yang. Attract or distract: Exploit the margin of open set. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7990–7999, 2019.

[28] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3429–3437, 2017.

[29] Geoff French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *Proceedings of the International Conference on Learning Representations*, 2018.

[30] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the International Conference on Machine Learning*, pages 1180–1189, 2015.

[31] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[32] Rui Gao, Xingsong Hou, Jie Qin, Jiaxin Chen, Li Liu, Fan Zhu, Zhao Zhang, and Ling Shao. Zero-vae-gan: Generating unseen features for generalized and transductive zero-shot learning. *IEEE Transactions on Image Processing*, 29:3665–3680, 2020.

[33] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 597–613. Springer, 2016.

[34] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[35] 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.

[36] 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.

[37] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Proceedings of the European Conference on Computer Vision*, pages 630–645. Springer, 2016.

[38] Judy Hoffman, Sergio Guadarrama, Eric S Tzeng, Ronghang Hu, Jeff Donahue, Ross Girshick, Trevor Darrell, and Kate Saenko. Lsda: Large scale detection through adaptation. In *Advances in Neural Information Processing Systems*, pages 3536–3544, 2014.

[39] Yan Hong, Li Niu, Jianfu Zhang, Weijie Zhao, Chen Fu, and Liqing Zhang. F2gan: Fusing-and-filling gan for few-shot image generation. In *Proceedings of the ACM International Conference on Multimedia*, pages 2535–2543, 2020.

[40] He Huang, Changhu Wang, Philip S Yu, and Chang-Dong Wang. Generative dual adversarial network for generalized zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 801–810, 2019.

[41] Johan Ludwig William Valdemar Jensen et al. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta mathematica*, 30:175–193, 1906.

[42] Huajie Jiang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Transferable contrastive network for generalized zero-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9765–9774, 2019.

[43] Xiang Jiang, Qicheng Lao, Stan Matwin, and Mohammad Havaei. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning*, pages 4816–4827. PMLR, 2020.

[44] Taotao Jing and Zhengming Ding. Adversarial dual distinct classifiers for unsupervised domain adaptation. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 605–614, 2021.

[45] Taotao Jing, Hongfu Liu, and Zhengming Ding. Towards novel target discovery through open-set domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9322–9331, 2021.

[46] Taotao Jing, Haifeng Xia, and Zhengming Ding. Adaptively-accumulated knowledge transfer for partial domain adaptation. In *Proceedings of the ACM International Conference on Multimedia*, pages 1606–1614, 2020.

[47] Taotao Jing, Haifeng Xia, Jihun Hamm, and Zhengming Ding. Augmented multi-modality fusion for generalized zero-shot sketch-based visual retrieval. *IEEE Transactions on Image Processing*, 2022.

[48] Taotao Jing, Haifeng Xia, Jihun Hamm, and Zhengming Ding. Marginalized augmented few-shot domain adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[49] Taotao Jing, Haifeng Xia, Renran Tian, Haoran Ding, Xiao Luo, Joshua Domeyer, Rini Sherony, and Zhengming Ding. Inaction: Interpretable action decision making for autonomous driving. In *Proceedings of the European Conference on Computer Vision*, pages 370–387. Springer, 2022.

[50] Taotao Jing, Bingrong Xu, and Zhengming Ding. Towards fair knowledge transfer for imbalanced domain adaptation. *IEEE Transactions on Image Processing*, 30:8200–8211, 2021.

[51] Eunji Kim, Siwon Kim, Minji Seo, and Sungroh Yoon. Xprotonet: diagnosis in chest radiography with global and local explanations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 15719–15728, 2021.

[52] Jinkyu Kim and John Canny. Interpretable learning for self-driving cars by visualizing causal attention. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2942–2950, 2017.

[53] Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. Textual explanations for self-driving vehicles. In *Proceedings of the European Conference on Computer Vision*, pages 563–578, 2018.

[54] Y. Kim and S. Hong. Adaptive graph adversarial networks for partial domain adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2021.

[55] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2015.

[56] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[57] Jogendra Nath Kundu, Naveen Venkat, Ambareesh Revanur, R Venkatesh Babu, et al. Towards inheritable models for open-set domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12376–12385, 2020.

[58] Vinod Kumar Kurmi, Shanu Kumar, and Vinay P Namboodiri. Attending to discriminative certainty for domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 491–500, 2019.

[59] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958. IEEE, 2009.

[60] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2013.

[61] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[62] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10285–10295, 2019.

[63] Wee Sun Lee, Peter L Bartlett, and Robert C Williamson. Lower bounds on the vc dimension of smoothly parameterized function classes. *Neural Computation*, 7(5):1040–1053, 1995.

[64] Jiangtong Li, Zhixin Ling, Li Niu, and Liqing Zhang. Zero-shot sketch-based image retrieval with structure-aware asymmetric disentanglement. *arXiv preprint arXiv:1911.13251*, 2019.

[65] Jingjing Li, Ke Lu, Zi Huang, Lei Zhu, and Heng Tao Shen. Transfer independently together: A generalized framework for domain adaptation. *IEEE Transactions on Cybernetics*, 49(6):2144–2155, 2018.

[66] Shuang Li, Chi Harold Liu, Qiuxia Lin, Qi Wen, Limin Su, Gao Huang, and Zhengming Ding. Deep residual correction network for partial domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020.

[67] Shuang Li, Chi Harold Liu, Binhui Xie, Limin Su, Zhengming Ding, and Gao Huang. Joint adversarial domain adaptation. In *Proceedings of the ACM International Conference on Multimedia*, MM '19, page 729–737, New York, NY, USA, 2019. Association for Computing Machinery.

[68] Shuang Li, Shiji Song, Gao Huang, Zhengming Ding, and Cheng Wu. Domain invariant and class discriminative feature learning for visual domain adaptation. *IEEE Transactions on Image Processing*, 27(9):4260–4273, 2018.

[69] Shuang Li, Mixue Xie, Kaixiong Gong, Chi Harold Liu, Yulin Wang, and Wei Li. Transferable semantic augmentation for domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11516–11525, 2021.

[70] Yu-Jhe Li, Xiaoliang Dai, Chih-Yao Ma, Yen-Cheng Liu, Kan Chen, Bichen Wu, Zijian He, Kris Kitani, and Peter Vajda. Cross-domain adaptive teacher for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7581–7590, 2022.

[71] Jian Liang, Ran He, Zhenan Sun, and Tieniu Tan. Distant supervised centroid shift: A simple and efficient approach to visual domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[72] Hong Liu, Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Qiang Yang. Separate to adapt: Open set domain adaptation via progressive separation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2927–2936, 2019.

[73] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang. Learning to propagate labels: Transductive propagation network for few-shot learning. *arXiv preprint arXiv:1805.10002*, 2018.

[74] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. In *Proceedings of the International Conference on Machine Learning*, pages 97–105, 2015.

[75] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 1640–1650, 2018.

[76] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 1647–1657, 2018.

[77] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S Yu. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2200–2207, 2013.

[78] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*, pages 136–144, 2016.

[79] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the International Conference on Machine Learning*, pages 2208–2217. JMLR. org, 2017.

[80] Yadan Luo, Zijian Wang, Zi Huang, and Mahsa Baktashmotlagh. Progressive graph learning for open-set domain adaptation. In *Proceedings of the International Conference on Machine Learning*, pages 6468–6478. PMLR, 2020.

[81] M Mancini, MF Naeem, Y Xian, and Zeynep Akata. Open world compositional zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2021.

[82] Devraj Mandal, Kunal N. Chaudhury, and Soma Biswas. Generalized semantic preserving hashing for cross-modal retrieval. *IEEE Transactions on Image Processing*, 28(1):102–112, 2019.

[83] Yao Ming, Panpan Xu, Huamin Qu, and Liu Ren. Interpretable and steerable sequence learning via prototypes. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 903–913, 2019.

[84] Saeid Motiian, Quinn Jones, Seyed Iranmanesh, and Gianfranco Doretto. Few-shot adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 6670–6680, 2017.

[85] Saeid Motiian, Quinn Jones, Seyed Iranmanesh, and Gianfranco Doretto. Few-shot adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 6670–6680, 2017.

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

[87] Zak Murez, Soheil Kolouri, David Kriegman, Ravi Ramamoorthi, and Kyungnam Kim. Image to image translation for domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4500–4509, 2018.

[88] Sanath Narayan, Akshita Gupta, Fahad Shahbaz Khan, Cees GM Snoek, and Ling Shao. Latent embedding feedback and discriminative features for zero-shot classification. In *Proceedings of the European Conference on Computer Vision*, 2020.

[89] Meike Nauta, Ron van Bree, and Christin Seifert. Neural prototype trees for interpretable fine-grained image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 14933–14943, 2021.

[90] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. *Advances in Neural Information Processing Systems*, 29:3387–3395, 2016.

[91] Tuan Nguyen, Trung Le, He Zhao, Quan Hung Tran, Truyen Nguyen, and Dinh Phung. Most: Multi-source domain adaptation via optimal transport for student-teacher learning. In *Uncertainty in Artificial Intelligence*, pages 225–235. PMLR, 2021.

[92] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11):e7, 2017.

[93] Daniel Omeiza, Helena Webb, Marina Jirotka, and Lars Kunze. Explanations in autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–21, 2021.

[94] OpenAI. Gpt-4 technical report, 2023.

[95] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1717–1724, 2014.

[96] Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. Zero-shot learning with semantic output codes. *Advances in Neural Information Processing Systems*, 22, 2009.

[97] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *TKDE*, 22(10):1345–1359, 2010.

[98] Yingwei Pan, Ting Yao, Yehao Li, Chong-Wah Ngo, and Tao Mei. Exploring category-agnostic clusters for open-set domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 13867–13875, 2020.

[99] Pau Panareda Busto and Juergen Gall. Open set domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 754–763, 2017.

[100] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[101] Kuan-Chuan Peng, Ziyan Wu, and Jan Ernst. Zero-shot deep domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 764–781, 2018.

[102] 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 International Conference on Computer Vision*, pages 1406–1415, 2019.

[103] Fan Qi, Xiaoshan Yang, and Changsheng Xu. A unified framework for multimodal domain adaptation. In *Proceedings of the ACM International Conference on Multimedia*, pages 429–437, 2018.

[104] Sayan Rakshit, Dipesh Tamboli, Pragati Shuddhodhan Meshram, Biplab Banerjee, Gemma Roig, and Subhasis Chaudhuri. Multi-source open-set deep adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 735–750. Springer, 2020.

[105] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.

[106] Luca Robbiano, Muhammad Rameez Ur Rahman, Fabio Galasso, Barbara Caputo, and Fabio Maria Carlucci. Adversarial branch architecture search for unsupervised domain adaptation. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 2918–2928, 2022.

[107] Pau Rodríguez, Issam Laradji, Alexandre Drouin, and Alexandre Lacoste. Embedding propagation: Smoother manifold for few-shot classification. In *Proceedings of the European Conference on Computer Vision*, pages 121–138. Springer, 2020.

[108] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, June 2022.

[109] Subhankar Roy, Aliaksandr Siarohin, Enver Sangineto, Samuel Rota Bulo, Nicu Sebe, and Elisa Ricci. Unsupervised domain adaptation using feature-whitening and consensus loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9471–9480, 2019.

[110] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

[111] Dawid Rymarczyk, Łukasz Struski, Jacek Tabor, and Bartosz Zieliński. Protopshare: Prototypical parts sharing for similarity discovery in interpretable image classification. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1420–1430, 2021.

[112] Kuniaki Saito, Donghyun Kim, Piotr Teterwak, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Tune it the right way: Unsupervised validation of domain adaptation via soft neighborhood density. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9184–9193, 2021.

[113] 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.

[114] Kuniaki Saito, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada. Open set domain adaptation by backpropagation. In *Proceedings of the European Conference on Computer Vision*, pages 153–168, 2018.

[115] Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8503–8512, 2018.

[116] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8247–8255, 2019.

[117] Candice Schumann, Xuezhi Wang, Alex Beutel, Jilin Chen, Hai Qian, and Ed H Chi. Transfer of machine learning fairness across domains. *arXiv preprint arXiv:1906.09688*, 2019.

[118] Ozan Sener, Hyun Oh Song, Ashutosh Saxena, and Silvio Savarese. Learning transferrable representations for unsupervised domain adaptation. In *Advances in Neural Information Processing Systems*, pages 2110–2118, 2016.

[119] Yuming Shen, Li Liu, Fumin Shen, and Ling Shao. Zero-shot sketch-image hashing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3598–3607, 2018.

[120] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[121] Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017.

[122] Rosanna Soentpiet et al. *Advances in kernel methods: support vector learning*. MIT press, 1999.

[123] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[124] Ardi Tampuu, Tambet Matiisen, Maksym Semikin, Dmytro Fishman, and Naveed Muhammad. A survey of end-to-end driving: Architectures and training methods. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[125] Shuhan Tan, Jiening Jiao, and Wei-Shi Zheng. Weakly supervised open-set domain adaptation by dual-domain collaboration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5394–5403, 2019.

[126] Hui Tang, Ke Chen, and Kui Jia. Unsupervised domain adaptation via structurally regularized deep clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8725–8735, 2020.

[127] Hui Tang and Kui Jia. Discriminative adversarial domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5940–5947, 2020.

[128] William Thong, Pascal Mettes, and Cees GM Snoek. Open cross-domain visual search. *Computer Vision and Image Understanding*, 200:103045, 2020.

[129] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015.

[130] 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*, volume 1, page 4, 2017.

[131] Paul Upchurch, Jacob Gardner, Geoff Pleiss, Robert Pless, Noah Snavely, Kavita Bala, and Kilian Weinberger. Deep feature interpolation for image content changes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7064–7073, 2017.

[132] 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.

[133] Vinay Kumar Verma, Gundeep Arora, Ashish Mishra, and Piyush Rai. Generalized zero-shot learning via synthesized examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4281–4289, 2018.

[134] Maunil R Vyas, Hemanth Venkateswara, and Sethuraman Panchanathan. Leveraging seen and unseen semantic relationships for generative zero-shot learning. In *Proceedings of the European Conference on Computer Vision*, pages 70–86. Springer, 2020.

[135] Dequan Wang, Coline Devin, Qi-Zhi Cai, Philipp Krähenbühl, and Trevor Darrell. Monocular plan view networks for autonomous driving. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 2876–2883. IEEE, 2019.

[136] Dequan Wang, Coline Devin, Qi-Zhi Cai, Fisher Yu, and Trevor Darrell. Deep object-centric policies for autonomous driving. In *Proceedngs of the IEEE International Conference on Robotics and Automation*, pages 8853–8859. IEEE, 2019.

[137] Jinghua Wang, Ming-Ming Cheng, and Jianmin Jiang. Domain shift preservation for zero-shot domain adaptation. *IEEE Transactions on Image Processing*, 30:5505–5517, 2021.

[138] Lichen Wang, Bin Sun, Joseph Robinson, Taotao Jing, and Yun Fu. Ev-action: Electromyography-vision multi-modal action dataset. In *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition*, pages 160–167. IEEE, 2020.

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

[140] Yulin Wang, Xuran Pan, Shiji Song, Hong Zhang, Gao Huang, and Cheng Wu. Implicit semantic data augmentation for deep networks. In *Advances in Neural Information Processing Systems*, pages 12614–12623, 2019.

[141] Yuan Wu, Diana Inkpen, and Ahmed El-Roby. Dual mixup regularized learning for adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 540–555. Springer, 2020.

[142] Haifeng Xia and Zhengming Ding. Hgnet: Hybrid generative network for zero-shot domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 55–70. Springer, 2020.

[143] Haifeng Xia and Zhengming Ding. Structure preserving generative cross-domain learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4364–4373, 2020.

[144] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4582–4591, 2017.

[145] Guo-Sen Xie, Li Liu, Fan Zhu, Fang Zhao, Zheng Zhang, Yazhou Yao, Jie Qin, and Ling Shao. Region graph embedding network for zero-shot learning. In *Proceedings of the European Conference on Computer Vision*, pages 562–580. Springer, 2020.

[146] Huazhe Xu, Yang Gao, Fisher Yu, and Trevor Darrell. End-to-end learning of driving models from large-scale video datasets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2174–2182, 2017.

[147] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. *arXiv preprint arXiv:1912.01805*, 2019.

[148] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6502–6509, 2020.

[149] Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1426–1435, 2019.

[150] Xiang Xu, Xiong Zhou, Ragav Venkatesan, Gurumurthy Swaminathan, and Orchid Majumder. d-sne: Domain adaptation using stochastic neighborhood embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2497–2506, 2019.

[151] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2272–2281, 2017.

[152] Guanglei Yang, Haifeng Xia, Mingli Ding, and Zhengming Ding. Bi-directional generation for unsupervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6615–6622, 2020.

[153] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, pages 3320–3328, 2014.

[154] Kaichao You, Ximei Wang, Mingsheng Long, and Michael Jordan. Towards accurate model selection in deep unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning*, pages 7124–7133. PMLR, 2019.

[155] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2020.

[156] Hyeonwoo Yu and Beomhee Lee. Zero-shot learning via simultaneous generating and learning. *Advances in Neural Information Processing Systems*, 32:46–56, 2019.

[157] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, 8:58443–58469, 2020.

[158] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the European Conference on Computer Vision*, pages 818–833. Springer, 2014.

[159] Hanwang Zhang, Xindi Shang, Wenzhuo Yang, Huan Xu, Huanbo Luan, and Tat-Seng Chua. Online collaborative learning for open-vocabulary visual classifiers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2809–2817, 2016.

[160] Jing Zhang, Zewei Ding, Wanqing Li, and Philip Ogunbona. Importance weighted adversarial nets for partial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8156–8164, 2018.

[161] 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.

[162] Yabin Zhang, Bin Deng, Kui Jia, and Lei Zhang. Label propagation with augmented anchors: A simple semi-supervised learning baseline for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 781–797. Springer, 2020.

[163] Yabin Zhang, Hui Tang, Kui Jia, and Mingkui Tan. Domain-symmetric networks for adversarial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5031–5040, 2019.

[164] Yabin Zhang, Hui Tang, Kui Jia, and Mingkui Tan. Domain-symmetric networks for adversarial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5031–5040, 2019.

[165] Yu Zhang, Peter Tiňo, Aleš Leonardis, and Ke Tang. A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2021.

[166] Zhaolong Zhang, Yuejie Zhang, Rui Feng, Tao Zhang, and Weiguo Fan. Zero-shot sketch-based image retrieval via graph convolution network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12943–12950, 2020.

[167] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4166–4174, 2015.

[168] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via joint latent similarity embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6034–6042, 2016.

[169] Bo Zhao, Yanwei Fu, Rui Liang, Jiahong Wu, Yonggang Wang, and Yizhou Wang. A large-scale attribute dataset for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[170] Heliang Zheng, Jianlong Fu, Tao Mei, and Jiebo Luo. Learning multi-attention convolutional neural network for fine-grained image recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5209–5217, 2017.

[171] Erheng Zhong, Wei Fan, Qiang Yang, Olivier Verscheure, and Jiangtao Ren. Cross validation framework to choose amongst models and datasets for transfer learning. In *Proceedings of the Machine Learning and Knowledge Discovery in Databases: European Conference*, pages 547–562. Springer, 2010.

[172] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929, 2016.

[173] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Interpretable basis decomposition for visual explanation. In *Proceedings of the European Conference on Computer Vision*, pages 119–134, 2018.

[174] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. *Advances in Neural Information Processing Systems*, 16(16):321–328, 2004.

[175] Yizhe Zhu, Jianwen Xie, Bingchen Liu, and Ahmed Elgammal. Learning feature-to-feature translator by alternating back-propagation for generative zero-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9844–9854, 2019.

[176] Junbao Zhuo, Shuhui Wang, Shuhao Cui, and Qingming Huang. Unsupervised open domain recognition by semantic discrepancy minimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 750–759, 2019.