UNIVERSIDADE FEDERAL DE MINAS GERAIS
Instituto de Ciências Exatas
Programa de Pós-Graduação em Ciência da Computação

Renato Sérgio Lopes Júnior

Enhancing Domain Adaptation on Visual Data

Belo Horizonte
2023

Renato Sérgio Lopes Júnior

Enhancing Domain Adaptation on Visual Data

Final Version

Thesis presented to the Graduate Program in Computer Science of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Master in Computer Science.

Advisor: William Robson Schwartz

Belo Horizonte
2023

Lopes Júnior, Renato Sérgio.

L864e Enhancing domain adaptation on visual data [recurso eletrônico] / Renato Sérgio Lopes Júnior – 2023.
1 recurso online (66 f. il., color.) : pdf.

Orientador: William Robson Schwartz

Dissertação(Mestrado) - Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciências da Computação.
Referências: f.62-66

1. Computação – Teses. 2. Aprendizado do computador – Teses. 3. Visão por computador - Teses. 4. Processamento de imagens – Técnicas digitais – Teses. 5. Aprendizado profundo - teses. 6. Redes neurais convolucionais - Teses. I. Schwartz, William Robson. II. Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Computação. III.Título.

CDU 519.6*84(043)

Ficha catalográfica elaborada pela bibliotecária Irénquer Vismeg Lucas Cruz
CRB 6/819 - Universidade Federal de Minas Gerais - ICEx

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

ENHANCING DOMAIN ADAPTATION ON VISUAL DATA

RENATO SÉRGIO LOPES JÚNIOR

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

Prof. William Robson Schwartz - Orientador
Departamento de Ciência da Computação - UFMG

Prof. Guillermo Câmara Chávez
Departamento de Computação - UFOP

Prof. Pedro Olmo Stancioli Vaz de Melo
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 19 de julho de 2023.

Documento assinado eletronicamente por **William Robson Schwartz**, **Professor do Magistério Superior**, em 25/08/2023, às 15:17, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Pedro Olmo Stancioli Vaz de Melo**, **Professor do Magistério Superior**, em 30/08/2023, às 11:40, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Guillermo Camara Chavez**, **Usuário Externo**, em 14/09/2023, às 17:22, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

A autenticidade deste documento pode ser conferida no site https://sei.ufmg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **2474177** e o código CRC **88CC7CC2**.

I dedicate this work to my beloved parents, Renato and Vanda Lopes, who always supported me throughout my studies.

Acknowledgments

Many people contributed to the development of this work. I first would like to thank my beloved parents, Renato and Vanda Lopes, for assisting and encouraging me throughout my studies.

I also express my deep gratitude to Professor Dr. William Robson Schwartz for being my advisor and for always supporting me in the journey that resulted in this work. I also thank Professor Dr. Guillermo Cámara Chávez and Professor Dr. Pedro Olmo Stancioli Vaz de Melo for being part of the Examination Board of this thesis and for providing insightful comments that further enriched this work.

To all my colleagues in the Smart Sense Laboratory, thanks for all the discussions and all the support given to me during this period. It is a fantastic team, and I loved to share all those incredible moments with you all.

I want to thank all the faculty and staff of the Graduate Program in Computer Science of the Federal University of Minas Gerais for all their assistance during these years. I also thank Petrobras and all those who worked on the SS-SMS project developed in collaboration with the Smart Sense Laboratory for the images we used in our case study.

Finally, I also would like to thank the National Council for Scientific and Technological Development – CNPq (Grant 309953/2019-7) and the Minas Gerais Research Foundation – FAPEMIG (Grant PPM-00540-17).

Resumo

Recentemente, as redes neurais profundas têm sido amplamente utilizadas para resolver uma variedade de problemas em diferentes áreas. Por exemplo, as redes neurais convolucionais mudaram completamente o cenário da Visão Computacional, alcançando resultados notáveis em tarefas como classificação de imagens e detecção de objetos. No entanto, para se obter bons resultados, é necessária uma grande quantidade de dados rotulados para treinar estas redes, o que constitui um dos principais obstáculos na sua adoção, uma vez que coletar e rotular esta grande quantidade de dados pode consumir muito tempo e recursos. Portanto, os métodos de adaptação de domínio usam dados rotulados que já estão disponíveis em um domínio de origem diferente, mas semanticamente relacionado, para treinar um modelo que possa fazer previsões corretas sobre os dados nos quais estamos interessados, o domínio de destino, evitando assim o alto custo de rotulagem. Este trabalho apresenta duas novas abordagens para melhorar ainda mais o desempenho de adaptação em domínios visuais na tarefa de classificação de imagens. Além disso, também realizamos um estudo de caso para investigar a viabilidade de realizar adaptação de domínio em um cenário do mundo real, considerando a tarefa de detecção automática do uso de Equipamentos de Proteção Individual com redes neurais convolucionais. Experimentos demonstram que nossas abordagens propostas são capazes de melhorar os resultados dos seus métodos base e fornecer *insights* significativos para trabalhos futuros sobre adaptação de domínio.

Palavras-chave: Adaptação de Domínio. Transferência de Aprendizado. Aprendizado de Máquina. Aprendizado Profundo. Visão Computacional. Processamento Digital de Imagens.

Abstract

Recently, deep neural networks have been extensively used to solve a variety of problems in different areas. For instance, convolutional neural networks have completely changed the landscape of the Computer Vision field by achieving remarkable results in tasks such as image classification and object detection. However, to obtain good results, a large amount of labeled data is necessary to train these networks, thus constituting one of the main obstacles in their adoption, as gathering and labeling this large amount of data can be very time and resource consuming. Therefore, domain adaptation methods leverage labeled data that are already available from a different, but semantically related, source domain to train a model that can correctly make predictions on the data in which we are interested, the target domain, thus skipping the high labeling cost. This work presents two new approaches for further enhancing the adaptation performance on visual domains in the image classification task. Furthermore, we also conduct a case study to investigate the viability of performing domain adaptation in a real-world scenario considering the task of automatic Personal Protective Equipment usage detection with convolutional neural networks. Experiments demonstrate that our proposed approaches are able to improve their baseline results and provide meaningful insights for future works on domain adaptation.

Keywords: Domain Adaptation. Transfer Learning. Machine Learning. Deep Learning. Computer Vision. Digital Image Processing.

List of Figures

List of Tables

Contents

Chapter 1

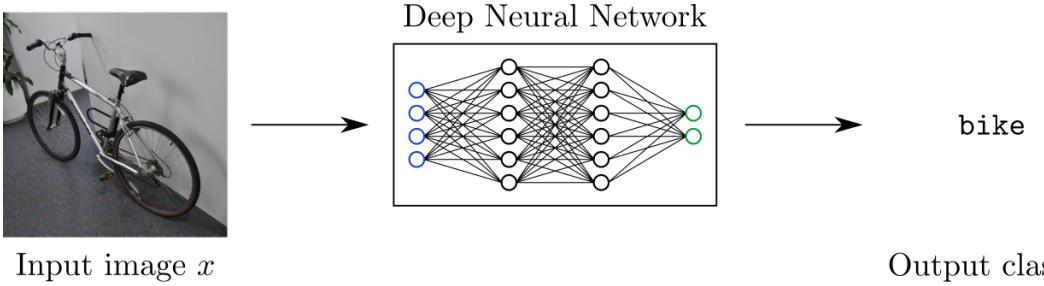
Introduction

In recent times, the development of smart computer systems that automate tasks in different areas has been on the rise. Examples of these systems can be found throughout the modern world, such as self-driving cars, recommendation engines on online stores, and smart surveillance setups. Machine learning models and algorithms are extensively used to power those systems, providing ways to extract knowledge from large data sets that are then used to automate various tasks [45].

Many real-world applications are based on visual data, such as images and videos. Hence, computer vision techniques are also employed in combination with machine learning in order to successfully extract the information from this kind of data. One such application is image classification, in which the computer system must be able to correctly classify an image based on what it depicts. In practice, such a system can be used to automatically organize images of a store's catalog or to automate the sorting of documents in a government department.

Currently, the main machine learning model employed by intelligent systems is the Deep Neural Networks. These networks are modeled based on how the human brain is structured and are able to achieve remarkable results, which can even be super-human levels in some scenarios [16]. Convolutional Neural Networks (CNNs) are a variant of deep neural networks devised to work on visual data and are extensively used in different computer vision applications [44]. Figure 1.1 illustrates how a deep neural network can perform image classification by assigning class labels to the input picture based on what it depicts.

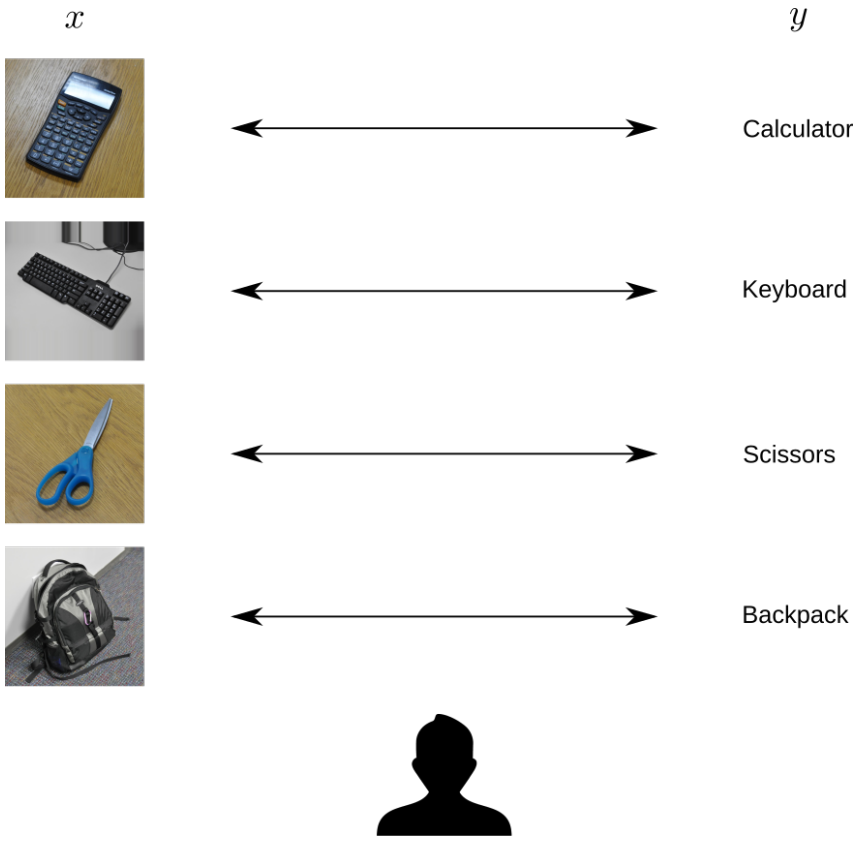
Figure 1.1: Deep Neural Network for image classification.



In order for a neural network to achieve great results in the task being performed,

it must be trained on a large data set. During this training, learning algorithms are employed to optimize the network's inner parameters so that it is able to extract the knowledge that is embedded in the data, thus learning how to perform the task. The training of neural networks constitutes one of the greatest obstacles in the development of a smart system, as gathering, organizing, and labeling the large amount of data that is needed during this procedure is very time and resource consuming [32, 44]. For instance, in the previously presented image classification task, in order to train the network, one would need a large data set containing pairs of images and labels, where each label indicates the class portrayed in each image. Labeling these data is a very demanding task, as a human operator needs to manually assign the label for each picture, thus significantly impacting the time and cost of the development of the smart system [44]. This manual labeling procedure is illustrated in Figure 1.2.

Figure 1.2: Manually labeling training instances.



A possible way to bypass this obstacle is to train the neural network using annotated data that are already available from different, but semantically related, domains and tasks. This is a viable solution as large amounts of data from different domains and with different characteristics are available on the Internet and in public repositories due to the *big data phenomenon* of the past decades [44].

An issue with the aforementioned solution is that, in most scenarios, the data that are available and will be used during training, the **source** domain data, and the data

on which the model will be used to make predictions, the **target** domain data, will have different characteristics, in what is known as *Domain Shift*. In visual data, this shift can be observed in the difference in the characteristics of the images, such as quality, illumination, and pose [44].

Figure 1.3 shows an example of the domain shift in an image data set, in which the source domain images are taken from an online store catalog, while the target images are taken with a digital camera in an office. The visual differences between the images, like their quality, the presence of background, and the pose/orientation of the object, illustrate the shift across the domains. Note that even though the data distributions are different, the images in each column are semantically related, as they represent the same object category: bike, monitor, and scissors, respectively.

Figure 1.3: Example of domain shift.

(a) Samples from the Source domain.

(b) Samples from the Target domain.

Images extracted from the Office-31 data set [37].

The shift in the data distribution will severely impact the performance on target samples of a model trained using only source domain data, as the assumption made by most machine learning algorithms that the training and test data are independent and identically distributed will not hold [16]. Even deep neural networks, which have great generalization capacity and learn more transferable features when compared to other learning methods, suffer from this shift in the distributions, as the deep features will eventually transition from general to specific in the higher layers of the network, diminishing their transferability [4, 44].

Domain Adaptation (DA) methods offer a solution to the domain shift problem, allowing for an effective transfer of the source domain knowledge into the target domain. These methods propose changes to the network's architecture and training procedure to circumvent the negative effects of the data shift, hence adapting the knowledge embedded in the data across the domains. The main goal of DA is to achieve a great performance in target data without the need to label a large number of samples, thus skipping the previously mentioned high labeling cost and making the development of smart systems more accessible.

In the literature, the DA task is categorized based on the level of divergence between the domains and the availability of labeled data in the target domain [32, 44]. Concerning the level of divergence, the different DA scenarios can be divided into *homogeneous*, in which the input space has the same dimensionality in both source and target domains, and the number of classes and each class concept does not change between the domains; and *heterogeneous*, in which the input and label spaces can be different across the domains. Note that heterogeneous DA is a more general setting in which there are no restrictions on how the domains can diverge, while in homogeneous DA the dimensionality of both input and output spaces do not vary between the domains. Hence, in the homogeneous setting, the domain shift is restricted to a shift in the data distributions, making the adaptation task more tangible [16].

The different DA settings can be further categorized based on the availability of data in the target domain: in *unsupervised* DA, there are no target labeled samples and adaptation is done using only target unlabeled samples and the source labeled ones; in a *semi-supervised* setting, it is assumed that a small amount of target labeled samples is available and it is used together with the unlabeled ones during the adaptation procedure; finally, there is *supervised* DA, which is similar to the semi-supervised setting, but there are no target unlabeled samples, hence adaptation is done using only the source domain data and the few target labeled samples that are available. Notice that in all these scenarios it is assumed that a large amount of source labeled data is available and that in the semi-supervised and supervised settings, the amount of target labeled samples available is not enough to train the model from scratch, hence adaptation from source data is still required.

In this work, an overview of commonly used approaches for performing DA using convolutional neural networks is presented and two new methods that build upon previous literature works for solving the DA task with visual domains are proposed. One of these approaches leverages dimensionality reduction and better use of the data available in each domain to increase the efficacy and robustness of the adaptation procedure, thus leading to better performance on the target domain data. The second approach improves the adversarial framework for DA by incorporating the source ground-truth labels and target estimated pseudo-labels into the domain discrimination procedure, leading to class-aware

domain confusion. The approaches proposed in this work tackle specifically the image classification task and the homogeneous and unsupervised DA scenario. Experimental results show that the proposed methods were able to enhance the baseline classification accuracy on target data in commonly used DA benchmarks.

Furthermore, we also conducted a study case to investigate the viability of performing unsupervised DA in a real-world setting. To this end, we experiment with training convolutional neural networks to perform automatic Personal Protective Equipment usage detection by using labeled data captured in a simulated environment and unlabeled data captured using surveillance cameras in a real-world workshop where actual workers are performing their usual routines.

1.1 Motivation

As noted in the previous section, the *big data phenomenon* of the recent decades made large collections of data readily available online. Therefore, DA methods that can robustly and effectively leverage these data while training a model for a target task are very desired, as they would eliminate the high cost of manually labeling samples, which constitutes one of the main obstacles in the development of smart systems.

Many methods that deal with the DA problem in the context of image classification with deep convolutional neural networks have been proposed in the literature in recent years. Although some of these approaches achieve great results in many commonly used data sets and test configurations, experiments show that there is still room for improvement.

The methods [14, 9, 29, 40] that currently have the overall best results still cannot achieve a great classification accuracy on the target samples in some data sets, especially when the domains are very dissimilar. For instance, these methods failed to consistently achieve a high classification accuracy in all settings of the Office-Home data set [43], one of the main benchmark data sets used to evaluate DA performance for image classification. This can cast doubts on the robustness of performing DA in a production setting, as it would make the resulting smart system unreliable.

Given the aforementioned reasons, in this work new methods that build upon past literature works that address the adaptation problem with visual data are proposed and experimented with, aiming at improving the performance on target data, particularly in the scenarios in which the existing DA methods do not achieve good results, thus making the overall adaptation more robust.

1.2 Objectives

This work targets the DA problem in the homogeneous and unsupervised scenario for the image classification task. The main goal is to propose modifications to existing literature methods to enhance their robustness and effectiveness while performing adaptation from source to target data. Additional objectives include:

- provide an overview of the most commonly used approaches for performing DA with visual data;
- evaluate the methods proposed in this work with commonly used DA benchmark data sets;
- experiment with performing DA in a real-world setting by conducting a case study in which data from a simulated source domain is adapted to a real target one.

1.3 Scientific Contibutions

This work contributes to the Domain Adaptation and Transfer Learning research topics by proposing two new approaches that enhance the adaptation performance of their baselines by improving the robustness of the pseudo-labeling strategy for DA and by exploring the concept of class-aware domain discrimination. Besides improving upon the baseline results, our proposed approaches also provide meaningful insights for future works on DA with visual data.

We have presented one of the approaches proposed in this work in the paper titled *Analyzing the Effects of Dimensionality Reduction for Unsupervised Domain Adaptation* that was accepted and presented in the Technical Sessions of SIBGRAPI 2021, the 34th Conference on Graphics, Patterns and Images [27].

Finally, this work also presents the results of a case study that investigates the viability of performing domain adaptation in a real-world setting by adapting data from a simulated data set to a real one for performing automatic Personal Protective Equipment (PPE) detection with deep convolutional neural networks.

1.4 Work Organization

In Chapter 2, the main approaches that deal with the DA problem are described, along with examples of methods previously proposed in the literature. In Chapter 3, a formal definition of the DA problem is presented together with a description of the baseline method for the approaches proposed in this work. In Chapter 4, our proposed approaches are described in detail. In Chapter 5, the conducted experiments and their results are presented and discussed, and also the results of the PPE detection case study are presented. Finally, in Chapter 6, the conclusions of this work and insights for future works are presented.

Chapter 2

Related Work

Several methods that deal with the homogeneous and unsupervised domain adaptation problem on visual data have been proposed in the literature. [44] suggested that these approaches can be summarized in three main categories: Discrepancy-based, Reconstruction-based, and Adversarial-based. All these methods share the same goal, which is to diminish the effects of the domain shift by introducing domain-invariability in the training of the model. However, they differ on how this invariability is achieved. In the following sections, each of these approaches is discussed and some methods based on them are described.

2.1 Discrepancy-based Approaches

Discrepancy-based methods adapt the models with fine-tuning and regularization terms that measure the discrepancy between the distributions [44]. In [25], the authors proposed the Deep Adaptation Networks (DAN), in which a regularizer term based on the multiple kernel Maximum Mean Discrepancy (MK-MMD) is added to the higher layers of the network, which are generally less transferable. The MK-MMD is formalized to jointly maximize the two-sample test power and minimize the Type II error, hence the added regularizer term matches the shift in marginal distributions across the domains during training. One of the main weaknesses of DAN is that it assumes that the conditional distributions are the same between the domains, which can be false in real-world applications, thus leading to a less effective adaptation. Some methods use different discrepancy metrics other than the MK-MMD, such as Deep CORAL, another discrepancy-based method proposed by [39], which aligns the second-order statistics between the domains based on a Coral loss that is given by the Frobenius norm of the difference between the covariance matrices of the source and target data.

One of the main challenges of the unsupervised scenario is that the target data conditional distribution cannot be directly estimated, as there are no target labels. Due to this restriction, many methods, such as the previously mentioned DAN [25], will in-

correctly assume that there is no shift in the conditional distribution between source and target data. To this end, some discrepancy-based methods will use pseudo-labeling heuristics to be able to align the conditional distributions. Pseudo-labeling consists of developing a strategy to automatically assign labels to target unlabeled samples. These pseudo-labeled samples can then be used to compute a discrepancy metric or to fine-tune the network's weights to implicitly align the distributions. A disadvantage that comes with this approach is that the pseudo-labels can be incorrectly assigned, which could lead to a worse performance after the adaptation. Therefore, methods that employ a pseudo-labeling strategy must implement ways to suppress the negative effect introduced by wrongly assigned pseudo-labels.

In [49], the authors expanded the idea behind DAN [25] by using pseudo-labels to compute a conditional MMD metric to align both marginal and conditional distributions. In [15], the authors used the K-Means clustering algorithm to assign the pseudo-labels based on the target samples' feature representation. The clusters are initialized with the class centroids computed using source data. Then, the assigned pseudo-labels, which are given by the cluster assignments, are filtered based on the distance to the group center and the number of samples in each group. The filtered pseudo-labels are then used in a Contrastive Domain Discrepancy (CDD) metric that is based on the MMD and is designed to align the distributions by taking into account the inter-class and intra-class discrepancies across the domains.

In [14], the authors proposed a method that also uses pseudo-labels to perform the adaptation. Their goal was to adapt the model from a class-conditioned domain alignment perspective in order to address the challenge of within-domain class imbalance and between-domains class distribution shift. To this end, they used an implicit class-conditioned alignment that removed the need for explicit pseudo-label-based optimization, as the pseudo-labels are instead implicitly used to sample class-conditioned data in a way that aligns the joint distribution between features and labels.

Category Contrast (CaCo), introduced by [12], explores the idea of instance contrast for unsupervised DA, in which a dictionary look-up task trains a visual encoder by matching encoded queries and keys. CaCo builds category-aware and domain-mixed dictionaries by assigning pseudo-labels for the target unlabeled samples, which allows learning invariant representations within and across the source and target domains. During training, a new category contrastive loss between target queries and dictionary keys is minimized, which will pull close samples from the same category and push away those of different categories. This learning objective will ultimately lead to category-discriminative yet domain-invariant representations, thus achieving the adaptation goal.

[29] introduced CoVi, a Contrastive Vicinal space-based DA method that enhances the discrepancy-based strategy by leveraging the vicinal space from the perspective of self-training. CoVi estimates pseudo-labels for the target unlabeled images and explores the

"equilibrium collapse of labels" between vicinal instances, as defined by the authors. To this end, the proposed method divides the vicinal space into a contrastive and a consensus space, based on an entropy maximization point (EMP) that is estimated with a new EMP-Mixup algorithm inspired by the minimax strategy. Ultimately, CoVi will alleviate the inter-domain discrepancy in the contrastive space and will simultaneously resolve intra-domain categorical confusion in the consensus space, thus improving the overall robustness of the adaptation procedure.

In summary, discrepancy-based DA methods leverage different distribution discrepancy metrics and techniques, such as target pseudo-labeling, to introduce domain-invariability by aligning the data distributions across the domains.

2.2 Reconstruction-based Approaches

Reconstruction-based methods use a reconstruction task in a multi-task setting to ensure feature invariance. As image reconstruction is an unsupervised task, it can be accomplished without the target labels. In [6], the authors proposed the Deep Reconstruction-Classification Networks (DRCN). DRCN consists of two pipelines with a shared feature encoder: a label prediction one, which is trained using source domain supervision, and an image reconstruction one that reconstructs the target images. The features produced by the encoder network are fed into both pipelines and the whole network is trained in an end-to-end fashion, with the reconstruction task introducing feature invariability throughout the training.

The Deep Separation Networks (DSNs), proposed by [1], explicitly model both private and shared components from each domain by using three separate encoders, one shared between the domains and a private one for each domain. A shared decoder network learns to reconstruct samples from both domains by using both private and shared representations. Soft subspace orthogonality constraints are used to push apart the private and shared representations, while a similarity loss keeps the shared representations similar. Finally, the label classifier is trained on the shared representation using source supervision, and prediction is done by feeding the classifier with the feature produced by the shared encoder.

Overall, reconstruction-based approaches are able to improve the model's performance on target data. However, by introducing an image reconstruction task, the training of the neural network becomes more challenging and resource-consuming.

2.3 Adversarial-based Approaches

Adversarial-based methods are built upon the concept of domain confusion, that is, the inability to distinguish between the domains [44]. To achieve this confusion, they incorporate a domain discriminator network that classifies the samples based on their domain to the original model, creating an adversarial training framework. [5] proposed the Domain Adversarial Neural Network (DANN), in which the adversarial max-min objective is directly implemented with a gradient reversal layer that multiplies the discriminator's gradient by a negative constant during backpropagation. This effectively makes the feature extractor produce more domain-invariant features in order to maximize the classification loss of the discriminator. The whole network is trained jointly, in a multi-task end-to-end fashion, with the domain classification task executed by the domain discriminator and the label prediction task performed by the classifier network. In [41], Adversarial Discriminative Domain Adaptation (ADDA) is proposed. In ADDA, the adversarial objective is implemented using a GAN-like loss with real and fake labels instead of using a gradient reversal layer. Furthermore, ADDA uses separate feature extractors for each domain, which enables them to learn domain-specific characteristics more freely.

Some adversarial-based methods use a generator task in addition to the discriminator one in a configuration similar to the Generative Adversarial Networks (GANs) [7]. The GANs are comprised of a generator network, which generates fake images, and a discriminator network, which discriminates the images between real and fake. These networks are trained adversarially so that the generator can produce convincing images. The goal of these methods is to generate synthetic samples with target domain characteristics that share labels with source samples, thus obtaining pairs of synthetic images and their respective labels in an unsupervised manner. For instance, [24] proposed CoGAN, which consists of a pair of GANs, where each will produce synthetic images from a single domain. The weights of the initial layers of both generative networks and the final layers of both discriminators are shared between the GANs. This weight-sharing lets CoGAN achieve a domain-invariant feature space in both networks. After training, pairs of images that share the same label will be produced by CoGAN, where one image has the characteristics of the source domain and the other has those of the target domain. The synthetic and labeled images can then be used to train a model for classifying target samples.

[31] proposed the Conditional Domain Adaptation Generative Adversarial Network (CoDAGAN) for performing DA considering the image segmentation task, specifically for segmenting medical X-ray images, where there are usually many samples available for training, but only a few of them are actually labeled, constituting an ideal scenario for unsupervised and semi-supervised DA. CoDAGAN uses an encoder-decoder network that performs image translation and an adversarial discriminator that classifies the images

between real and synthetic. Adaptation is achieved by leveraging the common isomorphic representation created by the encoder-decoder architecture, which allows multiple related data sets to be used conjointly during training, hence leading to overall better results.

[22] proposed 3C-GAN, a method that also uses a generator network to produce synthetic data from the target domain's data distribution to deal with the unsupervised domain adaptation problem in a setting in which the source data are not available during adaptation, instead only the network's pre-trained weights are provided. Therefore, 3C-GAN uses the target unlabeled samples to train a generator network that generates images conditioned both on the target data distribution, via the GAN discriminator, and a random label. In order to incorporate semantic meaning in the generated images, the classification loss of the synthetic image applied to the pre-trained classifier is embedded into the generator's loss, causing the generated images to portray the random labels fed to the generator. The produced pairs of images and labels are then used to fine-tune the classifier network for the target data.

[48] introduced the Spectral Unsupervised Domain Adaptation (SUDA) method, which learns domain-invariant spectral features through an adversarial objective. SUDA uses a spectral transformer network to create spectral views of each image, in which the inter-domain discrepancies are reduced by enhancing domain-invariant feature components. To find such components, SUDA leverages contextual information with a novel Adversarial Spectrum Attention (ASA), in which domain-variant feature components are suppressed via an adversarial loss with a domain discriminator. ASA takes the Fast Fourier Transform spectral representation of each image decomposed into N components using a band pass filter as input and outputs a recomposed spatial-space image, which is then forwarded to a discriminator that classifies each sample based on its domain. By employing this attention mechanism, SUDA is able to learn domain-invariant spectral features, which allow it to achieve great accuracies in the target domain.

2.4 Hybrid Approaches

Some domain adaptation methods combine the aforementioned approaches in hybrid architectures with the goal of achieving even greater performance. For instance, some methods combine the discrepancy-based approach with a dimensionality reduction strategy. Joint Distribution Adaptation (JDA), proposed by [26], aims to jointly align both marginal and conditional distributions between the source and target domains by integrating the Maximum Mean Discrepancy (MMD) metric with the Principal Component Analysis (PCA) algorithm for dimensionality reduction. This allows the creation of

a feature representation that is effective for calculating the distribution differences across the domains. [13] also proposed a method that integrates the MMD metric with a dimensionality reduction strategy. Their goal is to build a latent space in which the MMD value between the source and target data is as small as possible while maintaining the local geometry properties of the source domain. In OTDR, proposed by [21], a dimensionality reduction framework similar to PCA is combined with an optimal transport strategy in a two-stage solution. In the first stage, the samples' features are transformed to a lower dimensional space in which the intradomain dispersion is maximized and the source intraclass compactness is minimized. In the second stage, an optimal transport plan based on the Wasserstein distance is learned, which transports the source samples to the target domain while keeping local information, thus maintaining the class discriminability of the source data.

Recently, [40] proposed Safe Self-Refinement for Transformer-based Domain Adaptation (SSRT), which integrates a transformer network to the adversarial framework for performing DA. Visual transformers process an image by transforming it into a sequence of tokens and using global self-attention to build and refine this tokenized representation. In SSRT, a visual transformer is incorporated into the adversarial framework for DA, as the authors sustain that such networks can produce strong transferable feature representations, leading to a more effective adaptation.

The Robust Spherical Domain Adaptation (RSDA) method proposed by [9] combines the adversarial and discrepancy-based approaches by using both a domain discriminator and a pseudo-labeling technique. RSDA uses Gaussian-uniform mixture models to estimate the probability of a given pseudo-label being correctly assigned to a target sample based on the distance between the feature representation of each sample and the class centroid in a spherical space. After estimating the pseudo-labels and the mixtures' parameters, the network is trained in a multi-task setting with the label prediction loss for source samples, the adversarial loss of the domain discriminator, and the proposed robust pseudo-label loss that takes into account the estimated correct pseudo-labeling probabilities for the target samples.

RSDA achieves great results in the main DA benchmarks and is the baseline for our proposed approaches. In this work, we modify some aspects of the original RSDA's training procedure and network architecture to improve the robustness of the estimated pseudo-labels and how they are used, thus leading to better accuracy on target data. Further explanation of how RSDA operates and the enhancements that this work proposes are presented in the following chapters.

Chapter 3

Theoretical Framework

This chapter presents the definitions of the image classification task, the unsupervised domain adaptation problem, and the baseline of the methods proposed in this work.

3.1 Image Classification with Convolutional Neural Networks

Image classification is one of the main tasks in Computer Vision and can be used in a variety of scenarios and applications [2]. The goal is to associate correct labels to images based on what is depicted on them. More formally, we are looking for a model h that receives an image x as input and is able to correctly associate a label $\hat{y} = h(x), \hat{y} \in \{1, 2, 3, .., k\}$, where $\hat{y}$ indicates the class, from a total of k classes, that is portrayed on image x. The list of classes will vary depending on the application: it can be a list of product categories that are sold in an online store, or it can be a list of animal species that are in a zoo. Notice that, even though this task may seem simple, it is the building block for solving more complex Computer Vision problems, such as object detection, image segmentation, and super-resolution technology [2].

Most image classification methods are comprised of two steps: feature extraction and classification. The first one consists of transforming the input images into a feature space, in which the semantic characteristics of the pictures will be more prominent. In the classification stage, class labels are assigned to each image feature.

Before neural networks, manually-designed algorithms, also called *handcrafted*, such as the Scale Invariant Feature Transform (SIFT) [28] and the Oriented FAST and Rotated BRIEF (ORB) [36], were used for feature extraction and representation [30]. Each step of these algorithms was carefully elaborated in order to extract as much knowledge from the picture elements and overcome variations in scale and illumination [30]. For the classification step, traditional machine learning algorithms were used, like the

Support Vector Machine (SVM) [3], which took the extracted feature or some intermediary representation, such as the Bag of Visual Words (BOVW), as input and outputted the predicted class for each image. Although these methods that used handcrafted feature extraction techniques and traditional classification algorithms were able to achieve good results in some tasks, they showed a poor generalization ability, which led to lower accuracy, especially on more challenging data sets [30].

To overcome the limitations of the traditional approaches for image classification, researchers started employing neural networks to solve this task. Although Artificial Neural Networks are not a novel concept, being first proposed in the 1950s [35], only in recent years has it become possible to use them practically in real-world applications, due to the rapid development of more powerful hardware components. This is mainly because these networks mimic the way in which neurons are connected in a biological brain, which in turn creates a very complex model that needs a lot of computing power to execute [30].

For image classification, a special type of neural network, the Convolutional Neural Network (CNN), is employed. The modern framework of CNNs was first introduced by [20] and took inspiration from how biological vision works [8]. At its center is the convolutional layer, which allows the extraction of abstract features from the image pixels. These features are then combined through the multiple layers of the network to achieve more high-level representations of what is depicted on the image, allowing these networks to learn how to perform tasks with remarkable results [30, 16]. Many CNN architectures have been proposed, such as LeNet-5 [19], AlexNet [17] and VGGNet [38]. Currently, the architectures that achieve the highest results employ a residual learning technique, which improves the network's ability to learn, such as the ResNet [10] and the DenseNet [11].

Regardless of the CNN's architecture, it needs to be trained to learn how to classify the images. During this training, an optimizer algorithm, such as the Stochastic Gradient Descent (SGD), and a loss function, such as the Cross-entropy loss, are used to optimize the network's inner parameters using the available data. The goal is to minimize the classification error on the training images so that the final model is able to assign the correct labels to each image. A common challenge found while training is *model-overfitting*, that is, the model's incapacity to generalize for images outside the training set. When a network is overfitted, it will not be able to perform the classification effectively, even though a high accuracy may be achieved in the training images. The main causes for it are the high complexity of the network's architecture and a limited data set, which may not represent the real distribution of the target images [2]. Some techniques can be used to overcome this issue, such as using simpler network architectures, different activation functions, or the previously mentioned residual learning. However, the main procedure to detect and possibly overcome overfitting is to split the data set into a training and a test set. The main idea is to train the model using only the training set and validate the classification accuracy in the test set. This way, if overfitting happens, one will notice the

bad performance on the test set. The split between training and test data can be done using random sampling or techniques such as cross-validation.

The main assumption that is made during the model's training is that the training data and the data on which the model will be used to make predictions, the test data, are independent and identically distributed, in what is commonly known as the *i.i.d. assumption*. The current algorithms that are used to train the CNNs only work if this assumption holds true. Therefore, if there is a bias in the training set that does not exist in the real data distribution, it will severely impact the model's performance at test time. The aforementioned train/test split technique may help to guarantee that this assumption holds if the split is done in a truly random way, which may help identify hidden bias in the data, especially if the model is trained multiple times with different splits. Nevertheless, the *i.i.d. assumption* still requires the data set used for training and validating the model to be carefully constructed in order to avoid the introduction of biases or any kind of shift across the data distributions.

In summary, deep convolutional neural networks are currently the machine learning models that achieve the highest accuracy in the image classification task [2]. Their ability to learn complex representations of the patterns in an image allows them to obtain remarkable results. However, to train these networks, a large amount of data, which must be correctly labeled and representative of the real data distribution, has to be collected, organized, and processed. The gathering and labeling of this large amount of data are very time and resource-consuming, thus constituting one of the major constraints on the development of smart systems that use neural networks for image classification.

3.2 Domain Adaptation Definition

As discussed in the previous section, labeling the large amount of data that is necessary to train the Convolutional Neural Networks is one of the major challenges for the development of smart systems that perform image classification. Therefore, domain adaptation methods offer a solution to allow for data that are already available and labeled to be used to train models for performing different, but semantically-related tasks. These methods will look for ways to effectively transfer knowledge from one domain to another, thus eliminating the need to label a new large set of data and reducing the costs of developing a smart system.

A domain $\mathcal{D} = (X, Y, p)$ is defined as a combination of an input space X, an output space Y, and an associated probability distribution p and the domain adaptation problem consists of, given a source domain $\mathcal{S} = (X^s, Y^s, p^s)$ and a target domain $\mathcal{T} = (X^t, Y^t, p^t)$,

training a model $h(x)$ that is able to correctly make predictions on target data $x \in X^t$ [32, 16]. This problem can be divided into different categories based on the level of divergence between the domains and the availability of labeled data, as proposed by [33] and [44].

The setting in which the input spaces and output spaces are the same across the domains, i.e., $X^s = X^t$ and $Y^s = Y^t$, is referred to as *Homogeneous* Domain Adaptation. Note that in this setting, for a classification task, it is assumed that each class represented in Y^s and Y^t has the same semantic meaning in both domains and that only the probability distributions p^s and p^t vary across the domains. On the other hand, if the input and output spaces are different across the domains, i.e., $X^s \neq X^t$ or $Y^s \neq Y^t$, then it is called *Heterogeneous* Domain Adaptation and constitutes a more general scenario, where the domains may vary significantly. For instance, in the heterogeneous setting, one may want to adapt knowledge from a text domain to a visual one, as is done by [50].

According to the availability of labeled data from the target domain, we can further categorize the domain adaptation problem into [44]:

- Unsupervised Domain Adaptation: there are no labeled samples available in the target domain. Therefore, the model is trained using only source labeled data ($\{(x_i^s, y_i^s)\}_{i=1}^{n_s}$, where $x_i^s \in X^s$ and $y_i^s \in Y^s$) and target unlabeled data ($\{x_j^{tu}\}_{j=1}^{n_{tu}}$, where $x_j^{tu} \in X^t$);

- Semi-supervised Domain Adaptation: a small number of labeled target data is available, in addition to the target unlabeled data. Note that it is assumed that the amount of unlabeled data is far greater than the amount of labeled data. In this setting, the adaptation procedure uses the source labeled data ($\{(x_i^s, y_i^s)\}_{i=1}^{n_s}$, where $x_i^s \in X^s$ and $y_i^s \in Y^s$) together with target unlabeled ($\{x_j^{tu}\}_{j=1}^{n_{tu}}$, where $x_j^{tu} \in X^t$) and labeled samples ($\{(x_k^{tl}, y_k^{tl})\}_{k=1}^{n_{tl}}$, where $x_k^{tl} \in X^t$, $y_k^{tl} \in Y^t$ and $n_{tl} \ll n_{tu}$);

- Supervised Domain Adaptation: only a small number of labeled target data is available. Note that, different from the semi-supervised scenario, there are no target unlabeled samples. Therefore, the training of the model is done with the source labeled data ($\{(x_i^s, y_i^s)\}_{i=1}^{n_s}$, where $x_i^s \in X^s$ and $y_i^s \in Y^s$) and the few target labeled samples available ($\{(x_k^{tl}, y_k^{tl})\}_{k=1}^{n_{tl}}$, where $x_k^{tl} \in X^t$, $y_k^{tl} \in Y^t$ and $n_{tl} \ll n_s$).

This work focuses on the homogeneous and unsupervised setting.

3.3 Robust Spherical Domain Adaptation (RSDA)

In this section, the Robust Spherical Domain Adaptation (RSDA) method, the baseline for the new approaches proposed in this work, is described. RSDA was proposed by [9] and is based on the adversarial framework for domain adaptation, which was first introduced by [5] and is presented in Figure 3.1. In this framework, a domain discriminator G_d classifies the features produced by the feature extractor G_f based on their domain, and a Gradient Reversal Layer, which is added just before this domain classifier, directly implements the min-max adversarial goal by multiplying G_d's gradient by a negative factor. Throughout the complete network's training, this adversarial game will make the feature extractor G_f produce more domain-invariant features, as those will impair the domain classifier's ability to discriminate across the domains, thus maximizing its loss due to the introduced gradient reversal. This domain-invariability promoted by the adversarial goal will ultimately lead to better image classification accuracy on the target data.

Figure 3.1: The DANN adversarial framework for DA.

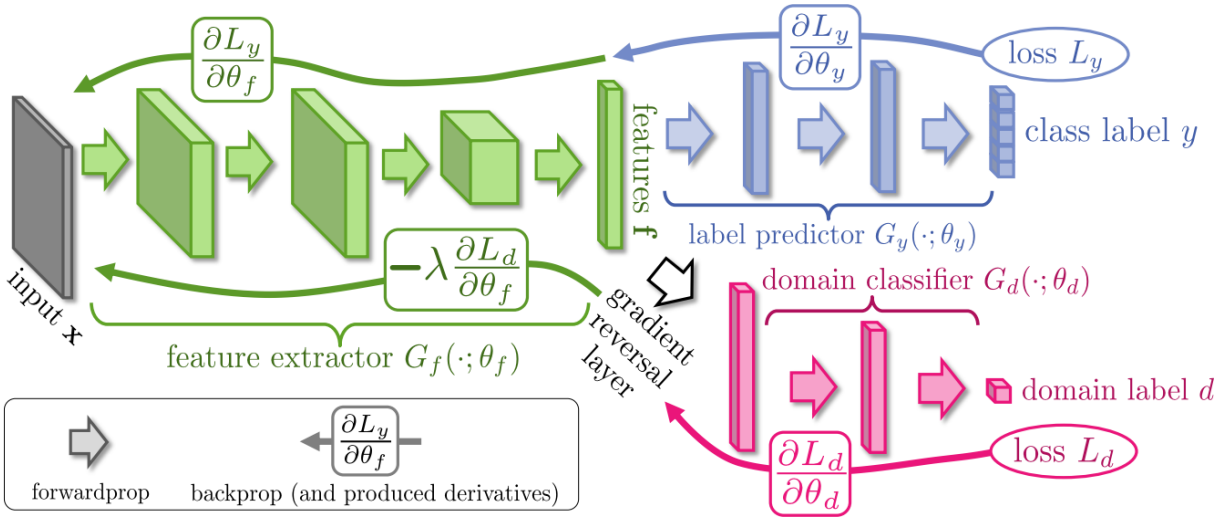


Figure adapted from [5].

RSDA expands the adversarial framework by introducing a new robust pseudo-label loss formulation. This loss weights the classification error of the target pseudo-labeled samples based on a correct pseudo-labeling probability, which is estimated using a Gaussian-uniform mixture model. This makes adaptation more robust by filtering out wrongly assigned pseudo-labels. Furthermore, RSDA also transforms the features produced by the feature extractor network into a spherical, l2-normalized, space, which the authors sustain makes adaptation easier.

Figure 3.2: Baseline RSDA method.

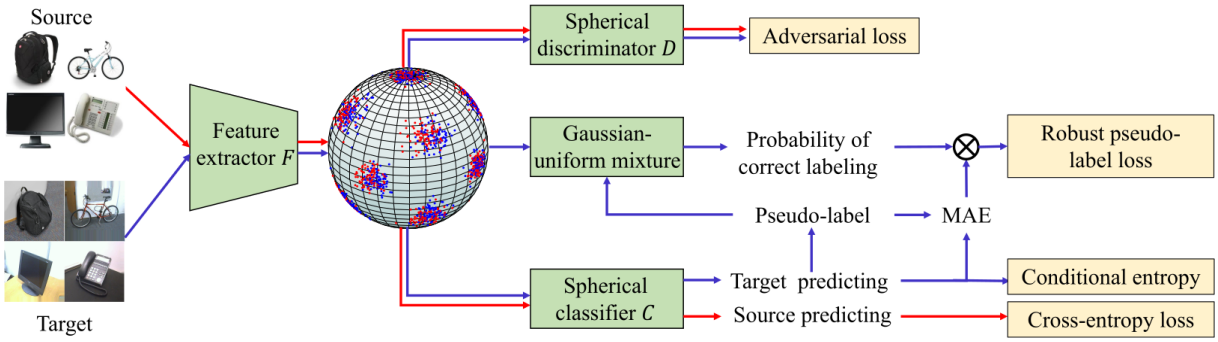


Figure extracted from [9].

An overview of the complete RSDA architecture is presented in Figure 3.2. In summary, RSDA trains a network comprised of a feature extractor F, a classifier C, and a domain discriminator D using the following spherical adversarial training loss:

$$\mathcal{L} = \mathcal{L}_{bas}(F, C, D) + \mathcal{L}_{rob}(F, C, \phi) + \gamma \mathcal{L}_{ent}(F), \tag{3.1}$$

where the basic loss $\mathcal{L}_{bas}$ is based on the adversarial loss of DANN [5] and is given by the sum of the cross-entropy loss of the source samples' classification and the adversarial loss of the domain discriminator:

$$\mathcal{L}_{bas}(F, C, D) = \mathcal{L}_{src}(F, C) + \lambda \mathcal{L}_{adv}(F, D), \tag{3.2}$$

where λ is a negative constant. As previously mentioned, the adversarial goal is achieved by inverting the discriminator's classification loss using a Gradient Reversal Layer, a direct implementation of the min-max objective of the adversarial framework as was originally proposed by [5]. By inverting this gradient, the feature extractor will be stimulated to produce more domain-invariant features, as these will impair D's ability to discriminate between the domains, maximizing its loss.

The proposed robust pseudo-label loss $\mathcal{L}_{rob}$ is defined as:

$$\mathcal{L}_{rob}(F, C, \phi) = \frac{1}{N_0} \sum_{j=1}^{N_t} w_\phi(x_j^t) \mathcal{J}(C(F(x_j^t)), \tilde{y}_j^t), \tag{3.3}$$

where $N_0 = \sum_{j=1}^{N_t} w_\phi(x_j^t)$, $\mathcal{J}$ is the mean absolute error and $w_\phi(x_j^t)$ is given by

$$w_\phi(x_j^t) = \begin{cases} \gamma_j, & \text{if } \gamma_j \geq 0.5 \\ 0, & \text{otherwise} \end{cases}, \tag{3.4}$$

where $\gamma_j = P_\phi(z_j = 1 | x_j^t, \tilde{y}_j^t)$ is the correct labeling probability associated with the target sample x_j^t and the pseudo-label $\tilde{y}_j^t$. The pseudo-labels are assigned based on the output of

the classifier C trained in the previous stage of RSDA and the correct labeling probability γ_j is estimated by the Gaussian-uniform mixture model for the assigned class, using the mixture parameters $\phi_{\tilde{y}_j^t} = (\pi_{\tilde{y}_j^t}, \sigma_{\tilde{y}_j^t}, \delta_{\tilde{y}_j^t})$. Note that there is one mixture for each of the K classes and that they operate independently. Each mixture takes as input the cosine distance between the spherical feature representation f_j^t of x_j^t produced by F and the centroid $\mathcal{C}_{\tilde{y}_j^t}$ of the features of all target samples assigned to the same class:

$$d_j^t = \text{dist}(f_j^t, \mathcal{C}_{\tilde{y}_j^t}) \tag{3.5}$$

The posterior correct labeling probability is then given by the mixture model:

$$P_\phi(z_j = 1 | x_j^t, \tilde{y}_j^t) = \frac{\pi_{\tilde{y}_j^t} \mathcal{N}^+(d_j^t | 0, \sigma_{\tilde{y}_j^t})}{\pi_{\tilde{y}_j^t} \mathcal{N}^+(d_j^t | 0, \sigma_{\tilde{y}_j^t}) + (1 - \pi_{\tilde{y}_j^t}) \mathcal{U}(0, \delta_{\tilde{y}_j^t})} \tag{3.6}$$

where $\mathcal{N}^+$ and $\mathcal{U}$ are the Gaussian and the Uniform components of the mixture, respectively, and $\pi_{\tilde{y}_j^t}$, $\sigma_{\tilde{y}_j^t}$, and $\delta_{\tilde{y}_j^t}$ are the mixture parameters estimated with an Expectation-Maximization algorithm. $z_j \in \{0, 1\}$ is a random variable that indicates whether the pseudo-label $\tilde{y}_j^t$ was correctly assigned to the target sample x_j^t.

The assumption made by [9] is that samples with a smaller distance to the class centroid in the spherical feature space have a higher probability of being correctly assigned. The Gaussian component of the mixture models the samples that are closer to the centroid, while the Uniform one models the samples that are further away, assigning a lower probability to them.

The training of the network, as defined by [9], is comprised of a number of stages that is defined via a new hyper-parameter. In the initial stage, F, C, and D are optimized using only the basic loss $\mathcal{L}_{bas}$, without using any pseudo-labels. After that, each stage has two main steps:

1. First, the weights of F and C that were trained in the previous stage are frozen and the pseudo-labels are assigned to all target samples based on the output of C. F is used to obtain the feature representation of the target samples and the parameters of the mixture models are estimated using an Expectation-Maximization algorithm;

2. Then, the weights of F and C are unfrozen and the complete network is trained again using the estimated pseudo-labels and their respective correct labeling probability in the complete loss formulation $\mathcal{L}$, as defined in Equation 3.1.

The authors of RSDA sustain that the proposed robust-pseudo label loss and the transformation of the features produced by F into a spherical space are responsible for the results achieved in the experiments with commonly used benchmark data sets for domain adaptation, in which RSDA obtained a higher accuracy than other adversarial-based methods in almost all configurations.

In this work, we propose some modifications to the original RSDA method with the goal of further improving its robustness during the estimation of the correct pseudo-labeling probabilities with the Gaussian-uniform mixture models, thus enhancing the target classification accuracy. We also propose incorporating a multi-class discriminator into the adversarial framework of RSDA. These changes are detailed and discussed in the following chapters.

Chapter 4

Methodology

In this work, we propose two new approaches for performing domain adaptation with visual data in an unsupervised setting. The first one is described in Section 4.1 and enhances the pseudo-labeling strategy of Robust Spherical Domain Adaptation (RSDA) [9] by combining it with a dimensionality reduction strategy and better use of the source and target domain data. The second one is described in Section 4.2 and builds upon the concept of a multi-class domain discriminator network to introduce class-aware domain confusion to further improve the final adaptation performance.

4.1 Enhancing Pseudo-label Robustness

As presented in Chapter 2, a common approach for performing DA on visual data is to develop a strategy to automatically assign labels to the target domain's unlabeled images. These pseudo-labeled samples can then be used during the network's training to enhance its classification accuracy in the target domain. The main issue with this approach is guaranteeing that the pseudo-labels assigned to each sample correspond to the actual class portrayed in the image, i.e., the pseudo-labels are correct. As in most scenarios this can not be guaranteed, DA methods have to proactively deal with these wrongly assigned pseudo-labels, as they can negatively impact the adaptation performance.

The Robust Spherical Domain Adaptation (RSDA) method, introduced by [9] and described in Section 3.3, deals with the wrongly assigned target pseudo-labels by proposing a new loss formulation that takes into account the correct labeling probability of the target samples, which is estimated using Gaussian-uniform mixture models based on the distance between the sample and the class centroid in a spherical feature space. The main goal of RSDA is to improve the overall robustness of the adaptation procedure by filtering out target samples that may be incorrectly pseudo-labeled based on this estimated correct labeling probability.

We propose two modifications to RSDA's training procedure, specifically by en-

hancing the estimation of the correct labeling probability introduced by [9], with the goal to further improve its robustness, thus enhancing the overall adaptation performance:

- We change how data from the source domain are handled during the estimation of the parameters of the Gaussian-uniform mixture models, based on the hypothesis that the source ground-truth labels can serve as anchors to the true-class meanings during this procedure;
- We apply a dimensionality reduction algorithm to the feature representation of the samples before computing the class centroid distances that are used during the estimation of the correct labeling probability, based on the hypothesis that transferability could be more easily achieved in lower-dimensional spaces.

In the following sub-sections, the proposed enhanced pipeline for estimating the correct labeling probability for target samples and the hypotheses that guided it are described in detail.

4.1.1 Source Labeled Data as Anchors During the Estimation of the Mixture Model's Parameters

In the unsupervised DA scenario, only source labeled samples and target unlabeled ones are available to train the model. As previously discussed, by using a pseudo-labeling strategy, we can obtain labels for the target samples that can be used during training. In the original RSDA [9] method, the pseudo-labels $\widetilde{y}_j^t$ are assigned based on the output of a class classifier network C that is initially trained using the adversarial framework proposed by [5], which does not use pseudo-labels. Then, these pseudo-labeled target samples are used to estimate the parameters of K Gaussian-uniform mixture models, one for each class from a total of K classes, that predict the probability of a pseudo-label being correctly assigned to a target sample. This correct labeling probability P_ϕ is given by the respective mixture model

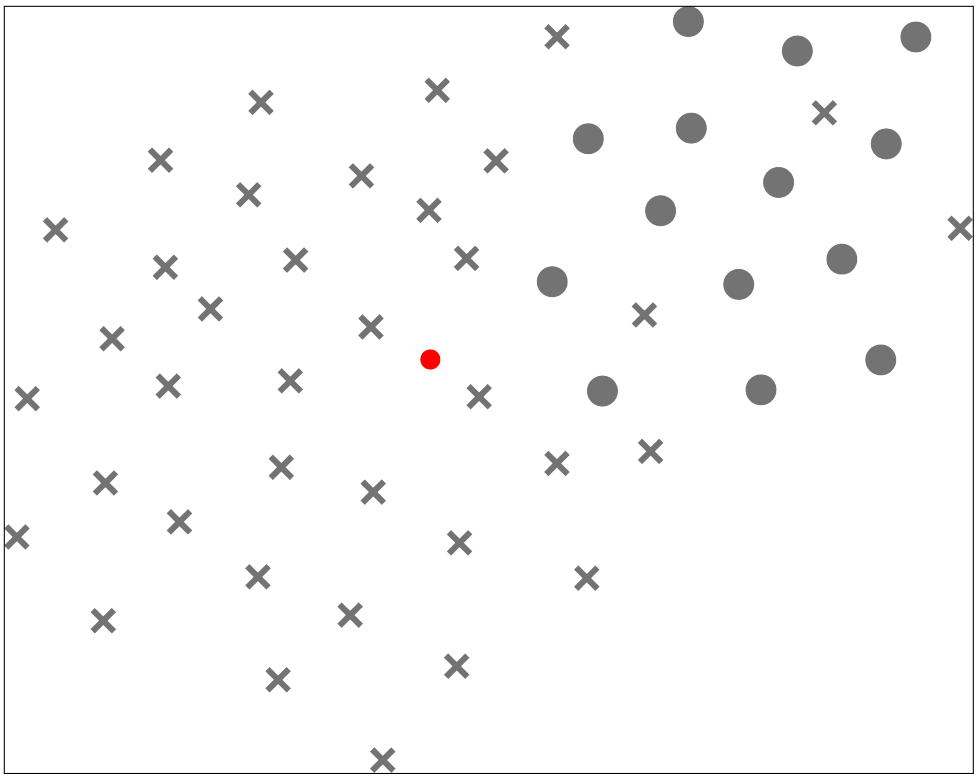
$$P_\phi(z_j = 1 | x_j^t, \widetilde{y}_j^t) = \frac{\pi_{\widetilde{y}_j^t} \mathcal{N}^+(d_j^t | 0, \sigma_{\widetilde{y}_j^t})}{\pi_{\widetilde{y}_j^t} \mathcal{N}^+(d_j^t | 0, \sigma_{\widetilde{y}_j^t}) + (1 - \pi_{\widetilde{y}_j^t}) \mathcal{U}(0, \delta_{\widetilde{y}_j^t})}, \tag{4.1}$$

where $d_j^t = \text{dist}(f_j^t, \mathcal{C}_{\widetilde{y}_j^t})$ is the cosine distance between the latent representation f_j^t of the j-th target sample and $\mathcal{C}_{\widetilde{y}_j^t}$ is the feature-space centroid of the assigned class $\widetilde{y}_j^t$. The parameters $\phi = \{\pi_k, \sigma_k, \delta_k\}_{k=1}^K$ for each of the K mixture models are estimated using an Expectation-Maximization (EM) algorithm that is derived based on the definition of

the Gaussian-Uniform mixture model presented in Equation 4.1. During this estimation, the target domain samples are grouped based on the assigned pseudo-labels, and the centroid for each class is computed. Then, the distance between each sample's feature representation and its class centroid is calculated. Finally, these distances $\{d_j^t | \widetilde{y}_j^t = k\}$ are inputted to the EM algorithm, which will estimate the parameters of the k-th mixture model, related to the k-th class.

Notice that in the original formulation of RSDA, only the features from target samples are used to compute the distances that are inputted to the EM algorithm. This can lead to some issues during the estimation of the mixture's parameters, as the correct labeling probability given by the mixture model will lose its meaning if enough target samples have wrongly assigned pseudo-labels and create a cluster in the feature space. These *incorrect clusters* would result in a loss of concept problem that would not be captured during the network's training, as ground-truth target labels are not available. This scenario is illustrated in Figure 4.1, where all the target samples that were assigned to a given class are plotted in a 2-dimensional feature space with a circle, if the pseudo-label is correct, i.e., these samples really are from the assigned class, or an x, which indicates that they were assigned the wrong pseudo-label. Notice that when the centroid for this class is computed, the red dot in the figure, many incorrectly-labeled samples will have a small distance to it, causing the mixture model to estimate a high correct labeling probability for them, even though they are incorrect, thus negatively impacting the adaptation performance.

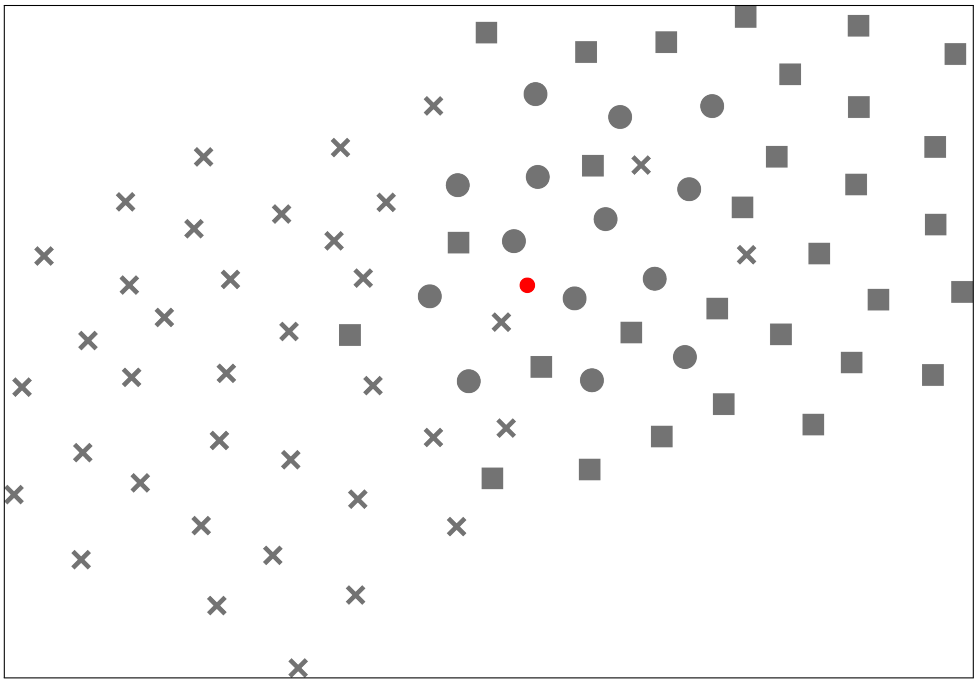
Figure 4.1: Failed correct labeling probability estimation.



The described issue can undermine the robustness and effectiveness of the adaptation, as the wrong pseudo-labeled samples would not be filtered out during training. Therefore, in this work, we propose a modification to the original RSDA procedure in order to mitigate this problem by incorporating the source samples as anchors to the true class meanings. This way, even if a great number of target samples are not correctly pseudo-labeled, the estimation of the mixture's parameters would not suffer a great impact as the ground-truth labels of the source samples would balance out the impact of the incorrectly-assigned target pseudo-labels.

The proposed source anchors are implemented by using data from both source and target domains during the estimation of the mixture's parameters: the EM algorithm now receives as input the distances calculated from the feature representation of the samples from both domains. As the ground-truth labels of the source samples are known, the estimation of the mixture's parameters becomes more robust, thus making the previously presented deviation caused by incorrectly pseudo-labeled target samples more unlikely. This is illustrated in Figure 4.2. Notice how, with the addition of the labeled source data (represented by the squares in the figure), the class centroid shifts towards the correctly labeled samples. Therefore, even though there are still many target mislabeled samples, the distance calculation and the subsequent correct labeling probability estimation will still be able to filter out most of these incorrectly-labeled target samples, thus improving the overall adaptation robustness.

Figure 4.2: Source samples as class anchors.



The use of data from both domains is possible due to the domain invariance promoted by the adversarial domain discriminator D, which was first proposed by [5] and

is also employed by the original RSDA method. This discriminator takes the features produced by the feature extractor F and classifies them based on which domain they are from, creating a binary classification task that is trained along with the other components of the RSDA architecture in a multi-task end-to-end setup. Feature domain-invariance is achieved with the adversarial game played between D and F, in which D has to be able to correctly identify the domain of each feature and F has to produce domain-invariant features in order to impair D's ability to discriminate the samples between the domains. This adversarial min-max objective is implemented directly via a Gradient Reversal Layer added just before the initial layer of D, as originally proposed by [5]. Throughout the training, the adversarial objective will make the features produced by F more domain-invariant by introducing domain confusion. This enables the use of the source samples in the estimation of the mixture's parameters, as we can assume that the distribution shift between the source and target features will diminish as the training progresses due to the domain invariability introduced by D.

In summary, we propose that both labeled data from the source $\{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ and pseudo-labeled data from the target $\{(x_j^t, \widetilde{y}_j^t)\}\}_{j=1}^{N_t}$ domains should be used when estimating the correct labeling probability for the target samples with the Gaussian-uniform mixture models. This will avoid a class-concept shift problem, as we sustain that the ground-truth labels of the source samples will serve as anchors to the true class meanings.

4.1.2 Dimensionality Reduction

As presented in Chapter 2, many methods employ a dimensionality reduction strategy in their solutions to the DA problem [26, 13, 23, 21]. The main assumption made by these methods is that adaptation can be more easily achieved in a lower-dimensional space. Many of them will use some kind of distribution discrepancy metric, such as the Maximum Mean Discrepancy (MMD), to explicitly model the distribution gap between the domains, so that the reduced feature space can be built in a way that minimizes the domain shift in the lower-dimensional representations.

Based on the aforementioned works and following the assumption that adaptation can be more easily achieved in lower dimensional spaces, we also propose incorporating a dimensionality reduction strategy to the RSDA [9] method. To this end, we transform the image features f produced by the feature extractor F to a reduced space by applying a dimensionality reduction algorithm, and then use these lower-dimensional representations r_i when estimating the correct labeling probability for the target samples.

The transformation of the image features to a lower-dimensional representation

for estimating the mixture's parameters is guided by the hypothesis that *semantic-related* information would be privileged in the image representations in a reduced space. Being that we consider the homogeneous DA scenario, in which the label space is the same across the domains and the domain shift is caused by the difference in image conditions, privileging information related to the semantic structure of the problem should lead to a more robust estimation of the correct labeling probability, as the class centroids would be more precise to each class meaning, thus enhancing the accuracy of the estimation performed by each Gaussian-uniform mixture model.

We experiment with different dimensionality reduction algorithms and configurations. The obtained results are presented and discussed in the following chapters. Note that, different from the aforementioned literature methods, we do not explicitly model the distribution discrepancy between the domains while creating the reduced representations. Instead, we rely on the domain invariability introduced by the domain discriminator D during training and use common dimensionality reduction algorithms, such as Principal Component Analysis (PCA) [34] and Partial Least Squares (PLS) [46], to obtain the reduced representations. This is done mainly to evaluate the efficacy of employing a dimensionality reduction strategy without explicitly calculating the distribution discrepancy across the domains, as this discrepancy is already implicitly modeled by the domain discriminator and the adversarial framework for DA.

With the addition of the dimensionality reduction strategy, the estimation of the correct labeling probability can be summarized in the following steps:

1. Given the samples x and their feature representation f, a dimensionality reduction algorithm is applied to f, generating the reduced representations r.

2. Then, the distance d between each representation r and the respective reduced-space class centroid $\mathcal{C}_k^r$ is calculated with respect to each sample.

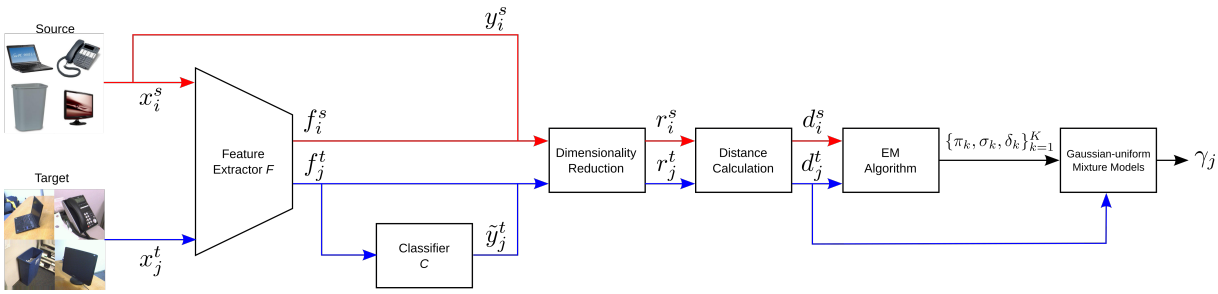
3. Next, the calculated distances d are used in the EM algorithm to estimate the parameters for the mixture model for each class.

4. Finally, the correct labeling probability γ is estimated for the target samples using the Gaussian-uniform mixture models and the parameters estimated by the EM algorithm.

4.1.3 Enhanced Correct Labeling Probability Estimation Pipeline

The complete procedure to estimate the correct pseudo-labeling probability for the target samples, updated with the proposed enhancements, is presented in the diagram in Figure 4.3. The features $\{f_i^s\}$ and $\{f_j^t\}$ for the source and target samples, respectively, are obtained by applying the feature extractor F trained on the previous iteration of RSDA to the images from each domain, $\{x_i^s\}$ and $\{x_j^t\}$. Then, the pseudo-labels $\tilde{y}_j^t$ of the target samples are assigned based on the output of the classifier C, which was also trained on the previous iteration of RSDA. The dimensionality reduction algorithm is then applied on $\{f_i^s\}$ and $\{f_j^t\}$ to obtain the reduced representations $\{r_i^s\}$ and $\{r_j^t\}$ of each sample. Note that the source ground-truth labels and target pseudo-labels can be used during this step in some algorithms, such as in the Partial Least Squares (PLS) one.

Figure 4.3: Enhanced correct labeling probability estimation pipeline.



After the reduced representations are obtained, the estimation follows the same steps as originally proposed in RSDA [9]: the centroids for each class are computed with respect to the reduced representations and the distances $\{d_i^s\}$ and $\{d_j^t\}$ ($d_i^s \in \mathbb{R}$ and $d_j^t \in \mathbb{R}$) between each reduced feature and its respective class centroid, based on the ground-truth labels for source samples and on the pseudo-labels for target ones, are calculated. These distances are then inputted to the EM algorithm, as devised by [9], to estimate the parameters $\{\pi_k, \sigma_k, \delta_k\}_{k=1}^K$ for each Gaussian-uniform mixture model. Finally, these mixture models are used to estimate the correct pseudo-labeling probability $\{\gamma_j\}$ for the target samples, which will then be used during training in the robust pseudo-label loss formulation.

The complete steps of the RSDA training procedure updated with our proposed enhancements are presented in Algorithm 1. Note that the previously discussed changes are made to the estimation of the correct labeling probabilities of the target pseudo-labeled samples and that the training of the feature extractor F, classifier C, and domain discriminator D is performed using the original loss formulation, as proposed by [9] and [5].

Algorithm 1 Modified RSDA Training Procedure.

procedure TRAIN(F, C, D)
    Optimize F, C and D using the Basic Loss $\mathcal{L}_{bas}$ (Equation 3.2) for N_{epochs} epochs.
    for $stage = 1$ to N_{stages} **do**
        $\tilde{y}_j^t, \gamma_j$ = GETCORRECTLABELINGPROBABILITY(F, C)
        Unfreeze and randomly reinitialize the weights of F, C, and D.
        Train F, C and D with the $\mathcal{L}$ loss (Equation 3.1), for N_{epochs} epochs.

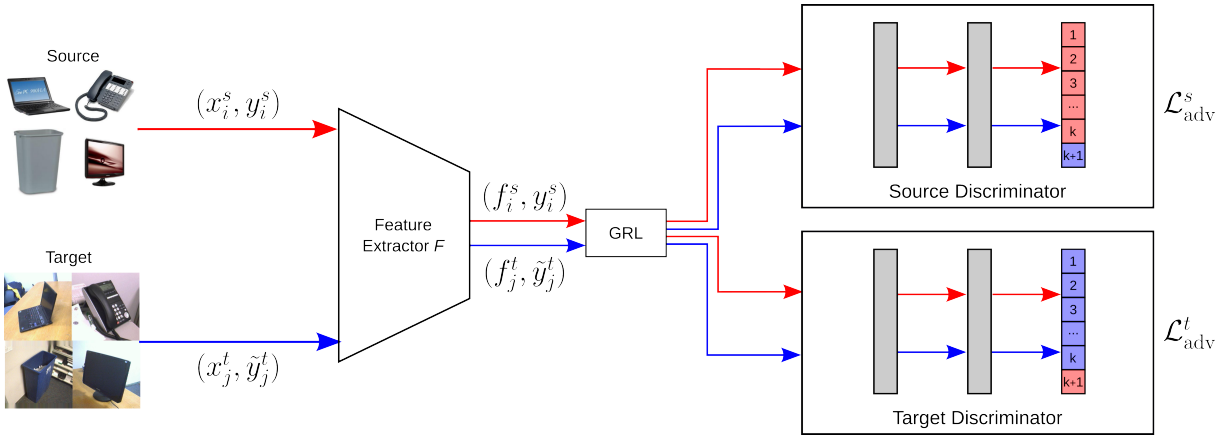
procedure GETCORRECTLABELINGPROBABILITY(F, C)
    Freeze the weights of F and C.
    Obtain the features f_i^s and f_j^t produced by F for all source x_i^s and target x_j^t samples.
    Assign the pseudo-labels $\tilde{y}_j^t$ based on the output of C.
    Run a dimensionality reduction algorithm on f_i^s and f_j^t to obtain the reduced features r_i^s and r_j^t.
    Compute the centroid $\mathcal{C}_k$ for each class $k = \{1, 2, \ldots, K\}$ in the reduced space.
    Compute the distances d_i^s and d_j^t between each reduced feature and the respective class centroid (ground-truth labels for source samples and pseudo-labels for target ones).
    Run the EM algorithm, as defined in [9], to obtain the parameters π_k, σ_k, and δ_k for each of the k mixture models.
    Estimate the correct labeling probability for the target samples γ_j with the Gaussian-uniform mixture models.
    return the estimated pseudo-labels $\tilde{y}_j^t$ with their respective correct labeling probability γ_j.

4.2 Multi-class Domain Discriminator

One of the best-performing strategies for DA consists of an adversarial framework with a discriminator that will classify the samples based on their domain. This discriminator will be trained via an adversarial objective in which it will get better at discriminating the samples, while the feature extractor network will also get better at producing more domain-invariant features, thus diminishing the effects of the data shift between the source and target domains. This strategy for DA was made popular by [5] with their proposal of the Domain Adversarial Neural Network (DANN), in which the min-max adversarial objective is directly implemented using a Gradient Reversal Layer, and the whole network is trained in an end-to-end fashion, as discussed in Chapter 2.

The discriminator network is usually comprised of a set of fully-connected layers that take the feature vector produced by the feature extractor network as input and perform a binary classification task by outputting the probability of the sample being from the source or target domains. The Gradient Reversal Layer (GRL) proposed by [5] multiplies the discriminator's gradient by a negative factor during backpropagation,

Figure 4.4: Proposed multi-class discriminator architecture.



thus inverting it. As during training the overall network error will be minimized, the discriminator error will actually be maximized due to the gradient inversion performed by the GRL. This adversarial objective will translate to more domain-invariant features, as these will lead to higher domain confusion, i.e., the inability of the discriminator network to correctly classify the samples based on their domain [5].

Some methods propose changes to how this discriminator network is structured. For instance, [18] propose that the discriminator should classify the samples into $k + 1$ classes rather than the original binary classification: the source samples will be classified into one of the k classes based on their ground-truth labels and the target ones will be classified into an additional class representing the *Other Domain*. The authors sustain that this allows for better use of the label information during the adaptation procedure, thus leading to better performance on target data.

Based on the multi-class discriminator concept, we propose leveraging the pseudo-labels estimated in RSDA [9] to incorporate the class information into the domain discrimination process. Originally, RSDA uses the same adversarial framework as proposed by [5]: a single binary domain classifier, and no class-label information is used in the adversarial loss, which takes into account only the domain label. As class labels are available for both domains due to the pseudo-label estimation for the target samples, we propose changing the original binary discriminator of RSDA to two separate multi-class discriminator networks:

- a **Source Discriminator**, which will classify the source samples based on their ground-truth labels and will assign all target samples to an extra *Other Domain* class;
- a **Target Discriminator**, which will classify the target samples based on their pseudo-labels and will assign all source samples to the extra *Other Domain* class.

In this configuration, there will be two separate discriminator networks, each with

$k+1$ classes: the k classes from the original image classification problem, plus an extra *Other Domain* class. Both discriminators will receive the feature vectors produced by the feature extractor network for samples from both domains and both discriminators will be connected to a Gradient Reversal Layer (GRL), which will implement the adversarial objective. This setup is illustrated in Figure 4.4.

During training, each discriminator will have its adversarial loss calculated separately: $\mathcal{L}^s_{\text{adv}}$ and $\mathcal{L}^t_{\text{adv}}$ for the source and target discriminators, respectively. In order to incorporate these losses into the original RSDA loss formulation presented in Equation 3.1, $\mathcal{L}^s_{\text{adv}}$ and $\mathcal{L}^t_{\text{adv}}$ should be aggregated. We propose using a dynamic linear operation to sum these losses together during training:

$$\mathcal{L}_{\text{adv}} = \omega\mathcal{L}^s_{\text{adv}} + (1-\omega)\mathcal{L}^t_{\text{adv}}, \tag{4.2}$$

where ω is a weight that will select which loss will have a higher influence on the complete adversarial loss.

At the start of the training $\omega = \omega_o = 1$, meaning that only the source discriminator's loss will be taken into account in the complete loss formulation. Then, in each following training epoch, ω will be decreased until reaching a final value $\omega = \omega_f$ in the last epoch. ω_f is a new hyper-parameter that must be manually set by the user. ω will have linear decrements in each epoch, and the actual value of this delta will be determined by the number of training epochs and the value of ω_f:

$$\begin{aligned} \omega_{n+1} &= \omega_n - \Delta_\omega \\ \Delta_\omega &= \frac{\omega_f - \omega_o}{E-1}, \end{aligned} \tag{4.3}$$

where n is the current epoch and E is the total amount of epochs.

Equations 4.2 and 4.3 were devised with the goal to give a higher importance to the source discriminator at the beginning of the training and balancing the discriminators' losses throughout the epochs by iteratively decreasing the source discriminator's importance while also increasing the target discriminator's one. This is done primarily to avoid any negative effects that may be caused by the inherent error associated with the target pseudo-labels. By starting out with a higher importance on the source discriminator, the network will be able to better capture the correct class information embedded in the source samples and then slowly incorporate the information from the target pseudo-labels.

As the correct pseudo-labeling probabilities estimated by the Gaussian-uniform mixture models are also available, the classification error of the pseudo-labeled samples on the target discriminator is weighted by this estimated correct labeling probability. This allows for more robust training, as it will reduce the negative influence of wrongly-assigned pseudo-labels.

The main motivation behind the proposal of this multi-class discriminator setup is the hypothesis that performing the domain discrimination on a local, class-based, level will result in a class-aware domain confusion that will ultimately lead to better adaptation results. The original binary discriminator as proposed by [5] does not leverage the class information and performs domain discrimination on a global level. This will lead to domain-invariability, but it may also have a negative impact on the feature's discriminability for the original classification task. Therefore, we sustain that incorporating class labels in this procedure using the estimated pseudo-labels for the target samples may lead to an overall better adaptation, as the domain-invariability will be introduced while also maintaining the class structures necessary for successfully performing the original image classification task.

Chapter 5

Experimental Results

In this chapter, the results achieved with the approaches proposed in this work are presented and discussed. We also present the results from our case study on performing domain adaptation in a real-world setting.

5.1 Experimental Setup

We first describe the setup employed during our experiments with the methods proposed in this work. This includes the data sets, neural network architecture, hyper-parameters, and other training configurations that were used.

Data Sets

We evaluate the proposed approaches presented in Chapter 4 using the Office-31 [37] and Office-Home [43] data sets, which are common benchmark data sets used throughout the DA literature.

The Office-31 data set contains 4,110 images of 31 object categories commonly found in an office environment. These images are distributed across 3 domains: Amazon, DSLR, and Webcam. The Amazon domain consists of images scrapped from the online store, hence the pictures usually do not have a background and have more uniform illumination and quality. The DSLR and Webcam domains are comprised of images taken in an actual office with a DSLR camera and a webcam, respectively. Therefore, the images from these two domains have complex backgrounds and also have more variation in quality and overall conditions, such as illumination and the object pose. Some sample images from this data set are presented in Figure 5.1.

Figure 5.1: Sample images from the Office-31 data set [37].

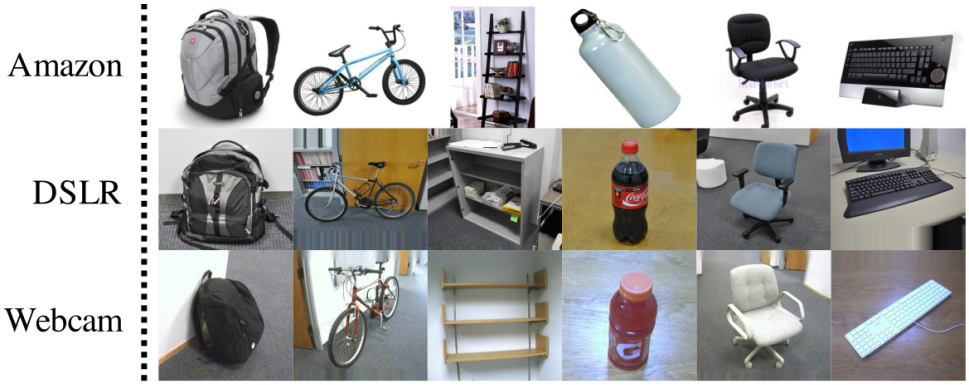


Figure adapted from [47].

Figure 5.2: Sample images from the Office-Home data set [43].

Figure extracted from [43].

The Office-Home data set is similar to the Office-31 one, but is bigger and more challenging. It has 15,500 images of 65 object categories divided into 4 domains: Artistic, Clipart, Product, and Real-World. The Clipart domain stands out, as it is comprised of vector drawings of objects, making it very different than the images in the other domains. The Product and Real-World domains are very similar to the Amazon and DSLR domains of Office-31, respectively. The Art domain consists of images that contain some post-processing and other visual artifacts that give them a unique style. Figure 5.2 contains some images from these 4 domains. Note that there is a big data shift across these domains for images in the same class, which can be perceived in the difference in image conditions and in the structure of each object itself.

We follow the standard unsupervised DA evaluation protocol for these data sets, in which the maximum accuracy achieved on target data is reported by varying the source-target pairs to cover all possible combinations in the data set. Furthermore, all the available data on both source and target domains are used during training, except for the target ground-truth labels. The maximum test accuracy achieved during training is reported by comparing the model's prediction for all target samples against the target ground-truth labels.

Implementation details

The proposed approaches were implemented using the PyTorch Python library, based on the code distributed by the authors of RSDA [9]. As in the original RSDA method, we also use the ResNet-50 [10] architecture with weights pre-trained on ImageNet as the feature extractor F. For the classifier C and domain discriminator D, we use the same spherical layers and activation functions described in the original RSDA paper [9], with a bottleneck dimension of 256. All hyper-parameters, including the network learning rate and the ones related to the Expectation-Maximization algorithm, are defined as in the original RSDA paper [9], so we can correctly evaluate the impact of the proposed enhancements.

Dimensionality Reduction

In order to evaluate the proposed addition of a dimensionality reduction step to the correct labeling probability estimation pipeline, we use two popular algorithms:

- Principal Component Analysis (PCA) [34], which reduces the dimensionality of data while also preserving the data's variance by finding principal components, i.e., the dimensions with a higher variance;
- Partial Least Squares (PLS) [46], which builds a lower-dimensional space by taking into account both the data and the labels by finding the multidimensional direction in the data space that will better explain the variance in the label space.

Both algorithms take the 256-dimensional feature vector produced by the feature extractor F and reduce it before calculating the centroid distances in the reduced feature space, as described in Chapter 4. For PLS, we experiment with different settings of output dimensions. For PCA, the dimensionality is reduced until a threshold of 95% of explained variance is met. We use the PCA and PLS implementations available in the scikit-learn Python package. We chose these two popular dimensionality reduction algorithms to easily incorporate this step into the original correct labeling probability estimation pipeline of [9]. Note that we also did not change the bottleneck dimensionality to not interfere with the network's ability to perform the original image classification task.

5.2 Enhancements to Pseudo-label Robustness

We first experiment with the enhancements to the correct labeling probability estimation pipeline, as described in Section 4.1. The results achieved for all the Source-Target pairs in the Office-31 and Office-Home[1] data sets are reported in Tables 5.1 and 5.2, respectively. We report the average classification accuracy and its standard deviation achieved on the target data with each method in three independent runs. In the first three rows, we present the baseline results for a no-adaptation scenario, in which the network is trained using only source-domain data, for the DANN [5] method, which is the baseline for RSDA, and the original RSDA method as proposed by [9], respectively.[2] Then, in the next two rows we present the results achieved with the incorporation of a dimensionality reduction algorithm into the correct labeling estimation pipeline (where c is the number of output dimensions when using the PLS algorithm). In the next row, the results achieved using data from *both* domains in this estimation are reported. Finally, in the last two rows, we present the results with the fully modified pipeline with the use of data from both domains and the addition of the dimensionality reduction step.

Analyzing the results in Tables 5.1 and 5.2, we can see that the introduction of a dimensionality reduction step to the correct labeling estimation pipeline resulted in a small improvement in the model's performance on target data in most scenarios, with the PLS algorithm having a slight advantage over the PCA one. In some scenarios, however, there was a small reduction in the achieved accuracy, which is probably due to the loss of information implicated by the lower dimensionality. Regarding the use of data from both domains in the correct labeling probability estimation, the results indicate that this change was able to considerably improve the target accuracy in most scenarios, with a gain of up to 4 percentage points. However, even as the addition of the dimensionality reduction alone was not able to significantly improve the baseline results, when it was combined with the use of data from both domains in the fully enhanced correct labeling probability estimation pipeline proposed in this work, we get an even greater improvement on the adaptation result, increasing up to 7 percentage points when compared to the results achieved by the baseline RSDA method.

In the fully enhanced correct labeling estimation pipeline, we can see that better target accuracies were achieved using the PLS dimensionality reduction algorithm in most scenarios. This is probably due to PLS taking into account the label information, ground-truth labels for source samples and pseudo-labels for target ones, when creating the reduced space. This shows that leveraging semantic class information has a positive

[1] Office-Home domain names have been shortened for better visualization: Art (Ar), Clipart (Cl), Product (Pr), and Realworld (Rw).

[2] We report the results achieved with the implementation made available by the RSDA authors.

Table 5.1: Target classification accuracy on the Office-31 data set with the enhanced correct labeling estimation pipeline

Method	Amazon-DSLR	Amazon-Webcam	DSLR-Amazon	DSLR-Webcam	Webcam-Amazon	Webcam-DSLR
No Adapt (ResNet-50 [10])	$80.0_{\pm 3.6}$	$79.6_{\pm 0.3}$	$59.5_{\pm 2.4}$	$91.4_{\pm 1.1}$	$61.9_{\pm 0.9}$	$99.0_{\pm 0.8}$
DANN [5]	$79.7_{\pm 0.4}$	$82.0_{\pm 0.4}$	$68.2_{\pm 0.4}$	$96.9_{\pm 0.2}$	$67.4_{\pm 0.5}$	$99.1_{\pm 0.1}$
RSDA [9]	$90.6_{\pm 0.1}$	$92.0_{\pm 0.7}$	$72.3_{\pm 1.1}$	$97.6_{\pm 0.2}$	$75.7_{\pm 0.5}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + PCA	$89.4_{\pm 0.2}$	$92.2_{\pm 0.1}$	$72.0_{\pm 0.8}$	$98.1_{\pm 0.1}$	$74.5_{\pm 0.1}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + PLS ($c = 10$)	$90.0_{\pm 0.3}$	$\mathbf{93.8_{\pm 0.2}}$	$71.0_{\pm 0.3}$	$97.9_{\pm 0.2}$	$75.0_{\pm 0.4}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + BOTH	$93.0_{\pm 0.3}$	$93.4_{\pm 0.3}$	$75.8_{\pm 0.5}$	$98.7_{\pm 0.5}$	$77.5_{\pm 0.8}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + BOTH + PCA	$92.3_{\pm 0.3}$	$93.0_{\pm 0.9}$	$75.9_{\pm 0.4}$	$\mathbf{99.2_{\pm 0.1}}$	$77.9_{\pm 0.1}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + BOTH + PLS ($c = 10$)	$\mathbf{93.4_{\pm 0.2}}$	$\mathbf{93.8_{\pm 0.4}}$	$\mathbf{79.3_{\pm 0.4}}$	$\mathbf{99.2_{\pm 0.1}}$	$\mathbf{78.8_{\pm 0.4}}$	$\mathbf{100.0_{\pm 0.0}}$

Table 5.2: Target classification accuracy on the Office-Home data set with the enhanced correct labeling estimation pipeline

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
No Adapt (ResNet-50 [10])	$36.4_{\pm 0.4}$	$58.3_{\pm 0.8}$	$68.7_{\pm 0.6}$	$44.1_{\pm 2.0}$	$52.1_{\pm 1.1}$	$55.8_{\pm 1.6}$	$46.1_{\pm 1.2}$	$32.6_{\pm 0.8}$	$67.0_{\pm 1.0}$	$63.0_{\pm 0.5}$	$40.7_{\pm 1.3}$	$74.1_{\pm 0.7}$
DANN [5]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8
RSDA [9]	$51.1_{\pm 0.6}$	$73.5_{\pm 0.4}$	$78.3_{\pm 0.2}$	$62.5_{\pm 0.4}$	$70.1_{\pm 0.3}$	$73.3_{\pm 0.4}$	$61.8_{\pm 0.5}$	$50.6_{\pm 0.7}$	$79.4_{\pm 0.3}$	$72.3_{\pm 0.4}$	$55.6_{\pm 0.5}$	$82.8_{\pm 0.6}$
RSDA + PCA	$50.9_{\pm 0.3}$	$73.6_{\pm 0.4}$	$78.7_{\pm 0.4}$	$62.4_{\pm 0.8}$	$71.9_{\pm 0.4}$	$73.8_{\pm 0.5}$	$62.6_{\pm 0.2}$	$50.7_{\pm 0.1}$	$79.1_{\pm 0.3}$	$72.1_{\pm 0.2}$	$55.7_{\pm 0.5}$	$82.7_{\pm 0.4}$
RSDA + PLS ($c = 20$)	$52.1_{\pm 0.2}$	$74.1_{\pm 0.3}$	$79.1_{\pm 0.3}$	$63.9_{\pm 0.4}$	$71.6_{\pm 0.3}$	$74.9_{\pm 0.3}$	$63.3_{\pm 0.1}$	$51.1_{\pm 0.2}$	$79.6_{\pm 0.1}$	$72.4_{\pm 0.5}$	$54.1_{\pm 0.2}$	$83.6_{\pm 0.3}$
RSDA + BOTH	$52.6_{\pm 0.2}$	$74.9_{\pm 0.5}$	$80.6_{\pm 0.3}$	$64.4_{\pm 0.4}$	$74.6_{\pm 0.4}$	$74.7_{\pm 0.1}$	$66.1_{\pm 0.4}$	$52.6_{\pm 0.1}$	$80.5_{\pm 0.1}$	$73.3_{\pm 0.4}$	$56.1_{\pm 0.2}$	$84.1_{\pm 0.2}$
RSDA + BOTH + PCA	$54.1_{\pm 0.4}$	$75.5_{\pm 0.2}$	$80.9_{\pm 0.4}$	$65.1_{\pm 0.5}$	$\mathbf{75.3_{\pm 0.2}}$	$75.1_{\pm 0.5}$	$66.3_{\pm 0.8}$	$\mathbf{53.6_{\pm 0.4}}$	$\mathbf{80.9_{\pm 0.1}}$	$73.7_{\pm 0.2}$	$\mathbf{58.2_{\pm 0.3}}$	$84.3_{\pm 0.3}$
RSDA + BOTH + PLS ($c = 20$)	$\mathbf{54.8_{\pm 0.3}}$	$\mathbf{75.9_{\pm 0.3}}$	$\mathbf{81.1_{\pm 0.1}}$	$\mathbf{66.3_{\pm 0.5}}$	$75.1_{\pm 0.2}$	$\mathbf{75.4_{\pm 0.4}}$	$\mathbf{67.2_{\pm 0.1}}$	$53.4_{\pm 0.3}$	$80.6_{\pm 0.2}$	$\mathbf{73.9_{\pm 0.1}}$	$57.8_{\pm 0.2}$	$\mathbf{84.7_{\pm 0.2}}$

impact when creating the reduced feature representations, leading to better adaptation.

By taking a look at the more challenging scenarios, such as the *DSLR-Amazon* and the *Webcam-Amazon* ones in the Office-31 data set and the ones involving the Clipart domain in the Office-Home data set, we can see that the proposed enhancements were also able to improve the overall results achieved on these harder settings. Specifically, in the *Art-Clipart* and *Product-Clipart* settings, in which the baselines have their worst results, our proposed enhancements were able to improve the baseline RSDA's accuracy by up to 3.7 percentage points.

In order to better understand how the size of the reduced space impacts the overall results, we perform an experiment varying the output dimensions of PLS. In Tables 5.3 and 5.4, the target classification accuracies obtained with different PLS output dimensions $c = 10, 15, 20, 25$ are presented.

Note that in the Office-31 data set, Table 5.3, the best accuracy was achieved using 10 dimensions, while in the Office-Home data set, Table 5.4, a higher number of dimensions, 20, achieved the best results. This illustrates how the Office-Home data set is more challenging than the Office-31 one, due to its higher number of classes and the more diverse collection of images. Therefore, more information is necessary to effectively perform the correct labeling probability estimation, requiring more dimensions. Note, however, that with more dimensions, $c = 25$, there is a slight decrease in the performance achieved in the Office-Home data set, indicating that a lower-dimensional space allows for a better estimation of the correct labeling probability, as long as there are enough dimensions to carry the semantic class-related information.

Figure 5.3 presents the t-SNE [42] visualizations of the features produced by the

Table 5.3: Varying PLS output dimensionality in the enhanced correct labeling estimation pipeline - Office-31 results

	Amazon-DSLR	Amazon-Webcam	DSLR-Amazon	DSLR-Webcam	Webcam-Amazon	Webcam-DSLR
$c = 10$	**93.4±0.2**	**93.8±0.4**	**79.3±0.4**	**99.2±0.1**	**78.8±0.4**	**100.0±0.0**
$c = 15$	92.5±0.2	93.1±0.3	78.4±0.2	99.1±0.1	78.1±0.5	**100.0±0.0**
$c = 20$	91.9±0.3	93.2±0.2	78.1±0.4	99.1±0.2	77.6±0.3	**100.0±0.0**
$c = 25$	92.1±0.1	92.8±0.2	77.9±0.3	99.0±0.2	77.8±0.9	**100.0±0.0**

Table 5.4: Varying PLS output dimensionality in the enhanced correct labeling estimation pipeline - Office-Home results

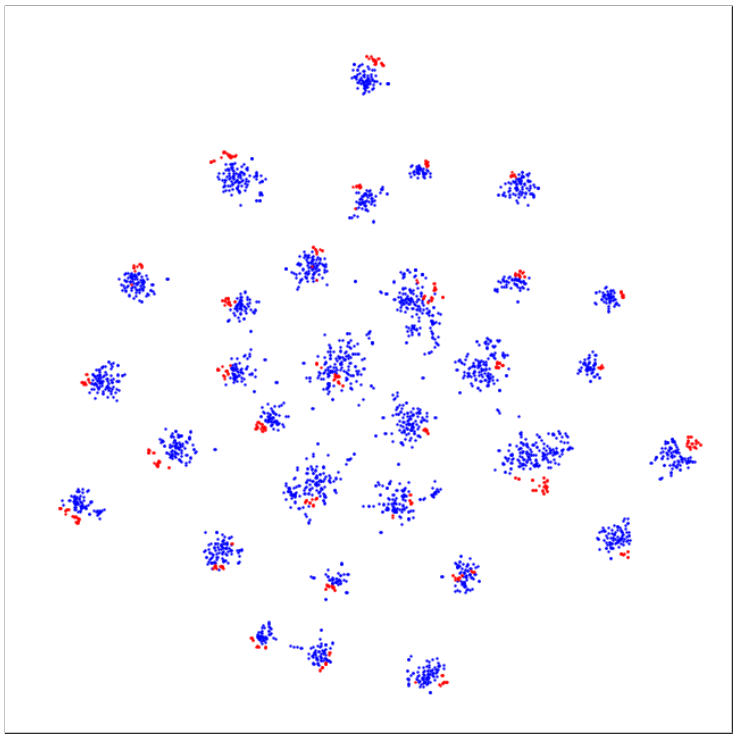
	Ar-Cl	Ar-Pr	Ar-Rw	Cl-Ar	Cl-Pr	Cl-Rw	Pr-Ar	Pr-Cl	Pr-Rw	Rw-Ar	Rw-Cl	Rw-Pr
$c = 10$	53.1±0.2	74.8±0.3	80.1±0.3	65.7±0.2	73.4±0.3	74.8±0.1	65.3±0.8	53.1±0.2	**81.1±0.5**	72.1±0.3	57.2±0.1	84.1±0.3
$c = 15$	53.7±0.3	75.1±0.1	80.6±0.2	65.9±0.3	74.4±0.8	75.2±0.3	65.8±0.5	52.9±0.1	80.5±0.2	72.8±0.2	57.7±0.1	84.6±0.1
$c = 20$	**54.8±0.3**	**75.9±0.3**	**81.1±0.1**	**66.3±0.5**	75.1±0.2	**75.4±0.4**	**67.2±0.1**	**53.4±0.3**	80.6±0.2	**73.9±0.1**	**57.8±0.2**	**84.7±0.2**
$c = 25$	54.2±0.2	75.5±0.2	**81.1±0.2**	66.1±0.3	**76.1±0.2**	75.1±0.2	66.9±0.2	52.5±0.2	80.5±0.2	73.2±0.3	57.1±0.2	83.3±0.2

feature extractor network F after the complete training procedure in the DSLR-Amazon scenario for both the original RSDA method and the enhanced one. The source samples are shown in red and the target ones in blue. Notice how the proposed enhancements led to features with better inter-class separability in both source and target domains, illustrated by the lower amount of points in the low-density areas between each class cluster. These observations corroborate that the proposed modifications to the original RSDA method resulted in an even more robust calculation of the correct labeling probabilities, which in turn led to better classification results, as presented in the previous tables.

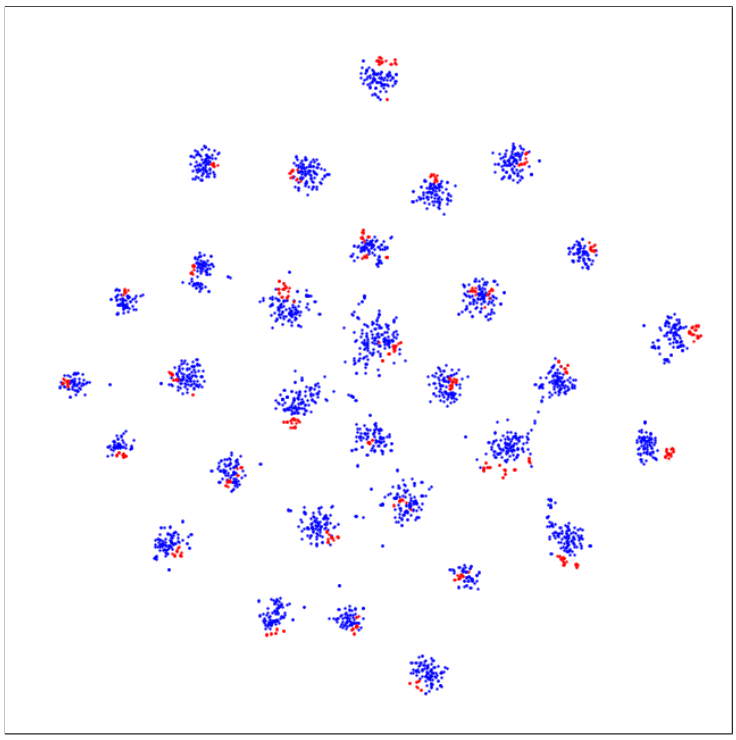
In summary, the overall analysis of the achieved results shows that the proposed enhanced pipeline for estimating the correct labeling probabilities, with the addition of a dimensionality reduction step and the use of data from both domains in the Expectation-Maximization algorithm, led to an improvement over the original method, thus indicating that the hypotheses that guided these modifications hold true. These results then demonstrate that it is beneficial for DA that we make effective use of the available data in both source and target domains and that the transformation of the features to a lower dimensional space can indeed lead to a more robust adaptation of the semantic knowledge across the domains.

Figure 5.3: Feature t-SNE visualization.

(a) Baseline RSDA [9].



(b) RSDA with the proposed enhanced correct labeling probability estimation pipeline.



5.3 Multi-class Discriminator Results

In this section, the results achieved with the multi-class discriminator architecture described in Section 4.2 are presented and discussed. The classification accuracy obtained in the target domain by replacing the original adversarial loss of RSDA [9] with the proposed multi-class discriminator one for each Source-Target pair in the Office-31 and Office-Home data sets are presented in Tables 5.5 and 5.6, respectively. We present the target classification accuracy obtained for different values of $\omega_f = 0.0, 0.25, 0.5, 0.75, 1.0$. As described in Section 4.2, ω_f is a hyper-parameter that controls the importance given to the source and target discriminators. As ω_f increases, the overall importance given to the target discriminator throughout the training decreases. Therefore, in the scenario with $\omega_f = 0.0$, the target discriminator will have a higher importance at the end of the training than when $\omega_f = 1.0$, in which only the source discriminator will be taken into account as the target one will have a weight of 0.0 throughout the whole training.

Table 5.5: Target classification accuracy on Office-31 data set with the proposed multi-class discriminator

	Amazon-DSLR	Amazon-Webcam	DSLR-Amazon	DSLR-Webcam	Webcam-Amazon	Webcam-DSLR
No Adapt (ResNet-50 [10])	$80.0_{\pm 3.6}$	$79.6_{\pm 0.3}$	$59.5_{\pm 2.4}$	$91.4_{\pm 1.1}$	$61.9_{\pm 0.9}$	$99.0_{\pm 0.8}$
DANN [5]	$79.7_{\pm 0.4}$	$82.0_{\pm 0.4}$	$68.2_{\pm 0.4}$	$96.9_{\pm 0.2}$	$67.4_{\pm 0.5}$	$99.1_{\pm 0.1}$
RSDA [9]	$90.6_{\pm 0.1}$	$92.0_{\pm 0.7}$	$72.3_{\pm 1.1}$	$97.6_{\pm 0.2}$	$75.7_{\pm 0.5}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + Multi-class Discriminator ($\omega_f = 0.0$)	$91.1_{\pm 0.1}$	$92.1_{\pm 0.3}$	$71.5_{\pm 0.1}$	$97.1_{\pm 0.2}$	$74.2_{\pm 0.2}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + Multi-class Discriminator ($\omega_f = 0.25$)	$92.1_{\pm 0.2}$	$93.1_{\pm 0.4}$	$72.2_{\pm 0.3}$	$97.8_{\pm 0.3}$	$75.4_{\pm 0.2}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + Multi-class Discriminator ($\omega_f = 0.5$)	$92.2_{\pm 0.1}$	$93.4_{\pm 0.2}$	$73.2_{\pm 0.3}$	$98.2_{\pm 0.1}$	$76.8_{\pm 0.3}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + Multi-class Discriminator ($\omega_f = 0.75$)	$\mathbf{92.7_{\pm 0.2}}$	$\mathbf{94.1_{\pm 0.3}}$	$\mathbf{74.1_{\pm 0.4}}$	$\mathbf{98.9_{\pm 0.1}}$	$\mathbf{77.9_{\pm 0.3}}$	$\mathbf{100.0_{\pm 0.0}}$
RSDA + Multi-class Discriminator ($\omega_f = 1.0$)	$91.9_{\pm 0.2}$	$92.6_{\pm 0.2}$	$73.1_{\pm 0.2}$	$97.9_{\pm 0.1}$	$76.1_{\pm 0.1}$	$\mathbf{100.0_{\pm 0.0}}$

Table 5.6: Target classification accuracy on Office-Home data set with the proposed multi-class discriminator

	Ar-Cl	Ar-Pr	Ar-Rw	Cl-Ar	Cl-Pr	Cl-Rw	Pr-Ar	Pr-Cl	Pr-Rw	Rw-Ar	Rw-Cl	Rw-Pr
No Adapt (ResNet-50 [10])	$36.4_{\pm 0.4}$	$58.3_{\pm 0.8}$	$68.7_{\pm 0.6}$	$44.1_{\pm 2.0}$	$52.1_{\pm 1.1}$	$55.8_{\pm 1.6}$	$46.1_{\pm 1.2}$	$32.6_{\pm 0.8}$	$67.0_{\pm 1.0}$	$63.0_{\pm 0.5}$	$40.7_{\pm 1.3}$	$74.1_{\pm 0.7}$
DANN [5]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8
RSDA [9]	$51.1_{\pm 0.6}$	$73.5_{\pm 0.4}$	$78.3_{\pm 0.2}$	$62.5_{\pm 0.4}$	$70.1_{\pm 0.3}$	$73.3_{\pm 0.4}$	$61.8_{\pm 0.5}$	$50.6_{\pm 0.7}$	$79.4_{\pm 0.3}$	$72.3_{\pm 0.4}$	$55.6_{\pm 0.5}$	$82.8_{\pm 0.6}$
RSDA + Multi-class Discriminator ($\omega_f = 0.0$)	$50.4_{\pm 0.1}$	$71.9_{\pm 0.2}$	$77.8_{\pm 0.3}$	$62.1_{\pm 0.2}$	$69.7_{\pm 0.2}$	$72.3_{\pm 0.1}$	$61.6_{\pm 0.2}$	$49.8_{\pm 0.1}$	$79.2_{\pm 0.2}$	$71.8_{\pm 0.1}$	$55.2_{\pm 0.2}$	$81.7_{\pm 0.3}$
RSDA + Multi-class Discriminator ($\omega_f = 0.25$)	$50.8_{\pm 0.3}$	$72.1_{\pm 0.4}$	$78.4_{\pm 0.1}$	$62.5_{\pm 0.4}$	$70.4_{\pm 0.2}$	$72.8_{\pm 0.2}$	$61.9_{\pm 0.2}$	$50.2_{\pm 0.3}$	$79.5_{\pm 0.1}$	$71.9_{\pm 0.2}$	$55.7_{\pm 0.3}$	$82.2_{\pm 0.2}$
RSDA + Multi-class Discriminator ($\omega_f = 0.5$)	$51.2_{\pm 0.3}$	$73.4_{\pm 0.2}$	$78.7_{\pm 0.2}$	$63.2_{\pm 0.3}$	$71.9_{\pm 0.1}$	$73.9_{\pm 0.1}$	$63.1_{\pm 0.4}$	$51.5_{\pm 0.2}$	$79.7_{\pm 0.3}$	$72.2_{\pm 0.2}$	$56.1_{\pm 0.4}$	$82.7_{\pm 0.3}$
RSDA + Multi-class Discriminator ($\omega_f = 0.75$)	$\mathbf{52.1_{\pm 0.2}}$	$\mathbf{74.4_{\pm 0.2}}$	$\mathbf{79.5_{\pm 0.2}}$	$\mathbf{64.1_{\pm 0.3}}$	$\mathbf{72.5_{\pm 0.3}}$	$\mathbf{74.7_{\pm 0.5}}$	$\mathbf{64.2_{\pm 0.2}}$	$\mathbf{52.9_{\pm 0.3}}$	$\mathbf{80.2_{\pm 0.2}}$	$\mathbf{72.4_{\pm 0.1}}$	$\mathbf{56.4_{\pm 0.3}}$	$\mathbf{83.2_{\pm 0.2}}$
RSDA + Multi-class Discriminator ($\omega_f = 1.0$)	$51.3_{\pm 0.2}$	$73.8_{\pm 0.2}$	$78.6_{\pm 0.2}$	$63.6_{\pm 0.4}$	$72.1_{\pm 0.1}$	$74.1_{\pm 0.2}$	$63.3_{\pm 0.2}$	$51.7_{\pm 0.2}$	$79.7_{\pm 0.2}$	$72.1_{\pm 0.2}$	$55.9_{\pm 0.2}$	$82.8_{\pm 0.1}$

The results in Tables 5.5 and 5.6 indicate that the proposed multi-class discriminator architecture is able to improve the baseline RSDA results in all tested scenarios when $\omega_f = 0.75$, i.e., when we give higher importance to the source discriminator but still take the target one into consideration. Note, however, that when we increase the importance given to the target discriminator by decreasing ω_f, the method achieves considerably lower accuracies, sometimes even worse than the baseline RSDA, especially when $\omega_f = 0.0$. This may be explained by the inherent error embedded in the pseudo-labels that are used in the target discriminator. Even though the correct labeling probabilities

Figure 5.4: Varying ω_f for Office-31's Webcam-Amazon setting.

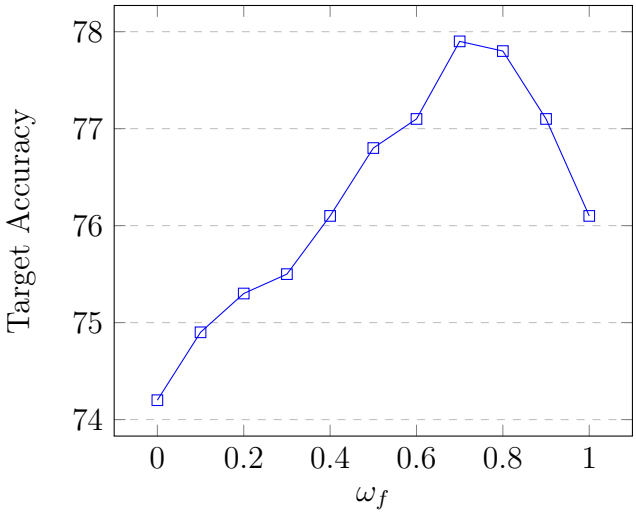
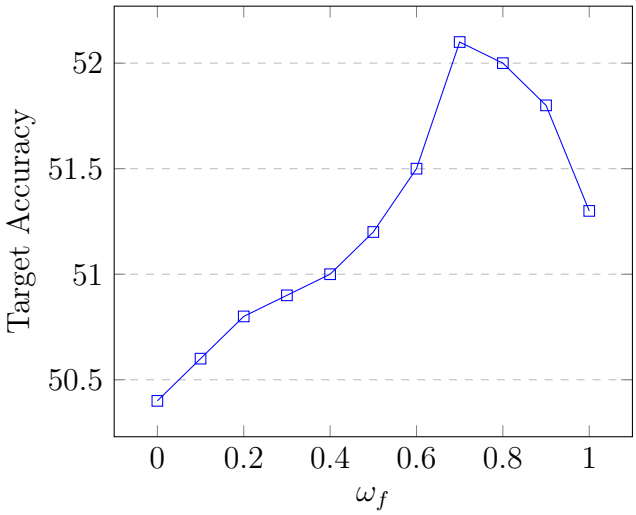


Figure 5.5: Varying ω_f for Office-Home's Art-Clipart setting.



are used to weight the classification error in the target discriminator, the results show that the imprecision associated with the pseudo-labels still causes a negative impact on the training procedure. By reducing the overall importance of the target discriminator in the $\omega_f = 0.75$ scenario, we can balance the loss in performance due to using the pseudo-labels with the gain caused by class-aware domain discrimination, hence leading to better results. Note, however, that when we fully remove the target discriminator's influence by setting $\omega_f = 1.0$, we get lower results than with $\omega_f = 0.75$, thus indicating that using the pseudo-labeled target samples in the adversarial domain discrimination loss does, in fact, contributes positively to the adaptation.

To better evaluate the impact of the ω_f parameter, we experiment with a larger range of values for some scenarios. In Figures 5.4 and 5.5, the accuracy achieved using different values of ω_f are reported. In these graphics, we can clearly see how the accuracy behaves as we modify the value of ω_f: as we increase ω_f, the accuracy also increases

Table 5.7: Target classification accuracy on Office-31 data set combining the enhanced RSDA correct labeling estimation pipeline (PLS, $c = 10$) and the multi-class discriminator

	Amazon-DSLR	Amazon-Webcam	DSLR-Amazon	DSLR-Webcam	Webcam-Amazon	Webcam-DSLR
No Adapt (ResNet-50 [10])	$80.0_{\pm 3.6}$	$79.6_{\pm 0.3}$	$59.5_{\pm 2.4}$	$91.4_{\pm 1.1}$	$61.9_{\pm 0.9}$	$99.0_{\pm 0.8}$
DANN [5]	$79.7_{\pm 0.4}$	$82.0_{\pm 0.4}$	$68.2_{\pm 0.4}$	$96.9_{\pm 0.2}$	$67.4_{\pm 0.5}$	$99.1_{\pm 0.1}$
RSDA [9]	$90.6_{\pm 0.1}$	$92.0_{\pm 0.7}$	$72.3_{\pm 1.1}$	$97.6_{\pm 0.2}$	$75.7_{\pm 0.5}$	$\mathbf{100.0_{\pm 0.0}}$
Enhanced RSDA	$93.4_{\pm 0.2}$	$93.8_{\pm 0.4}$	$79.3_{\pm 0.4}$	$99.2_{\pm 0.1}$	$78.8_{\pm 0.4}$	$\mathbf{100.0_{\pm 0.0}}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 0.0$)	$92.6_{\pm 0.3}$	$93.2_{\pm 0.2}$	$78.8_{\pm 0.3}$	$98.9_{\pm 0.1}$	$77.1_{\pm 0.2}$	$\mathbf{100.0_{\pm 0.0}}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 0.25$)	$92.9_{\pm 0.5}$	$93.5_{\pm 0.2}$	$79.1_{\pm 0.2}$	$99.1_{\pm 0.2}$	$77.5_{\pm 0.2}$	$\mathbf{100.0_{\pm 0.0}}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 0.5$)	$93.2_{\pm 0.4}$	$94.2_{\pm 0.3}$	$79.3_{\pm 0.3}$	$99.3_{\pm 0.1}$	$78.9_{\pm 0.3}$	$\mathbf{100.0_{\pm 0.0}}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 0.75$)	$\mathbf{93.8_{\pm 0.2}}$	$\mathbf{94.6_{\pm 0.1}}$	$\mathbf{79.7_{\pm 0.3}}$	$\mathbf{99.5_{\pm 0.1}}$	$\mathbf{79.2_{\pm 0.2}}$	$\mathbf{100.0_{\pm 0.0}}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 1.0$)	$93.5_{\pm 0.3}$	$94.1_{\pm 0.2}$	$79.2_{\pm 0.1}$	$99.3_{\pm 0.1}$	$78.9_{\pm 0.4}$	$\mathbf{100.0_{\pm 0.0}}$

Table 5.8: Target classification accuracy on Office-Home data set combining the enhanced RSDA correct labeling estimation pipeline (PLS, $c = 20$) and the multi-class discriminator

	Ar-Cl	Ar-Pr	Ar-Rw	Cl-Ar	Cl-Pr	Cl-Rw	Pr-Ar	Pr-Cl	Pr-Rw	Rw-Ar	Rw-Cl	Rw-Pr
No Adapt (ResNet-50 [10])	$36.4_{\pm 0.4}$	$58.3_{\pm 0.8}$	$68.7_{\pm 0.6}$	$44.1_{\pm 2.0}$	$52.1_{\pm 1.1}$	$55.8_{\pm 1.6}$	$46.1_{\pm 1.2}$	$32.6_{\pm 0.8}$	$67.0_{\pm 1.0}$	$63.0_{\pm 0.5}$	$40.7_{\pm 1.3}$	$74.1_{\pm 0.7}$
DANN [5]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8
RSDA [9]	$51.1_{\pm 0.6}$	$73.5_{\pm 0.4}$	$78.3_{\pm 0.2}$	$62.5_{\pm 0.4}$	$70.1_{\pm 0.3}$	$73.3_{\pm 0.4}$	$61.8_{\pm 0.5}$	$50.6_{\pm 0.7}$	$79.4_{\pm 0.3}$	$72.3_{\pm 0.4}$	$55.6_{\pm 0.5}$	$82.8_{\pm 0.6}$
Enhanced RSDA	$54.8_{\pm 0.3}$	$75.9_{\pm 0.3}$	$81.1_{\pm 0.1}$	$66.3_{\pm 0.5}$	$75.1_{\pm 0.2}$	$75.4_{\pm 0.4}$	$67.2_{\pm 0.1}$	$53.4_{\pm 0.3}$	$80.6_{\pm 0.2}$	$\mathbf{73.9_{\pm 0.1}}$	$57.8_{\pm 0.2}$	$84.7_{\pm 0.2}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 0.0$)	$54.2_{\pm 0.2}$	$76.2_{\pm 0.2}$	$79.9_{\pm 0.1}$	$66.2_{\pm 0.3}$	$74.1_{\pm 0.2}$	$73.9_{\pm 0.4}$	$66.3_{\pm 0.1}$	$53.1_{\pm 0.2}$	$79.8_{\pm 0.3}$	$72.7_{\pm 0.2}$	$57.1_{\pm 0.3}$	$83.9_{\pm 0.2}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 0.25$)	$54.6_{\pm 0.4}$	$76.8_{\pm 0.3}$	$80.2_{\pm 0.2}$	$66.9_{\pm 0.4}$	$74.8_{\pm 0.1}$	$74.9_{\pm 0.3}$	$66.9_{\pm 0.2}$	$53.3_{\pm 0.1}$	$80.2_{\pm 0.2}$	$73.1_{\pm 0.3}$	$57.4_{\pm 0.2}$	$84.2_{\pm 0.1}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 0.5$)	$54.3_{\pm 0.4}$	$\mathbf{77.1_{\pm 0.3}}$	$80.5_{\pm 0.0}$	$67.1_{\pm 0.6}$	$75.5_{\pm 0.1}$	$75.6_{\pm 0.2}$	$67.5_{\pm 0.3}$	$53.6_{\pm 0.3}$	$80.3_{\pm 0.1}$	$73.5_{\pm 0.2}$	$57.7_{\pm 0.3}$	$84.5_{\pm 0.2}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 0.75$)	$\mathbf{54.9_{\pm 0.3}}$	$76.8_{\pm 0.5}$	$\mathbf{81.2_{\pm 0.5}}$	$\mathbf{67.0_{\pm 0.2}}$	$\mathbf{76.1_{\pm 0.4}}$	$\mathbf{76.0_{\pm 0.2}}$	$\mathbf{67.8_{\pm 0.7}}$	$\mathbf{54.1_{\pm 0.3}}$	$\mathbf{80.9_{\pm 0.2}}$	$\mathbf{73.9_{\pm 0.4}}$	$\mathbf{58.1_{\pm 0.2}}$	$\mathbf{84.9_{\pm 0.2}}$
Enhanced RSDA + Multi-class Discriminator ($\omega_f = 1.0$)	$54.4_{\pm 0.2}$	$76.4_{\pm 0.2}$	$79.7_{\pm 0.2}$	$66.6_{\pm 0.1}$	$75.4_{\pm 0.2}$	$75.5_{\pm 0.3}$	$67.3_{\pm 0.2}$	$53.2_{\pm 0.2}$	$80.7_{\pm 0.1}$	$73.6_{\pm 0.2}$	$57.6_{\pm 0.1}$	$84.8_{\pm 0.2}$

until achieving a maximum value between $\omega_f = 0.7$ and $\omega_f = 0.8$. Then, the accuracy drops sharply as we completely remove the target discriminator's influence on the complete adversarial loss when $\omega_f = 1.0$. These results corroborate the previously made observation that the negative influence of wrongly assigned pseudo-labels and the positive effect of class-aware domain discrimination including target samples seems to balance out resulting in an overall better adaptation when $\omega_f \sim 0.75$.

We also run experiments combining the proposed enhanced correct labeling estimation pipeline described in Section 4.1 and the multi-class discriminator architecture. The results are presented in Tables 5.7 and 5.8, where we compare the baselines against the combined method.

The addition of the multi-class discriminator architecture to the enhanced RSDA pipeline was able to further increase the target classification accuracy by a slight margin in most scenarios. We can also see the same pattern regarding the ω_f parameter, in which we get better results with a higher value of ω_f. These results show that, while each enhancement can considerably improve the target accuracy over the baseline RSDA method, the combination of both enhancements does not result in a big improvement over the results achieved by each one separately. Therefore, this indicates that there may be other ways to better incorporate the more robust correct pseudo-labeling probability estimation into the multi-class discrimination architecture, in order to bypass the negative effects of wrongly assigned pseudo-labels. Nevertheless, the overall results suggest that both approaches could indeed improve the results achieved by the baseline RSDA method.

5.4 Case Study: Domain Adaptation for Automatic PPE Detection

Domain adaptation methods can be leveraged in real-world scenarios to greatly reduce the development cost of smart systems, by allowing semantically-related data that are already available to be used to train machine learning models. To test the effectiveness of performing DA in a real-world setting, we conduct an experiment based on automatic Personal Protective Equipment (PPE) usage detection.

PPEs, such as hard hats, vests, and protection glasses, are a fundamental part of work safety as they protect workers against serious injuries that may be caused by workplace accidents. Therefore, their use is mandatory in places such as construction sites, oil platforms, and factories. Enforcing PPE usage can be a tedious task for managers and work safety professionals, as there are usually many workers on a job site, and checking if each one of them is using all the required PPEs takes a lot of time. To this end, a smart system can be used to automate this task, by processing images captured from surveillance cameras placed across the site and automatically detecting which workers are not wearing the full set of required PPEs.

As much as a smart system that is able to automatically and accurately check for PPE usage is very desired, gathering and labeling the data for training the machine learning models that would perform this detection is a very time and resource-consuming task: one would need to record many hours of footage at each work-site where the system would be active and then manually label each image based on the PPEs that each subject is wearing. This will be a huge obstacle in the development of such a system. Therefore, a solution would be to apply a DA method to adapt data that are already available to avoid having to collect and label multiple data sets, thus skipping the previously mentioned high cost.

To explore the DA solution in the aforementioned scenario, we created a test setup consisting of surveillance images from two different data sets:

- **Simulation**: images captured with surveillance cameras in a simulated work environment, in which test subjects wear different PPE combinations and perform a pre-defined routine in front of the cameras;
- **Real**: images captured with surveillance cameras installed in a real workshop, where actual workers are performing their duties as usual. These images were recorded over multiple weeks and no intervention was made to the workers' usual day-to-day tasks.

Figure 5.6: Sample images from the **Simulation** data set.

Figure 5.7: Sample images from the **Real** data set.

Some images from these data sets are presented in Figures 5.6 and 5.7. Notice how there is a considerable difference in image conditions between the simulated environment and the real one. Furthermore, we can see from these samples that the real data set contains a lot more variation in the worker's pose and the aspects of each PPE, as these images were collected in an uncontrolled environment with the workers going through their usual day-to-day routines without any kind of intervention.

The main goal of this case study is to train a PPE detection model using *labeled* data from the simulated environment and *unlabeled* data from the real domain, thus simulating a real application of unsupervised DA. To this end, we train models for detecting two types of PPE: hard hat and vest. The model should take an image of the worker, as shown in Figures 5.6 and 5.7, and perform a binary classification task by outputting 0 if the PPE is not present and 1 if it is. The amount of images in each class in each data set is presented in Tables 5.9 and 5.10.

For this experiment, we evaluate the adaptation performance of the original RSDA

Table 5.9: Number of images with and without **Hard hat** in the Simulated and Real data sets

	Simulated	Real
With Hard hat	5000	15000
Without Hard hat	5000	15000

Table 5.10: Number of images with and without **Vest** in the Simulated and Real data sets

	Simulated	Real
With Vest	2500	15000
Without Vest	2500	15000

[9] method and the enhanced version with the changes discussed in Chapter 4, by comparing the achieved target accuracy to that of a no-adaptation scenario, in which the model is trained using only simulated data and tested on the real data.

We use the ResNet-50 [10] architecture with weights pre-trained on ImageNet as the feature extractor network and we use the same protocol used for the Office-31 and Office-Home data sets: all labeled source data and all unlabeled target data are used during training and we evaluate the model on all target data using the ground-truth labels.

We first evaluate the accuracy achieved on each data set in a traditional machine learning setup, in which we train the neural network using only labeled data from each domain. Therefore, there is no mix between data from different data sets, that is, each one is trained separately and no transfer or adaptation is performed. These results are presented in Table 5.11. Notice how we are able to achieve high accuracies in both data sets, with a slightly lower accuracy on Real data, due to the higher variety present in this domain, which makes classifying its samples harder than the Simulated ones.

Table 5.11: Classification accuracy achieved with traditional training

	Hard hat	Vest
Simulated	$98.7_{\pm 0.1}$	$97.4_{\pm 0.2}$
Real	$97.6_{\pm 0.1}$	$95.3_{\pm 0.1}$

Then we experiment with actually performing the adaptation from the source Simulated domain to the target Real one. The adaptation results are summarized in Table 5.12, where we report the accuracy achieved on target data, i.e., the Real data set images, after training the model using the labeled Simulated images and the unlabeled Real ones. By analyzing these results, we can see that the original RSDA method as proposed by [9] was able to considerably improve the accuracy over the no-adaptation

Table 5.12: Classification accuracy on **Real** data set images after performing the adaptation task **Simulated→Real**

	Hard hat	Vest
No Adapt (ResNet-50 [10])	$68.2_{\pm 0.2}$	$54.9_{\pm 0.2}$
RSDA [9]	$73.8_{\pm 0.2}$	$60.2_{\pm 0.3}$
Enhanced RSDA (PLS, $c = 20$)	$75.6_{\pm 0.3}$	$62.3_{\pm 0.2}$
Enhanced RSDA (PLS, $c = 20$) + Multi-class Discriminator ($\omega_f = 0.75$)	$\mathbf{76.9_{\pm 0.1}}$	$\mathbf{62.9_{\pm 0.3}}$

scenario. We can also see that the accuracy obtained with the Vest PPE is significantly lower than the one achieved for the Hard hat. This can be explained by the higher variety of vest types that exist on the Real data, as is illustrated by the samples in Figure 5.7, and also by the lower amount of images available in the Simulated domain, as shown in Tables 5.9 and 5.10.

Nevertheless, the enhancements proposed in this work were also able to further improve the target domain accuracy in this real-world example. When compared to the original RSDA baseline, the method including the proposed enhancements to the correct labeling estimation pipeline and the multi-class discriminator architecture was able to improve the target accuracy by up to 3 p.p., thus showing how the proposed enhancements could in fact improve the overall robustness of the original RSDA method.

However, even though the results show that DA methods can improve the accuracy over a no-adaptation scenario, the efficacy of such approaches decreases as the data shift across the domains increases. This is evidenced by the lower overall results achieved for the Vest PPE, where, as previously mentioned, we have lower data variety in the Simulated data set, leading to a larger data shift when compared to the Real domain's data distribution. Furthermore, when comparing the adaptation results to those of the traditional learning setup in Table 5.11, we notice a considerable gap between the accuracies, with the traditional setup achieving upwards of 30 more percentage points than the adaptation-based approach. This shows that even though the experimented DA methods can improve the results over a no-adaptation scenario, actually labeling the target data, hence dealing with the associated high labeling cost, and performing a traditional training procedure still leads to far greater results.

In summary, this case study shows that DA methods can be leveraged in real-world scenarios to improve the performance on target data, with a considerable improvement over a no-adaptation scenario. However, it also shows that these methods still cannot achieve the same results as in a traditional learning setup that uses actual target labeled data. Therefore, when developing a smart system that relies on machine learning models, one must ponder the savings of skipping the high labeling cost by adapting available data and the associated reduction of the system's overall accuracy caused by the lack of ground-truth target labeled data.

5.5 Discussion

Overall, the experimental results show that our proposed changes to the RSDA [9] method could consistently improve the accuracy achieved on target data in the commonly used Office-31 and Office-Home benchmark data sets. The better use of the available data by incorporating the source labeled samples in the correct labeling probability estimation pipeline allied with the addition of a dimensionality reduction step before calculating the sample class centroid distance in the feature space was able to considerably improve the results achieved by the baseline method by up to 7 percentage points. These results indicate that exploring ways to better incorporate the available data in the source and target domains can, in fact, lead to better adaptation.

Furthermore, the proposed multi-class discriminator architecture could also improve the baseline results, showing that taking the label information, including the pseudo-labeled target data, into consideration while performing the domain discrimination can introduce a class-aware domain confusion, which will ultimately lead to better accuracy on the target domain. The incorporation of this multi-class discriminator architecture into the RSDA method with our enhancements to the correct labeling estimation pipeline further improved the results over the baseline, albeit by a small margin, indicating that future works can explore better ways to incorporate the correct labeling probability into the multi-class discriminator architecture to achieve even greater results.

Since RSDA was first introduced by [9], other methods that also build upon the adversarial framework and the pseudo-labeling strategy for DA have been proposed. Recently, [29] proposed CoVi, in which the vicinal label space is leveraged for an even more robust use of the source labels and target pseudo-labels. [40] enhanced the adversarial framework by integrating it with a visual transformer network, which processes the images via a tokenized representation and refines the extracted features using global self-attention. In Tables 5.13 and 5.14, we compare the results achieved by our baselines, our proposed enhancements (using PLS, with $c = 10$ for Office-31 and $c = 20$ for Office-Home, and $\omega_f = 0.75$ for both data sets), and these recently introduced methods. Note that our proposed enhancements to RSDA still beat the results achieved by CoVi [29] in some settings from both data sets. However, SSRT, with its use of a modern visual transformer, was able to greatly improve the results in all settings from these data sets, especially in the hardest ones of Office-Home.

The results achieved by the newer methods show that research on the DA topic is progressing very rapidly, as there is a great desire to have a robust method that is able to leverage the large publicly-available data repositories for training deep learning models. Nevertheless, the enhancements proposed in this work still provide meaningful insights that can further improve the DA performance of these newer methods:

Table 5.13: Target classification accuracy on Office-31 data set for different methods (highest accuracy in **bold**, second-best is underlined)

Method	Amazon-DSLR	Amazon-Webcam	DSLR-Amazon	DSLR-Webcam	Webcam-Amazon	Webcam-DSLR
No Adapt (ResNet-50 [10])	$80.0_{\pm 3.6}$	$79.6_{\pm 0.3}$	$59.5_{\pm 2.4}$	$91.4_{\pm 1.1}$	$61.9_{\pm 0.9}$	$99.0_{\pm 0.8}$
DANN [5]	$79.7_{\pm 0.4}$	$82.0_{\pm 0.4}$	$68.2_{\pm 0.4}$	$96.9_{\pm 0.2}$	$67.4_{\pm 0.5}$	$99.1_{\pm 0.1}$
RSDA [9]	$90.6_{\pm 0.1}$	$92.0_{\pm 0.7}$	$72.3_{\pm 1.1}$	$97.6_{\pm 0.2}$	$75.7_{\pm 0.5}$	$\mathbf{100.0_{\pm 0.0}}$
Enhanced RSDA	$93.4_{\pm 0.2}$	$93.8_{\pm 0.4}$	$79.3_{\pm 0.4}$	$99.2_{\pm 0.1}$	$78.8_{\pm 0.4}$	$\mathbf{100.0_{\pm 0.0}}$
Enhanced RSDA + Multi-class Discriminator	$93.8_{\pm 0.2}$	$94.6_{\pm 0.1}$	$79.7_{\pm 0.3}$	$\mathbf{99.5_{\pm 0.1}}$	$79.2_{\pm 0.2}$	$\mathbf{100.0_{\pm 0.0}}$
CoVi [29]	$98.0_{\pm 0.3}$	$97.6_{\pm 0.2}$	$77.5_{\pm 0.3}$	$99.3_{\pm 0.1}$	$78.4_{\pm 0.3}$	$\mathbf{100.0_{\pm 0.0}}$
SSRT [40]	**98.6**	**97.7**	**83.5**	99.2	**82.2**	**100.0**

Table 5.14: Target classification accuracy on Office-Home data set for different methods (highest accuracy in **bold**, second-best is underlined)

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
No Adapt (ResNet-50 [10])	$36.4_{\pm 0.4}$	$58.3_{\pm 0.8}$	$68.7_{\pm 0.6}$	$44.1_{\pm 2.0}$	$52.1_{\pm 1.1}$	$55.8_{\pm 1.6}$	$46.1_{\pm 1.2}$	$32.6_{\pm 0.8}$	$67.0_{\pm 1.0}$	$63.0_{\pm 0.5}$	$40.7_{\pm 1.3}$	$74.1_{\pm 0.7}$
DANN [5]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8
RSDA [9]	$51.1_{\pm 0.6}$	$73.5_{\pm 0.4}$	$78.3_{\pm 0.2}$	$62.5_{\pm 0.4}$	$70.1_{\pm 0.3}$	$73.3_{\pm 0.4}$	$61.8_{\pm 0.5}$	$50.6_{\pm 0.7}$	$79.4_{\pm 0.3}$	$72.3_{\pm 0.4}$	$55.6_{\pm 0.5}$	$82.8_{\pm 0.6}$
Enhanced RSDA	$54.8_{\pm 0.3}$	$75.9_{\pm 0.3}$	$81.1_{\pm 0.1}$	$66.3_{\pm 0.5}$	$75.1_{\pm 0.2}$	$75.4_{\pm 0.4}$	$67.2_{\pm 0.1}$	$53.4_{\pm 0.3}$	$80.6_{\pm 0.2}$	$73.9_{\pm 0.1}$	$57.8_{\pm 0.2}$	$84.7_{\pm 0.2}$
Enhanced RSDA + Multi-class Discriminator	$54.9_{\pm 0.3}$	$76.8_{\pm 0.5}$	$81.2_{\pm 0.5}$	$67.0_{\pm 0.2}$	$76.1_{\pm 0.4}$	$76.0_{\pm 0.2}$	$67.8_{\pm 0.7}$	$54.1_{\pm 0.3}$	$80.9_{\pm 0.2}$	$73.9_{\pm 0.4}$	$58.1_{\pm 0.2}$	$84.9_{\pm 0.2}$
CoVi [29]	58.5	78.1	80.0	68.1	80.0	77.0	66.4	60.2	82.1	76.6	63.6	86.5
SSRT [40]	**75.1**	**88.9**	**91.0**	**85.1**	**88.2**	**89.9**	**85.0**	**74.2**	**91.2**	**85.7**	**78.5**	**91.7**

- strategies that leverage both source labeled data and target pseudo-labeled data can lead to better adaptation;
- performing DA in lower dimensional feature spaces can improve the adaptation performance;
- and, finally, incorporating class-aware domain confusion into the adversarial framework for DA can also contribute to better performance on the target domain.

Furthermore, we also demonstrated through our case study that DA methods can also be leveraged in real-world applications, such as the PPE detection example that we explored. The results from the case study showed that our proposed enhancements to RSDA could also improve the baseline results in this real-world setting. However, they also exposed some challenges for performing DA in a production setting, such as the considerably lower accuracy achieved over the traditional learning setup, in which labels are available, demonstrating that there is still a necessity to improve the performance and robustness of DA in order for it to be a complete replacement for labeling data in a real-world setting. As previously mentioned, our work provides meaningful directions for future works to further improve their adaptation performance so that, as research on the DA topic progresses, we may be able to avoid the high labeling cost without causing a big impact on the model's accuracy in real-world scenarios.

Chapter 6

Conclusions and Future Works

In this work, we explored the Unsupervised Homogeneous Domain Adaptation problem for image classification, in which a machine learning model that will classify images from a target domain is trained using labeled data from a semantic-related source domain. This allows training a model without having to label a large number of samples, thus skipping that high labeling cost, which constitutes one of the main obstacles in the development of machine learning-powered smart systems.

We provided an overview of commonly used approaches for dealing with domain adaptation and proposed two new methods that enhanced the Robust Spherical Domain Adaptation (RSDA) baseline, introduced by [9], by improving its adaptation performance and robustness: we introduced an enhanced correct labeling probability estimation pipeline that incorporated a dimensionality reduction step and better use of the available data in each domain to further enhance the robustness of the target pseudo-labels. In addition, we also proposed a multi-class discriminator architecture to the domain adaptation adversarial framework that leveraged ground-truth labeled source samples and pseudo-labeled target ones to introduce class-aware domain confusion to the extracted image features, leading to an improved adaptation.

Both methods introduced in this work were able to improve the classification accuracy on target data over the baseline RSDA, indicating that the proposed changes could successfully enhance the robustness of the overall adaptation procedure. Furthermore, the achieved results also demonstrated that exploring better ways to leverage the pseudo-labeled target data, as we did in our proposed approaches, can indeed translate into a more effective transfer of the knowledge from the source domain to the target one.

We also demonstrated through a case study that our proposed domain adaptation methods could also enhance the classification accuracy in the target domain in a real-world setting, although there is still a considerable accuracy gap when compared to the results obtained with the traditional learning procedure, which requires the target data to be completely labeled. Therefore, one should weigh the savings that come from using an adaptation method to skip the high labeling cost against the hit in the model's accuracy caused by the lack of target labeled data.

Domain adaptation is currently a very actively researched topic, and progress is

being made rapidly. Nevertheless, domain adaptation methods still need to be more robust to be effectively employed in real-world applications. Our work makes contributions towards this goal by proposing enhancements that, besides improving our baseline results, also provide meaningful insights for further enhancing the adaptation performance. Specifically, future works can combine our proposed enhanced correct labeling probability estimation pipeline with novel concepts in the machine learning literature, such as visual transformers, which produce powerful feature representations and can ultimately lead to better performance on the target data. Furthermore, future methods can look for alternative ways to introduce class-aware domain confusion in the features produced by the neural network, based on the intuition that aligning the features across the domains while also taking into account the class information can result in better adaptation, as demonstrated by our second proposed method. Exploring different weighting strategies or even different architectures for each discriminator can lead to an even more effective domain confusion, hence better overall adaptation.

References

[1] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 343–351, Red Hook, NY, USA, 2016. Curran Associates Inc.

[2] Leiyu Chen, Shaobo Li, Qiang Bai, Jing Yang, Sanlong Jiang, and Yanming Miao. Review of image classification algorithms based on convolutional neural networks. *Remote Sensing*, 13(22), 2021.

[3] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines: And Other Kernel-Based Learning Methods*. Cambridge University Press, USA, 1999.

[4] 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 Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 647–655, Bejing, China, 22–24 Jun 2014. PMLR.

[5] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1180–1189, Lille, France, 07–09 Jul 2015. PMLR.

[6] Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 597–613, Cham, 2016. Springer International Publishing.

[7] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

[8] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, and Tsuhan Chen. Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377, 2018.

[9] Xiang Gu, Jian Sun, and Zongben Xu. Spherical space domain adaptation with robust pseudo-label loss. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9098–9107, June 2020.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[11] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017.

[12] Jiaxing Huang, Dayan Guan, Aoran Xiao, Shijian Lu, and Ling Shao. Category contrast for unsupervised domain adaptation in visual tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1203–1214, June 2022.

[13] Min Jiang, Wenzhen Huang, Zhongqiang Huang, and Gary G. Yen. Integration of global and local metrics for domain adaptation learning via dimensionality reduction. *IEEE Transactions on Cybernetics*, 47(1):38–51, 2017.

[14] Xiang Jiang, Qicheng Lao, Stan Matwin, and Mohammad Havaei. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4816–4827. PMLR, 13–18 Jul 2020.

[15] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4893–4902, 2019.

[16] Wouter M. Kouw and Marco Loog. An introduction to domain adaptation and transfer learning, 2019.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[18] Vinod Kurmi, Venkatesh Subramanian, and Vinay Namboodiri. Informative discriminator for domain adaptation. *Image and Vision Computing*, 111:104180, 04 2021.

[19] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[20] Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, R. Howard, Wayne Hubbard, and Lawrence Jackel. Handwritten digit recognition with a back-propagation network. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1989.

[21] Ping Li, Zhiwei Ni, Xuhui Zhu, Juan Song, and Wenying Wu. Optimal transport with dimensionality reduction for domain adaptation. *Symmetry*, 12(12), 2020.

[22] R. Li, Q. Jiao, W. Cao, H. S. Wong, and S. Wu. Model adaptation: Unsupervised domain adaptation without source data. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9638–9647, 2020.

[23] 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.

[24] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[25] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 97–105, Lille, France, 07–09 Jul 2015. PMLR.

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

[27] Renato Sergio Lopes Junior and William Robson Schwartz. Analyzing the effects of dimensionality reduction for unsupervised domain adaptation. In *2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 73–80, 2021.

[28] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[29] Jaemin Na, Dongyoon Han, Hyung Jin Chang, and Wonjun Hwang. Contrastive vicinal space for unsupervised domain adaptation. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 92–110, Cham, 2022. Springer Nature Switzerland.

[30] Loris Nanni, Stefano Ghidoni, and Sheryl Brahnam. Handcrafted vs. non-handcrafted features for computer vision classification. *Pattern Recognition*, 71:158–172, 2017.

[31] Hugo N. Oliveira, Edemir Ferreira, and Jefersson A. Dos Santos. Truly generalizable radiograph segmentation with conditional domain adaptation. *IEEE Access*, 8:84037–84062, 2020.

[32] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[33] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa. Visual domain adaptation: A survey of recent advances. *IEEE Signal Processing Magazine*, 32(3):53–69, 2015.

[34] Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[35] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408, 1958.

[36] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, 2011.

[37] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Proceedings of the 11th European Conference on Computer Vision: Part IV*, ECCV'10, page 213–226, Berlin, Heidelberg, 2010. Springer-Verlag.

[38] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[39] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, pages 443–450, Cham, 2016. Springer International Publishing.

[40] Tao Sun, Cheng Lu, Tianshuo Zhang, and Haibin Ling. Safe self-refinement for transformer-based domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7191–7200, June 2022.

[41] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2962–2971, 2017.

[42] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[43] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan. Deep hashing network for unsupervised domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5385–5394, 2017.

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

[45] Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):9, May 2016.

[46] H. Wold. Partial least squares. In *Encyclopedia of Statistical Sciences*. John Wiley & Sons, Ltd, 1985.

[47] Xiang Xu, Xiong Zhou, Ragav Venkatesan, Gurumurthy Swaminathan, and Orchid Majumder. d-sne: Domain adaptation using stochastic neighborhood embedding. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 2497–2506, June 16-20 2019.

[48] Jingyi Zhang, Jiaxing Huang, Zichen Tian, and Shijian Lu. Spectral unsupervised domain adaptation for visual recognition. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9819–9830, 2022.

[49] Xu Zhang, Felix X. Yu, Shih-Fu Chang, and Shengjin Wang. Deep transfer network: Unsupervised domain adaptation. *ArXiv*, abs/1503.00591, 2015.

[50] Yin Zhu, Yuqiang Chen, Zhongqi Lu, Sinno Jialin Pan, Gui-Rong Xue, Yong Yu, and Qiang Yang. Heterogeneous transfer learning for image classification. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, AAAI'11, page 1304–1309. AAAI Press, 2011.