SELF-TRAINING FOR UNSUPERVISED DOMAIN ADAPTATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

İBRAHIM BATUHAN AKKAYA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

AUGUST 2022

Approval of the thesis:

SELF-TRAINING FOR UNSUPERVISED DOMAIN ADAPTATION

submitted by **İBRAHIM BATUHAN AKKAYA** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil KALIPÇILAR
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İlkay ULUSOY
Head of Department, **Electrical and Electronics Engineering** _____

Prof. Dr. Uğur HALICI
Supervisor, **Electrical and Electronics Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Gözde BOZDAĞI AKAR
Electrical and Electronics Engineering, METU _____

Prof. Dr. Uğur HALICI
Electrical and Electronics Engineering, METU _____

Assoc. Prof. Dr. Aydın KAYA
Computer Engineering, Hacettepe University _____

Assist. Prof. Dr. Özgür ERKENT
Computer Engineering, Hacettepe University _____

Assist. Prof. Dr. Ramazan Gökberk CİNBİŞ
Computer Engineering, METU _____

Date: 31.08.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: İbrahim Batuhan Akkaya

Signature :

ABSTRACT

SELF-TRAINING FOR UNSUPERVISED DOMAIN ADAPTATION

Akkaya, İbrahim Batuhan
Ph.D., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. Uğur HALICI

August 2022, 134 pages

Despite the outstanding performance of deep learning techniques, achieving high performance generally demands large amounts of labeled data. Because of the labeling costs, people consider utilizing public datasets or synthetic images with freely generated labels. Unfortunately, deep neural networks are notably sensitive to domain misalignment. The methods to reduce domain misalignment are studied under domain adaptation (DA). Self-training, which selects a subset of the unlabeled data for pseudo-labeling, has been exploited for DA methods lately. These studies usually exploit a confidence threshold to eliminate inaccurate pseudo-labels. Confidence-based approaches rely on the low-density separation hypothesis, which assumes data is independent and identically distributed. However, the low-density separation hypothesis for the target domain for the model trained in the source domain may not hold since the source, and target domains do not share the same distribution. This situation reveals the necessity of a pseudo-labeling metric specific to the target domain.

In this thesis, we propose several self-training-based unsupervised DA methods. We evaluate our methods in different modalities such as visible and thermal spectrum, for different tasks such as classification and semantic segmentation, and in different sce-

narios such as classical and source-free DA. First, we propose a self-training guided adversarial DA method to promote the generalization capabilities of adversarial DA methods in thermal to RGB modalities. Then, as the main contribution of this thesis, we design a metric learning approach defined in the target domain to enable better guidance for self-training. We use our metric for semantic segmentation tasks in classical and source-free DA scenarios. The experimental results show the superiority of the proposed metric and the effectiveness of the self-training.

Keywords: Domain adaptation, metric learning, self-training, adversarial training

ÖZ

DENETİMSİZ ALAN UYARLAMA İÇİN KENDİ-KENDİNE ÖĞRENME

Akkaya, İbrahim Batuhan
Doktora, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Uğur HALICI

Ağustos 2022 , 134 sayfa

Derin öğrenme tekniklerinin olağanüstü performansına rağmen, yüksek performansa ulaşmak genellikle büyük miktarda etiketlenmiş veri gerektirir. Etiketleme maliyetleri nedeniyle, insanlar halka açık veri kümelerini veya maliyetsizce oluşturulmuş etiketlerle sentetik görüntüleri kullanmayı düşünürler. Ne yazık ki, derin sinir ağları, alan kaymasına karşı özellikle hassastır. Alan kaymasını azaltma yöntemleri, alan uyarlaması (AU) altında incelenir. Sözde etiketleme için etiketlenmemiş verilerin bir alt kümesini seçen kendi kendine eğitim yöntemleri, son zamanlarda AU yöntemleri için kullanılmıştır. Bu çalışmalar genellikle yanlış sözde etiketleri ortadan kaldırmak için bir güven eşiğinden yararlanır. Güvene dayalı yaklaşımlar, verilerin bağımsız ve aynı şekilde dağıtıldığını varsayan düşük yoğunluklu ayırma hipotezine dayanır. Ancak, kaynak ve hedef alanları aynı dağılımı paylaşmadığından kaynak alanında eğitilen model için hedef alanda düşük yoğunluklu ayırma hipotezi geçerli olmayabilir. Bu durum, hedef alanına özgü bir sözde etiketleme metriğinin gerekliliğini ortaya koymaktadır.

Bu tezde, birkaç kendi kendine eğitim tabanlı denetimsiz AU yöntemi öneriyoruz.

Yöntemlerimizi görünür ve termal spektrum gibi farklı modalitelerde, sınıflandırma ve semantik segmentasyon gibi farklı görevler için ve klasik ve kaynaksız AU gibi farklı senaryolarda değerlendiriyoruz. İlk olarak, termal ve RGB modalitelerinde çekişmeli AU yöntemlerinin genelleştirme yeteneklerini teşvik etmek için kendi kendine eğitim rehberli bir çekişmeli AU yöntemi öneriyoruz. Ardından, bu tezin ana katkısı olarak, kendi kendine eğitime daha iyi rehberlik sağlamak için hedef alanda tanımlanan bir metrik öğrenme yaklaşımı tasarlıyoruz. Klasik ve kaynaksız AU senaryolarında anlamsal bölütleme görevleri için metriğimizi kullanıyoruz. Deneysel sonuçlar, önerilen metriğin üstünlüğünü ve kendi kendine eğitimin etkinliğini göstermektedir.

Anahtar Kelimeler: Alan uyarlama, metrik öğrenme, kendi-kendine eğitim, çekişmeli öğrenme

To my beloved daughter, Elif Duru AKKAYA

ACKNOWLEDGMENTS

I want to start by thanking my supervisor, Prof. Dr. Uğur Halıcı, for all of her academic support and encouragement, as well as for serving as a terrific mentor and demonstrating how to conduct a respectable research project. I am privileged to have worked with her on various projects, which gave me priceless experience. I am honored to have been her student. This study would not have been possible without her encouragement and guidance.

I also want to express my gratitude to the committee's professor, Prof. Dr. Gözde Bozdağı Akar, and Assist. Prof. Dr. Ramazan Gökberk Cinbiş for their insightful remarks and debates. They closely followed the development of the research, and their insightful comments and conversations have significantly raised the caliber of the thesis.

I am thankful for the support of TUBITAK (The Scientific and Technological Research Council of Turkey) with the 2211-C National Ph.D. Scholarship Program in the Priority Fields in Science and Technology fellowship during my Ph.D. education.

I must express my gratitude to my dear wife, Şerife Burcu Akkaya, for her unending support and understanding. She is the unsung hero of the study as she's been a constant source of motivation and encouragement for me.

My daughter Elif Duru, you have no idea how much you improved me as a person. When I was going through difficult and stressful moments, you made me laugh and gave me emotional support with your pure love.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

[ABN] Adaptive Batch Normalization

[AdaIN] Adaptive Instance Normalization

[ADDA] Adversarial Discriminative Domain Adaptation

[ASPP] Atrous Spatial Pyramid Pooling

[BSP] Batch Spectral Penalization

[CAN] Collaborative and Adversarial Network

[CDAN] Conditional Domain Adaptation

[CNN] Convolutional Neural Networks

[CORAL] Correlation Alignment

[DA] Domain Adaptation

[DANN] Domain Adversarial Neural Network

[DM-ADA] Domain Mixup Adversarial Domain Adaptation

[DML] Deep Metric Learning

[DtA] Drop to Adapt

[FCN] Fully Convolutional Network

[FFT] Fast Fourier Transform

[GAN] Generative Adversarial Networks

[GtA] Generate To Adapt

[IB] Information Bottleneck

[IID] Independent and Identically Distributed

[KL] Kullback-Leibler

[MCD] Maximum Classifier Discrepancy

[MMD] Maximum Mean Divergence

[MOCM] Metric-based Online ClassMix

[MS] Multi Similarity

[MT] Mean Teacher

[NCA] Neighboring Component Analysis

[ProxyNCA] Proxy Neighbor Component Analysis

[SF-UDASS] Source-Free Unsupervised Domain Adaptation for Semantic Segmentation

[SGADA] Self-training Guided Adversarial Domain Adaptation

[SGD] Stochastic Gradient Descent

[SIM] Stuff and Instance Matching

[SPLS] Similarity-based Pseudo-Label Selection

[SSL] Self-Supervised Learning

[ST] Self-Training

[STVM] Self-Training via Metric learning

[TAT] Transferable Adversarial Training

[UDA-SS] Unsupervised Domain Adaptation for Semantic Segmentation

[UDA] Unsupervised Domain Adaptation

CHAPTER 1

INTRODUCTION

1.1 Motivation of the Thesis

Deep learning techniques, which have gained popularity both in industry and academy, have been widely used especially in the domains of computer vision and pattern recognition in recent years [191]. And significant successes have been achieved in these domains. As a specialized type of deep neural networks, convolutional neural networks stand out with their hierarchical structure in terms of semantic feature extraction from images. These features play a key role in achieving high performance in almost all areas of computer vision, such as classification [191], semantic segmentation [17], face detection [184] and recognition [160], human pose estimation [144], and object tracking [1].

Many computer vision tasks require the output to be pixel-by-pixel dense labeling for a given image. Semantic segmentation is an excellent example of dense image labeling. The purpose of semantic segmentation is to assign a label to each pixel in an image based on the object classes to which it belongs to. There are numerous uses for semantic segmentation. It has made its way into practically all image and video-related tasks. Image manipulation, 3D modeling, facial segmentation, the healthcare industry and precision agriculture are some examples to use of semantic segmentation [151].

The ability to model the appearance of a variety of objects in the scene, such as buildings, trees, roads, billboards and pedestrians, is required for scene understanding applications. The model must learn and comprehend the spatial relationships between various items. Semantic segmentation makes it possible to distinguish different ob-

jects in the scene [38]. Segmentation masks categorizing pedestrians crossing the road, for example, will trigger the automobile to stop, whereas segmentation masks classifying roads and lane markers will lead the car to continue on a specific path. Aerial image processing is comparable to scene understanding, except it entails semantic segmentation of the terrain from above [75]. This technology is quite valuable when drones can spread out to examine different locations to locate people and animals in need of rescue in times of disaster, such as a flood. Medical image diagnosis has also benefited from semantic segmentation [58]. Radiologists nowadays find it quite valuable to classify anomalies on CT scans. CT scans and other medical images are highly complicated, making it challenging to spot irregularities. Semantic segmentation can be used as a diagnostic tool to assess such images, allowing doctors and radiologists to make critical treatment decisions.

Despite deep learning's outstanding performance in computer vision, achieving high performance demands large amounts of data. Obtaining labels are typically costly. It is even harder for dense prediction tasks. Manually annotating a Cityscapes [28] image, for example, takes roughly 90 minutes. The big data era provides us data in many areas and applications. For example, the ImageNet dataset contains a total of 14,197,122 tagged images in 1,000 different classes, including classes such as vehicles, goods, animal species. The GTA5 dataset [120] includes 24966 synthetic images that have been annotated at the pixel level for 19 semantic classes. Besides, with the development of simulation technologies, an unlimited number of labeled synthetic images that are similar to the desired application can be created. As a result, people consider utilizing many photo-realistic synthetic images with freely generated labels. On the other hand, deep neural networks are notably sensitive to domain misalignment. Any subtle unrealism in generated visuals will result in poor generalization to real-world data. This phenomena is called the domain shift which may occur due to some factors such as lighting, image quality, exposure change, and seasonal differences. It is considered that adapting the knowledge obtained from the source domain to the target domain will be beneficial in terms of increasing the performance in real-world applications. With this perspective, it is feasible to use domain adaptation approaches.

Self-training is one of the first semi-supervised learning approaches, but it has gained

favor recently [175]. The self-training selects a subset of the unlabeled data for pseudo-labeling. The pseudo-labels are added to the labeled dataset and a model is retrained with the extended labeled dataset. This training cycle is repeated a couple of iterations until the model converges. As a result, when self-training is implemented, the challenge of deciding the subset of examples to pseudo-label arises. The low-density separation hypothesis [14] assumes that the classifier taught at each step makes most of its mistakes on observations near the decision boundary. Semi-supervised learning research recommends that pseudo-labels be assigned only to unlabeled data for which the present classifier has high confidence [153]. In an unsupervised domain adaptation scenario, the source domain is labeled, and the target domain is unlabeled, which shows similarity with the semi-supervised learning settings. Recently, self-training has been adopted for UDA and has shown successful results [195, 103, 196]. Therefore, in this thesis, the importance of domain adaptation and the success of self-training approaches motivated us to focus on self-training approaches in unsupervised domain adaptation.

1.2 Contributions of the Thesis

It is important to apply deep models to real-world problems. However, the models trained with data from visible spectrum suffer from a performance bottleneck under illumination changes. Thermal IR cameras are more robust against such changes, and thus can be very useful for the real-world problems. In order to investigate efficacy of combining feature-rich visible spectrum and thermal image modalities, we propose an unsupervised domain adaptation method which does not require RGB-to-thermal image pairs. We employ large-scale RGB dataset MS-COCO as source domain and thermal dataset FLIR ADAS as target domain to demonstrate results of our method. Although adversarial domain adaptation methods aim to align the distributions of source and target domains, simply aligning the distributions cannot guarantee generalization to the target domain. To this end, we propose a self-training guided adversarial domain adaptation method to promote generalization capabilities of adversarial domain adaptation methods in chapter 3. To perform self-training, pseudo labels are assigned to the samples on the target thermal domain to learn more general-

ized representations for the target domain. Extensive experimental analyses show that self-training guides adversarial-training for better alignment between domains. Our proposed method achieves better results than the state-of-the-art adversarial domain adaptation methods.

Deep learning approaches assume the training and test data are independent and identically distributed (IID). IID is a valid argument for a semi-supervised learning scenario because both labeled and unlabeled data are sampled from the same domain. Therefore, the trained model with the labeled data represents the unlabeled data, making the low-density separation hypothesis practical [14]. However, there is no IID assumption on the source and the target domain in an unsupervised domain adaptation scenario. Even though some parts of the distribution of the source and target domains align, it is assumed that the marginal distribution is different, which is called the domain gap. Therefore, the low-density separation hypothesis for the target domain for the model trained in the source domain may not hold. This situation reveals the necessity of a pseudo-labeling metric specific to the target domain.

One approach to learning a metric in the target domain is to introduce a new head to the neural network model and train it with target domain samples. As the name suggests, in unsupervised domain adaptation, there is no label in the target domain. To be able to train the new head, self-predictions can be used. However, only a few highly confident predictions should be used since the model output contains false predictions. SoftMax output is a commonly used confidence metric. However, Horiguchi et. al. [64] show that if the size of the training dataset is small, deep metric learning approaches perform better than SoftMax confidence.

Deep Metric Learning is a collection of approaches for measuring the similarity of data samples. The goal is to learn an embedding space where similar data pairs stay close together and different sample pairs stay away. Even though some works utilized deep metric learning loss functions to align source and target domain features, utilization of deep metric learning for self-training is an open question. In chapter 4, we analyze the deep metric learning techniques to learn a metric space in the target domain that can be used as a measure for pseudo-labeling.

We propose a pseudo-label selection method to improve self-training performance in

unsupervised domain adaptation for semantic segmentation. To enhance the quality and diversity of the pseudo-labels, we develop a novel metric learning-based approach. Specifically, we train a metric learning network in the target domain that learns the semantic similarity metric between pixels such that it learns a mapping to form a dense cluster in feature space for each class, decreasing the intra-domain gap. Finally, we filter the segmentation network's prediction based on the similarity to the class-proxy, which represents the class distribution in the target domain's metric feature space. Experiments on GTAV-to-CityScapes and SYNTHIA-to-CityScapes have shown the effectiveness of our method against existing state-of-the-art approaches.

Traditional UDA methods assumes that cross-domain data is accessible during model training so that the domain discrepancy may be successfully assessed and eliminated. Despite their success, it is easy to see how these strategies rely on the coexistence of source and target data. Data is distributed on various devices nowadays, and most of them contain private information, such as those on personal devices or from security cameras. Traditional UDA approaches require access to the source data during the learning process in order to adapt, which is inefficient for data transmission and may breach data privacy policies. Due to data privacy and limited device memory constraints, many application cases cannot always match the basic assumption of UDA. The mismatch between practical demand and the UDA setting drives a new research area known as Source-free Domain Adaptation, in which only well-trained source models are provided rather than well-annotated source data to enable adaptation to the target domain. A few research studies have recently begun investigating this novel scenario on cross-domain classification tasks, given that the source classifier has sufficient information. As a common framework, they try to modify features for target domain samples directly to adapt to the source classifier. However, when the source domain data is unbalanced or insufficient, the frozen classifier becomes vulnerable due to the source classifier's weaker generalization. These methods have difficulty adapting to a plethora of target features with much variance within the limited source classification boundary. In a source-dissimilar set, this results in poor classification performance.

Unsupervised source-free domain adaptation methods aim to train a model to be used in the target domain utilizing the pretrained source-domain model and unla-

beled target-domain data, where the source data may not be accessible due to intellectual property or privacy issues. These methods frequently utilize self-training with pseudo-labeling thresholded by prediction confidence. In a source-free scenario, only supervision comes from target data, and thresholding limits the contribution of the self-training. In chapter 5, we utilize self-training with a mean-teacher approach for source-free domain adaptation. The student network is trained with all predictions of the teacher network. Instead of thresholding the predictions, the gradients calculated from the pseudo-labels are weighted based on the reliability of the teacher's predictions. We propose a novel method that uses proxy-based metric learning to estimate reliability. We train a metric network on the encoder features of the teacher network. Since the teacher is updated with the moving average, the encoder feature space is slowly changing. Therefore, the metric network can be updated in training time, which enables end-to-end training. We also propose a metric-based online ClassMix method to augment the input of the student network where the patches to be mixed are decided based on the metric reliability. We evaluated our method in synthetic-to-real and cross-city scenarios. The benchmarks show that our method significantly outperforms the existing state-of-the-art methods.

1.3 Organization of the Thesis

This thesis is divided into six main chapters. The first chapter, Introduction, defines the motivation for self-training for UDA and source-free UDA settings. The contributions of the thesis are also presented in the introduction. In the second chapter, we explain the concepts and approaches related to the methods proposed in this thesis for better comprehension. In order to show the effectiveness of the self-training approaches in an UDA setting, we first focus on the classification task. Considering the lack of research in visible-to-thermal domain adaptation, we choose visible and thermal domains for the adaptation scenario. We use self-training as a rectifier to alleviate the negative transfer of the adversarial alignment. We propose the self-training guided adversarial domain adaptation (SGADA) method for domain adaptation from the visible spectrum to the thermal spectrum. The third chapter describes our proposed SGADA method. Domain adaptation is valuable, especially in dense

prediction tasks. Chapters four and five focus on employing self-training for domain adaptation in semantic segmentation tasks as a dense prediction task. Pseudo-label selection is crucial for a good performance in self-training. We observe that confidence-based pseudo-label selection methods have some issues. As a solution, we propose a metric learning based selection method. In the fourth chapter, a novel similarity-based pseudo-label selection method (SPLS) is described, where we propose a target domain-specific metric using a metric learning approach. The common self-training approaches follow an iterative training strategy that requires training the same model multiple times. It is desirable to have an end-to-end training strategy since the training of a deep model is often time-consuming. Therefore, we focus on improving our self-training method by proposing an end-to-end self-training based domain adaptation method for a significant source-free scenario that considers the source dataset is unavailable due to privacy or intellectual property issues. The fifth chapter demonstrates our self-training via metric learning for source-free domain adaptation of the semantic segmentation (STVM) method. We propose a mean-teacher based method for UDA in a source-free scenario. We utilize a model trained using the target domain images using a metric learning approach to predict the reliability of the predictions of the teacher model. The pseudo-labels are weighted based on reliability. Finally, conclusions and future work are given in the sixth chapter.

CHAPTER 2

LITERATURE SURVEY

This chapter presents the current literature on unsupervised domain adaptation, unsupervised domain adaptation for semantic segmentation, and metric learning. In the first section, we explain the unsupervised domain adaptation (UDA) problem and present the notations and definitions. Then we explain the taxonomy and the state-of-the-art methods. In the second section, we will focus on the semantic segmentation task for UDA. First, we will explain the semantic segmentation techniques used in evaluating the UDA methods and related works referenced by these methods. Then, we will review the state-of-the-art methods and present some insights that we gained from the modern approaches. In this chapter, we will describe a particular case of the UDA for semantic segmentation called source-free UDA for semantic segmentation. In the final section, we will explain the metric learning approaches we utilize in the methods we propose in this thesis.

2.1 Unsupervised Domain Adaptation

Unsupervised domain adaptation is a special transfer learning technique that uses labeled data from one or more relevant source domains to perform tasks in the unlabeled target domain. In other words, it is a learning technique that deals with the lack of large amounts of labeled data.

In the literature, transfer learning methods are examined under three categories namely inductive, transductive and unsupervised transfer learning. In inductive transfer learning, while the source and target areas are the same, the tasks are different. In transductive transfer learning, the source and target areas are different but the tasks are the

same. In unsupervised transfer learning, both domains and tasks are different. Given this classification, domain adaptation methods are included under the title of transductive transfer learning. Domain adaptation is a special transfer learning technique that uses labeled data from one or more relevant source domains to perform the same task in a target domain. It aims to transfer knowledge learned from source domain to target domain with minimal performance loss to overcome domain shift. In other words, as emphasized in the examples above, it is a learning technique that deals with the lack of large amounts of labeled data. If no labaled data is available in the target domain, it is called Unsupervised Domain Adaptation (UDA).

2.1.1 Notations and Definitions

Domain $\mathcal{D}$ is composed of feature space $\mathcal{X}$ ($X = x_1, ..., x_n \in \mathcal{X}$) and a marginal probability distribution $P(X)$. For a given domain $\mathcal{D} = \mathcal{X}, P(X)$, Task $\mathcal{T}$ is composed of a feature space $\mathcal{Y}$ and predictive function $f(\cdot)$. The predictive function can be represented as conditional probability distribution $P(Y|X)$.

In the light of these definitions, suppose we have two domains :

- Domain with sufficient labeled data: Source Domain $\mathcal{D}_s = \mathcal{X}_s, P(X)_s$
- Domain without labeled data: Target Domain $\mathcal{D}_t = \mathcal{X}_t, P(X)_t$

Domain adaptation is a transductive transfer learning approach. The tasks are the same ($\mathcal{T}_s = \mathcal{T}_t = \mathcal{T}$) but the domains are different ($\mathcal{D}_s \neq \mathcal{D}_t$) in domain adaptation. The aim of the unsupervised domain adaptation is to train a model that achieves high performance on task $\mathcal{T}$ in the target domain. During training, both source domain $\mathcal{D}_s$ and target domain $\mathcal{D}_t$ are utilized where we only have labels in the source domain.

2.1.2 Taxonomy of Domain Adaptation

In recent years, various shallow domain adaptation methods have been proposed to solve a domain shift between source and target domains. For shallow domain adaptation, algorithms can be basically divided into two classes: sample-based domain

adaptation [11, 27] and feature-based domain adaptation [45, 109, 40] methods. The first class reduces the inconsistency by re-weighing samples in the source domain based on the similarity to the target domain. Then, weighted source domain samples are used for adaptation. For the second class, a common domain is trained from a shared distribution of two data clusters. In this way, the model obtained becomes usable for both domains.

Recently, Deep learning approaches have achieved very successful results in many computed vision applications, which is also the case for domain adaptation studies. Although some studies have shown that deep networks can learn more transferable represents, Donahue et al. [31] have shown that the domain shift problem still affects performance.

Methods in the domain adaptation literature are categorized as one-step and multi-step domain adaptation. It is called a one-step if knowledge is transferred directly from the source domain to the target domain, and multi-step domain adaptation if knowledge transfer is performed by using one or more intermediate domains. If recent deep learning-based methods are analyzed, it is seen that the researchers mostly prefer the one-step domain adaptation approach. The reason is that deep neural networks usually have enough capacity to transfer knowledge across domains without needing intermediate domains. Based on the input distribution of the source and target domain, the methods are categorized into two categories. If the input space of the source and the target domains are the same, the methods are called homogeneous DA methods. If they are different, the methods are called heterogeneous DA methods. In computer vision applications, the input images are resized to a specific size as a common approach. Therefore, the methods are usually studied under homogeneous DA methods. The taxonomoy of the DA methods are shown in Figure 2.1.

One-step homogeneous domain adaptation methods are examined under three categories. Brief explanations of these approaches are given below with their subcategories.

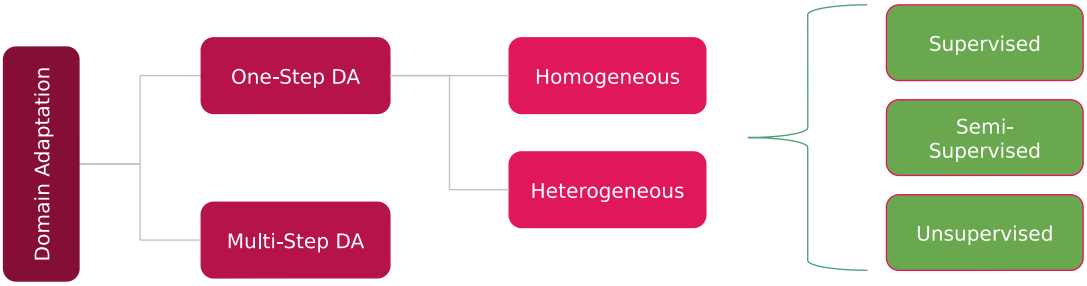


Figure 2.1: Taxonomy of Domain Adaptation

2.1.2.1 Discrepancy Based Methods

Discrepancy-based deep DA approach aim to reduce the domain shift by fine-tuning the deep neural network with target data. It uses source data to train a base network, then reuses the initial layers to build a target network. The target network's remaining layers are randomly initialized and trained with a loss based on disparity. Depending on the size of the target dataset and its resemblance to the source dataset, the initial layers of the target network can be fine-tuned or frozen during training. These methods have four sub-categories:

Class Critera The most fundamental training loss in deep DA is the class criterion. The target model use the class label information as a guide to train the network after pre-training it with source data. Therefore, it's assumed that just a few labeled data from the target dataset are accessible.

If there is labeled data in the target domain, soft label and metric learning techniques are very effective in the target domain [148, 65, 59, 104]. If labeled data is not available, different techniques such as pseudo-labels [99, 172] and attributes [39, 148] can be used in place of class labeled data.

Statistical Critera Although some discrepancy-based algorithms look for pseudo labels, attribute labels, or other labels to replace labeled target data, many research is focusing on learning domain-invariant representations by minimizing the domain distribution disparity in unsupervised DA.

The statistical criterion aims to reduce the domain shift by aligning the statistical distribution between the source and target domain features. The most common methods for comparing and reducing distribution shift are maximum mean divergence (MMD) [100, 172, 97, 99, 150], correlation alignment (CORAL) [136, 113], Kullback-Leibler (KL) divergence, and $\mathcal{H}$ divergence.

Architectural Critera Other ways to reduce the distribution discrepancy is to optimize the network's architecture. The Architectural Criterion aims to learn more transferable features by modifying the architectures of deep networks. These techniques include adaptive batch normalization (ABN) [87, 70, 86], weak-related weight [124], domain guided dropout [166].

Geometric Criteria By integrating intermediary subspaces from the source to the target domains, the geometric criteria mitigates domain shift. One can build intermediate subspaces by selecting a fixed or infinite number of subspaces along the path from the source to the target domain to aid in the discovery of domain correlations. The distribution is then aligned by projecting both source and target data to the resulting intermediate subspaces [26].

2.1.2.2 Adversarial Based Methods

The use of generative adversarial networks, first proposed in 2014 [47], in different domains of deep learning is increasing rapidly. Generative adversarial networks consist of two networks, a generator and a discriminator. These two networks compete with each other on a "minimax game". The generator network tries to produce data similar to the real data to deceive the discriminator network. The discriminator network tries to distinguish between generated and real data. As the training progresses, the generative network becomes more successful in producing realistic data, and the discriminator network becomes more successful in separating this data from the real ones. The success of generative adversarial networks is also reflected in domain adaptation methods. The most advanced domain adaptation methods in classification have generally used these approaches. These studies are divided into two groups called

generative and non-generative methods.

Generative methods Using generative adversarial networks, synthetic data is generated in the target domain. Creating simulated instances that are comparable to target instances is a common example [93, 9, 72]. The source image, as well as the noise vector, are provided to the generative network as input to preserve the label information in the source domain to generate images for the target domain to specific class.

Non-Generative methods These models consist of two networks, one is the discriminator network that classifies whether the data is sampled from the source or the target domain, and the other is a feature extractor network that will generate features from the image given as an input. The feature extractor network is trained to extract indistinguishable features for the two domains, and the discriminator network is trained to distinguish these features. It is aimed to reduce the domain shift with the help of the adversarial training by generating more similar features [149, 148, 36, 37].

2.1.2.3 Reconstruction Based Methods

The reconstruction of target or source data is an auxiliary task that focuses on producing a shared representation between the two domains while maintaining each domain's individual properties. Reconstruction of source and/or target instances helps to improve domain adaptation performance. Reconstruction networks ensure both the authenticity of within-domain representations and the indistinguishability of cross-domain representations. There are two types of reconstruction based methods namely encoder-decoder based methods and adversarial methods.

Encoder-Decoder based methods Typically encoder-decoder based methods uses autoencoder architecture for reconstruction. The autoencoder [60] converts an input into a hidden representation, which it then decodes into a reconstructed version. Commonly, in the source and target domains, DA techniques based on encoder-decoder reconstruction learn the domain-invariant representation using a shared encoder and keep the domain-special representation with a reconstruction loss [10, 42, 41, 194].

Adversarial methods Adversarial based methods share the same goal with encoder-decoder based methods. Instead of autoencoder architecture, adversarial networks are used. Image-to-image translation methods that uses cyclic-consistency loss is one of the popular approach. The reconstruction error is measured as the difference between the original image and the image resulting from the cyclic transformation using methods such as Dual GAN [177], Cycle GAN [192] and Disco GAN [78]. Compared to encoder-decoder based methods, adversarial methods perform better.

2.1.3 State-of-the-art Methods

In this section, the up-to-date domain adaptation methods and the studies that form the basis for these methods are explained.

The aim of domain adversarial neural network (DANN) [36, 37] is to integrate domain adaptation into the representation learning process. In this way, predictions can be made on features that are both distinctive and domain-independent. The artificial neural network used in DANN [36, 37] consists of three components. These are feature extractor, class label predictor and domain classifier networks. The feature extractor network extracts features for subsequent neural networks. Label predictor and domain classifier network use these features to make predictions in line with their own tasks. There are two separate targets in the training. The first of these is to determine the correct label of the image from the source domain. The second is to determine whether the incoming image is from the target domain or the source domain. The gradient descent method is used during training. The only non-standard component of the proposed architecture is a gradient reversal layer placed before domain classifier that leaves the input unchanged during forward propagation and inverts the gradient value by multiplying it by a negative scalar during backpropagation. Thanks to this layer, the feature extractor learns to create hard-to-classify features for the network domain classifier. In this way, domain-independent features are obtained and the performance is increased in the target domain. This study is one of the pioneering studies of the adversarial domain adaptation class.

The ADDA method [149] consists of three training stages performed sequentially. First of all, using the labeled data in the source domain, the feature extractor and

classifier network are trained in the source domain. At the end of this stage, the classifier network can perform a successful classification with the features created by the source domain's feature extractor network. After training in the source domain, the second stage called adversarial adaptation is performed. At this stage, the network parameters of the feature extractor trained on the source domain are fixed and a feature extractor specific to the target domain is trained in an unsupervised manner. During the training, the discriminator network described in the Generative adversarial network architecture is used. Thanks to the adversarial training, the distributions of the features coming from the target domain and source domain are aligned. In this way, the classifier trained on the source domain can also be used on the source domain. The final stage of the ADDA method is the testing stage. The parameters of the feature extractor trained in the source domain and the classifier trained in the target domain are fixed. By using these two networks together, classification is made on the target domain.

Although ADDA [149] and DANN [36] methods are applications that successfully reduce domain contrast using an adversarial metric and domain classifier, the vanishing gradient problem is observed in these studies due to the characteristics of the metrics used. The main contribution of [133] is to eliminate the problems encountered in previous studies by using the advantages of the gradient properties of the Wasserstein distance criterion [3] for domain adaptation. To solve the vanishing gradient problem, it is proposed to replace the domain contrast criterion with the Wasserstein distance criterion [3]. In this way, it has been argued that more stable gradient values will be obtained even if the two distributions are far from each other. A network called 'Domain Critic' is trained to estimate the Wasserstein distance between source and target feature representations. Next, the feature extractor network is optimized adversarially to minimize this estimated Wasserstein distance.

Saito et al. [127] argue that general domain discriminators do not take into account task-specific cross-class decision boundaries. They argue that this leads to the uncertainty of features near the class boundaries of feature extractor structures. In order to solve the aforementioned problem, a method named Maximum Classifier Discrepancy (MCD) is proposed. In MCD, there are two task-specific classifiers that follow a common feature extractor. The discriminator is a combination of these two classi-

fiers. The feature extractor is trained to minimize the discrepancy metric, while the classifiers are trained to maximize the discrepancy of the predictions by two classifier corresponding to the target sample.

Long et. al. [98] states that adversarial domain adaptation methods face two bottlenecks. First, when data distributions contain complex and multimodal structures, adversarial adaptation methods may have difficulty capturing these structures. Second, when the discriminator information is ambiguous, it is risky to modulate domain discriminators over this information. For the solution of these two bottlenecks, Long et al. [98] proposed a conditional domain adaptation framework (CDAN). The discriminator information passed inside the classifier outputs is used by CDAN to support adversarial alignment.

CAN [186] proposes two mechanism for domain alignment each of which complements and improves the other. It takes the Domain Adversarial Neural Network (DANN) [36, 37] method as baseline. CAN states that DANN learns domain uninformative features. In addition to these features that DANN learned in its last layer and are not informative about the domain, in this study, domain informative features, which are representations in initial layers that provide information such as corners and edges, are used to achieve a better visual recognition performance with domain collaborative learning. In order to further increase the performance, Zhang et. al. proposes the incremental version of the existing Collaborative and Adversarial Network (CAN) structure, Incremental CAN (iCAN). In each iteration of iCAN, pseudo-labels are assigned to the target samples. The CAN model is retrained with the expanded training set with these new samples.

Drop to Adapt (DtA) [85] argue that the features obtained via the domain adversarial training will not only be domain-independent but also non-discriminative for the class labels since the domain discriminator aligns without considering the class labels. For this reason, it is difficult to obtain optimal classification performance. In DtA, the clustering assumption was taken as the basis. According to this assumption, decision boundaries should be located in low-density regions of the feature space. Therefore, the target model is trained by pushing the decision boundaries away from the features of the target domain. Drop to Adapt (DTA) uses the adversarial dropout [110] method

to apply the clustering assumption to the target domain. Adversarial dropout applied in DTA is proposed element-wise for fully connected layers and channel-wise for convolutional layers. In element-wise adversarial dropout neurons are removed independently of spatial location, while in channel-wise adversarial dilution, all neurons in a channel of feature map are removed.

Liu et al. [91] discussed that adversarial domain adaptation methods have various problems. The most important of these problems, according to the domain adaptation theory [6], is that in cases where the adaptability metric defined between source and target domains is weak, it cannot guarantee learning a classifier with low target error by minimizing the source error. Converting feature representations to be domain independent alter the original feature distributions that weakens the adaptability metric. Therefore, the adversarial feature alignmend ias defined as risky and Transferable Adversarial Training (TAT) is proposed. TAT generates examples that can be transferred to the category classifier and domain discriminator without changing the feature representations.

Batch Spectral Penalization (BSP) [19] analyze how to learn transferable and discriminable feature representations for deep domain adaptation. Transferability and discriminability are criteria that characterize the goodness of feature representations used in the domain adaptation problem. Transferability is the ability of features to combine the differences between domains. Discriminability is the capability of feature representations being separated by a supervised classifier trained on them. In addition to transferability, which was studied in previous domain adaptation studies using adversarial learning, discriminability, which was not studied in previous studies, is examined in detail in this study. BSP states that adversarial domain adaptation methods tend to improve transferability at the expense of worsening discriminability. BSP proposes to penalize the largest singular values in order to increase feature discriminability to solve this dilemma. The reason for this penalty is that large singular eigenvectors dominate the transferability of feature representations. By adding the BSP method to the existing adversarial domain adaptation methods, it has been shown by experimental studies that these methods help them learn both transferable and discriminable representations.

In adversarial domain adaptation studies, although the feature extractor is well trained to generate domain-independent features of the source and target samples, the classifier trained on the source samples cannot perform well on the target samples. Zhang et al. [187] have tried to solve this problem by assigning pseudo-labels to target samples, proposing the method named SymNet. SymNet is based on a symmetrical design of the source and target task classifiers. In addition to this symmetric structure, there is an extra classifier that shares neurons with the classifiers. Domain discrimination and domain confusion loses are implemented by this additional classifier. In order to train the SymNet structure, a two-level domain confusion loss-based adversarial learning method is applied.

Xu et al. [171] defined two common restrictions in the adversearial domain adaptation domain and suggested the domain mixup (DM-ADA) method to solve them. The first of these limitations is that the separate sampling of source and target domains is insufficient to ensure domain independence in the entire latent space. The second is that the domain discriminator used in these methods distinguishes between real and fake instances using only hard labels. In DM-ADA, a variation of VAE-GAN [83] was applied to the domain adaptation problem. Mixup images obtained by pixel-based merging of source and target domain images were used as inputs. The encoder converts the source and target inputs into two separate hidden spaces. The feature embeddings of these two domains are combined to create the blended features. After this stage, the network is divided into two branches. In the first branch, source domain embeddings is used for object classification. In the other branch, the encoded embeddings are decoded by the decoder. Thanks to the adversarial learning between the decoder and the discriminator, domain independence is achieved at the category level.

In adversarial domain adaptation methods, there may be a mod crash problem caused by the seperate design of the task and domain classifiers. This problem leads to limitations in aligning the distributions of features and categories on domains. The DADA method [139] aims to align distributions jointly. Thus, in addition to previous work, DADA provides an interaction between category and domain predictions.

In the M-ADDA [82] method, a domain adaptation technique based on metric learn-

ing is proposed. This method consists of two stages. First, a metric learning is performed on the source domain data using the triplet loss function. As a result of this training, a metric learning results in which the features of the images with the same label are close to each other and those with different labels are far from each other. In other words, a space in which the same classes are clustered is learned. Then, with the adversarial method mentioned in the ADDA method, the features of the target and source domain are aligned. During this training, in addition to the adversarial loss function, the magnet loss function, which was originally proposed in this study, draws the features of the target domain to the closest cluster of the source domain. The magnet loss function pulls the features towards the center of the nearest cluster. While determining the center of the cluster, the Euclidean average of the features belonging to a certain class is considered.

The PixelDA [9] method adapts the source domain images to appear as if they were taken from the target domain, using a Generative adversarial network-based model. While the model is trained to make the images in the source domain appear to be sampled from the target domain, it is aimed to preserve the original content at the same time. This approach has many advantages. The primary advantage is that the task-specific network does not need to be retrained. The task-specific network can be used as is, as images are aligned at the pixel level. The second advantage is to increase the stability of the training. Neural networks are sensitive to initial parameters. In classical adversarial training, problems such as stability and mode collapse can occur. However, in the PixelDA approach, adversarial training is guided by the knowledge of the task-specific network and a more stable training is provided. A third advantage is that unlike classical style transfer approaches, the style of the entire domain is transferred, not the style of a single image. All these advantages contribute to increasing the success of the task in the target domain.

In DupGAN [66], a dual generative adversarial network structure is proposed for domain independent feature learning and inter-domain conversion learning. This method is called DupGAN for short. The DupGAN method consists of four parts, an encoder, generator, and two classifiers. The encoder network learns representations of images from both domains. The generator converts the image representations back into images using the code containing the network domain information. In other

words, it learns the conversion between domains. In this method, discriminator networks are not used only for real and fake discrimination. They are also conditioned to predict class information. An extra class for the discriminator network has been added to detect fake images. In this way, the features created by the feature extractor network also contain domain independent category information, which is important to avoid negative transfer.

In the CyCADA [61] method, domain adaptation is performed at two levels, namely pixel and feature levels. Adversarial domain adaptation techniques show higher performance than classical methods. However, there are two factors that limits the performance of adversarial methods. First of all, aligning marginal distributions does not mean that they will be semantically consistent. For example, a feature that corresponds to a car in the target feature space can be aligned to the bicycle class in the source domain feature space. Second, alignments in deep features may not be able to model appearance differences at the pixel level. For this reason, aligning at two levels at the same time will increase the domain adaptation performance. Image to image translation methods such as CycleGAN [192] are able to create realistic images while preserving regional content in natural scenes. However, while developing these methods, the network related to the task was not taken into account. For this reason, such methods may not preserve semantic information. For example, a model trained to translate numbers from Google Street View to handwritten numbers may learn to translate a printed 8 to a handwritten 1. The proposed method aims to eliminate the mentioned problems by making domain adaptation at both pixel level and feature level.

Some of the methods used in previous domain adaptation studies are to adapt the source data to the target data either by image representation transformations or by generating new source images. Some of the researchers using these methods have used Generative adversarial networks to do this. Another method, which is similar but inverse of these methods, transforms the target domain, which has little or no labeled data, into the source domain. Russo et al. [125] argued that transforming in both directions (from source to target and from target to source) within the same architecture will produce more general and more consistent results. In the light of this information, bidirectional image translation mapping and loss of class consistency

are proposed in this study. In the SBADA-GAN structure proposed in this study, two generative adversarial losses are used to enable generating target images from source samples and source images from target samples. In addition, two classification losses are used, one using the original source images and the other using the target view source images together. In this structure, the source classifier is also used to label target images with source view. It has been shown that such a pseudo-label regulates the classifier. Also, a new semantic constraint on source images, class consistency loss, is proposed. In the test phase, two trained classifiers were used on the original target images and their source-like versions, respectively. Two estimates from these were linearly integrated to form the final labels.

Generate To Adapt (GtA) [128] aims to learn an embedding robust to shifts between source and target distributions. An adversarial image generation approach is proposed to directly learn shared feature embedding using labeled data from the source and unlabeled data from the target. Although there are already studies using adversarial structures to solve the problem of domain adaptation, the fundamental contribution of this paper is an adversarial image generation strategy for unsupervised domain adaptation that effectively learns a joint feature space in which the distance between source and target distributions is reduced.

2.2 Unsupervised Domain Adaptation for Semantic Segmentation (UDA-SS)

In this section, unsupervised domain adaptation for semantic segmentation (UDA-SS) problem will be explained, and recent methods that try to solve this problem are summarized. Several methods are proposed to address the domain shift in semantic segmentation recently. Domain adaptation for semantic segmentation aims to train a segmentation network that can make a reliable and accurate prediction of the dense label for each image in the target domain. In the source domain, the ground-truth labels are available along with images. In the target domain, only unlabeled images exist.

2.2.1 Semantic Segmentation Methods

Image segmentation is the practice of classifying each pixel in an image into a particular category and can thus be thought of as a pixel-by-pixel classification problem. Segmentation techniques are divided into two categories: Semantic segmentation is the first type. The method of categorizing each pixel as belonging to a specific label is known as semantic segmentation. It is consistent across different occurrences of the same object. For example, semantic segmentation assigns the same label to each vehicle's pixels if a picture contains two vehicles. Instance segmentation is the second type. Instance segmentation differs from semantic segmentation in that each instance of a particular object in the image is assigned a unique label.

Classic machine learning approaches such as Genetic Algorithm and K-means Clustering [182] were utilized to handle the problem of image segmentation before the advent of deep learning. However, like with other image-related research problems, deep learning has outperformed previous algorithms and has become the standard for semantic segmentation.

This section provides an overview of some of the important deep learning approaches for semantic segmentation to better understand the recent domain adaptation of semantic segmentation methods.

A CNN's basic design comprises a few convolutional and pooling layers, followed by a few fully linked layers. According to Fully Convolutional Network (FCN) [96], the final fully connected layer can be conceived as doing a 1x1 convolution that spans the entire region. As a result, the last dense layers can be substituted by a convolution layer, and the outcome will be the same. Furthermore, the benefit of doing so is that the input size does not have to be fixed.

When working with dense layers, the input size is fixed. Thus if a different size input is required, it must be scaled. This constraint is removed when a dense layer is replaced with convolution. Also, when a larger image is used as an input, the result is a feature map rather than a class output, as is the case with a smaller image. The final feature map shows the required class's heatmap, with the object's position highlighted in the feature map. Because the feature map's output is a heatmap of the

required object, it is valuable information for segmentation.

One may want to use an interpolation technique to up-sample the feature map acquired at the output layer because it is downsampled due to the set of convolutions performed. FCN suggested utilizing deconvolution and learned upsampling to learn non-linear upsampling. The network's encoder is responsible for downsampling, while the decoder is responsible for upsampling. This is a common trend in several designs, with the encoder reducing the size and the decoder increasing the sample rate. In an ideal scenario, we would not use pooling to downsample and maintain the sample size constant throughout. However, this would result in a massive number of parameters and would be computationally impossible.

Even though the output results were satisfactory, they may be seen rough and not smooth. This is because downsampling by 32 times using convolution layers causes information loss at the final feature layer. It is now exceptionally hard for the network to perform 32x upsampling with this bit of data. FCN-32 is the name of this architecture used in FCN.

The study presented two other architectures to handle this problem: FCN-16 and FCN-8. The input from the previous pooling layer is combined with the final feature map in FCN-16, and the network's objective is now to learn 16x upsampling, which is better than FCN-32. FCN-8 seeks to improve it by incorporating data from a prior pooling layer.

U-net [121] is an another network for semantic segmentation built on top of a fully convolutional network. It is designed to detect tumors in the lungs or brain for medicinal purposes. It has an encoder that down-samples the input images to a feature map and a decoder that uses learned deconvolution layers to up-sample the feature map to the input image size.

Because FCN downsamples an image as part of the encoder, it loses much information that is difficult to recover in the encoder. FCN attempts to address this by including input from pooling layers before the final feature layer. The U-Net architecture's essential contribution is the introduction of shortcut connections. It proposes feeding information to each upsampling layer in the decoder from the matching downsampling

layer in the encoder, capturing more delicate information while keeping computation minimal. Because the layers at the start of the encoder have more information compared to later layers, they will help the decoder's upsampling process by supplying fine details corresponding to the input images, significantly boosting the results.

Deeplab [15] has offered a variety of approaches to improve previous results and obtain finer output at reduced processing costs. The research suggests three primary changes for enhancing final output: atrous convolutions, atrous spatial pyramidal pooling, and conditional random fields.

The excessive shrinking caused by consecutive pooling processes is one of the primary issues with the FCN technique. The input image is downsampled by 32x due to a series of pooling operations, then upsampled to obtain the segmentation result. The loss of information caused by downsampling by 32x is critical for obtaining fine output in a segmentation operation. Also, because additional factors are involved in constructing a learned upsample, deconvolution to upsample by 32x is a memory and computation-intensive operation.

DeepLab suggests employing atrous convolution, also known as hole convolution or dilated convolution, to help utilize significant contexts with few parameters. Dilated convolution fills the space between parameters by expanding the size of the filter via inserting zeros. A term dilation rate is the number of zeros filled in between the filter parameters.

When the rate is equal to 1, a convolution operation is nothing more than a standard convolution. When the rate equals 2, a zero is put between each of the other parameters, giving the filter the appearance of a 5x5 convolution for a 3x3 convolution. It is now possible to obtain the context of a 5x5 convolution with only 3x3 convolution parameters. Similarly, the receptive field for rate 3 is 7x7.

The latest pooling layers in Deeplab are replaced with stride 1 instead of 2, lowering the downsampling rate to only 8x. The larger context is then captured using a succession of atrous convolutions. The output labeled mask is downsampled by 8x for training to compare each pixel. In order to provide an output of the same size for inference, bilinear upsampling is utilized, which produces acceptable results at lower

computational and memory costs because bilinear upsampling does not require any parameters, unlike deconvolution for upsampling.

Pooling is a technique for lowering the number of parameters in a neural network while also introducing an invariance property. The quality of a neural network unaffected by minor input translations is known as invariance. The segmentation output provided by a neural network is imprecise, and the borders are not precisely defined due to the characteristic obtained with pooling. The paper recommends using a graphical model called CRF to cope with this problem. A conditional random field is used as a post-processing technique to help determine shaper borders and improve results. It works by identifying pixels based on their labels and the labels of neighboring pixels.

DeepLabv2 [16] is a DeepLab's extension study. The critical change from DeepLabv1 is that it incorporates an extra design called Atrous Spatial Pyramid Pooling (ASPP). SPPNet [53] established the concept of ASPP Spatial Pyramidal Pooling to capture multi-scale information from a feature map. Previously, input images of various resolutions were supplied, and the computed feature maps were combined to obtain multi-scale information, but this required more computation and time. Multi-scale information can be obtained with a single input image using Spatial Pyramid Pooling.

ASPP applies the concept of combining information from various scales to Atrous convolutions. Kernels with different dilation rates are convolved with the input, and the results are merged. The input is convolved with four 3x3 filters with dilation rates of 6, 12, 18, and 24, and the outputs are concatenated as they all have the same size. The fused output of 3x3 varying dilated outputs is processed via 1x1 convolution to generate at the desired number of channels. Because the segmented image might be any size in the input, ASPP's multi-scale information aids in improving the outcomes.

Deeplab-v3 [16] added batch normalization and proposed multiplying the dilation rate by 1, 2, and 4 inside consecutive Resnet layers. Image level features have also been added to the ASPP module. Instead of bilinear upsampling 16x, Deeplab-v3+ proposed using a decoder. The decoder is inspired by the decoders used in designs like U-Net, which leverage encoder layer information to improve the output. The encoder

output is upsampled four times using bilinear upsampling, then concatenated with the encoder's features, which are upsampled four times following a 3x3 convolution. This method outperforms straight 16x upsampling in terms of results.

2.2.2 State-of-the-art Domain Adaptation Methods

In this section, the up-to-date domain adaptation methods for semantic segmentation and the studies that form the basis for these methods are explained.

Adapted representations frequently fail to capture pixel-level domain shifts, which are critical in dense prediction tasks. Because there is no explicit limit set on the predictions in the target domain, aligning marginal distributions does not always result in sufficient performance. A novel pixel-wise adversarial domain adaptation algorithm is presented in CrDoCo [22]. The approach is made up of two key components: an image-to-image translation network and two domain-specific task networks (one for target and the other for source). While the original and translated images in two separate domains may have unique looks, CrDoCo ensures that the domain-specific task network predictions is same.

The association between depth and semantics is more domain-invariant and suffers less from domain shift than the wide difference between synthetic and actual images. Chen et. al. [20] proposes a method for cross-domain semantic segmentation using auxiliary geometry information obtained from virtual data. Geometric information is used to reduce domain shift on two levels: on the input level, it is used to augment the standard image translation network with geometric information to translate synthetic data into realistic style; on the output level, it is used to build a task network that performs semantic segmentation and depth estimation simultaneously.

AdvEnt [154] use an entropy loss and an adversarial loss to present a complimentary domain adaptation technique. It reduces the entropy of target predictions by aligning the target and source domains' weighted self-information distributions with adversarial domain adaptation technique. The self-information or "surprisal" is defined as $-\log P_x(h, w, c)$ for a given a pixel-wise class score $P_x(h, w, c)$. Entropy is

effectively the expected evalue of self-information $E_x(h, w)$. However, entropy minimization tend to favor simple classes. As a result, it can be advantageous to steer the learning process using some prior knowledge. They achieve this by employing a simple class-prior based on the distribution of classes across source labels. The class-prior vector is calculated as an L1-normalized histogram of the number of pixels per class over the source labels.

Unlike AdvEnt, Chen et. al. [18] propose a maximum squares loss to balance the gradient of well-classified target samples to overcome the bias towards samples that are easy to transfer when using entropy minimization.

The discriminator in output-level adversarial domain adaptation is not supervised to capture various modes in the data distribution, and it may end up learning only low-level distinctions across domains such as tone or texture. Tsai et al. [147] present an unsupervised domain adaption technique for learning a better discriminator between the two domains by intentionally discovering many modes (K modes are generated using the K means procedure) in the structured output space of semantic segmentation.

Most previous research has treated semantic segmentation as the only mode of supervision for source domain data, ignoring additional, potentially available variables such as depth. Vu et. al. [155] believe that adding more depth-specific adaptation will have complementary effects, bridging the performance difference between source and target at test time. They achieve this by using an extra depth regression job to alter the segmentation backbone so that the depth information is included into a specialized deep architecture.

Semantic segmentation objects are separated into two types. Things (i.e. object instances) have far larger changes in appearance among images of different domains than stuff categories. To address the intra-class domain shift problem, Wang et. al. [164] first propose a stuff and instance matching (SIM) approach. Second, they present a self-supervised learning framework that works in conjunction with the suggested SIM structure to enable label-level transferring, which improves performance even further. They minimize the distance between each background class's stuff representation and the nearest intra-class source stuff feature representation for each tar-

get domain image. For the stuff representation for every background class, they average the features corresponding to the same background semantic class throughout the width and height of the image. Because the ground truth lacks instance-level annotations, they create the foreground instance mask by looking for disconnected regions in the label map L for each foreground class.

Previous studies have focused on adapting models straight from source data to unlabeled target data. Nonetheless, these methods ignore the significant distribution gap among the target data which is called an intra-domain gap. Pan et al. [108] proposed a two-step self-supervised domain adaptation technique to reduce both the inter-domain and intra-domain gap. First, they apply the adversarial domain adaptation approach to decrease the inter-domain gap. Then, using an entropy-based ranking mechanism, they divide the target domain into an easy and hard split. They propose using a self-supervised adaptation strategy from the easy to the hard split to reduce the intra-domain gap.

The category-level joint distribution is not taken into account in the global alignment approach. CLAN [102] proposes a category-level adversarial network with the goal of ensuring local semantic consistency throughout the global alignment process. First, they find classes with features that are already well aligned between the source and target domains, and they preserve this category-level alignment from adversarial learning's side effects. Second, they detect classes with differing distributions of features between the two domains and increase the weight of the adversarial loss during training. Generator G is separated into feature extractor E and two classifiers C1 and C2. E extracts features from the images it receives. C1 and C2 classify the features generated by E into pre-defined semantic categories like car, tree, and road. Following the co-training approach, they use a cosine distance loss to ensure that the weights of C1 and C2 are different. The final prediction map is created by adding the two different predictions together.

A Semantic-wise Separable Discriminator (SS-D) [32] is a technique for adapting semantic characteristics across target and source domains independently. It aims to solve the problem of inconsistent adaptation in class-wise adversarial learning. To create a more reliable separation, SS-D includes a progressive confidence technique.

To balance the class-wise adversarial learning process, an efficient Class-wise Adversarial loss Reweighting module (CA-R) is implemented, which causes the generator to focus more on poorly adapted classes. Their advantages over state-of-the-art class-wise adaptation techniques are two-folds. First, they use the downsampled pseudo label to isolate various semantic characteristics from the full feature space, making class-wise adaption independent. Second, during the adaptation phase, the progressive confidence technique will reduce inaccurate adaption.

Images' structural content is the most informative and decisive aspect in semantic segmentation, and it may easily be transferred between domains. The DISE [13] uses a series of common and private encoders to separate an image's high-level, domain-invariant structural information from its low-level, domain-specific texture information. The method's novelty is highlighted by the emphasis on deliberately regularizing the common and private encoders toward capturing structure and texture information, as well as the capacity to transform images from one domain to another for label transfer. Adversarial learning is applied based on spatial contextual resemblances between the source and target domains on segmentation map level.

Traditional image translation methods map the image directly from the source domain to the target domain, however the DLOW [46] proposes a method to generate a series of intermediate domains that shift from the source domain to the target domain. Images can be translated into intermediate domains and then used to perform domain adaptation. DLOW shows that using intermediate domain images, standard domain adaptation methods can be improved to obtain superior performance in the target domain. The conditional instance normalization layer is utilized to injects the domainness variable into the image translation network. The domainness variable is also utilized as discriminator weights to balance the output images' relatedness to distinct domains.

The quality of image-to-image translation is essential to the segmentation model. The segmentation adaptionmodel and the image translation model can be trained alternately and they improve to each other. BDL [88] proposes bidirectional learning to train image translation and segmentation model. It utilize a self-supervised learning (SSL) technique in training their segmentation adaption model in the forward direc-

tion. And the translation model is iteratively improved by the segmentation adaption model in the reverse way.

There are two drawbacks to using cycle-consistency in unsupervised domain adaptation. First, redundant components such as a target-to-source generator and associated computational cost are required. Second, when target data is limited compared to source data, as is the case with unsupervised domain adaptation, cycle consistency may be overly strong. One of the most successful strategies for domain adaptation in categorization is the self-ensembling technique. Choi et. al. [25] propose a computationally efficient and effective data augmentation strategy based on Generative Adversarial Networks (GANs) for domain alignment. They use self-ensembling to improve the performance of the segmentation network on the target domain using those augmented images. They use a source encoder to generate content representation and a target encoder to extract style representation, based on the notion that an image may be divided into two disentangled representations, content and style. They use Adaptive Instance Normalization (AdaIN) on feature maps of source pictures to correctly blend these two representations.

In order to manage low-level variability, sophisticated adversarial learning models are ineffective. FDA [176] investigates if simple alignment of low-level statistics between the source and target distributions can improve UDA performance without requiring further training beyond the basic job of semantic segmentation. It simply computes the (Fast) Fourier Transform (FFT) of each input image and inserts the low-level frequencies of the target images into the source images before reassembling the image for training, utilizing the original annotations in the source domain, via the inverse FFT (iFFT). They use entropy minimization to regularize segmentation network training. UDA becomes a semisupervised learning (SSL) problem since FDA aligns the two domains. By averaging the model weights, the mean teacher enhances semisupervised learning performance. FDA utilizes the mean teacher approachs to improve the alignment performance.

Because of the source domain's dominant data size, source-to-target translation increases the bias in translated photos. Yang et. al. [173] propose a method that uses image reconstruction to eliminate image translation bias and align cross-domain fea-

ture representations. The goal of a reconstruction network is to rebuild both source and target images using their predicted labels. This technique enforces cross-domain features with the same label to be close to one another.

Kim et. al. [76] suggest a method for learning texture invariant representation for better domain adaptation. They use a style transfer algorithm (Style-swap) to diversify the texture of the original source dataset, and use an image translation technique (CycleGAN) to translate the original source dataset. After that, their model goes through two levels of training. They learn texture-invariant representation in stage 1 by training a segmentation model with the texture-diversified dataset. They fine-tune the model to the texture of the target domain in stage 2 using the texture-invariant representation.

PyCDA [89] is based on two previously studied technique: curriculum domain adaptation and self-training. It builds a pyramid curriculum that includes numerous properties related to the target domain. These attributes are mostly concerned with the required label distributions among target domain images, image areas, and pixels. It incorporates the self-training pseudo labels into the curriculum as the finest layer of attributes about the target domain images.

One of the effective strategies for domain adaptation is self-ensembling. The mean-teacher method is commonly used in existing cross-domain semantic segmentation approaches. They use a consistency regularization on the target prediction of the student and teacher models under various perturbations. Previous studies, on the other hand, have not taken into account the accuracy of the predicted target samples, which could hinder the learning process by providing the student model with inaccurate guidance. More meaningful and trustworthy knowledge from the teacher model would be conveyed to the student model by utilizing the target samples' latent uncertainty information. Zhou et. al. [190] propose a stochastic forward pass method for capturing uncertainty. They do this by injecting a Gaussian noise into the target predictions prior evaluating the uncertainty. They also employ dropout for weight perturbation.

Even if two images are from separate domains, some parts, such as the car and road region, have similar structures. Sun et. al. [137] propose a domain adaptation method

to concentrate on these related areas in order to increase efficiency. They present a hierarchical transfer learning system for learning real image segmentation from synthetic images at three levels: pixel, region, and image. To assign greater weights to synthetic image granularities that are similar to actual ones, three weighting networks are learned together. They use hierarchical weighting networks to score similarity at the pixel, region, and complete image levels, taking into consideration both local and global information.

Although the method of aligning the two domains in latent feature space through adversarial learning has made significant progress in image classification, it frequently fails in semantic segmentation problems with overcomplex latent representations. SIBAN [101] allows for significance-aware feature purification prior to adversarial adaptation, making feature alignment easier and the adversarial training more stable. Their method is based on the information bottleneck (IB) theory, which states that the learned latent representation Z must produce a consistent prediction with the ground-truth labels Y while containing the least mutual information I(X, Z) with the provided input X. It is capable of balancing the information constraint across distinct classes in order to preserve ultimate performance on datasets with classes that are uncommon.

Context dependency is critical for semantic segmentation, but its transferability remains unclear. Yang et al. [174] propose a self-attention-based cross-attention mechanism to capture and adapt transferable context dependencies between two domains. They build two cross-domain attention modules to modify context dependencies from both spatial and channel views to achieve this goal. The spatial attention module, in particular, captures the local feature relationships between each place in the source and target images. The semantic dependencies between each pair of cross-domain channel maps are modelled by the channel attention module. They selectively combine context information from two domains to modify context dependencies.

2.2.3 Effective approaches for UDA-SS

The related works in UDA-SS show that the proposed methods show commonalities. Most of the state-of-the-art methods follow a similar framework. An example data-flow is illustrated in Figure 2.2. The effective approaches will be explained by the

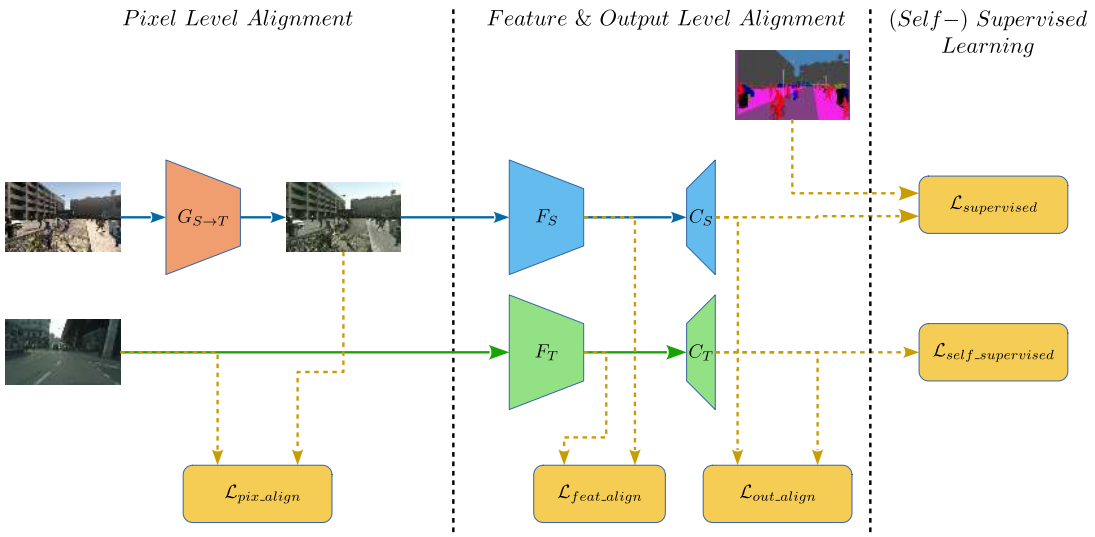


Figure 2.2: Unsupervised Domain Adaptation for Semantic Segmentation Common Framework

help of this figure.

There are two data flows, correspond to the source and the target domain. The dataflow of the image that belongs to the source domain is shown with blue lines in the figure. This flow can be summarized as follows. Firstly, the image is translated to the target domain with an image-to-image translation (IIT) method. Then the translated image is fed to the segmentation network. The semantic segmentation network can be trained with ground-truth labels as if the images are sampled from the target domain since ideally, the translated image matches to the marginal distribution of the target domain. The dataflow of the image that belongs to the target domain is shown with green lines in the figure. The segmentation map of the target image can be calculated directly by feeding it to the segmentation network. The segmentation network is composed of feature extraction network (F) and classification network (C). The feature extraction network is used to learn representative features for segmentation tasks. The two popular feature extraction backbone architectures are ResNet-101 and VGG16. The classification network is used the upscale the feature map to the input image size so that a segmentation map can be achieved. Even though two different segmentation network is shown in the figure for source and target domain, they usually share weights in recent works.

The paths to calculate loss functions are shown with dashed orange lines. The typical loss function ($\mathcal{L}_{supervised}$) for supervised learning is a standard pixel-wise cross-entropy loss. The knowledge in the source domain can be acquired with this loss function. In order to exploit images that belong to the target domain, several methods are studied. The major approach is to use an adversarial training to align the features, extracted from the two domains. The adversarial training achieved great success in the domain adaptation task. Therefore, this method is widely studied by researchers. The goal in adversarial training is to generate similar outputs to source domain image when a target image is forwarded to the segmentation network. Alignment can be done in three-level, namely pixel, feature, and output levels. The pixel-level alignment ($\mathcal{L}_{pix\_align}$) corresponds to image-to-image translation from the source domain to the target domain. If the features extracted from the feature extraction network are aligned, this approach is called a feature level alignment($\mathcal{L}_{feat\_align}$) . The output level alignment ($\mathcal{L}_{out\_align}$) is applied to the output of the classification network. Most of the methods use vanilla GAN loss to align features between domains. However, these approaches may cause training instability and negative transfer. In order to solve negative transfer and achieve better alignment across all segmentation classes, some methods are proposed, such as class-conditional, region/pixel level, and discriminative patch alignment.

The most effective way of boosting the performance of the segmentation network is to train it in a supervised manner in the target domain. However, it is not possible due to a lack of labels in the target domain. Self-training aims to train network based on the self-predictions. Self-training uses confident predictions of the segmentation network as a ground-truth label and trains network with these labels in a supervised manner. Curriculum learning splits target dataset based on the predictions of the segmentation network. The easy data are trained first, which helps better convergence. The disadvantage of self-supervised learning is that the overconfident mistakes and propagated errors due to the generation of pseudo-labels lead erroneous training.

There are also some studies, that doesn't fit the pipeline explained above. Information bottleneck aims a significance-aware feature purification before the adversarial adaptation in the feature extraction network F. Another approach is to use a self-ensembling method. A self-ensembling is composed of a teacher and a student

network, where the student is compelled to produce consistent predictions provided by the teacher on target data. The teacher's weights are updated by the student's with exponential moving average, generally. In this way, the teacher network learns more domain-independent features gradually. Entropy minimization defines a loss function that minimizes the entropy of the network prediction, which forces the network to make more confident predictions. Context-aware domain adaptation uses self-attention to detect context dependencies between source and target domain. It adapts the transferable context with this cross-attention mechanism.

In summary, the main research goals of the domain adaptation for semantic segmentation in the literature can be collected in four categories, which are listed as follows:

- Design image-to-image translation method which both preserves the semantic content and better align to the distribution of target domain.
- Learn domain invariant representation by alignment on feature or output level that can also avoid negative transfer.
- Learn domain invariant representation by non-adversarial methods like information bottleneck, mean teacher and entropy minimization.
- Better self-training techniques that better decides confident predictions and exploit them. In this thesis, we focus on the self-training techniques.

2.2.4 Source-Free UDA-SS

Traditional unsupervised domain adaptation exploits information from both labeled source data and unlabeled target data. However, data cannot be freely shared due to Intellectual Property or privacy concerns in some applications such as medical image processing. A new task called source-free domain adaptation for semantic segmentation (SF-UDASS) is introduced recently to eliminate the problem. In source-free scenarios, the annotated source dataset is unavailable. The task is to apply domain adaptation using only the trained source model instead of the source data as well as unlabeled target data.

During adaptation training, recovering and conserving the source knowledge learned

by a source model is critical due to missing source data and unclear target pseudo-labels. This is because the target model will vary from the source domain due to the unclear supervision information in target pseudo-labels. SFDA [95] proposes a solution. Knowledge transfer and model adaptation are two stages of the SFDA framework. It employs a generator in the knowledge transfer stage to estimate the source domain and synthesize fake samples that are similar to the real source data in distribution, which is used to transfer domain knowledge from a well-trained source model to a target model. Furthermore, the source model may predict proper labels on a target domain. As a result, Liu et. al. [95] propose an intra-domain patch-level self-supervision module (IPSM) based on entropy to use appropriately segmented patches as self-supervision during the domain adaptation stage.

Fleuret et. al. [34] proposes a method that enforces confident predictions in target domain under the feature noise. The model in training is initialized with the source model. It uses multiple decoders and introduces feature noise with dropout. The noise resilience is satisfied by uncertainty loss which is the squared difference of the decoder outputs. The training stability is improved with entropy minimization and self-training with threshold-based pseudo-labeling.

Many self-training approaches tend to fall into the "winner-takes-all" issue, where the majority classes completely dominate the segmentation networks and the networks fail to identify the minority classes since there is no supervision from the source domain data. You et. al. [178] propose a system that includes two mutually reinforcing elements: positive and negative learning. To avoid "winner-takes-all," they use an intra-class threshold to select the class-balanced pseudo-labeled predictions in positive learning. They offer a heuristic complementary label selection (HCLS) to create the complementary label for each pixel in negative learning. The pixel's complimentary label specifies which category it does not belong to.

2.3 Metric Learning

Distance metric learning is a machine learning technique for autonomously creating task-specific distance metrics from supervised data. The goal is to learn a transfor-

mation from input image space to an embedding space where semantic similarity is inversely proportional to the learned distance. Face recognition [94, 157, 156], few-shot learning [138, 116], person re-identification [167, 56], representation learning [181, 161], and visual tracking [84, 140] are just a few of the applications that it has been used for. In terms of the loss function used, distance metric learning approaches are divided into two groups [77]. These are pair and proxy-based approaches.

2.3.1 Pair-Based Approaches

Methods that utilizes a pairwise relationship in data are referred to as pair-based methods. They are defined in a dataset over tuples such as pair, triplet, and N-tuplet.

The most common loss function in pair-based approaches is triplet loss. An anchor sample, a positive sample from the same class as the anchor, and a negative sample from a different class make up a triplet. The embedding vector of anchors, positives, and negatives is represented by z^a, z^p, and z^n, respectively.

2.3.1.1 Triplet Loss

Triplet loss enables the distance between anchor-positive and anchor-negative pairs to be smaller by a defined margin $\alpha_{triplet}$. As a distance metric in the loss, Euclidean distance (ℓ_2) or squared Euclidean distance (ℓ_2^2) might be used. The triplet loss function is used to train the model, where d is a distance function over embedding vectors generated by the neural network.

$$L_{Triplet} = \sum_{a,p,n \subset T} [d(z^a, z^p) - d(z^a, z^n) + \alpha_{triplet}]_+ \tag{2.1}$$

A model's collapsing risk is defined as the collection of all embedding vectors at the same location. Because of hard negatives, there is a possibility of a model collapsing in ℓ_2^2 triplet loss [165]. Triplet with ℓ_2^2 solves the collapsing model problem by mining all triplets whose distance between anchor-negative and anchor-positive pairs is greater than the distance between anchor-positive pair but falls within the set margin.

Although this mining, known as semi-hard mining, solves the collapsing problem, it causes the hard triplet to be skipped, which is required to learn fine-grained details. In addition, due to the mining procedure, triplet loss suffers from excessive training complexity.

2.3.1.2 Multi-Similarity Loss

The Multi similarity (MS) loss develops a pair weighting framework in which the weights are the partial derivative of the loss with respect to the similarity matrix entry. Each distinction denotes the relative importance of the two pairs. It distinguishes between three new types of similarity: self, positive relative, and negative relative. These similarities provide the information needed to assess the pair's significance.

Let's look at three different similarities using a negative pair as an example. Negative pair cosine similarity is equivalent to self-similarity. To learn discriminative space, a negative pair with a high self-similarity contains subtle distinctions. As a result, negative pairs of this nature are given more weight. The disparity between the self-similarity of the negative pair and that of other positive pairs is characterized as positive relative similarity. When a positive pair gets away from a negative pair, for example, positive relative similarity grows. Negative pairs with a high positive relative similarity are difficult to differentiate from the anchor class. As a result, they are given large weights as well. The discrepancy between the self-similarity of the negative pair and that of other negative pairings is known as negative relative similarity. When a pair has a high negative relative similarity, it suggests that it is more informative than other negatives.

$$\begin{aligned} w_{ij}^{-} &= \frac{1}{e^{\beta(\lambda - S(z_i,z_j))} + \sum\limits_{k \in N_i} e^{\beta(S(z_i,z_k) - S(z_i,z_j))}} \\ w_{ij}^{+} &= \frac{1}{e^{-\alpha_{ms}(\lambda - S(z_i,z_j))} + \sum\limits_{k \in P_i} e^{-\alpha_{ms}(S(z_i,z_j) - S(z_i,z_k))}} \end{aligned} \tag{2.2}$$

We obtain a new pair-based loss function, multi-similarity (MS) loss, whose partial derivative with respect to S is the weights, when we combine pair mining and weighting schemes into a single framework.

$$L_{MS} = \sum_{i=1}^{m} \left\{ \frac{1}{\alpha_{ms}} \log \left[1 + \sum_{k \in P_i} e^{-\alpha_{ms}(S(z_i,z_k)-\lambda)} \right] + \frac{1}{\beta} \log \left[1 + \sum_{k \in N_i} e^{\beta(S(z_i,z_k)-\lambda)} \right] \right\} \tag{2.3}$$

2.3.2 Proxy-Based Approaches

To see all possible combinations of samples that comprise a given type of tuple in a dataset, pair-based losses require several iterations. Due to the mini-batch size limitation, pair-based losses only cover tuples in the mini-batch, regardless of how they are combined with instances from outside the mini-batch. As a result, seeing all combinations is difficult because it necessitates locating all mini-batches that result in all combinations.

Assume that data from a specific class is chosen at random and referred to as an anchor. Positive pair refers to a pair of samples from the same class of anchor, whereas negative pair refers to samples from different class of anchor. Negative pairs gain in value when their negative sample moves closer to the anchor, because closer negative samples resemble the anchor and have distinguishing characteristics. This value is calculated using negative hardness. As a result, mining harder negatives is a good way to teach the network. The hard negatives, on the other hand, may lead the network to collapse into a single point [165]. As a result, mining has an impact on not only the pace of network convergence, but also the point at which the network converges. Proxy-based loss replaces data-data relations with data-proxy relations, where a proxy is defined as a trainable representative of a class. As a result, proxy-based losses minimize the overall number of pair combinations, reducing mining effort. It also allows for inter-batch interactions via proxies.

The proxies are data placeholders that express the distribution of classes. In general, each class has just one proxy that corresponds to a single semantic class. However, for a case like multi-modality, it is conceivable to have more than one proxy for a class to increase intra-class variance. Although proxy losses can be employed in a variety of loss formulations, they are most commonly used in Neighboring Component Analysis (NCA) [44]. This goal is to increase the likelihood of instances from the same class

being in the same region while reducing the likelihood of examples from different classes being in the same neighborhood. It calculates the likelihood of z_i and z_j being neighbors. NCA is typically formulated as a log-likelihood minimization task.

$$p_{ij} = -\sum \log \frac{e^{-d(z_i,z_j)}}{\sum_{i \neq k} e^{-d(z_i,z_k)}} \tag{2.4}$$

2.3.2.1 Proxy Neighbor Component Analysis (ProxyNCA)

As a remedy to computational cost in pair based losses owing to many comparisons, ProxyNCA [105] associates each data with its proxy to mimic pairwise associations. By comparing each example to its proxy rather than all positive pairs, the proxy is integrated into NCA loss. Each piece of data is drawn to its own proxy and repulsed by others.

$$L_{ProxyNCA} = -\sum_{i=0}^{N_{mb}} \log \left(\frac{e^{-d(z_i,p(z_i))}}{\sum_{p^- \in P \setminus p(z_i)} e^{-d(z_i,p^-)}} \right) \tag{2.5}$$

2.3.2.2 ProxyNCA++

ProxyNCA++ is a version of ProxyNCA that has been enhanced. It has several improvements over ProxyNCA. To begin with, ProxyNCA++ considers the relationship between data and all proxies when determining the likelihood of data being attributed to one of the proxy. When compared to the ProxyNCA, which examines the distance ratio to negative proxies, the sum of probability of data being allocated to each proxy equals one. This attribute of ProxyNCA++ makes the loss more stable.

Secondly, ProxyNCA++ discovered that due to ℓ_2-Normalization, proxies have a little gradient. It improves the convergence of proxies by increasing the learning rate of proxies as a solution. Proxies are updated at the same time as the embedding layer. As a result, rapidly moving proxies enable stronger semantic class generalization.

Third, temperature scaling, which is utilized in soft-max, prevents the network from becoming overconfident by transforming hard probabilities into soft probabilities dur-

ing network calibration [50]. When a network is calibrated, the total number of examples predicted with a certain confidence includes true predictions equal to the confidence ratio times the total number of examples predicted with that confidence. ProxyNCA++ searches for an acceptable temperature scaling coefficient in view of this fact. This temperature scaling, on the other hand, is regarded as a gradient boost that exaggerates the distance between data and its proxy.

$$L_{ProxyNCA++} = -\sum_{i=0}^{N_{mb}} \log \left(\frac{e^{-D(z_i, p(z_i))/T}}{\sum_{p \in P} e^{-D(z_i, p)/T}} \right) \tag{2.6}$$

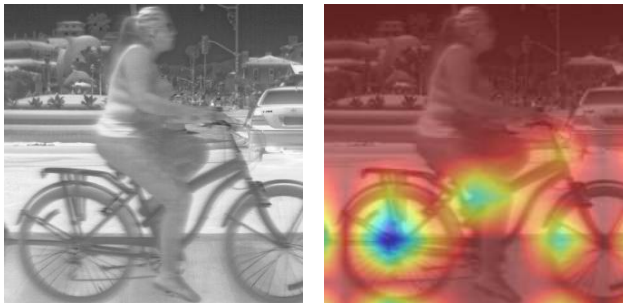
Last but not least, ProxyNCA++ discovered that using layer normalization at the end of backbone network and using global max pooling rather than global average pooling improves classification accuracy.

CHAPTER 3

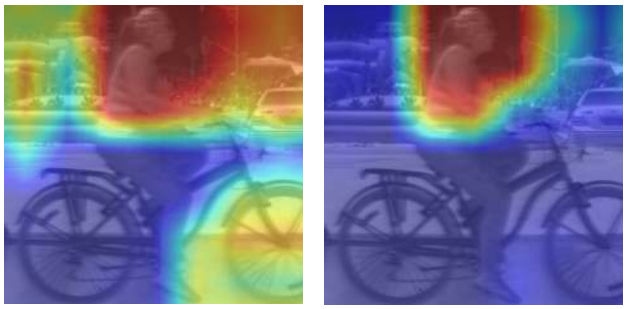
SELF-TRAINING GUIDED ADVERSARIAL DOMAIN ADAPTATION FOR THERMAL IMAGERY

Significant improvements have been made by using RGB images for image classification and detection problems [43, 54, 80, 118, 119] recently. The state-of-the-art methods have been trained on large-scale RGB datasets such as MS-COCO [90], ImageNet [29], Pascal-VOC [33], etc. However, low-lighting conditions hinder current state-of-the-art deep learning methods trained on visible spectrum images from performing well on computer vision tasks such as image classification, object detection, etc. Since thermal IR cameras are more robust against these conditions, exploiting them is useful for real-world applications. Therefore, usage of thermal IR cameras has become more common in the tasks related to autonomous driving, military operations, security surveillance, etc. Since such large-scale thermal datasets are not publicly available, it still remains an important challenge to achieve same level of performance on thermal image datasets. Therefore, exploiting complementary information offered by visible spectrum images is a straightforward technique to improve performance of the methods which work on thermal images for classification and detection problems. Unfortunately, recent studies demonstrated that performance of a deep model well-trained on visible spectrum images may significantly drop when applying to thermal images [30, 49, 51, 69, 73].

Since deep networks are sensitive to domain shift, a deep model trained on a large amount of labeled source domain data may fail at generalizing to unlabeled target domain data which are not similar to source domain data. To overcome these issues, unsupervised domain adaptation (UDA) aims to learn a model which maps both domains into a common feature space without requiring image pairs. Among the



(a) Input image (b) Source only



(c) Base method (d) Our method

Figure 3.1: Visualization of class activation maps on a target domain image using occlusion sensitivity [183]. Given a person image (a), our proposed model (d) activates semantically more meaningful parts of the image compared to our base method (c) [149], while the model trained on only source domain images (b) misclassifies the image as bicycle and activates wrong regions.

recent UDA methods, adversarial domain adaptation methods have become popular [149, 36, 91, 98]. These approaches incorporate adversarial learning as a two-player game similar to generative adversarial networks (GANs) [47]. Adversarial domain adaptation methods utilize a domain discriminator to distinguish source domain from target domain and a feature extractor to learn domain invariant representations to fool the domain discriminator. By learning domain invariant feature representations, adversarial domain adaptation methods assume that a classifier trained on source domain features is able to successfully classify target domain samples as well.

In this chapter, we propose an unsupervised adversarial domain adaptation method to align source and target domain distributions as described in Section 3.2. We employ Adversarial Discriminative Domain Adaptation (ADDA) [149] method as our base method. Although ADDA and other adversarial domain adaptation methods have achieved successful results, these methods face a major generalization limitation.

The limitation is that even though the distributions are aligned by learning domain invariant representations with a feature extractor, theoretically, the classifier may not work well on the target domain as shown in [6]. Therefore, learning discriminative representations for the unlabeled target domain is a difficult problem.

Based on the assumption of self-training, a classifiers' own high-confidence predictions are correct [193]. Since we assume that the predictions are mostly correct, exploiting the samples with high confidence values and retraining the classifier further improves the performance of the classifier. To this end, recent adversarial domain adaptation methods proposed to use pseudo-labels obtained from a classifier and retrain the model using the pseudo-labeled samples [126, 169, 186]. With these in mind, in this study, we propose a self-training guided adversarial domain adaptation (SGADA) method to overcome the generalization problems of adversarial domain adaptation methods (Figure 3.2). To perform self-training, pseudo labels obtained after warm-up phase of our method are assigned to the samples on the target domain to learn more generalized representations for the target domain. Pseudo-labels are assigned if confidences of the classifier trained on source domain and the domain discriminator reaches to threshold values for a target domain sample.

Our proposed method makes use of features obtained from visual spectrum images to improve classification performance on thermal domain. Moreover, our method does not need paired samples of RGB and thermal datasets. In order to train and test our proposed method, we use large-scale RGB dataset MS-COCO [90] and thermal imagery dataset FLIR ADAS [48]. We evaluate the proposed method quantitatively and qualitatively. We demonstrate our methods' success compared to the state-of-the-art unsupervised domain adaptation methods in Section 3.3. The results show that our method improves the performance of our base model and outperforms the state-of-the-art methods. Moreover, Figure 3.1 depicts that given a thermal image, our method classifies the image correctly by activating semantically more meaningful regions compared to our base method ADDA [149] and the model trained only on source domain data.

Effective classification for imbalanced data is an important field of research since class imbalance exists in many real-world applications [12, 74]. Therefore, it is im-

portant to address the class imbalance problem. In our experimental studies (Section 3.3), we show that class imbalanced datasets cause UDA methods to over-classify the majority category. We show that our proposed method achieves better results compared to the state-of-the-art UDA methods when class imbalance exists.

Our contributions are summarized as follows:

- We demonstrate the efficacy of combining visible spectrum and thermal image modalities by using unsupervised domain adaptation without requiring RGB-to-thermal image pairs.
- We propose a self-training guided adversarial domain adaptation method for thermal imagery. In order to learn more generalized feature representations for target thermal domain, we employ pseudo-labels generated by the classifier trained on RGB images and the discriminator, and train our model with these pseudo-labels.
- In order to demonstrate results of our method, we employ the large-scale RGB dataset MS-COCO as source domain and the thermal dataset FLIR ADAS as target domain. Extensive experimental analyses show that our proposed method outperforms the state-of-the-art unsupervised domain adaptation methods.

3.1 Related Work

By using RGB images, deep neural networks have gained popularity on computer vision tasks such as object detection, classification etc. Although significant improvements have been accomplished by using visible spectrum images, it still remains a critical problem to train a deep model robust to real-world problems, e.g. low-lighting conditions. To overcome these problems, thermal imagery has been used for object detection and classification problems [51, 69, 130].

Some of recent studies investigated the effects of combining RGB and thermal images to the performance on object detection problem [30, 51, 69, 73]. Since large-scale thermal datasets are not publicly available, in this study, we exploit complementary information offered by visible spectrum images to improve classification performance

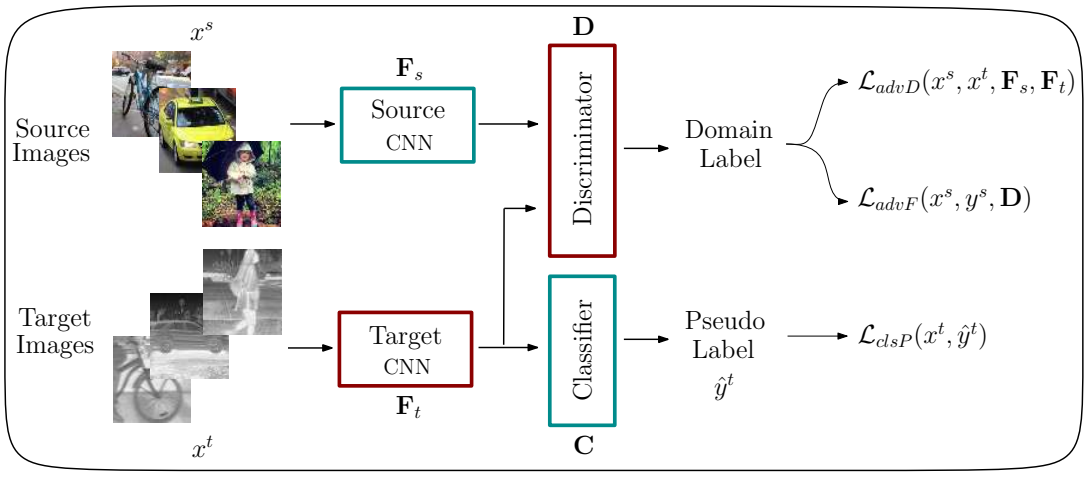


Figure 3.2: An overview of our proposed self-training guided adversarial domain adaptation (SGADA) method. Pseudo-labels generated after our methods' warm-up phase are assigned to target thermal images. Then, the target CNN $\mathbf{F}_t$ is trained using the pseudo-labels. The classifier $\mathbf{C}$ and the source CNN $\mathbf{F}_s$ are reused from our base model, and thus they are fixed. Target feature representations are learned by updating parameters of the target CNN $\mathbf{F}_t$ with respect to losses generated by the discriminator $\mathbf{D}$ and the classifier $\mathbf{C}$. Blue boxes indicate fixed network parameters while red boxes indicate trainable network parameters.

on thermal imagery without requiring RGB-to-thermal image pairs. We propose an unsupervised domain adaptation method in order to investigate efficacy of combining visible spectrum and thermal image modalities.

Numerous recent methods attempted to address domain adaptation problem. Recently, Generative Adversarial Networks (GANs) [47] inspired the field of domain adaptation, and thus deep adversarial domain adaptation methods have become popular [149, 9, 36, 91, 98].

Feature-level adversarial domain adaptation methods incorporate a domain discriminator to distinguish source and target domains while feature extractor learns features to fool the discriminator. Ganin et al. [36] proposed a gradient reversal layer to learn a feature extractor which generates features that maximize domain discriminator loss while minimizing label prediction loss. More recently, Tzeng et al. [149] proposed a method to learn a discriminative mapping of target images to source feature space by

fooling a domain discriminator which distinguishes the encoded target images from source samples. Many recent works employ adversarial training paradigm in their domain adaptation procedure [91, 98]. Although feature-level adversarial domain adaptation methods have accomplished successful empirical results, these methods suffer from a major limitation. As shown in [6], even if a feature extractor is well learned to generate domain invariant features, theoretically, the classifier may not work well on the target domain. Therefore, learning discriminative representations for the unlabeled target domain is considered difficult.

On the other hand, pixel-level adversarial domain adaptation methods translate source domain data into target domain data or vice versa by using image-to-image translation [92]. Bousmalis et al. [9] proposed an approach to learn a transformation in pixel-level from one domain to the other. Inspired by CycleGAN [192], Hoffman et al. [61] proposed CyCADA to increase semantic consistency of the image translation to improve the pixel-level methods. Even though pixel-level adversarial domain adaptation studies present remarkable results, image-to-image translation sometimes performs poorly on the datasets which have objects with many complex structures.

To overcome the limitations of adversarial domain adaptation methods, recent studies propose to directly deal with relationship between decision boundary and learned feature representations [85, 127]. Saito et al. [127] introduced to use a minimax training method to push target feature distributions away. Lee et al. [85] proposed to exploit adversarial dropout mechanism to learn more discriminative features by enforcing cluster assumption [14]. However, our experimental studies show that these methods and aforementioned adversarial domain adaptation methods have a drawback: Class imbalanced datasets [12, 74] lead to a performance drop for these methods.

We employ a self-training guided adversarial domain adaptation method to deal with the generalization problems of adversarial domain adaptation methods for thermal imagery. To the best of our knowledge, there is no self-training guided domain adaptation study in the literature of thermal image classification. Self-training is a technique to assign pseudo-labels to unlabeled samples using predictions of a classifier and retrain the model including the pseudo-labeled samples. Based on the assumption of self-training, a classifiers' own high-confidence predictions are cor-

rect [193]. Recent adversarial domain adaptation methods proposed to use pseudo-labels [126, 169, 186]. In our experiments, we show that our self-training guided method performs better than previous domain adaptation methods under class imbalance problem by learning more generalized representations for target thermal domain.

3.2 Proposed Method

Our proposed self-training guided adversarial domain adaptation method is illustrated in Figure 3.2. Before performing self-training, we extract pseudo-labels for the target domain samples. The pseudo-label extraction mechanism is depicted in Figure 3.3.

First, a feature extractor $\mathbf{F}_s$ and a classifier $\mathbf{C}$ are trained on source domain using labeled source domain RGB images (Figure 3.3-(a)). This step is named as pre-training. After this step, the classifier network $\mathbf{C}$ can successfully classify the source domain images by exploiting the features which are extracted by the source Convolutional Neural Network (CNN) $\mathbf{F}_s$. After the training on the source domain, we perform the second step: the warm-up phase for pseudo-label generation (Figure 3.3-(b)). In this step, we fix the parameters of the feature extractor $\mathbf{F}_s$ trained on the source domain. A target specific feature extractor $\mathbf{F}_t$ is learned in an unsupervised manner. By performing this step, features extracted from the source domain and the target domain are aligned with adversarial training. Therefore, we can use the classifier $\mathbf{C}$ trained on the source domain to classify target domain samples. We perform aforementioned two steps by following the training process of our base method ADDA [149]. In the last step, we fix the parameters of the feature extractor $\mathbf{F}_t$ trained on the target domain, the classifier $\mathbf{C}$ trained on the source domain and the discriminator $\mathbf{D}$. Then, we obtain predictions from the classifier and confidences from both the classifier and the discriminator for the target domain samples.

Once the predictions and the confidences are obtained, we utilize the predictions to give pseudo-labels for target domain samples using the confidences obtained from the classifier $\mathbf{C}$ and the discriminator $\mathbf{D}$ as shown in Figure 3.3-(c). We use prediction of the classifier for a target sample if classifiers' confidence value is higher than a threshold and domain label prediction of discriminator is close to source domain. That

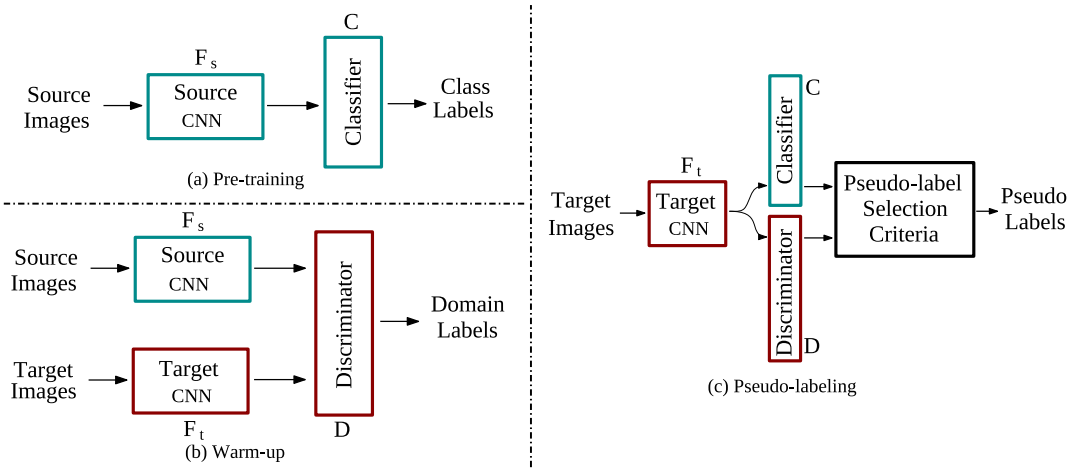


Figure 3.3: Illustration of our pseudo-label generation mechanism.

is, we can use the prediction of the classifier if the discriminator incorrectly classifies target samples. By using this pseudo-label selection mechanism, intuitively, we select samples with feature representations which are close to data with known labels. Next, as illustrated in Figure 3.2, we train our proposed method using extracted pseudo-labels.

A general definition of unsupervised domain adaptation, and self-training guided adversarial domain adaptation procedures of our proposed method are described in Section 3.2.1 and 3.2.2, respectively.

3.2.1 Unsupervised Domain Adaptation

In the general definition of unsupervised domain adaptation (UDA) problem, we are given n_s labeled samples from a source domain $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ and n_t unlabeled samples from a target domain $\mathcal{D}_t = \{(x_j^t)\}_{j=1}^{n_t}$. The goal of UDA problem is to learn a feature extractor $\mathbf{F}_t$ for the target domain and a classifier $\mathbf{C}_t$ which correctly classifies the features. It is not possible to perform supervised training since there is no labeled samples in the target domain. Therefore, UDA learns to adapt the source feature extractor $\mathbf{F}_s$ and the source classifier $\mathbf{C}_s$ to be able to use them on target domain.

3.2.2 Self-training Guided Adversarial Domain Adaptation (SGADA)

The task of adversarial domain adaptation methods is to adversarially align source and target domain representations. For this purpose, adversarial domain adaptation methods propose to reduce the gap between $\mathbf{F}_s(x^s)$ and $\mathbf{F}_t(x^t)$. Thus, the classifier $\mathbf{C}_s$ trained on source domain can be applied to the representations on target domain, and necessity to train a separate $\mathbf{C}_t$ can be eliminated. As a result, we obtain $\mathbf{C} = \mathbf{C}_s = \mathbf{C}_t$ [149]. We employ the feature extractor $\mathbf{F}_s$ and the classifier $\mathbf{C}$ which are learned during the warm-up phase. In this subsection, we elaborate our training scheme shown in Figure 3.2.

We use the following loss function for the domain discriminator $\mathbf{D}$ which distinguishes source domain from target domain:

$$\begin{aligned}\mathcal{L}_{advD}(x^s, x^t, \mathbf{F}_s, \mathbf{F}_t) = &-\frac{1}{n_s}\sum_{i=1}^{n_s}\log[\mathbf{D}(\mathbf{F}_s(x_i^s))] \\ &-\frac{1}{n_t}\sum_{i=1}^{n_t}\log[1-\mathbf{D}(\mathbf{F}_t(x_i^t))].\end{aligned} \tag{3.1}$$

Given source images x^s and target images x^t, we update the parameters of the domain discriminator $\mathbf{D}$ with respect to outputs of the feature extractors $\mathbf{F}_s$ and $\mathbf{F}_t$. While updating the parameters, we fix and reuse the source feature extractor $\mathbf{F}_s$ which is trained in the pre-training step of our pseudo-label generation.

We employ two loss functions to train the target CNN $\mathbf{F}_t$: adversarial loss $\mathcal{L}_{advF}$ and self-training loss $\mathcal{L}_{clsP}$. The adversarial loss is formulated as follows:

$$\mathcal{L}_{advF}(x^s, y^s, \mathbf{D}) = \frac{1}{n_t}\sum_{i=1}^{n_t}\log[\mathbf{D}(\mathbf{F}_t(x_i^t))]. \tag{3.2}$$

Note that we reuse the parameters of the source feature extractor $\mathbf{F}_s$ from the previous step to initialize $\mathbf{F}_t$.

We exploit pseudo-labeled target domain samples to perform self-training guided adversarial learning. After the learning of the classifier $\mathbf{C}$ and the domain discriminator $\mathbf{D}$ is completed during the warm-up phase, we obtain predictions from the classifier

and confidences for these predictions. Given an unlabeled target domain sample, if confidence of the classifier $\mathbf{C}$ is higher than pre-defined threshold and the domain discriminator $\mathbf{D}$ classifies the sample as source domain, we include the sample during our self-training guided adversarial domain adaptation step. Also, if the domain discriminator $\mathbf{D}$ classifies the sample as target domain with a confidence lower than a pre-defined threshold, we assign the pseudo-label $\hat{y}^t$ generated by the classifier $\mathbf{C}$ to the sample as well. By using $\hat{n}_t$ pseudo-labeled samples on target domain, we aim to train a target specific feature extractor (Figure 3.2). We use the new self-training loss function for our proposed method:

$$\mathcal{L}_{clsP}(x^t, \hat{y}^t) = \frac{1}{\hat{n}^t} \sum_{i=1}^{\hat{n}^t} \ell_{ce}(\mathbf{C}(\mathbf{F}_t(x_i^t)), \hat{y}_i^t). \tag{3.3}$$

The overall objective function to train our proposed method SGADA is defined as:

$$\begin{aligned} \min_{\mathbf{D}} \mathcal{L}_{advD}(x^s, x^t, \mathbf{F}_s, \mathbf{F}_t) \\ \min_{\mathbf{F}_t} \mathcal{L}_{advF}(x^s, y^s, \mathbf{D}) \\ +\lambda \mathcal{L}_{clsP}(x^t, \hat{y}^t), \end{aligned} \tag{3.4}$$

where λ is a trade-off parameter. We set the trade-off parameter λ and thresholds based on validation split (see Section 3.3.2 for further details).

3.3 Experiments

We perform extensive evaluations and compare our proposed method with several state-of-the-art unsupervised domain adaptation methods.

3.3.1 Datasets

We prepare a new RGB-to-thermal domain adaptation setting for classification using FLIR ADAS [48] as thermal dataset and MS-COCO [90] as visible spectrum dataset for our experimental studies.

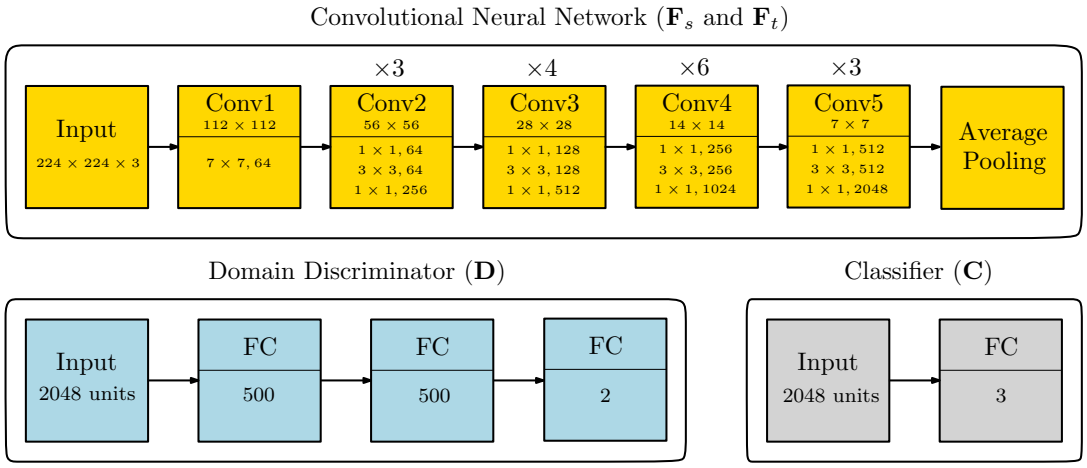


Figure 3.4: The network architectures used for our experimental analyses.

FLIR ADAS [48] consists of 9,214 thermal images with bounding box annotations. Each image has a resolution of 640 $\times$ 512 and obtained from FLIR Tau2 camera. 60% of the images are captured during daytime and the remaining 40% of the images are captured during night. The dataset provides both visible spectrum (RGB) images and thermal images. We consider only the thermal images of the dataset for our experiments. We use the training and test splits as suggested in the dataset documentation for our experiments. The objects in the dataset are classified into four categories i.e. bicycle, car, dog, and person. However, the dog class has very few annotations. Therefore, the dog class is not considered in our experimental studies. We crop square images using bounding box annotations for objects. After objects are extracted, we resize the images to 224 $\times$ 224. Finally, our thermal dataset consists of 4,137 samples of bicycle, 43,734 samples of car, and 26,294 samples of person images. Example images from FLIR ADAS dataset are shown in the second row of Figure 3.5.

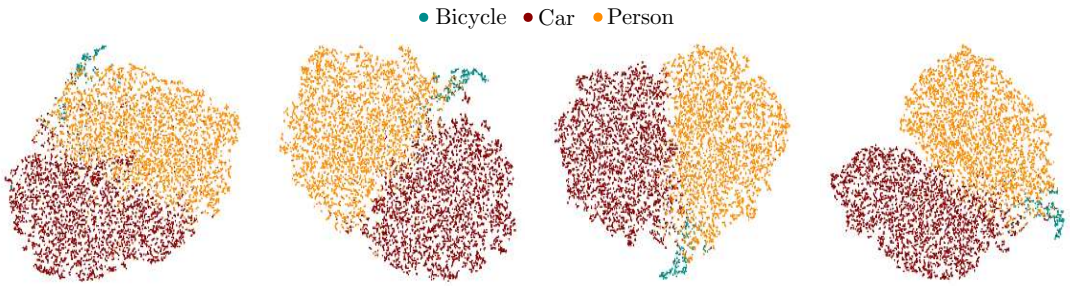
Our proposed method incorporates publicly available large-scale visible spectrum datasets to improve classification performance on thermal dataset. Therefore, we consider using an RGB dataset which includes the same classes as FLIR dataset [48] (bicycle, car, and person). For this purpose, we use MS-COCO dataset [90] as our visible spectrum dataset. In the first row of Figure 3.5, we show some example images from MS-COCO dataset. MS-COCO dataset contains 91 object categories (airplane, bicycle, bird, car, person, etc.). In total, there are 123,287 images and around 886,000

Figure 3.5: Example images from MS-COCO dataset [90] and FLIR ADAS thermal dataset [48].

bounding boxes. 118,287 of the images are for training while 5,000 of the images are for validation split. We apply standard training and test splits as provided in the dataset documentation for our experiments. We use only bicycle, car, and person classes to match with our thermal dataset. We cropped the annotated objects with the same procedure as applied to FLIR dataset. Once objects are extracted, we resize the images to 224 $\times$ 224. Our visible spectrum image dataset extracted from MS-COCO consists of 5,732 samples of bicycle, 38,453 samples of car, and 209,162 samples of person images.

3.3.2 Implementation Details

For our experiments, we used same training procedure with ADDA [149]. For a fair comparison with other methods, we employed ResNet-50 [54] pre-trained on ImageNet [29] as backbone for all methods. Our network architectures are given in Figure 3.4. The architecture of our feature extractors (source CNN $\mathbf{F}_s$ and target CNN $\mathbf{F}_t$) is the ResNet-50 without the last fully connected (FC) layer. In the figure, each convolutional residual unit is depicted with the size of filters at the top and the outputs



(a) Source only (b) ADDA [32] (c) SGADA-Cls (d) SGADA-Cls+Disc

Figure 3.6: The t-SNE visualization of network activations on target thermal domain generated by source only model (a), our base method ADDA [149] (b), our proposed method SGADA with classifier confidences only (c), and our proposed method SGADA with classifier and discriminator confidences (d).

of each convolutional layer at the bottom. The notation k $\times$ k, n in the convolutional layer blocks represents a filter of size k and n channel. The number on the top of the convolutional layer blocks denotes the number of repetition for the unit. The domain discriminator $\mathbf{D}$ consists of three FC layers: two consecutive hidden units of 500 neurons and the discriminator output. The classifier $\mathbf{C}$ has only one FC layer.

To train our method, we set batch size to 32. Number of epochs was set to 15 in our experiments. Parameters were updated using ADAM optimization algorithm [79]. For pre-training of our method on source domain only, we set learning rate to 5e-4. For adversarial adaptation step of our method, we set learning rate as 1e-5 and discriminator learning rate as 1e-3. We used same learning rates as used in the previous step for self-training guided adversarial adaptation step of our method. We set λ value as 0.7 and threshold value as 0.87 for our method with classifier confidences only (SGADA-Cls). For our method with classifier and discriminator confidences (SGADA-Cls+Disc), we used same learning rates as used in the previous step, and λ value of 0.25, classifier threshold value of 0.79, and discriminator threshold value of 0.87. We used the same experimental settings for training and testing. We exploit classification accuracy to compare our proposed method with other methods. We implemented our proposed method using PyTorch framework [112].

3.3.3 Evaluation of SGADA

In our experiments, we select visual spectrum (RGB) domain as the source domain, and thermal domain as the target domain. As a general practice in the field of domain adaptation, we denote *source only* as the target domain performance of a model trained using only source domain images, and *target only* as that of a model trained on the target domain. Performances of *source only* and *target only* models serve as baselines for the lower and upper bound performances.

Table 3.1: Per-class classification performance comparison.

Method	Bicycle	Car	Person	**Average**
Source only	69.89	83.89	86.52	80.10
Pixel-DA [9]	62.53	89.99	76.73	76.42
DTA [85]	75.45	**97.65**	92.45	88.52
MCD-DA [127]	81.71	94.90	91.83	89.48
DANN [36]	78.16	95.07	**96.24**	89.82
CDAN [98]	78.16	97.10	94.82	90.03
ADDA [149]	86.67	96.95	89.10	90.90
SGADA (ours)	**87.13**	94.44	92.03	**91.20**
Target only	87.59	98.78	96.35	94.24

Quantitative Analysis. We compare our proposed method SGADA with several state-of-the-art unsupervised domain adaptation methods: Unsupervised Pixel-Level Domain Adaptation with Generative Adversarial Networks (Pixel-DA) [9], Drop to Adapt (DTA) [85], Maximum Classifier Discrepancy for Unsupervised Domain Adaptation (MCD-DA) [127], Domain Adversarial Neural Network (DANN) [36], Conditional Domain Adversarial Adaptation (CDAN) [98], and Adversarial Discriminative Domain Adaptation (ADDA) [149]. Since these methods do not consider domain adaptation problem for thermal datasets, there exist no reported results on their paper for our dataset. Therefore, we trained and evaluated all these methods for our dataset.

Table 3.2: Ablation studies of different pseudo-label selection scenarios for self-training guided adversarial domain adaptation.

		Bicycle	Car	Person
	Number of samples	3702	38657	21081
Classifier confidences only	Number of selected samples	3995	35494	20323
	Number of correctly selected samples	2901	34905	18911
	Accuracy of selected samples (%)	72.62	98.34	93.05
Discriminator confidences only	Number of selected samples	3598	36024	3800
	Number of correctly selected samples	2874	35251	3549
	Accuracy of selected samples (%)	79.88	97.85	93.39
Classifier and discriminator confidences together	Number of selected samples	3557	35123	3558
	Number of correctly selected samples	2873	34558	3454
	Accuracy of selected samples (%)	80.77	98.39	97.08

Table 3.3: Ablation experiments.

Method	Bicycle	Car	Person	**Average**
SGADA-Cls	87.36	95.27	90.62	91.08
SGADA-Cls+Disc	87.13	94.44	92.03	**91.20**

Per-class domain adaptation performances are reported in Table 3.1. The results show that our proposed method which uses classifier and discriminator confidences together for self-training outperforms the state-of-the-art methods. Although DTA [85] and DANN [36] perform well for *car* and *person* classes respectively, their performance for *bicycle* class cannot reach the top performances since the number of samples for *bicycle* class is much less than the other classes. On the other hand, our proposed method achieves more balanced performance scores for all classes and outperforms other methods. It is important to address this problem since datasets for real-world problems usually include imbalanced classes [12, 74]. As shown in the table, our proposed method achieves more balanced class-wise accuracies compared to our base method ADDA [149], and furthermore our method increases the average accuracy over our base method.

Qualitative Analysis. We visualize the feature representations on target thermal domain with t-SNE [152] for qualitative analysis in Figure 3.6. The features of source only model on target domain can not be discriminated very well while ADDA [149] discriminate some overlapping points in the feature space. Our proposed model which uses only classifier confidences for self-training (SGADA-Cls) learns more discriminative representations. As shown in the figure, our proposed model which uses classifier and discriminator confidences together for self-training (SGADA-Cls+Disc) further enlarges inter-class distances, especially for *car* and *person* classes.

Ablation Study. To evaluate the contributions of our proposed method, we perform ablation studies. We examine effects of using classifier and/or discriminator confi-

dences for pseudo-label selection. As described in Section 3.2, we select samples using our base method. Table 3.2 shows three cases where we select target domain samples using only the confidences of the classifier, using only the confidences of the discriminator, and using the confidences of both the classifier and the discriminator. As shown in the table, if we utilize the discriminator confidences, the number of selected samples for the *person* class decreases. Moreover, when we use both discriminator and classifier confidences to select target domain samples, accuracy of the pseudo-labels increases significantly for all classes. This results in better separation of feature representations as depicted in Figure 3.6 (c)-(d). Furthermore, since accuracy of selected samples for all classes using classifier and discriminator confidences is higher than the other cases, class imbalance of our proposed method (SGADA-Cls+Disc) reduces compared to SGADA-Cls as shown in Table 3.3. And thus, the overall accuracy of our proposed method surpasses SGADA-Cls, resulting in the best overall performance.

CHAPTER 4

A SIMILARITY-BASED PSEUDO-LABEL SELECTION METHOD FOR DOMAIN ADAPTATION OF SEMANTIC SEGMENTATION

Semantic segmentation is one of the main computer vision tasks, aiming to assign each pixel to a semantic class. Researchers devoted significant effort to this area in recent years, leading to substantial progress. Recent approaches are driven by deep neural networks trained by using large annotated datasets in a supervised manner [96, 16]. However, manually annotating a large dataset with dense labels requires a significant amount of human effort, making it costly. One alternative to manual annotation is using photo-realistic annotated images rendered from simulators or game engines. However, the model's performance drops significantly when the learned source model is directly applied to the target data. The main cause of this problem is the domain mismatch between the real and the synthetic data. Various domain adaptation techniques are proposed to address this problem. This study focuses on the case where there is no label from the target domain called unsupervised domain adaptation (UDA).

Recently, self-training methods are gaining popularity in the UDA of semantic segmentation [103, 131, 195, 196]. In self-training approaches, one-hot pseudo-labels are generated for target domain images based on the segmentation network's prediction, and the network is retrained with pseudo-labels, exploiting them for better alignment. The model is updated iteratively with pseudo-label generation and retraining cycle until task performance converges. Even though self-training helps to decrease the domain gap, false labels have an adverse effect on features. Several pseudo-label selection strategies have been proposed to choose a proper subset of target prediction to address this problem. Thresholding the cross-entropy response of the segmentation

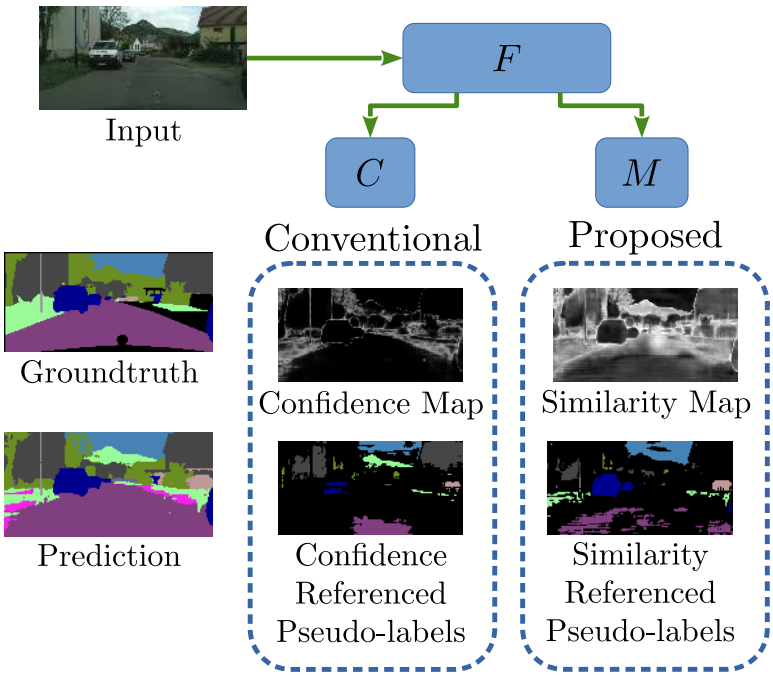


Figure 4.1: Comparison of our proposed method with a conventional pseudo-label selection approach. We utilize similarity metric to class reference learned in the target domain with a metric learning technique instead of prediction confidence.

network is one of the primary methods that assume the target samples with a larger prediction probability have better accuracy.

Most of the UDA methods focus on the transfer of knowledge from the source to the target domain. However, real-world target data have varied distribution due to several factors such as weather conditions or architectural differences of cities. This phenomenon is called the intra-domain gap. Even though the domain gap is significantly reduced by UDA methods for some samples in the target domain, the others are not well aligned with the source domain, leading to low confidence scores even though the prediction is correct. When confidence-based pseudo selection methods are used in self-training, a considerable amount of these samples are eliminated, therefore causing only a limited contribution to the domain alignment.

In this study, instead of using a confidence score, we propose a novel pseudo-label selection method that evaluates correctness based on a similarity metric defined in the target domain, as illustrated in Figure 4.1. Our method is a three-step domain adaptation technique: 1) Similarity metric learning: We apply deep metric learning to

the segmentation network's encoder outputs. The primary purpose of metric learning is to learn a transformation to map semantically similar samples closer and dissimilar examples far away in a metric space so that the model forms a dense cluster in the metric space for each class. Motivated by this, we introduce a metric network to learn a similarity metric for each pixel. The triplet loss function is used as a training criterion that decreases the intra-domain gap in metric space. 2) Pseudo-label generation: We generate pseudo-labels by filtering the network prediction, using the distance to class-proxies, which are the vectors representing the classes in metric feature space. To reduce the computation time, we computed class-proxies during training with an outlier elimination mechanism. 3) Adversarial alignment: We use an adversarial method to align source and target domain as well as supervised and self-supervised losses calculated from source labels and target pseudo-labels, respectively.

The proposed method, Similarity based Pseudo-Label Selection (SPLS), is evaluated on large-scale rendered images to real image adaptation scenarios. Our method is orthogonal to existing methods and can be easily generalized for further performance gains. In summary, our main contributions are:

1. We propose a novel pseudo-label selection method based on a similarity metric.
2. We train a deep metric network to reduce the intra-domain gap in the target domain.
3. We propose an online proxy calculation method that eliminates erroneous features with an outlier elimination mechanism to increase the robustness of proxy calculation and to reduce the computation time of our method.

4.1 Related Works

Unsupervised Domain Adaptation Unsupervised domain adaptation is widely studied in image classification [159]. Recently, adversarial learning based UDA methods have achieved much progress in learning domain-invariant features [23, 37, 61, 85, 114, 146, 149]. Adversarial methods are composed of two neural networks, namely

generator, and discriminator. For a given image, while the discriminator is trained to classify the image domain, the generator learns to extract invariant features for the source and the target domain to fool the discriminator. Despite their success in image classification [146], the feature-level alignment methods usually fail in dense prediction tasks due to over-complex representation. For the image segmentation task, the efficacy of image [13, 22, 46, 61, 88, 176] or output level [13, 20, 102, 146, 154] adversarial alignment is demonstrated. In previous studies, CyCADA [61] utilizes an image-to-image translation method based on cyclic-consistency proposed in CycleGAN [192] along with semantic consistency to align domains at the image-level. AdaptSegNet [146] proposes structured output space alignment to output level alignment. AdvEnt [154] improved output level alignment performance by exploiting the entropy of pixel-level predictions.

Self Training Self-training approaches, commonly used in semi-supervised learning literature [153], are gaining popularity in UDA recently. The model is iteratively trained to utilize both the labels already available in the source domain and the pseudo-labels generated by the trained network in the target domain. The standard approach is to use high-confident predictions as pseudo-labels. However, transferability from the source to the target domain is different for each class, making the model biased towards easy-to-transfer classes. Motivated by this, CBST [196] proposes a class-balanced self-training, which utilizes a class-wise confidence score for filtering target predictions. CRST combines CBST with diverse confidence regularizers, resulting in a better performance [195]. IAST presents an instance adaptive pseudo-label generation strategy to improve the quality of pseudo-labels [103]. All the previous studies employ prediction confidence as a correctness measure. However, real-world data have a multi-modal distribution. While some samples are well aligned with the source domain, others may not, leading to a low confidence score. Such methods eliminate correct predictions with low confidence that limit self-training efficacy. The technique that we propose uses a similarity metric learned in the target domain to select pseudo-labels.

Metric Learning Deep metric learning is a widely studied topic in computer vision that has various applications such as image retrieval [105], clustering [57], person re-identification [24] and face recognition [132]. Triplet loss [24] and variants are widely used in the metric learning literature. The main intent of deep metric learning is to learn a transformation from feature space to metric space so that semantically similar samples get closer and dissimilar samples become far away in predefined norm space. Metric learning is usually used to learn a similarity between images or patches. Unlike the common approach, Chen et al. [21] exploits pixel-wise metric learning for video object segmentation, which can be modeled as an image retrieval problem for interactive segmentation.

For the UDA task, metric learning is usually utilized in the image classification task. It is used to align the source and the target domain by learning metric space that semantically similar samples from both source and target domain are close to each other [67, 82, 114, 179].

Our proposed method uses pixel-wise metric learning to determine a class-wise similarity metric only in the target domain to guide pseudo-label selection.

4.2 Proposed Method

In this section, we present our similarity based pseudo-label selection (SPLS) approach to unsupervised domain adaptation problem for semantic segmentation. Let S denote a source domain consisting of a set of images X_S and corresponding label set Y_S, and T is a target domain containing only unlabeled image set X_T. The aim is to train a segmentation network G that can make a reliable and accurate prediction of the dense label for each image in the target domain T with the knowledge gained from source domain S, in which supervised training is possible. The segmentation network G is composed of two sub-networks called feature extraction (F) and classification (C) networks. The discriminator network (D) is utilized for adversarial training alongside G. In addition to these, we introduce a simple metric learning network M cascaded to F to form a similarity metric in the similarity metric learning phase.

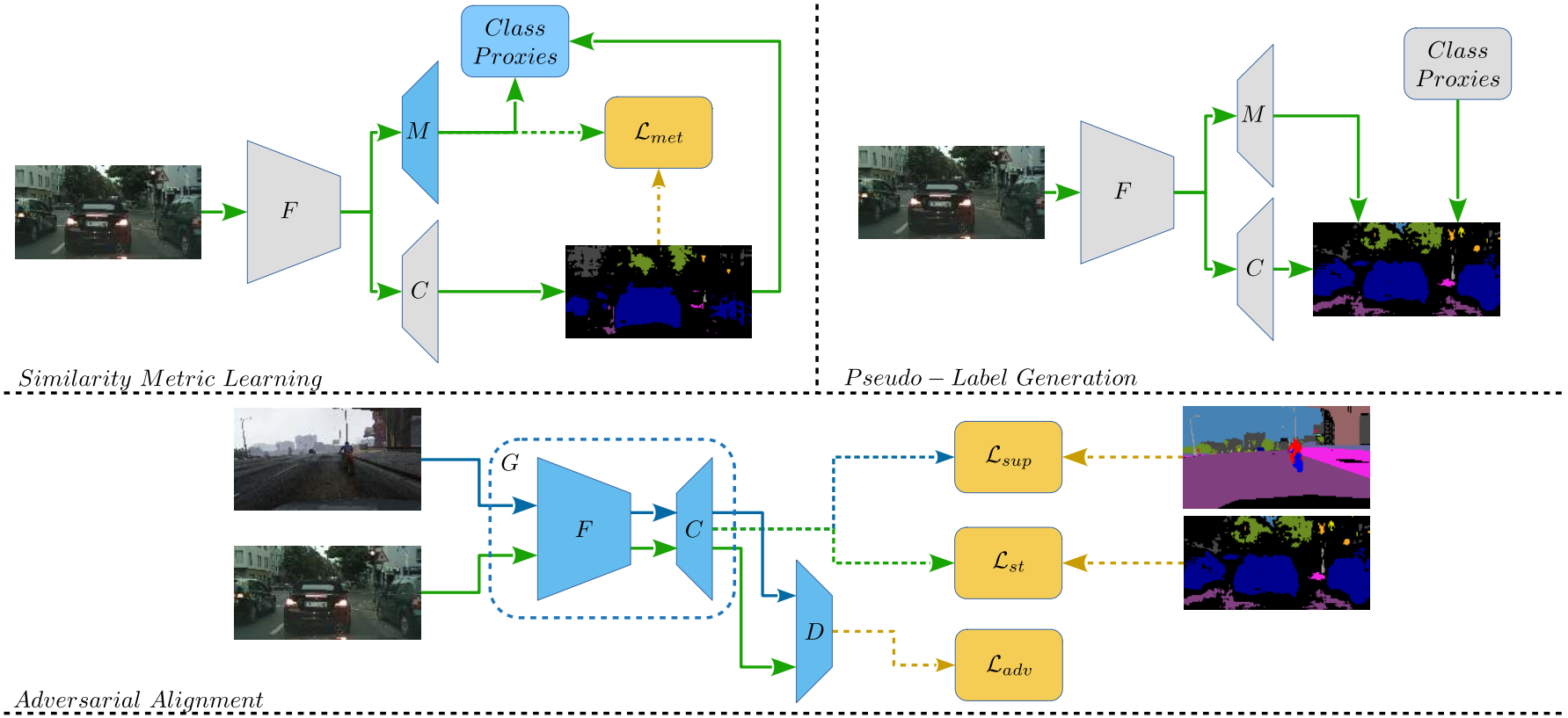


Figure 4.2: The overview of the three stages of SPLS. The dataflow for the images that belong to the source and target domains are shown with blue and green lines in the figure, respectively. The paths that calculate loss functions are shown with dashed lines. The segmentation network (G) is composed of two sub-network called feature extraction (F) and classification (C) networks. The feature extraction network is used to learn representative features for segmentation tasks, while the classification network is used to assign a class label for each pixel using features extracted by F. The discriminator (D) predicts from which domain the input is sampled. D is exploited in adversarial training. We introduce a simple metric learning network M cascaded to F. M learns a transformation to map semantically similar samples closer and dissimilar examples far away in the target domain to form a dense cluster in metric space for each-class. The gray boxes represent the networks whose parameters are fixed, while blue boxes represent those currently being updated. The orange boxes annotated with $\mathcal{L}$ represent the loss functions.

The dataflow of our proposed method is illustrated in Figure 4.2. In order to address the domain gap, we combined self-supervised training with adversarial learning. The proposed method comprises three stages: similarity metric learning, pseudo-label generation, and adversarial alignment. Our method aims at selecting pseudo-labels using a similarity metric to minimize the intra-domain gap and retrain the segmentation network with self-training. The metric learning network is trained with a confidence-based pseudo-label to learn a similarity metric between classes in the target domain. We select the normalized class threshold proposed in CBST [196] to provide class balanced training in metric learning. The metric learning model learns a transformation that brings semantically similar samples closer and pushes dissimilar examples far away. With the help of that, we obtain semantically clustered features for each class in the similarity metric space of the target domain, which decreases the intra-domain gap. The class representatives in metric space called class-proxies, are learned during M's training concurrently to reduce computation time. The pseudo-labels are generated by filtering the segmentation network's prediction. The selection is based on the similarity distance of metric feature to class-proxies in the pseudo-label generation phase. In the adversarial alignment phase, the segmentation network is trained with an output-level adversarial alignment technique to decrease the domain gap.

4.2.1 Similarity metric learning

In this study, we perform pixel-wise metric learning that learns the similarity between pixels in target images. We utilize the triplet loss to train the M network. Three samples, namely, an anchor, a positive, and a negative sample, contribute to metric learning training. These three samples form a required sample set, which is called a triplet: The anchor and the positive samples belong to the same class, while the negative sample belongs to one of the other classes. The triplet loss encourages the network to form a cluster such that positive samples are closer than negative samples to the anchor. The loss function also introduces a margin between the positive and the negative pairs to improve clustering performance. Formally, let $f \in \mathbb{R}^N$ is an N-dimensional normalized feature vector on which a specified metric is to be appeared for a specific pixel. The metric network takes features generated by feature extractor

network F and produces a metric map $m \in \mathbb{R}^{HxWxN}$. In the metric training phase, we select a batch of triplets from the features of the single target image and train metric network using triplet loss, represented as follows:

$$\mathcal{L}_{met} = \sum_{i}^{K} \max\left\{\left[\|f_{ai} - f_{pi}\|_2^2 - \|f_{ai} - f_{ni}\|_2^2 + \alpha\right], 0\right\}. \tag{4.1}$$

In Equation 4.1, f_a, f_p, f_n, K and α correspond to features of anchor, positive, negative pixels, batch size, and margin value, respectively.

In metric network training, anchor, positive and negative samples are selected using confidence-based pseudo-labels. It is essential to select the correct triplet to train the network since pseudo-labels contain erroneous labels. The straightforward approach is to randomly select anchor and positive from the same class and randomly select negative from one of the other classes. Although this strategy is simple and has low complexity: It may lead to bad local minima due to false pseudo-labels. In FaceNet [132], the authors proposed a semi-hard mining method. In semi-hard mining, the negative sample is selected to be further than the positive sample but lies in the margin: $\|f_a - f_p\|_2^2 < \|f_a - f_n\|_2^2 < \|f_a - f_p\|_2^2 + \alpha$. Note that, if the negative sample is closer to the anchor than the positive sample, there is a possibility that it may be a member of the same class as the anchor. Semi-hard selection methods contribute to noise-robustness in metric learning. Therefore, we utilize semi-hard mining method to select negative sample. False-labels in positive class also affect convergence. The metric loss function motivates that the L2 norm between every sample in different classes should be bigger or equal to the margin. If a sample is closer to anchor than a margin, it can be considered a positive sample in an ideal condition. Motivated by this, we selected the positive sample whose distance to the anchor is smaller than half of the margin to improve positive sample confidence. In order to avoid bias towards easy-to-transfer classes, the anchor sample is selected randomly from the least selected class so far.

4.2.2 Pseudo-label generation

Class-proxy is the feature vector representing class distribution in metric space. One method to compute proxy features of the classes is to take the mean of all feature vectors for each class after training. However, it requires the inference of all samples in the dataset after training, which is computationally expensive. In order to reduce the computation time of our method, we propose an online proxy calculation method. The proxy features of each class are updated during training on each iteration. For a given image x_T, the feature vector set for each class is selected with respect to the pseudo-labels used for metric learning. Naturally, some premature or erroneous features exist during the training, which can be considered as outlier. In order to eliminate them, we calculate z-score for each feature in metric space, defined as follows:

$$z^i = \frac{f^i - \mu^i}{\sigma^i}, \tag{4.2}$$

where μ and σ are the element-wise mean and standard deviation of features that belong to the same class with f, respectively. We use the arithmetic mean of absolute values of z, denoted as $\bar{z}$, to decide which samples are the outlier. Then instance proxy is calculated by taking the element-wise arithmetic mean of reliable features whose $\bar{z}$ value is below threshold θ_z. The element-wise exponential moving average of instance proxies is used for updating proxy values during training to calculate final class-proxies.

Pseudo-labels are selected based on the L2 distance between the feature of a pixel and the proxy of the predicted class. Similar to confidence-based methods, we use the class-wise metric threshold. Threshold values are also computed during training with an exponential moving average of instance thresholds. Instance thresholds are calculated so that θ_M percent of all pixels are selected for each class. If the distance is smaller than the threshold, the prediction is accepted as reliable and used as a pseudo-label.

4.2.3 Adversarial alignment

For segmentation, the source domain S is defined by a source image space X_S and the corresponding label space Y_S. The segmentation network takes an image $x_S \in X_S \subset \mathbb{R}^{H \times W \times 3}$ with its associated label $y_S \in Y_S \subset \mathbb{R}^{H \times W \times C}$, where y_S, H, W, and C correspond to a one-hot vector for each pixel in the image, the image height, the image width and the number of classes, respectively. The feature extractor generates feature representation for the input image, and the classifier network estimates soft-segmentation map corresponding class prediction probability for each class for each pixel. In accordance with the Figure 4.2, the segmentation network can be represented as $G(x_S) = C(F(x_S))$.

For a given image-label set from the source domain, G can be optimized in a supervised manner by cross-entropy loss:

$$\mathcal{L}_{sup}(x_S, y_S) = -\sum_{h,w}\sum_{c} y_S^{(h,w,c)} \log(G(x_S)^{(h,w,c)}). \tag{4.3}$$

Supervised training in the source domain is practical if both train and test data are sampled from the same distribution. However, this is not the case for several real-life applications due to the domain gap between train and test data. To address this problem, AdvEnt [154] proposed adversarial training on entropy maps. They observed that the trained model produces low-entropy predictions for source-like images. To minimize entropy in the target domain, they proposed adversarial training on weighted self-information I, where $I^{(w,h,c)} = -P^{(w,h,c)} \log P^{(w,h,c)}$. In adversarial training, a discriminator network denoted as D, is used to distinguish if the input belongs to the source or target domain, while a semantic segmentation network G tries to generate a source-like self-information map to fool the discriminator. The training objective to optimize G and D is as follows:

$$\mathcal{L}_{adv}(x_S, x_T) = -\sum_{h,w} \log(1 - D(I_s^{(h,w,c)})) + \log(D(I_T^{(h,w,c)})). \tag{4.4}$$

Self-training (ST) is a popular method in semi-supervised training. Recent studies in unsupervised domain adaptation show the efficacy of ST in this field of study.

The model is trained iteratively to achieve better alignment using pseudo-labels as a ground-truth. The commonly used loss function is pixel-wise cross-entropy loss as in supervised training from the source domain. The difference is that the pseudo-labels are used instead of ground-truth, which is not available in the target domain. In this study, we use pseudo-labels selected by similarity metric instead of prediction confidence. The self-training loss is given as follows:

$$\mathcal{L}_{st}(x_T, \hat{y}_T) = -\sum_{h,w}\sum_{c} \hat{y}_T^{(h,w,c)} \log(P_T^{(h,w,c)}). \tag{4.5}$$

Our complete loss function $\mathcal{L}$ for the semantic segmentation network (G) is composed of supervised, self-supervised, and adversarial loss. The training objective for G and M are given as:

$$\begin{aligned} G^* &= \arg\min_G \min_G \max_D \{\mathcal{L}_{sup} + \mathcal{L}_{st} + \lambda_{adv}\mathcal{L}_{adv}\} \\ M^* &= \arg\min_M \mathcal{L}_{met}. \end{aligned} \tag{4.6}$$

4.3 Experiments

In this section, we explain the experimental details and analyze the results of the proposed method.

4.3.1 Experimental Details

4.3.1.1 Datasets

In the experiments, we evaluate two popular synthetic-to-real adaptation scenarios, which are GTA5-to-Cityscapes and SYNTHIA-to-Cityscapes settings. The Cityscapes [28] is a real-world dataset composed of street-view images captured in 50 different cities. It consists of 5000 images of resolution 2048 x 1024 with high-quality pixel-level annotations. These images of street scenes were annotated with 19 semantic

labels for evaluation. This dataset is split into training, validation, and test sets with 2975, 500, and 1525 images, respectively. The UDA methods for semantic segmentation studies are evaluated on the validation set. The GTA5 [120] dataset consists of 24,966 synthetic images with pixel-level annotations of 33 categories. For training, only 19 categories compatible with the Cityscapes dataset are used. The images are the size of 1914 x 1052. SYNTHIA [122] is a dataset with synthetic images of urban scenes and pixel-wise annotations. In UDA for semantic segmentation studies, researchers adopt the SYNTHIA-RAND-CITYSCAPES subset, which contains 9,400 images of resolution 1280 x 760 compatible with the Cityscapes dataset categories. The 16 common categories with CityScapes are used for training.

4.3.1.2 Implementation Details

In our implementations, we utilize the network architectures that are used in the AdvEnt framework. Deeplab-v2 with ResNet-101 backbone network is employed for G network, where F and C correspond to ResNet-101 and ASPP modules. The sampling rate of the ASPP module is used as 6, 12, 18, and 24. ASPP module is also used for metric learning network M with the same sampling rate. For discriminator network D, the same architecture proposed in DCGAN [117] is used as in AdvEnt [154].

The model is initialized with pre-trained weights of the ImageNet dataset. Before training, we apply semantically regularized CycleGAN [192] based image translation to source images, as proposed in BDL [88], for additional performance gain. The translated images can be found in the BDL authors' repository[1].

The Pytorch framework [111] is employed in all implementations. Experiments are done on a single NVIDIA RTX2080TI GPU with 11 GB memory. To optimize the G network, SGD [8] optimizer is used with learning rate $2.5x10^{-4}$, weight decay 10^{-4}, and momentum 0.9. The Adam optimizer is used to train C and M networks with learning rates 10^{-4} and 10^{-3}, respectively. λ_{adv} is fixed to 0.001. All learning rates are scheduled with a polynomial annealing procedure. The G and D networks are trained for 120,000 iterations. The M network is trained for 5,000 iterations. The

[1] https://github.com/liyunsheng13/BDL

models are trained with a batch size of 1. The metric learning hyper-parameters, feature size N, batch size K, margin α, z-score threshold θ_z are set to 128, 32, 1, and 0.5, respectively. In order to train the metric network, the normalized class-wise confidence threshold proposed in CBST[196] is used. Thresholds are determined so that it equals to θ_T percentile of all prediction per class. θ_T is selected as 0.2 and increased 0.05 at each iteration, following the procedure in CBST [196]. Instance metric threshold percentile θ_M is selected as $\theta_T/2$.

4.3.2 Results

We compare our method with the state-of-the-art methods on two challenging synthetic-to-real domain adaptation tasks: GTAV-to-CityScapes and SYNTHIA-to-CityScapes. The results obtained with DeepLabv2 [16] architecture with a Resnet-101 backbone are reported for a fair comparison. We report the results in Table 4.1 and 4.2 in per-class IoU and mean IoU.

AdaptSegNet [146], SIBAN [101], CLAN [102], AdvEnt [154], AllAboutStructure [13], Intra-domain [108], and CrCDA [68] methods focus on adversarial alignment. CBST [196], CRST [195], MaxSquare [18], and LSE+FL [135] methods use self-training as a supervisory signal. SSF-DAN [32] and PatchAlign [147] methods use adversarial training and self-training approaches together. BDL [88] uses an image-translation module in addition to adversarial training and self-training to further reduce the domain gap.

Table 4.1: Comparison with state-of-the-art on GTAV-to-CityScapes in terms of per-class IoUs and mIoU (%).

Method	Road	Sidewalk	Building	Wall	Fence	Pole	Light	Sign	Veg.	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Mbike	Bicycle	mIoU
AdaptSegNet[146]	86.5	36.0	79.9	23.4	23.3	23.9	35.2	14.8	83.4	33.3	75.6	58.5	27.6	73.7	32.5	35.4	3.9	30.1	28.1	42.4
SIBAN[101]	88.5	35.4	79.5	26.3	24.3	28.5	32.5	18.3	81.2	40.0	76.5	58.1	25.8	82.6	30.3	34.4	3.4	21.6	21.5	42.6
CLAN[102]	87.0	27.1	79.6	27.3	23.3	28.3	35.5	24.2	83.6	27.4	74.2	58.6	28.0	76.2	33.1	36.7	6.7	31.9	31.4	43.2
AllAboutStructure[13]	91.5	47.5	82.5	31.3	25.6	33.0	33.7	25.8	82.7	28.8	82.7	62.4	30.8	85.2	27.7	34.5	6.4	25.2	24.4	45.4
SSF-DAN[32]	90.3	38.9	81.7	24.8	22.9	30.5	37.0	21.2	84.8	38.8	76.9	58.8	30.7	85.7	30.6	38.1	5.9	28.3	36.9	45.4
ADVENT[154]	89.4	33.1	81.0	26.6	26.8	27.2	33.5	24.7	83.9	36.7	78.8	58.7	30.5	84.8	38.5	44.5	1.7	31.6	32.4	45.5
CBST[196]	91.8	53.5	80.5	32.7	21.0	34.0	28.9	20.4	83.9	34.2	80.9	53.1	24.0	82.7	30.3	35.9	16.0	25.9	**42.8**	45.9
Intra-domain[108]	90.6	36.1	82.6	29.5	21.3	27.6	31.4	23.1	85.2	39.3	80.2	59.3	29.4	**86.4**	33.6	**53.9**	0.0	32.7	37.6	46.3
MaxSquare[18]	89.4	43.0	82.1	30.5	21.3	30.3	34.7	24.0	**85.3**	39.4	78.2	**63.0**	22.9	84.6	36.4	43.0	5.5	**34.7**	33.5	46.4
PatchAlign[147]	92.3	51.9	82.1	29.2	25.1	24.5	33.8	33.0	82.4	32.8	82.2	58.6	27.2	84.3	33.4	46.3	2.2	29.5	32.3	46.5
CRST[195]	91.0	55.4	80.0	33.7	21.4	**37.3**	32.9	24.5	85.0	34.1	80.8	57.7	24.6	84.1	27.8	30.1	**26.9**	26.0	42.3	47.1
LSE + FL[135]	90.2	40.0	83.5	31.9	26.4	32.6	38.7	37.5	81.0	34.2	**84.6**	61.6	**33.4**	82.5	32.8	45.9	6.7	29.1	30.6	47.5
BDL[88]	91.0	44.7	84.2	34.6	27.6	30.2	36.0	36.0	85.0	**43.6**	83.0	58.6	31.6	83.3	35.3	49.7	3.3	28.8	35.6	48.5
CrCDA[68]	92.4	55.3	82.3	31.2	**29.1**	32.5	33.2	35.6	83.5	34.8	84.2	58.9	32.2	84.7	**40.6**	46.1	2.1	31.1	32.7	48.6
SPLS	**93.6**	59.4	84.8	34.0	22.8	33.2	41.2	44.5	83.9	39.2	82.4	61.5	32.6	84.7	34.9	49.7	0.0	25.9	39.3	49.9
SPLS-MST	**93.6**	**59.6**	**85.9**	**35.6**	25.1	34.4	**44.7**	**45.3**	84.8	41.2	83.5	62.9	33.1	84.7	37.4	50.4	0.0	27.6	40.6	**51.1**

Table 4.2: Comparison with state-of-the-art on SYNTHIA-to-CityScapes in terms of per-class IoUs and mIoU (%). The mIoU* column denotes the mean IoU over 13 categories excluding those marked by *.

Method	Road	Sidewalk	Building	Wall*	Fence*	Pole*	Light	Sign	Veg.	Sky	Person	Rider	Car	Bus	Mbike	Bicycle	mIoU	mIoU*
SIBAN[101]	82.5	24.0	79.4	-	-	-	16.5	12.7	79.2	82.8	58.3	18.0	79.3	25.3	17.6	25.9	-	46.3
PatchAlign[147]	82.4	38.0	78.6	8.7	0.6	26.0	3.9	11.1	75.5	84.6	53.5	21.6	71.4	32.6	19.3	31.7	40.0	46.5
AdaptSegNet[146]	84.3	42.7	77.5	-	-	-	4.7	7.0	77.9	82.5	54.3	21.0	72.3	32.2	18.9	32.3	-	46.7
CLAN[102]	81.3	37.0	80.1	-	-	-	16.1	13.7	78.2	81.5	53.4	21.2	73.0	32.9	22.6	30.7	-	47.8
ADVENT[154]	85.6	42.2	79.7	8.7	0.4	25.9	5.4	8.1	80.4	84.1	57.9	23.8	73.3	36.4	14.2	33.0	41.2	48.0
MaxSquare[18]	82.9	40.7	80.3	10.2	0.8	25.8	12.8	18.2	**82.5**	82.2	53.1	18.0	79.0	31.4	10.4	35.6	41.5	48.2
AllAboutStructure[13]	**91.7**	**53.5**	77.1	2.5	0.2	27.1	6.2	7.6	78.4	81.2	55.8	19.2	82.3	30.3	17.1	34.3	41.5	48.8
CBST[196]	68.0	29.9	76.3	10.8	1.4	33.9	22.8	29.5	77.6	78.3	60.6	28.3	81.6	23.5	18.8	39.8	42.6	48.8
Intra-domain[108]	84.3	37.7	79.5	5.3	0.4	24.9	9.2	8.4	80.0	84.1	57.2	23.0	78.0	38.1	20.3	36.5	41.7	48.9
LSE + FL[135]	82.9	43.1	78.1	9.3	0.6	28.2	9.1	14.4	77.0	83.5	58.1	25.9	71.9	38.0	**29.4**	31.2	42.5	49.4
SSF-DAN[32]	84.6	41.7	80.8	-	-	-	11.5	14.7	80.8	85.3	57.5	21.6	82.0	36.0	19.3	34.5	-	50.0
CrCDA[68]	86.2	44.9	79.5	8.3	0.7	27.8	9.4	11.8	78.6	**86.5**	57.2	26.1	76.8	39.9	21.5	32.1	43.0	50.0
CRST[195]	67.7	32.2	73.9	10.7	**1.6**	**37.4**	22.2	**31.2**	80.8	80.5	**60.8**	**29.1**	**82.8**	25.0	19.4	45.3	43.8	50.1
BDL[88]	86.0	46.7	80.3	-	-	-	14.1	11.6	79.2	81.3	54.1	27.9	73.7	**42.2**	25.7	45.3	-	51.4
SPLS	82.1	42.3	80.3	20.0	0.8	28.3	28.3	24.4	80.5	84.9	51.7	18.4	76.3	31.4	10.2	45.5	44.1	50.5
SPLS-MST	81.4	41.9	**82.2**	**21.8**	0.9	33.1	**28.4**	23.8	81.9	86.1	53.7	17.8	77.3	38.2	11.9	**47.2**	**45.5**	**51.7**

Table 4.3: Results of ablation study on the GTA5-to-CityScapes

Method	AT	IT	ST_{conf}	ST_{sim}	$mIOU$	Δ
Source	-	-	-	-	36.6	
AdvEnt	$\surd$	-	-	-	43.1	+6.5
$\text{AdvEnt}_{\text{I2I}}$	$\surd$	$\surd$	-	-	45.7	+2.6
$\text{SPLS}_{\text{conf}}$	$\surd$	$\surd$	$\surd$	-	48.5	+2.8
SPLS	$\surd$	$\surd$	-	$\surd$	49.9	+4.2

Our method exploits image translation, adversarial alignment, and self-training, similar to the BDL method. Unlike BDL, we use adversarial entropy minimization proposed in AdvEnt[154] instead of prediction alignment. To generate pseudo-labels, we introduce a novel similarity-based selection mechanism. The similarity metric is predicted with the M network, which is trained with confidence-based pseudo-labels calculated by normalized class-wise threshold proposed in CBST [196]. We do not train the image-translation network in our framework. The translated images proposed by BDL are directly used as a source domain dataset in our method. Although the translated images improve the mIoU score from 43.1% to 45.7%, meaning 2.6% improvement for GTA5, it decreases mIoU* score from 46.8% to 46.2% for the SYNTHIA dataset. To present consistent results, the translated images are also used in the SYNTHIA experiments even though it performs worse than original images. We anticipate that if original images were used in training, SPLS would perform better for SYNTHIA dataset. In the Tables 4.1 and 4.2, SPLS, and SPLS-MST rows correspond to the results of the proposed self-training via metric learning method, and its multi-scale testing results, respectively. Our method achieves superior performance compared to the state-of-the-art.

4.3.2.1 Ablation studies

We report the result of the ablation study in Table 4.3. In the table, AT, IT, ST_{conf}, ST_{sim}, $mIOU$, and Δ correspond to adversarial training, image-translation, confidence-based self-training, similarity-based self-training, mean intersection over union and relative performance gain, respectively.

Table 4.4: Results of component analysis on the GTA5-to-CityScapes

Method	OE	NM	PM	$mIOU$	Δ
SPLS	$\surd$	$\surd$	$\surd$	49.9	
- Outlier Elimination (OE)	-	$\surd$	$\surd$	49.2	- 0.7
- Neg. Mining (NM)	$\surd$	-	$\surd$	48.5	- 1.4
- Pos. Mining (PM)	$\surd$	$\surd$	-	48.4	- 1.5
- All	-	-	-	48.1	- 1.8

If G network is trained in a supervised manner using only the source dataset, it obtains 36.6% of mIoU. When an adversarial alignment technique is utilized alongside supervised learning, the model achieves 43.1% of mIoU, corresponding to 6.5% performance gain with respect to the source-only version. As an adversarial alignment technique, we utilize AdvEnt [154] in our framework. Unlike original work, we use single-stage training instead of multi-stage training, resulting 0.7% performance drop compared to original work, which is reported as 43.8% in the original study with the multi-stage setting.

If the translated images are used as source domain images, the model, which we call $\text{AdvEnt}_{\text{I2I}}$, achieves 45.7% of mIoU value, corresponding +2.6% performance gain compared to AdvEnt. In SPLS, the novel similarity-based self-training is used over the $\text{AdvEnt}_{\text{I2I}}$ baseline. To present the effectiveness of our method, the results of two different configurations, namely $\text{SPLS}_{\text{conf}}$ and SPLS, are presented in the Table 4.3. Both methods are trained using image translation, adversarial alignment, and self-training. The difference between the two models is that while the pseudo-labels used in $\text{SPLS}_{\text{conf}}$ are calculated by thresholding with the normalized class-wise confidence threshold proposed in CBST [196], the pseudo-labels that are used in SPLS are calculated by the proposed similarity-based method. Self-training using confidence-based pseudo-labels gives a 2.8% increase over $\text{AdvEnt}_{\text{I2I}}$. On the other hand, self-training using similarity-based pseudo-labels achieves 49.9% mIoU, which corresponds to a 4.2% and 1.4% increase over $\text{AdvEnt}_{\text{I2I}}$ and $\text{SPLS}_{\text{conf}}$, respectively. Besides, when we apply multi-scale testing on SPLS, the combined result accomplishes 51.1% mIoU.

Our method contains three mechanisms against label and feature noise, namely out-

Table 4.5: Class-wise and overall average accuracies of the pseudo-labels (%)

Pseudo-labels	Confidence-based	Similarity-based (Ours)	Sim. accepts, Conf. rejects
Road	99.9	99.6	99.4
Sidewalk	80.1	93	89.3
Building	99.6	98.3	97.8
Wall	70	56.4	36.4
Fence	66.1	62.1	42.5
Pole	74.7	67.6	52.1
Light	94.4	85.3	78.5
Sign	94.6	90.6	85.3
Veg.	99.7	99.6	99.3
Terrain	81.6	74	61
Sky	99.5	98.6	98.1
Person	99.4	93.9	89.9
Rider	73.9	63.8	55
Car	99.9	99.3	98.9
Truck	79.6	22.6	18.2
Bus	44.5	52.4	51.1
Train	2.6	19.2	21.5
Mbike	80.6	69.1	62.4
Bicycle	87.6	74.9	68.1
Avg. Acc.	96.6	96.6	95.9

lier elimination (OE), negative mining (NM), and positive mining (PM). In order to analyze the effect of each component, we remove one of them at a time. Random mining is used if any mining is removed, and the average of all features is used if outlier elimination is removed. The performance of each experiment setup is shown in Table 4.4. The method achieves 49.2%, 48.5%, and 48.4% for removing OE, NM and PM, respectively, while full training achieves 49.9%. If all components are removed, the performance of the method drops to 48.1%.

4.3.2.2 Analysis

Our method proposes a pseudo-label selection method that decreases the intra-domain gap in the pseudo-label selection stage, using a metric learning approach. The accuracy values of the pseudo-labels are presented in Table 4.5. The first and second rows correspond to the confidence and similarity-based pseudo-labels, respectively. The third row corresponds to the pseudo-labels that the similarity threshold accepts while the confidence threshold rejects. Because the third row's accuracy values are similar to the second, this justifies that the pseudo-labels in that areas are valid, and metric alleviates the intra-domain gap. In addition to that, even though similarity pseudo-labels' accuracy is similar to the confidence-based ones, the performance of SPLS is better than $\text{SPLS}_{\text{conf}}$. This is another indicator that the similarity-based pseudo-labels are more diverse.

In Figure 4.4, we visually compared the outputs of the methods for which pseudo-labels are selected by confidence-based and novel similarity-based approaches. The prediction corresponding to buses in the second row is a good illustration of the difference. While the low confidence predictions of the bus are eliminated by the confidence-based method, the similarity-based method successfully assigns pseudo-labels.

Recent self-training methods report that several iterations are required for maximum performance. We observe a similar pattern when the pseudo-labels are generated by a confidence-based method. $\text{SPLS}_{\text{conf}}$ method reached its maximum value on the second iteration. On the contrary, our novel similarity-based method SPLS converged on a single iteration, saving serious computational sources. Despite the lack of a

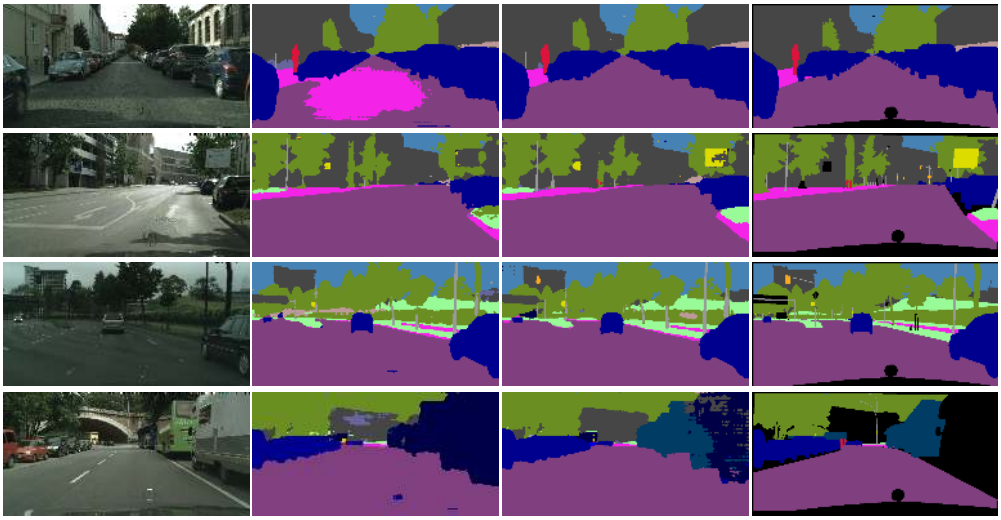


Figure 4.3: Adaptation results for the $\text{SPLS}_{\text{conf}}$ and SPLS. From left to right, columns correspond to input image, $\text{SPLS}_{\text{conf}}$ prediction, SPLS prediction and ground-truth.

theoretical guarantee, we observe the same behavior across the datasets.

In Figure 4.3, the results of SPLS is compared with $\text{SPLS}_{\text{conf}}$. SPLS performs better than $\text{SPLS}_{\text{conf}}$ in predicting confusing classes such as road/sidewalk, building/wall, and light/sign since metric learning transforms semantic feature space to class-wise clustered metric space, which creates dense clustering, leading to better supervisory signal. SPLS method provides significant performance gain, especially for sign class with 7.0% mIoU performance gain with respect to the closest method.

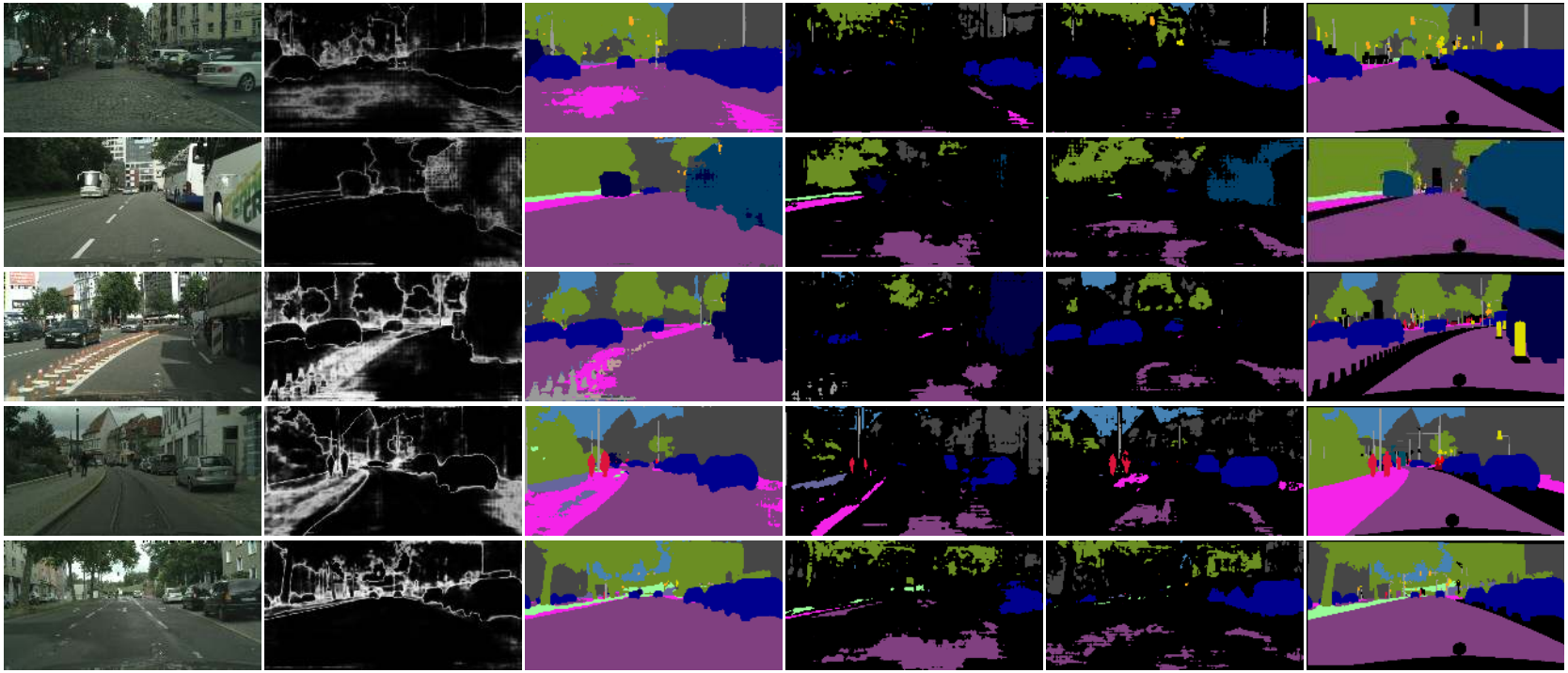


Figure 4.4: Pseudo-Label comparison. From left to right, columns correspond to input image, confidence map, $\text{AdvEnt}_{\text{I2I}}$ prediction, confidence-based pseudo-label, similarity-based pseudo-label and ground-truth.

CHAPTER 5

SELF-TRAINING VIA METRIC LEARNING FOR SOURCE-FREE DOMAIN ADAPTATION OF SEMANTIC SEGMENTATION

Thanks to advances in deep learning, computer vision has made significant progress in recent years. Deep learning networks trained in a supervised manner achieve high performance even in challenging tasks that require dense prediction, such as semantic segmentation [17, 16, 158, 189]. However, preparing a large dataset with dense labeling is very laborious and expensive [28]. An approach to decrease the workforce may be using public real [33, 28, 23] or synthetic datasets [120, 122]. However, deep learning approaches assume the IID distribution of train and test sets. Since the marginal distribution of the source domain and target domain are different, the trained model suffers from performance degradation when the model is evaluated on the target data. This problem is called the domain gap.

Recently, domain adaptation methods have been proposed to mitigate domain gap [129, 32, 146, 2, 108, 170, 191]. Such methods use labeled source domain data and unlabeled target domain data together when training the model. However, the source domain data may not be accessible due to intellectual property (IP) or privacy issues in some applications like medical imaging or autonomous driving. A possible solution to this problem is using the model trained in the source domain instead of data. The problem setting of using the source model and target data in order to decrease the domain gap is called source-free domain adaptation. In this study, we propose a method for source-free domain adaptation of semantic segmentation called Self-Training via Metric learning (STvM).

Self-training is widely used in both classical domain adaptation and source-free domain adaptation. Self-training is a training strategy in deep learning where the model

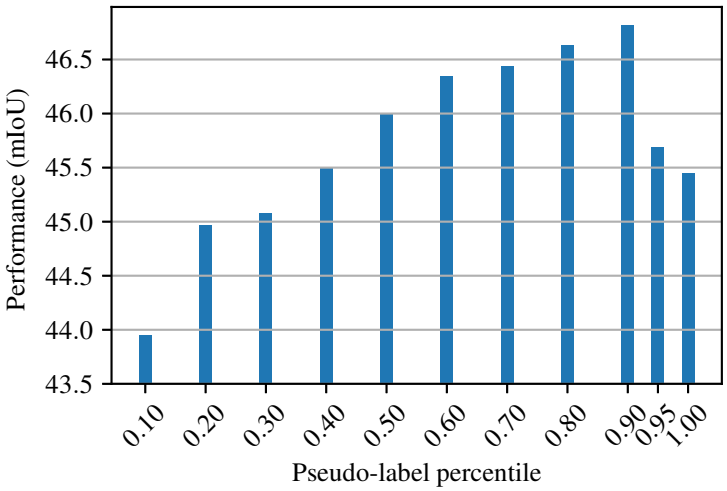


Figure 5.1: Segmentation performance of the self-training when different percentile of the predictions are used as pseudo-labels

fits the pseudo-labels predicted by itself. Most methods use prediction filtering to decide pseudo-labels [196, 195, 103, 88]. In source-free domain adaptation, the only supervision comes from the target dataset since the source dataset does not exist. Therefore, the predictions of the target images should be exploited in training as much as possible. However, using all predictions is not a good strategy since false predictions give wrong supervision to the training, which decreases performance, as shown in Figure 5.1. Therefore, instead of filtering them, we scale the gradients based on the reliability of the prediction.

Neural networks may make overconfident false predictions for data far away from training sets, such as out-of-distribution images [55]. The images belonging to the target dataset are out-of-distribution images for the model trained in the source domain. Therefore, a good measure that is valid in the target domain is needed. To this end, we propose a novel reliability measure defined in the target domain using pixel-level proxy-based metric learning. A network called a metric network is trained to predict a distance metric for each input image pixel, where pixels belonging to the same class are close to each other and far away from the pixels from the other classes. We adopt the proxy-based metric learning approach, where a proxy feature vector is trained for each class. These features can be considered as class prototypes. The

distance between the pixel feature and the proxy feature of the predicted class is used to calculate reliability.

Recently, the ClassMix [107] data augmentation technique showed its effectiveness in semi-supervised semantic segmentation tasks. It cuts half of the classes from one image and pastes on another by also transferring class labels. However, the data is unlabeled in a source-free domain adaptation setting which prevents using ClassMix directly. Therefore, we propose a metric-based online ClassMix method. In training time, we store a fixed-size patch buffer for each class separately. If the average metric distance of a patch is lower than a threshold, it is added to the patch buffer and used for mixing in training time.

In summary, our contributions are as follows:

- We propose a novel reliability metric that is directly learned in the target domain using a proxy-based metric learning approach.
- Instead of filtering the prediction to decide pseudo-labels, we use all predictions in self-training and scale the gradients based on the reliability metric.
- We propose a metric-based online ClassMix method to augment the input image.
- STvM significantly outperforms state-of-the-art methods on GTA5-to-CityScapes, SYNTHIA-to-CityScapes, and NTHU datasets.

5.1 Related works

Domain adaptation. Domain adaptation is a widely studied topic, especially for image classification [159] and semantic segmentation [143]. Adversarial learning [13, 62, 146, 20, 22, 25, 88] and self-training [18, 71, 103, 108, 185, 188, 195, 196] approaches are commonly utilized in semantic segmentation. Adversarial methods enforce the feature space of the source and target space to be aligned inspired by the GAN framework [47], where the source and the target domains are discriminated instead of real and fake samples. Adversarial learning is applied in image and feature levels. Some methods exploit the adversarial approach to the low-dimensional

output space to facilitate adversarial learning. Self-training methods use confident predictions as pseudo-labels, obtained by filtering-out noisy samples or applying constraints, and train in a supervised manner with them.

Source-free domain adaptation. Source-free domain adaptation applies domain adaptation using only the trained source model instead of the source data as well as unlabeled target data, where the source data cannot be freely shared due to Intellectual Property of privacy concerns. The task is recently introduced by two works [34, 95] concurrently. URMA [34] enforces confident predictions under the feature noise. It uses multiple decoders and introduces feature noise with dropout. The noise resilience is satisfied by uncertainty loss which is the squared difference of the decoder outputs. The training stability is improved with entropy minimization and self-training with threshold-based pseudo-labeling. SFDA [95] introduces a data-free knowledge distillation approach. It generates source domain synthetic images that preserve semantic information using batch-norm statistics of the model and dual-attention mechanism. They also use a self-supervision module to improve performance. GtA [81] categorized the SF-UDA methods as a vendor and client-side strategies, where vendor-side strategies focus on the improving source model for better adaptation. They focus on improving the vendor side performance and propose multiple augmentation techniques to train different source models using the leave-one-out technique on the vendor side.

Considering that it may not be possible to train the model on the vendor-side in source-free scenario, we propose a client-side source free domain adaptation method like SFDA and URMA, in which the source model is used as-is.

Metric learning. Deep metric learning (DML) is an approach to establish a distance metric between data to measure similarity. It learns an embedding space where similar examples are close to each other, and different examples are far away. It is a widely studied topic in computer vision that has various applications such as image retrieval [105], clustering [57], person re-identification [24] and face recognition [132]. In the unsupervised domain adaptation literature, metric learning is mostly utilized in the image classification task. The features of the semantically similar samples from both

source and target domain are aligned to mitigate the domain gap [67, 82, 114, 179]. Commonly, the similarity metric between images or patches is trained in deep metric learning. With a different perspective, Chen et al. [21] utilize pixel-wise deep metric learning for interactive object segmentation, where they model the task as an image retrieval problem. DML approaches are categorized into two classes based on the loss functions, namely pair-based and proxy-based methods. While pair-based loss functions [162, 163, 106, 134] exploit the data-to-data relations, the proxy-based loss functions [105, 142, 115, 4] exploit data-to-proxy relations. Generally, the number of proxies is substantially smaller than training data. Therefore, proxy-based methods converge faster, and the training complexity is smaller than pair-based methods. They are also more robust to label noise and outliers [77].

In STvM, we propose a proxy-based pixel-wise metric learning to estimate pseudo-label reliability, trained by limited and possibly noisy pseudo-labels.

Mixing. Mixing is an augmentation technique that combines pixels of two training images to create highly perturbed samples. It has been utilized in classification [7, 180, 63] and semantic segmentation [35, 107]. Especially in semi-supervised learning of semantic segmentation, mixing methods, such as CutMix [180] and ClassMix [107], achieved promising results. While the former cut rectangular regions from one image and paste them onto another, the latter use a binary mask belonging to some classes to cut. DACS [145] adapts ClassMix [107] for domain adaptation by mixing across domains.

5.2 Proposed method

In this section, we present our self-training via metric learning (STvM) approach for unsupervised source-free domain adaptation problem that we used for semantic segmentation. The source-free domain adaptation setting is composed of the following components. Let $\mathcal{D}_S$ is a labeled source dataset composed of source domain images x_S and corresponding pixel-wise labels y_S, $\mathcal{D}_S = \{(x_S^i, y_S^i)\}_{i=1}^{N_S}$ where N_S is the number of samples in $\mathcal{D}_S$. It is assumed that the source model is trained with $\mathcal{D}_S$ in a supervised manner so that it performs well in the source domain. Let $\mathcal{D}_T$ is an unla-

beled target dataset composed of target domain images, $\mathcal{D}_T = \{(x_T^i)\}_{i=1}^{N_T}$ where N_T is the number of samples in $\mathcal{D}_T$. The source-free domain adaptation trains a model using only the pretrained source domain model and an unlabeled target dataset $\mathcal{D}_T$. Our method utilizes the mean teacher model. It consists of two segmentation networks called the teacher network $\mathcal{T}$ and the student network $\mathcal{S}$. Both networks are initialized with the source model enabling knowledge transfer from the source domain to the target domain. They are trained with target dataset $\mathcal{D}_T$ in an unsupervised manner.

5.2.1 Framework overview

The proposed method is illustrated in the Figure 5.2. We utilize the mean-teacher approach to train a segmentation model. There are two data paths in STvM: One belongs to the teacher and metric networks, and the other belongs to the student.

The teacher data path is responsible for training the metric network, generating the pseudo-labels and the reliability map. An image belonging to the target dataset is fed to the teacher network. The predictions of the teacher model are used as pseudo-labels. In addition to that, the features generated by the feature extractor of the teacher network are fed to the metric network to form metric features for each pixel of the input image. The distance between the proxy feature of the predicted class and the metric feature is used as a class similarity map. A proxy feature is a vector representing the class distribution in metric space. The class similarity map is converted to a reliability score using the reverse sigmoid function. The metric network is trained with a proxy-based metric learning approach using the highly confident predictions of the teacher network.

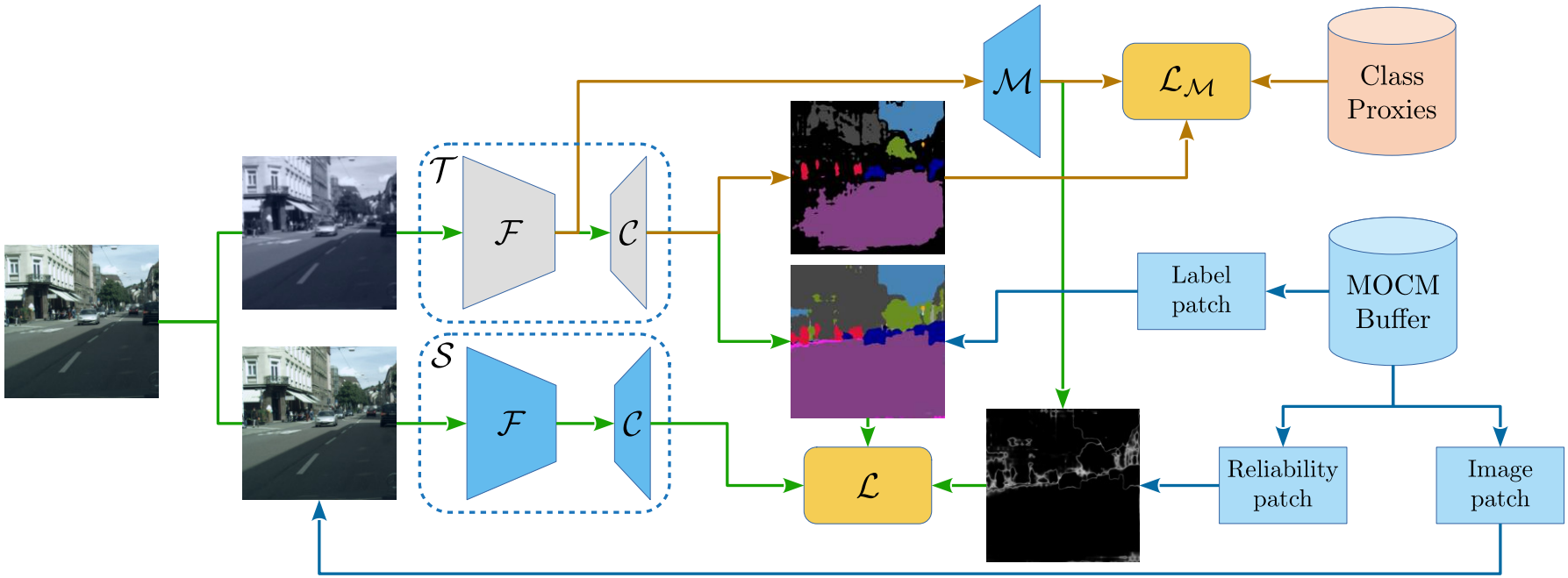


Figure 5.2: STvM comprises three networks, namely teacher, student, and metric network, represented as $\mathcal{T}$, $\mathcal{S}$, and $\mathcal{M}$, respectively. The teacher and the student network use the same segmentation network architecture. Each segmentation network is composed of feature extractor $\mathcal{F}$ and classifier $\mathcal{C}$. The metric network has the same architecture as $\mathcal{C}$, trained to learn metric feature space. Inspired by the mean-teacher approach, the student network is trained with a backpropagation algorithm. On the other hand, the parameters of the teacher model are updated with the moving average of the parameters of the student model. The metric network $\mathcal{M}$ and proxy features are also trained with a backpropagation algorithm.

The student data path is responsible for the training of the student network using pseudo-labels and reliability scores. Firstly, A photometric augmentation is applied to the input image, then Metric-based Online ClassMix (MOCM) augmentation is applied to the input image, the pseudo-labels, and the reliability map. The patches that are used in MOCM are determined and stored to the patch buffer in training time based on the metric similarity. The online patch update approach enables end-to-end training of the student model. The student model is trained with a classical pixel-wise cross-entropy loss function. However, the pixel-wise loss is multiplied with a reliability score to scale gradients which suppress erroneous pseudo-labels and allow under-confident accurate predictions to contribute to training.

5.2.2 Mean teacher with noisy student

The mean teacher method constitutes two networks called the teacher and the student networks [141]. The parameters of the teacher network are updated by moving average of the student network parameters where the student network is updated with backpropagation. The mean teacher approach can be integrated with the self-training concept by generating the pseudo-labels from the teacher network and training the student network with these pseudo-labels. In order to obtain pseudo-labels as accurately as possible, we do not apply augmentation to the input of the student network. The continuous update of the teacher network enables gradually improved pseudo-labels during training. Xie et al. [168] showed that deliberately adding noise to the student model leads to a better teacher model. Following the same principle, we applied an input noise using photometric augmentation and MOCM.

For the segmentation task, the teacher network $\mathcal{T}$ takes an image $x_T^i \in X_T \subset \mathbb{R}^{H\times W\times 3}$ and generate a class probability distribution $p_T \in P_T \subset \mathbb{R}^{H\times W\times C}$ for each pixel, where H, W, and C correspond to the image height, the image width and the number of classes, respectively. The pseudo-label map $\tilde{y} \in \mathbb{R}^{H\times W\times C}$ is a one-hot vector for each pixel, where the channel corresponding to the maximum prediction confidence is one and the others are zero.

The pixel-wise cross-entropy loss is a widely used criterion in semantic segmentation tasks. We adapted the cross-entropy loss to train the student network. Different from

the existing methods, all pseudo-labels are taken into account to train the student network. However, the gradients computed by pseudo-labels are scaled based on the label's reliability. We multiply pixel-wise loss values with the gradient scaling factor using Hadamard multiplication to scale the gradients. The gradient scaling factor $w^{(w,h)}$ is a real-valued map taking values between 0 and 1, and it is calculated based on the reliability metric, explained in section 5.2.3 in detail.

5.2.3 Reliability metric learning

The gradient scaling factor (w) plays an important role in the training of the student model. It has a strong effect on the parameters of the student model since it manipulates the gradients. In addition to that, the student model parameters directly affect the teacher model's performance and consequently the quality of the pseudo-labels. Therefore, It is essential to estimate a good reliability metric in the target domain to calculate the scaling factor. To this end, we proposed a novel reliability metric predicted by the metric network $\mathcal{M}$. The metric network utilizes the same architecture with the classifier $\mathcal{C}$ of the segmentation network. It takes the features of the feature extractor network $\mathcal{F}$ of the teacher network $\mathcal{T}$, and predicts a pixel-wise metric feature $f \in \mathbb{R}^{H \times W \times N_f}$, corresponding one metric feature for each input image pixel. The metric network $\mathcal{M}$ learns a transformation from the segmentation feature space to the metric feature space, where the features of the pixels belonging to the same class are pulled together, and those of the pixels belonging to different classes are pushed away.

The metric learning methods commonly use two different relations to train the model, namely pair-based and proxy-based relations. The pair-based methods use data-to-data similarity to train the network, whereas the proxy-based network uses proxy-to-data similarity. We exploit a proxy based relation in our study. The proxy feature $f_p \in \mathbb{R}^{N_f}$ is a vector that represents the distribution of a class of data points. We assign a different proxy feature to each segmentation class since we want to estimate how much the data point is associated with the predicted class. The proxy features are defined as a trainable parameter just as the metric learning network parameters, and it is trained concurrently with the metric network parameters.

Metric learning is a supervised learning method that requires class labels to be trained. Since the proxy-based method is known to be trained with a small number of samples [105, 142], we use a small subset of the predictions of the teacher network with high confidence values to train the metric network. This subset is called the metric pseudo-labels, and they are selected with class balanced thresholding strategy.

$$\tilde{y}_M^{h,w,c} = \begin{cases} 1 & p_T^{h,w,c} = \max(p_T^{h,w}) > \tau_c \\ 0 & elsewhere \end{cases} . \tag{5.1}$$

Threshold values (τ) are varied for each class. Class-balanced thresholding strategy [196] selects a certain percentile (q_M) of the prediction confidences. We choose a low percentile value to obtain highly confident predictions. Threshold values are calculated in training time with the moving average of the percentile of the prediction of the current frame.

One successful approach in proxy-based metric learning is to use neighborhood component analysis (NCA) in training, where the samples are compared against proxies. The proxy features and the metric network parameters are updated concurrently to attract the feature of a sample to the corresponding proxy feature and repel from the other proxy features. We apply a similar approach to train the metric network and proxy features. A temperature scaling is an effective method. Teh et al. [142] show that using small T values refine decision boundaries and help classify samples better. Motivated by these, we utilize NCA with temperature scaling. We select an equal number of samples in each class in $\tilde{y}_M^{h,w,c}$ to ensure balanced training. The metric network is optimized the following loss function:

$$\mathcal{L}_M = \sum_{h,w} \left[-\log \left(\frac{\exp(-d(f^{h,w}, f_p) * \frac{1}{T})}{\sum_c \exp(-d(f^{h,w}, f_{p_c}) * \frac{1}{T})} \right) \right], \tag{5.2}$$

where T is a temperature and $d(x, y)$ is a normalized squared L2-Norm:

$$d(x, y) = \sum_i \left(\frac{x_i}{\|x\|} - \frac{y_i}{\|y\|} \right)^2 \tag{5.3}$$

The same distance function is used in both training the metric network and calculating the reliability score, which is used as the gradient scaling factor. The smaller the distance gets, the more similar to the corresponding class it becomes. In order to calculate the similarity, we first feed the features of the feature extractor network of the teacher model to the metric network. The metric network outputs a metric feature for each pixel of the input image of the teacher network. Then we generate the similarity map by calculating the distance between the metric features and the proxy feature of the class predicted by the teacher network. We propose a reverse sigmoid function to transform similarity to the reliability, where α is sharpness constant, and β is offset of the sigmoid function:

$$w^{(h,w)} = \frac{1}{1 + e^{-\alpha * (\beta - d(f^{h,w}, \hat{f}_p))}} \tag{5.4}$$

5.2.4 Metric-based online ClassMix

The noise injection to the input of the student model leads to a better generalization for both the student and teacher model [168]. Data augmentation with photometric noise is a common approach in computer vision applications. Another effective augmentation technique for classification and semantic segmentation is the mixing method. The mixing methods combine pixels from two training images to create a hıghly perturbed sample. ClassMix algorithm [107] is a mixing data augmentation method that cuts half of the classes in the predicted image and pastes on the other image.

The straightforward integration of the ClassMix method would be cutting half of the classes based on the prediction of the teacher network and pasting on the input of the student network. However, such an approach would not be beneficial since the input of the student model is the same as of the teacher model's input except for the photometric noise. We propose a metric-based Online ClassMix (MOCM) method that stores reliable patches in training time using metric distance and uses these patches to mix the input data to overcome this problem.

We keep a patch buffer for each class separately. The patch buffers are fixed-sized and

employ a first-in-first-out scheme that decreases the memory requirement and ensures that always up-to-date patches are stored in the buffer. Unlike the standard ClassMix approach, we use distances predicted by the metric network to proxy features to select patches. If the average distance of the metric features to the corresponding class proxy (f_p^c) of a patch is smaller than a threshold (τ_{MOCM}), the patch is added to the buffer of the class c. The image, pseudo-label, and reliability map patch tuple is stored in the buffer since the mixing operation occurs in both image and label space.

To utilize the stored patches, we first apply a photometric noise to the input image, then apply the proposed MOCM method. We sample one patch from each N_{MOCM} classes randomly and paste them onto the image (x_{CM}), pseudo-label ($\tilde{y}_{CM}$), and reliability map (w_{MOCM}). The training of the student network is performed by the stochastic gradient descent in order to minimize the following loss function:

$$\mathcal{L} = \frac{1}{H \times W} \sum \left[\left(- \sum_{c} \tilde{y}_{MOCM}^{h,w,c} \times \log(T(x_{MOCM})^{h,w,c}) \right) \circ w_{MOCM}^{h,w} \right]. \tag{5.5}$$

5.3 Experiments

5.3.1 Datasets

We demonstrate the performance of STvM on three different source-to-target adaptation scenarios which are GTA5-to-Cityscapes, Synthia-to-Cityscapes, and Cityscapes-to-NTHU Cross-City.

The Cityscapes [28] dataset consists of 5000 real-world street-view images captured in 50 different cities. The images have high-quality pixel-level annotations with a resolution of 2048 x 1024. They are annotated with 19 semantic labels for semantic segmentation. The Cityscapes is split into training, validation, and test sets containing 2975, 500, and 1525 images, respectively. The methods are evaluated on the validation set following the standard setting used in the previous domain adaptation studies. The GTA5 [120] is a synthetic dataset where images and labels are automat-

ically grabbed from Grand Theft Auto V video game. It consists of 24,966 synthetic images with the size of 1914 x 1052. They have pixel-level annotations of 33 categories. The 19 classes compatible with the Cityscapes are used in the experiments. The Synthia[1] [122] is also a synthetic dataset. It is composed of urban scene images with pixel-level annotations. The commonly used SYNTHIA-RAND-CITYSCAPES subset contains 9,400 images with a resolution of 1280 x 760. The dataset has 16 shared categories with the Cityscapes dataset. Cross-City [23] is a real-world dataset. The images are recorded in four cities, which are Rome, Rio, Taipei, and Tokyo. The dataset has 3200 unlabeled images as a training set and 100 labeled images for the test set with the size of 2048 x 1024. Cross-City has 13 shared classes with the Cityscapes dataset.

5.3.2 Implementation details

We utilize two different segmentation networks to make a fair comparison with the previous unsupervised source-free domain adaptation for semantic segmentation methods. One is DeepLabV2 [16] with the ResNet-101 backbone, and the other is the DeepLabV3 [17] network with the ResNet–50 backbone. The same architecture is used for both the teacher ($\mathcal{T}$) and the student ($\mathcal{S}$) networks. ResNet-101 and ResNet-50 are used as feature extractor. The same architecture is used for the classifier ($\mathcal{C}$) and metric ($\mathcal{M}$) networks, which is the corresponding ASPP module of DeepLabV2 and DeepLabV3.

We perform the experiments on a single NVIDIA RTX 2080 Ti GPU using the PyTorch framework [111]. We use the same parameters in the training of both DeepLabV2 and DeepLabV3. We resize the input image to 1024×512 and cropped 512×512 patch randomly in our experiments. The batch size is set to 2.

Both the student and the teacher network is initialized with the source model which is trained in the source domain. The student network ($\mathcal{S}$) is trained with the Stochastic Gradient Descent (SGD) [8] optimizer with Nesterov acceleration. The initial learning rate is set to 2.5×10^{-4} and 2.5×10^{-3} for the feature extractor ($\mathcal{F}$) and the classifier ($\mathcal{C}$), respectively. The momentum is set to 0.9, and the weight decay is set

[1] This dataset is subject to the CC-BY-NC-SA 3.0

to 5.0×10^{-4}. The teacher network is updated once in 100 iterations with the parameters of the student network, setting the smoothing factor as 1.0×10^{-3}. The Adam [79] optimizer is utilized for training the metric network ($\mathcal{M}$) with the initial learning rate of 3.0×10^{-4}. The learning rates of both optimizers are scheduled with polynomial weight decay with the power of 0.9. All the networks are trained concurrently, enabling end-to-end training.

As for the hyper-parameters, metric feature size N_f, metric temperature T, metric pseudo-label threshold percentile q_m and the patch buffer size are set to 128, 0.25, 0.2 and 50, respectively. The metric distance to reliability transformation parameters α and β are set to 2 and 0.6. The metric distance threshold τ_{MOCM} is set to 0.8.

5.3.3 Results

We evaluated our proposed method STvM on two challenging synthetic-to-real and one cross-city domain adaptation scenarios, namely GTA5-to-CityScapes, SYNTHIA-to-CityScapes, and CityScapes-to-Cross-City settings. We compared our method with the state-of-the-art methods, containing both classical and source-free methods. The results, obtained with two network architectures, are given in Table 5.1, 5.2, and 5.3. MST corresponds to multi-scale testing.

Classical domain adaptation methods, utilizing the labeled source-domain dataset alongside the unlabeled target dataset, gain the advantage of better convergence than source-free domain adaptation methods. Therefore, they usually perform better than source-free settings.

STvM is a source-free domain adaptation method. The experimental results show that our method outperforms the state-of-the-art methods in the same class by a large margin. Specifically, It improves the performance of SRDA by 20% on the GTA5-to-CityScapes dataset with DeepLabV2, SFDA by %14 on SYNTHIA-to-CityScapes dataset with DeepLabV3, and SFDA by 13% on the GTA5-to-CityScapes dataset with DeepLabV3. We observed that metric learning is a powerful tool for discriminating confusing classes such as building/wall and light/sign. In addition, STvM shows better or comparable performance with the classical domain adaptation methods.

Table 5.1: Comparison with state-of-the-art on GTAV-to-CityScapes in terms of per-class IoUs and mIoU (%). SF represents if the method is in the source-free setting.

Method	Network	SF	Road	Sidewalk	Building	Wall	Fence	Pole	Light	Sign	Veg.	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Mbike	Bicycle	mIoU
AdvEnt [154]	DeepLabV2	✗	89.4	33.1	81.0	26.6	26.8	27.2	33.5	24.7	83.9	36.7	78.8	58.7	30.5	84.8	38.5	44.5	1.7	31.6	32.4	45.5
Intra-domain [108]		✗	90.6	37.1	82.6	30.1	19.1	29.5	32.4	20.6	85.7	40.5	79.7	58.7	31.1	86.3	31.5	48.3	0.0	30.2	35.8	45.8
MaxSquare [18]		✗	89.4	43.0	82.1	30.5	21.3	30.3	34.7	24.0	85.3	39.4	78.2	63.0	22.9	84.6	36.4	43.0	5.5	34.7	33.5	46.4
LSE + FL [135]		✗	90.2	40.0	83.5	31.9	26.4	32.6	38.7	37.5	81.0	34.2	84.6	61.6	33.4	82.5	32.8	45.9	6.7	29.1	30.6	47.5
BDL [88]		✗	91.0	44.7	84.2	34.6	27.6	30.2	36.0	36.0	85.0	43.6	83.0	58.6	31.6	83.3	35.3	49.7	3.3	28.8	35.6	48.5
Stuff and Things [164]		✗	90.6	44.7	84.8	34.3	28.7	31.6	35.0	37.6	84.7	43.3	85.3	57.0	31.5	83.8	42.6	48.5	1.9	30.4	39.0	49.2
Texture Invariant [76]		✗	**92.9**	55.0	85.3	34.2	31.1	34.9	40.7	34.0	85.2	40.1	87.1	61.0	31.1	82.5	32.3	42.9	0.3	36.4	46.1	50.2
FDA [176]		✗	92.5	53.3	82.4	26.5	27.6	36.4	40.6	38.9	82.3	39.8	78.0	62.6	**34.4**	84.9	34.1	53.1	**16.9**	27.7	46.4	50.4
DMLC [52]		✗	92.8	**58.1**	86.2	39.7	33.1	36.3	**42.0**	38.6	85.5	37.8	**87.6**	62.8	31.7	84.8	35.7	50.3	2.0	36.8	48.0	52.1
URMA [34]		✓	92.3	55.2	81.6	30.8	18.8	**37.1**	17.7	12.1	84.2	35.9	83.8	57.7	24.1	81.7	27.5	44.3	6.9	24.1	40.4	45.1
SRDA [5]		✓	90.5	47.1	82.8	32.8	28.0	29.9	35.9	34.8	83.3	39.7	76.1	57.3	23.6	79.5	30.7	40.2	0.0	26.6	30.9	45.8
STvM (w/o MST)		✓	91.4	52.9	87.3	41.5	**33.3**	35.9	40.8	48.5	87.3	49.2	87.4	62.2	9.3	**87.1**	45.5	58.7	0.0	47.4	59.8	54.0
STvM (w/ MST)		✓	92.1	55.1	**87.6**	**45.5**	**33.3**	36.1	41.8	**48.6**	**87.9**	**51.1**	**87.6**	**63.2**	8.6	87.0	**47.8**	**59.8**	0.0	**50.1**	**60.4**	**54.9**
MinEnt [154]	DeepLabV3	✗	80.2	31.9	81.4	25.1	20.8	24.6	30.2	17.5	83.2	18.0	76.2	55.2	24.6	75.5	33.2	31.2	4.4	27.4	22.9	40.2
AdaptSegNet [146]		✗	81.6	26.6	79.5	20.7	20.5	23.7	29.9	22.6	81.6	26.7	81.2	52.4	20.2	79.1	**36.0**	28.8	**7.5**	24.7	26.2	40.5
CBST [196]		✗	84.8	41.5	80.4	19.5	22.4	24.7	30.2	20.4	83.5	29.6	82.3	54.7	25.3	79.2	34.5	32.3	6.8	29.0	**34.9**	42.9
MaxSquare [18]		✗	85.8	33.6	82.4	25.3	25.0	26.5	33.3	18.7	83.2	32.9	79.8	57.8	22.2	81.0	32.1	32.6	5.2	**29.8**	32.4	43.1
SFDA [95]		✓	84.2	39.2	82.7	27.5	22.1	25.9	31.1	21.9	82.4	30.5	85.3	58.7	22.1	80.0	33.1	31.5	3.6	27.8	30.6	43.2
STvM (w/o MST)		✓	90.3	50.2	87.4	37.9	**33.0**	35.8	45.2	48.5	85.7	**44.1**	86.1	62.4	29.8	84.3	30.2	50.0	0.6	7.4	0.0	47.8
STvM (w/ MST)		✓	**90.8**	**50.7**	**87.7**	**40.9**	32.0	**36.1**	**47.2**	**48.6**	**85.9**	43.3	**87.0**	**62.9**	**31.5**	**85.2**	34.6	**54.0**	0.6	7.8	0.0	**48.8**

Table 5.2: Comparison with state-of-the-art on SYNTHIA-to-CityScapes in terms of per-class IoUs and mIoU (%). The mIoU* column denotes the mean IoU over 13 categories excluding those marked by *. SF represents if the method is in the source-free setting.

Method	Network	SF	Road	Sidewalk	Building	Wall*	Fence*	Pole*	Light	Sign	Veg.	Sky	Person	Rider	Car	Bus	Mbike	Bicycle	mIoU	mIoU*
AdvEnt [154]	DeepLabV2	✗	85.6	42.2	79.7	8.7	0.4	25.9	5.4	8.1	80.4	84.1	57.9	23.8	73.3	36.4	14.2	33.0	41.2	48.0
MaxSquare [18]		✗	82.9	40.7	80.3	10.2	0.8	25.8	12.8	18.2	82.5	82.2	53.1	18.0	79.0	31.4	10.4	35.6	41.5	48.2
Intra-domain [108]		✗	84.3	37.7	79.5	5.3	0.4	24.9	9.2	8.4	80.0	84.1	57.2	23.0	78.0	38.1	20.3	36.5	41.7	48.9
Texture Invariant [76]		✗	**92.6**	**53.2**	79.2	-	-	-	1.6	7.5	78.6	84.4	52.6	20.0	82.1	34.8	14.6	39.4	-	49.3
LSE + FL [135]		✗	82.9	43.1	78.1	9.3	0.6	28.2	9.1	14.4	77.0	83.5	58.1	25.9	71.9	38.0	29.4	31.2	42.5	49.4
BDL [88]		✗	86.0	46.7	80.3	-	-	-	14.1	11.6	79.2	81.3	54.1	27.9	73.7	42.2	25.7	45.3	-	51.4
Stuff and Things [164]		✗	83.0	44.0	80.3	-	-	-	17.1	15.8	80.5	81.8	59.9	**33.1**	70.2	37.3	28.5	45.8	-	52.1
FDA [176]		✗	79.3	35.0	73.2	-	-	-	19.9	24.0	61.7	82.6	**61.4**	31.1	83.9	40.8	**38.4**	51.1	-	52.5
DMLC [52]		✗	**92.6**	52.7	81.3	8.9	**2.4**	28.1	13.0	7.3	83.5	85.0	60.1	19.7	84.8	37.2	21.5	43.9	45.1	52.5
URMA [34]		✓	59.3	24.6	77.0	**14.0**	1.8	31.5	18.3	32.0	83.1	80.4	46.3	17.8	76.7	17.0	18.5	34.6	39.6	45.0
STvM (w/o MST)		✓	71.7	31.0	83.7	0.2	0.1	35.8	34.7	37.9	84.8	87.5	53.6	23.0	85.8	46.7	26.9	52.6	47.2	55.4
STvM (w/ MST)		✓	71.6	31.1	**84.2**	0.0	0.0	**36.7**	**36.0**	**38.4**	**85.4**	**87.8**	55.4	23.5	**85.9**	**47.8**	29.1	**53.6**	**47.9**	**56.1**
MinEnt[154]	DeepLabV3	✗	78.2	39.6	81.9	4.3	0.2	26.2	2.2	4.1	81.1	87.7	37.7	7.2	75.8	24.9	4.6	25.1	36.3	42.3
AdaptSegNet [146]		✗	79.7	38.6	79.3	5.6	0.8	25.4	3.6	5.5	80.0	85.4	40.8	11.7	79.8	21.4	5.2	30.5	37.1	43.2
CBST [196]		✗	81.4	44.2	80.4	7.9	0.7	25.6	5.2	12.4	81.4	89.5	39.7	10.6	82.1	21.9	6.3	32.9	38.9	45.2
MaxSquare [18]		✗	81.0	39.8	82.6	8.7	0.5	23.2	6.6	12.4	85.3	**90.1**	39.9	8.4	**84.7**	19.4	10.2	33.4	39.1	45.7
SFDA [95]		✓	**81.9**	**44.9**	81.7	4.0	0.5	26.2	3.3	10.7	**86.3**	89.4	37.9	13.4	80.6	25.6	9.6	31.3	39.2	45.9
STvM (w/o MST)		✓	49.3	20.7	84.2	14.0	1.2	33.1	42.4	45.9	82.9	82.8	50.4	22.1	81.1	38.4	21.2	46.5	44.8	51.4
STvM (w/ MST)		✓	45.9	19.8	**84.5**	**15.3**	**1.3**	**33.7**	**44.6**	**46.5**	83.3	84.0	**51.3**	**23.1**	80.9	**40.7**	**24.8**	**47.2**	**45.4**	**52.1**

Table 5.3: Comparison with state-of-the-art on CityScapes-to-NTHU in terms of mIoU (%). DL V2 and DL V3 represents DeepLabV2 with ResNet-101 backbone and DeepLabV3 with ResNet-50 backbone, respectively.

Method	Network	Rome	Rio	Tokyo	Taipei	Mean
URMA [34]	DL V2	53.8	53.5	49.8	50.1	51.8
STvM (w/o MST)		53.1	54.5	51.6	51.4	52.6
STvM (w/ MST)		**54.2**	**56.8**	**52.7**	**53.4**	**54.3**
SFDA [95]	DL V3	48.3	49.0	**46.4**	**47.2**	47.7
STvM (w/o MST)		50.8	51.0	46.1	45.0	48.2
STvM (w/ MST)		**51.2**	**52.6**	**46.4**	46.0	**49.0**

5.3.3.1 Ablation study

In order to analyze the effect of each component on the performance, we present the results of the ablation study in Table 5.4. We named the methods as $Source$, ST, ST_{Aug}, ST_{MT}, $STvM_{Raw}$, and $STvM$. $Source$ is a model trained in source-domain without any domain adaptation method is applied. ST model represents a network trained with the self-training (ST) method using all pseudo-labels. Conforming the common knowledge, the self-training boosts the performance of the source model by +6.8%. ST_{Aug}utilizes both self-training (ST) and photometric augmentation (Aug.) to the input image. Injection of the photometric noise to the self-training provides +2.1% improvement. ST_{MT} follows the mean teacher (MT) approach. All predıctıons of the teacher network are used as pseudo-labels to train the student network. While photometric augmentation is applied to the input of the student, no augmentation is applied to the input of the teacher network. Enabling mean-teacher with self-training by using all the predictions of the teacher model contributes +2.4%. As an extension to the ST_{MT}, $STvM_{Raw}$ benefits from the metric network which is used to estimate the reliability of the predictions of the teacher network. . gradient scaling (GS). The metric network that generates reliability to scale gradients of the predictions improves performance by +1.4% showing the effectiveness of the proposed method. $STvM$ also exploits the metric-based online ClassMix (MOCM). Using the metric distance in

the generation of the patches improves performance significantly. Generating patches and mixing with the input in training time using the reliability has a strong positive impact on the performance by +4.7%.

Table 5.4: Ablation Study on GTAV-to-CityScapes

Name	ST	Aug.	MT	MGS	MOCM	mIoU	Δ
$Source$	-	-	-	-	-	36.6	-
ST	✓	-	-	-	-	43.4	+6.8
ST_{Aug}	✓	✓	-	-	-	45.5	+2.1
ST_{MT}	✓	✓	✓	-	-	47.9	+2.4
$STvM_{Raw}$	✓	✓	✓	✓	-	49.3	+1.4
$STvM$	✓	✓	✓	✓	✓	**54.0**	**+4.7**

5.3.3.2 Hyper-Parameter analysis

In order to evaluate the influence of the hyper-parameters, we conduct experiments for four different values of all hyper-parameters: MOCM threshold (τ_{MOCM}), MOCM patch per image (N_{MOCM}), metric feature size (N_f), metric proxy temperature (T), metric pseudo-label quantile (q_M), reverse sigmoid alpha (α), reverse sigmoid beta (β). The experimental results are given in Table 5.5. The parameter stated in the name column is modified in the experiments. The default values are kept for the rest. We observed that no parameter has a noteworthy impact on the performance except large α and small N_{MOCM} values. Assigning large α leads to a sharp decrease in the reliability of the sample that limits the contribution of the samples that are far away to the corresponding proxy. This is a desirable situation for the perfectly trained metric network. However, the metric network is trained with highly-confident predictions of the teacher network. Therefore, the metric network is overconfident for the easy samples. If large α is selected, the student network is biased towards easy samples. That decreases diversity and has a negative impact on performance. Using a small N_{MOCM} value decreases the complexity of the augmentation, limiting the contribution of the proposed metric-based online ClassMix method.

Table 5.5: HyperParameter Analysis on GTAV-to-CityScapes

Name	Parameter Value / Performance			
τ_{MOCM}	0.4 / 52.9	0.6 / 53.07	0.8 / 54.0	1.0 / 53.0
N_{MOCM}	2 / 51.1	5 / 52.4	10 / 54.0	15 / 53.5
N_f	32 / 53.1	64 / 53.0	128 / 54.0	256 / 52.9
T	0.1 / 53.8	0.25 / 54.0	0.5 / 52.9	1.0 / 52.3
q_M	0.1 / 52.6	0.2 / 54.0	0.3 / 53.5	0.4 / 53.3
α	1 / 53.1	2 / 54.0	4 / 51.3	8 / 50.0
β	0.5 / 53.0	0.6 / 54.0	0.7 / 53.6	0.8 / 53.8

5.3.3.3 Variance analysis

We made a variance analysis for our STvM and compared it to state-of-the-art methods. Following URMA [34], we conduct five experiments with different random seeds, keeping the hyper-parameters in their default values. The mean and standard deviations obtained for GTA5-to-CityScapes with DeepLabV2 are reported in Table 5.6. Experiments show that STvM shows robust performance with the lowest standard deviation among the state-of-the-art methods.

Table 5.6: Variance Analysis on GTAV-to-CityScapes

Method	Performance Estimate	Min
AdaptSegnet	39.68 ± 1.49	37.70
ADVENT	42.56 ± 0.64	41.60
CBST	44.04 ± 0.88	42.80
UMRA	42.44 ± 2.18	39.71
STvM	**53.55 ± 0.25**	**53.26**

5.3.3.4 Metric network evaluation

The metric network clusters classes in the metric feature space, providing a reliable distance measure. The silhouette score is a widely used metric to calculate the good-

ness of the clustering method. We utilize the silhouette score to evaluate the performance of the metric network. Specifically, we computed the silhouette score [123] for each pixel of each image of the validation set. The overall mean and the class-wise mean silhouette scores of all validation set is presented in the Table 5.7. Note that the clustering performance correlates with the segmentation performance, which indicates the collaboration between the metric network and the segmentation network.

Table 5.7: Metric Evaluation

Name	Class Mean	Overall Mean
ST_{MT}	24.3	42.0
$STvM_{Raw}$	30.1	48.3
$STvM$	**33.5**	**49.1**

CHAPTER 6

CONCLUSION AND FUTURE WORK

In this theses, we propose several self-training based unsupervised domain adaptation method. We evaluated our methods in different modalities such as visible and thermal spectrum, for different tasks such as classification and semantic segmentation and in different scenarios such as classical and source-free domain adaptation.

In chapter three, we propose a self-training guided adversarial domain adaptation (SGADA) method in order to investigate the efficacy of combining visible spectrum and thermal image modalities by using unsupervised domain adaptation. To overcome the generalization problems of the current adversarial domain adaptation methods, we employ pseudo-labels obtained from a classifier trained on RGB images, and train our method with these pseudo-labels. The discriminator confidence is also utilized in the selection of pseudo-labels. In order to demonstrate results of our method, we generate a new RBG-to-thermal classification dataset by cropping the object from a large scale MS-COCO and FLIR ADAS dataset. MS-COCO dataset is used as a the source domain and thermal dataset FLIR ADAS as the target domain.

Quantitative and qualitative results show that our proposed method achieves better results than the state-of-the-art adversarial domain adaptation methods by learning more generalized feature representations for the target thermal domain. Even though existing methods show good performance in classes with many samples, SGADA shows performance improvement in rare classes. Our method outperformed other methods with respect to classwise average accuracy, indicating more balanced performance across classes. Visual results generated by t-SNE show that the learned features by SGADA are better separated, especially when classifier and discriminator confidences are used together. Quantitative results also support the benefit of using

these two confidences together. On the other hand, such an approach brings another hyper-parameter, the threshold value for discriminator confidence, which may require fine-tuning for other datasets. Our method can be improved with adaptive thresholding mechanisms.

In chapter four, we presented SPLS, which proposes a similarity-based pseudo-label selection method using metric learning for self-training in unsupervised domain adaptation of semantic segmentation. We first train a metric network to learn pixel-wise similarity metrics in the target domain so that the distance between two pixels belonging to the same class is small. Secondly, we use the similarity metric to select pseudo-labels to better utilize correct predictions with low confidence.

Comparative analyses show that using similarity metrics learned in the target domain achieves superior performance to confidence-based pseudo-label selection methods, and our method outperforms existing state-of-the-art domain adaptation methods. SPLS performs better in predicting confusing classes since the metric learning approach shows better clustering performance. In addition, unlike confidence-based methods, SPLS reached maximum performance in a single iteration in our experiments. The self-training via metric learning proposed here for UDA is a general-purpose method that can be used for any classification problem besides segmentation.

In chapter five, we proposed a self-training via metric learning (STvM) framework utilizing the mean-teacher approach for the source-free domain adaptation method of semantic segmentation. STvM learns a metric feature space in the target domain using a proxy-based metric learning technique. In training time, a reliability score is calculated for the teacher's predictions by using the distance of the metric features to the class-proxy features. The reliability score is used for scaling gradients and generating object patches. The generated patches are used to augment the input of the student network with the proposed Metric-based Online ClassMix (MOCM) method.

Unlike the standard approach in classical domain adaptation, the experimental results show that utilizing all the predictions is beneficial for self-training for source-free domain adaptation. We show that gradient scaling helps mitigate the negative impact of false positive pseudo-labels. Moreover, using learned metrics instead of confidence to calculate the scale is more beneficial. This self-training strategy also facilitates under-

confident positive samples to contribute to training. MOCM augmentation technique highly perturbs the input image, facilitating more robust training. STvM significantly outperforms state-of-the-art methods, and it is highly robust to the randomness in training.

The metric network is the essential component of the STvM. It directly impacts the performance since the gradient scaling factor, and the patch quality used in the MOCM are calculated using the metric distance. Even though we use highly reliable predictions of the teacher network, over-confident false predictions may harm the performance of the metric network. Therefore, better supervision would be beneficial. Recently, self-supervised training techniques have shown promising results. Note that metric learning does not need class labels but discriminative labels. Therefore, we believe training the metric network with self-supervised training techniques would help the overall performance.

REFERENCES

[1] İ. B. Akkaya and U. Halici. Mouse face tracking using convolutional neural networks. *IET Computer Vision*, 12(2):153–161, 2018. 1.1

[2] N. Araslanov and S. Roth. Self-supervised augmentation consistency for adapting semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15384–15394, 2021. 5

[3] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. 2.1.3

[4] N. Aziere and S. Todorovic. Ensemble deep manifold similarity learning using hard proxies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7299–7307, 2019. 5.1

[5] M. Bateson, H. Kervadec, J. Dolz, H. Lombaert, and I. B. Ayed. Source-relaxed domain adaptation for image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 490–499. Springer, 2020. 5.1

[6] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010. 2.1.3, 3, 3.1

[7] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 2019. 5.1

[8] L. Bottou. Large-scale machine learning with stochastic gradient descent. In

Proceedings of COMPSTAT'2010, pages 177–186. Springer, 2010. 4.3.1.2, 5.3.2

[9] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3722–3731, 2017. 2.1.2.2, 2.1.3, 3.1, 3.1, 3.3.3

[10] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan. Domain separation networks. In *Advances in neural information processing systems*, pages 343–351, 2016. 2.1.2.3

[11] L. Bruzzone and M. Marconcini. Domain adaptation problems: A dasvm classification technique and a circular validation strategy. *IEEE transactions on pattern analysis and machine intelligence*, 32(5):770–787, 2009. 2.1.2

[12] M. Buda, A. Maki, and M. A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 2018. 3, 3.1, 3.3.3

[13] W.-L. Chang, H.-P. Wang, W.-H. Peng, and W.-C. Chiu. All about structure: Adapting structural information across domains for boosting semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1900–1909, 2019. 2.2.2, 4.1, 4.3.2, 4.1, 4.2, 5.1

[14] O. Chapelle and A. Zien. Semi-supervised classification by low density separation. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2005. 1.1, 1.2, 3.1

[15] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. *arXiv:1412.7062 [cs]*, June 2016. arXiv: 1412.7062. 2.2.1

[16] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *arXiv:1606.00915 [cs]*, May 2017. arXiv: 1606.00915. 2.2.1, 4, 4.3.2, 5, 5.3.2

[17] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv preprint arXiv:1706.05587*, Dec. 2017. arXiv: 1706.05587. 1.1, 5, 5.3.2

[18] M. Chen, H. Xue, and D. Cai. Domain adaptation for semantic segmentation with maximum squares loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2090–2099, 2019. 2.2.2, 4.3.2, 4.1, 4.2, 5.1, 5.1, 5.2

[19] X. Chen, S. Wang, M. Long, and J. Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. volume 97 of *Proceedings of Machine Learning Research*, pages 1081–1090, Long Beach, California, USA, 09–15 Jun 2019. PMLR. 2.1.3

[20] Y. Chen, W. Li, X. Chen, and L. V. Gool. Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1841–1850, 2019. 2.2.2, 4.1, 5.1

[21] Y. Chen, J. Pont-Tuset, A. Montes, and L. Van Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1189–1198, 2018. 4.1, 5.1

[22] Y.-C. Chen, Y.-Y. Lin, M.-H. Yang, and J.-B. Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1791–1800, 2019. 2.2.2, 4.1, 5.1

[23] Y.-H. Chen, W.-Y. Chen, Y.-T. Chen, B.-C. Tsai, Y.-C. Frank Wang, and M. Sun. No more discrimination: Cross city adaptation of road scene segmenters. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1992–2001, 2017. 4.1, 5, 5.3.1

[24] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *Pro-*

ceedings of the iEEE conference on computer vision and pattern recognition, pages 1335–1344, 2016. 4.1, 5.1

[25] J. Choi, T. Kim, and C. Kim. Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6830–6840, 2019. 2.2.2, 5.1

[26] S. Chopra, S. Balakrishnan, and R. Gopalan. Dlid: Deep learning for domain adaptation by interpolating between domains. In *ICML workshop on challenges in representation learning*, volume 2, 2013. 2.1.2.1

[27] W.-S. Chu, F. De la Torre, and J. F. Cohn. Selective transfer machine for personalized facial action unit detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3515–3522, 2013. 2.1.2

[28] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 1.1, 4.3.1.1, 5, 5.3.1

[29] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 3, 3.3.2

[30] C. Devaguptapu, N. Akolekar, M. M Sharma, and V. N Balasubramanian. Borrow from anywhere: Pseudo multi-modal object detection in thermal imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019. 3, 3.1

[31] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014. 2.1.2

[32] L. Du, J. Tan, H. Yang, J. Feng, X. Xue, Q. Zheng, X. Ye, and X. Zhang. Ssf-dan: Separated semantic feature based domain adaptation network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 982–991, 2019. 2.2.2, 4.3.2, 4.1, 4.2, 5

[33] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 3, 5

[34] F. Fleuret et al. Uncertainty reduction for model adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9613–9623, 2021. 2.2.4, 5.1, 5.1, 5.2, 5.3, 5.3.3.3

[35] G. French, S. Laine, T. Aila, M. Mackiewicz, and G. Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. *British Machine Vision Conference*, 2020. 5.1

[36] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189, 2015. 2.1.2.2, 2.1.3, 3, 3.1, 3.1, 3.3.3, 3.3.3

[37] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 2.1.2.2, 2.1.3, 4.1

[38] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857*, 2017. 1.1

[39] T. Gebru, J. Hoffman, and L. Fei-Fei. Fine-grained recognition in the wild: A multi-task domain adaptation approach. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1349–1358, 2017. 2.1.2.1

[40] M. Gheisari and M. S. Baghshah. Unsupervised domain adaptation via representation learning and adaptive classifier learning. *Neurocomputing*, 165:300–311, 2015. 2.1.2

[41] M. Ghifary, W. Bastiaan Kleijn, M. Zhang, and D. Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *Proceedings of the IEEE international conference on computer vision*, pages 2551–2559, 2015. 2.1.2.3

[42] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016. 2.1.2.3

[43] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 3

[44] J. Goldberger, G. E. Hinton, S. T. Roweis, and R. R. Salakhutdinov. Neighbourhood components analysis. In *Advances in neural information processing systems*, volume 17, pages 513–520, 2005. 2.3.2

[45] B. Gong, K. Grauman, and F. Sha. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 222–230, 2013. 2.1.2

[46] R. Gong, W. Li, Y. Chen, and L. V. Gool. Dlow: Domain flow for adaptation and generalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2477–2486, 2019. 2.2.2, 4.1

[47] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 2.1.2.2, 3, 3.1, 5.1

[48] F. A. Group. Flir thermal dataset for algorithm training. `https://www.flir.com/oem/adas/adas-dataset-form/`, 2019. (document), 3, 3.3.1, 3.3.1, 3.3.1, 3.5

[49] D. Guan, X. Luo, Y. Cao, J. Yang, Y. Cao, G. Vosselman, and M. Ying Yang. Unsupervised domain adaptation for multispectral pedestrian detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019. 3

[50] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern

neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017. 2.3.2.2

[51] T. Guo, C. P. Huynh, and M. Solh. Domain-adaptive pedestrian detection in thermal images. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2019. 3, 3.1

[52] X. Guo, C. Yang, B. Li, and Y. Yuan. Metacorrection: Domain-aware meta loss correction for unsupervised domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3927–3936, 2021. 5.1, 5.2

[53] K. He, X. Zhang, S. Ren, and J. Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, Sept. 2015. 2.2.1

[54] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3, 3.3.2

[55] M. Hein, M. Andriushchenko, and J. Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 41–50, 2019. 5

[56] A. Hermans, L. Beyer, and B. Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017. 2.3

[57] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe. Deep clustering: Discriminative embeddings for segmentation and separation. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 31–35. IEEE, 2016. 4.1, 5.1

[58] M. H. Hesamian, W. Jia, X. He, and P. Kennedy. Deep learning techniques for medical image segmentation: achievements and challenges. *Journal of digital imaging*, 32(4):582–596, 2019. 1.1

[59] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2.1.2.1

[60] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006. 2.1.2.3

[61] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR, 2018. 2.1.3, 3.1, 4.1

[62] W. Hong, Z. Wang, M. Yang, and J. Yuan. Conditional generative adversarial network for structured domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1335–1344, 2018. 5.1

[63] Y. N. D. Hongyi Zhang, Moustapha Cisse and D. Lopez-Paz. mixup: Beyond empirical risk minimization. *International Conference on Learning Representations*, 2018. 5.1

[64] S. Horiguchi, D. Ikami, and K. Aizawa. Significance of softmax-based features in comparison to distance metric learning-based features. *IEEE transactions on pattern analysis and machine intelligence*, 42(5):1279–1285, 2019. 1.2

[65] J. Hu, J. Lu, and Y.-P. Tan. Deep transfer metric learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 325–333, 2015. 2.1.2.1

[66] L. Hu, M. Kan, S. Shan, and X. Chen. Duplex generative adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1498–1507, 2018. 2.1.3

[67] J. Huang, R. S. Feris, Q. Chen, and S. Yan. Cross-domain image retrieval with a dual attribute-aware ranking network. In *Proceedings of the IEEE international conference on computer vision*, pages 1062–1070, 2015. 4.1, 5.1

[68] J. Huang, S. Lu, D. Guan, and X. Zhang. Contextual-relation consistent domain adaptation for semantic segmentation. *Proceedings of the European Conference on Computer Vision*, 2020. 4.3.2, 4.1, 4.2

[69] S. Hwang, J. Park, N. Kim, Y. Choi, and I. So Kweon. Multispectral pedestrian detection: Benchmark dataset and baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3, 3.1

[70] N. Inoue, R. Furuta, T. Yamasaki, and K. Aizawa. Cross-domain weakly-supervised object detection through progressive domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5001–5009, 2018. 2.1.2.1

[71] J. Iqbal and M. Ali. Mlsl: Multi-level self-supervised learning for domain adaptation with spatially independent and semantically consistent labeling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1864–1873, 2020. 5.1

[72] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 2.1.2.2

[73] S. W. Jingjing Liu, Shaoting Zhang and D. Metaxas. Multispectral deep neural networks for pedestrian detection. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2016. 3, 3.1

[74] J. M. Johnson and T. M. Khoshgoftaar. Survey on deep learning with class imbalance. *Journal of Big Data*, 2019. 3, 3.1, 3.3.3

[75] P. Kaiser, J. D. Wegner, A. Lucchi, M. Jaggi, T. Hofmann, and K. Schindler. Learning aerial image segmentation from online maps. *IEEE Transactions on Geoscience and Remote Sensing*, 55(11):6054–6068, 2017. 1.1

[76] M. Kim and H. Byun. Learning texture invariant representation for domain adaptation of semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12975–12984, 2020. 2.2.2, 5.1, 5.2

[77] S. Kim, D. Kim, M. Cho, and S. Kwak. Proxy anchor loss for deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3238–3247, 2020. 2.3, 5.1

[78] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim. Learning to discover cross-domain relations with generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1857–1865. JMLR. org, 2017. 2.1.2.3

[79] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 3.3.2, 5.3.2

[80] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 2012. 3

[81] J. N. Kundu, A. Kulkarni, A. Singh, V. Jampani, and R. V. Babu. Generalize then adapt: Source-free domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7046–7056, 2021. 5.1

[82] I. H. Laradji and R. Babanezhad. M-adda: Unsupervised domain adaptation with deep metric learning. In *Domain Adaptation for Visual Understanding*, pages 17–31. Springer, 2020. 2.1.3, 4.1, 5.1

[83] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. In *International conference on machine learning*, pages 1558–1566. PMLR, 2016. 2.1.3

[84] L. Leal-Taixé, C. Canton-Ferrer, and K. Schindler. Learning by tracking: Siamese cnn for robust target association. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 33–40, 2016. 2.3

[85] S. Lee, D. Kim, N. Kim, and S.-G. Jeong. Drop to adapt: Learning discriminative features for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 91–100, 2019. 2.1.3, 3.1, 3.1, 3.3.3, 3.3.3, 4.1

[86] Y. Li, N. Wang, J. Liu, and X. Hou. Demystifying neural style transfer. *arXiv preprint arXiv:1701.01036*, 2017. 2.1.2.1

[87] Y. Li, N. Wang, J. Shi, J. Liu, and X. Hou. Revisiting batch normalization for practical domain adaptation. *arXiv preprint arXiv:1603.04779*, 2016. 2.1.2.1

[88] Y. Li, L. Yuan, and N. Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6936–6945, 2019. 2.2.2, 4.1, 4.3.1.2, 4.3.2, 4.1, 4.2, 5, 5.1, 5.1, 5.2

[89] Q. Lian, F. Lv, L. Duan, and B. Gong. Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6758–6767, 2019. 2.2.2

[90] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and L. Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2014. (document), 3, 3.3.1, 3.3.1, 3.5

[91] H. Liu, M. Long, J. Wang, and M. Jordan. Transferable adversarial training: A general approach to adapting deep classifiers. volume 97 of *Proceedings of Machine Learning Research*, pages 4013–4022, Long Beach, California, USA, 09–15 Jun 2019. PMLR. 2.1.3, 3, 3.1

[92] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 700–708, 2017. 3.1

[93] M.-Y. Liu and O. Tuzel. Coupled generative adversarial networks. In *Advances in neural information processing systems*, pages 469–477, 2016. 2.1.2.2

[94] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017. 2.3

[95] Y. Liu, W. Zhang, and J. Wang. Source-free domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1215–1224, 2021. 2.2.4, 5.1, 5.1, 5.2, 5.3

[96] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 2.2.1, 4

[97] M. Long, Y. Cao, J. Wang, and M. Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015. 2.1.2.1

[98] M. Long, Z. Cao, J. Wang, and M. I. Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1640–1650, 2018. 2.1.3, 3, 3.1, 3.1, 3.3.3

[99] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in neural information processing systems*, pages 136–144, 2016. 2.1.2.1, 2.1.2.1

[100] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Deep transfer learning with joint adaptation networks. In *International conference on machine learning*, pages 2208–2217, 2017. 2.1.2.1

[101] Y. Luo, P. Liu, T. Guan, J. Yu, and Y. Yang. Significance-aware information bottleneck for domain adaptive semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6778–6787, 2019. 2.2.2, 4.3.2, 4.1, 4.2

[102] Y. Luo, L. Zheng, T. Guan, J. Yu, and Y. Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2507–2516, 2019. 2.2.2, 4.1, 4.3.2, 4.1, 4.2

[103] K. Mei, C. Zhu, J. Zou, and S. Zhang. Instance adaptive self-training for unsupervised domain adaptation. *Proceedings of the European Conference on Computer Vision*, 2020. 1.1, 4, 4.1, 5, 5.1

[104] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5715–5725, 2017. 2.1.2.1

[105] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh. No fuss distance metric learning using proxies. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 360–368, 2017. 2.3.2.1, 4.1, 5.1, 5.2.3

[106] H. Oh Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4004–4012, 2016. 5.1

[107] V. Olsson, W. Tranheden, J. Pinto, and L. Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1369–1378, 2021. 5, 5.1, 5.2.4

[108] F. Pan, I. Shin, F. Rameau, S. Lee, and I. S. Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3764–3773, 2020. 2.2.2, 4.3.2, 4.1, 4.2, 5, 5.1, 5.1, 5.2

[109] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2010. 2.1.2

[110] S. Park, J. Park, S.-J. Shin, and I.-C. Moon. Adversarial dropout for supervised and semi-supervised learning, 2018. 2.1.3

[111] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017. 4.3.1.2, 5.3.2

[112] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chin-

tala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 3.3.2

[113] X. Peng and K. Saenko. Synthetic to real adaptation with generative correlation alignment networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1982–1991. IEEE, 2018. 2.1.2.1

[114] P. O. Pinheiro. Unsupervised domain adaptation with similarity learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8004–8013, 2018. 4.1, 4.1, 5.1

[115] Q. Qian, L. Shang, B. Sun, J. Hu, H. Li, and R. Jin. Softtriple loss: Deep metric learning without triplet sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6450–6458, 2019. 5.1

[116] L. Qiao, Y. Shi, J. Li, Y. Wang, T. Huang, and Y. Tian. Transductive episodic-wise adaptive metric for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3603–3612, 2019. 2.3

[117] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proceedings of International Conference on Learning Representations*, 2016. 4.3.1.2

[118] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3

[119] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015. 3

[120] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for data: Ground truth from computer games. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *Proceedings of the European Conference on Computer Vision*, volume 9906 of *LNCS*, pages 102–118. Springer International Publishing, 2016. 1.1, 4.3.1.1, 5, 5.3.1

[121] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv:1505.04597 [cs]*, May 2015. arXiv: 1505.04597. 2.2.1

[122] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 4.3.1.1, 5, 5.3.1

[123] P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987. 5.3.3.4

[124] A. Rozantsev, M. Salzmann, and P. Fua. Beyond sharing weights for deep domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 41(4):801–814, 2018. 2.1.2.1

[125] P. Russo, F. M. Carlucci, T. Tommasi, and B. Caputo. From source to target and back: Symmetric bi-directional adaptive gan. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8099–8108, 2018. 2.1.3

[126] K. Saito, Y. Ushiku, and T. Harada. Asymmetric tri-training for unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2017. 3, 3.1

[127] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2.1.3, 3.1, 3.1, 3.3.3

[128] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8503–8512, 2018. 2.1.3

[129] S. Sankaranarayanan, Y. Balaji, A. Jain, S. N. Lim, and R. Chellappa. Learning from synthetic data: Addressing domain shift for semantic segmentation. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3752–3761, 2018. 5

[130] P. Saponaro, S. Sorensen, A. Kolagunda, and C. Kambhamettu. Material classification with thermal imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3.1

[131] A. Saporta, T.-H. Vu, M. Cord, and P. Pérez. Esl: Entropy-guided self-supervised learning for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 4

[132] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 4.1, 4.2.1, 5.1

[133] J. Shen, Y. Qu, W. Zhang, and Y. Yu. Wasserstein distance guided representation learning for domain adaptation. In S. A. McIlraith and K. Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4058–4065. AAAI Press, 2018. 2.1.3

[134] K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in neural information processing systems*, pages 1857–1865, 2016. 5.1

[135] M. N. Subhani and M. Ali. Learning from scale-invariant examples for domain adaptation in semantic segmentation. *Proceedings of the European Conference on Computer Vision*, 2020. 4.3.2, 4.1, 4.2, 5.1, 5.2

[136] B. Sun and K. Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016. 2.1.2.1

[137] R. Sun, X. Zhu, C. Wu, C. Huang, J. Shi, and L. Ma. Not all areas are equal: Transfer learning for semantic segmentation via hierarchical region selection.

In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 4360–4369. Computer Vision Foundation / IEEE, 2019. 2.2.2

[138] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1199–1208, 2018. 2.3

[139] H. Tang and K. Jia. Discriminative adversarial domain adaptation. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2020. 2.1.3

[140] R. Tao, E. Gavves, and A. W. Smeulders. Siamese instance search for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1420–1429, 2016. 2.3

[141] A. Tarvainen and H. Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *arXiv preprint arXiv:1703.01780*, 2017. 5.2.2

[142] E. W. Teh, T. DeVries, and G. W. Taylor. Proxynca++: Revisiting and revitalizing proxy neighborhood component analysis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 448–464. Springer, 2020. 5.1, 5.2.3, 5.2.3

[143] M. Toldo, A. Maracani, U. Michieli, and P. Zanuttigh. Unsupervised domain adaptation in semantic segmentation: a review. *Technologies*, 8(2):35, 2020. 5.1

[144] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014. 1.1

[145] W. Tranheden, V. Olsson, J. Pinto, and L. Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1379–1389, 2021. 5.1

[146] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7472–7481, 2018. 4.1, 4.3.2, 4.1, 4.2, 5, 5.1, 5.1, 5.2

[147] Y.-H. Tsai, K. Sohn, S. Schulter, and M. Chandraker. Domain adaptation for structured output via discriminative patch representations. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1456–1465, 2019. 2.2.2, 4.3.2, 4.1, 4.2

[148] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015. 2.1.2.1, 2.1.2.2

[149] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017. 2.1.2.2, 2.1.3, 3.1, 3, 3.1, 3.2, 3.2.2, 3.3.2, 3.6, 3.1, 3.3.3, 3.3.3, 3.3.3, 4.1

[150] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014. 2.1.2.1

[151] I. Ulku and E. Akagündüz. A survey on deep learning-based architectures for semantic segmentation on 2d images. *Applied Artificial Intelligence*, pages 1–45, 2022. 1.1

[152] L. van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 2008. 3.3.3

[153] J. E. Van Engelen and H. H. Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020. 1.1, 4.1

[154] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2517–2526, 2019. 2.2.2, 4.1, 4.2.3, 4.3.1.2, 4.3.2, 4.1, 4.2, 4.3.2, 4.3.2.1, 5.1, 5.2

[155] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez. Dada: Depth-aware domain adaptation in semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7364–7373, 2019. 2.2.2

[156] F. Wang, J. Cheng, W. Liu, and H. Liu. Additive margin softmax for face verification. *IEEE Signal Processing Letters*, 25(7):926–930, 2018. 2.3

[157] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5265–5274, 2018. 2.3

[158] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 5

[159] M. Wang and W. Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018. 4.1, 5.1

[160] M. Wang and W. Deng. Deep face recognition: A survey. *Neurocomputing*, 429:215–244, 2021. 1.1

[161] X. Wang and A. Gupta. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2015. 2.3

[162] X. Wang, X. Han, W. Huang, D. Dong, and M. R. Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5022–5030, 2019. 5.1

[163] X. Wang, Y. Hua, E. Kodirov, G. Hu, R. Garnier, and N. M. Robertson. Ranked list loss for deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5207–5216, 2019. 5.1

[164] Z. Wang, M. Yu, Y. Wei, R. Feris, J. Xiong, W.-m. Hwu, T. S. Huang, and H. Shi. Differential treatment for stuff and things: A simple unsupervised

domain adaptation method for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12635–12644, 2020. 2.2.2, 5.1, 5.2

[165] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krahenbuhl. Sampling matters in deep embedding learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2840–2848, 2017. 2.3.1.1, 2.3.2

[166] T. Xiao, H. Li, W. Ouyang, and X. Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1249–1258, 2016. 2.1.2.1

[167] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang. Joint detection and identification feature learning for person search. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3415–3424, 2017. 2.3

[168] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020. 5.2.2, 5.2.4

[169] S. Xie, Z. Zheng, L. Chen, and C. Chen. Learning semantic representations for unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018. 3, 3.1

[170] H. Xu, M. Yang, L. Deng, Y. Qian, and C. Wang. Neutral cross-entropy loss based unsupervised domain adaptation for semantic segmentation. *IEEE Transactions on Image Processing*, 30:4516–4525, 2021. 5

[171] M. Xu, J. Zhang, B. Ni, T. Li, C. Wang, Q. Tian, and W. Zhang. Adversarial domain adaptation with domain mixup. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, pages 6502–6509. AAAI Press, 2020. 2.1.3

[172] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, and W. Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2272–2281, 2017. 2.1.2.1, 2.1.2.1

[173] J. Yang, W. An, S. Wang, X. Zhu, C. Yan, and J. Huang. Label-driven reconstruction for domain adaptation in semantic segmentation. *arXiv preprint arXiv:2003.04614*, 2020. 2.2.2

[174] J. Yang, W. An, C. Yan, P. Zhao, and J. Huang. Context-aware domain adaptation in semantic segmentation. *arXiv preprint arXiv:2003.04010*, 2020. 2.2.2

[175] Q. Yang, X. Wei, B. Wang, X.-S. Hua, and L. Zhang. Interactive self-training with mean teachers for semi-supervised object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5941–5950, 2021. 1.1

[176] Y. Yang and S. Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4085–4095, 2020. 2.2.2, 4.1, 5.1, 5.2

[177] Z. Yi, H. Zhang, P. Tan, and M. Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, pages 2849–2857, 2017. 2.1.2.3

[178] F. You, J. Li, L. Zhu, Z. Chen, and Z. Huang. Domain adaptive semantic segmentation without source data. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 3293–3302, 2021. 2.2.4

[179] B. Yu, T. Liu, M. Gong, C. Ding, and D. Tao. Correcting the triplet selection bias for triplet loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 71–87, 2018. 4.1, 5.1

[180] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019. 5.1

[181] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361, 2015. 2.3

[182] N. M. Zaitoun and M. J. Aqel. Survey on image segmentation techniques. *Procedia Computer Science*, 65:797–806, 2015. 2.2.1

[183] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, 2014. 3.1

[184] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503, 2016. 1.1

[185] Q. Zhang, J. Zhang, W. Liu, and D. Tao. Category anchor-guided unsupervised domain adaptation for semantic segmentation. *arXiv preprint arXiv:1910.13049*, 2019. 5.1

[186] W. Zhang, W. Ouyang, W. Li, and D. Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3801–3809, 2018. 2.1.3, 3, 3.1

[187] Y. Zhang, H. Tang, K. Jia, and M. Tan. Domain-symmetric networks for adversarial domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2.1.3

[188] Z. Zheng and Y. Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *International Journal of Computer Vision*, 129(4):1106–1120, 2021. 5.1

[189] Z. Zhong, Z. Q. Lin, R. Bidart, X. Hu, I. B. Daya, Z. Li, W.-S. Zheng, J. Li, and A. Wong. Squeeze-and-attention networks for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13065–13074, 2020. 5

[190] Q. Zhou, Z. Feng, G. Cheng, X. Tan, J. Shi, and L. Ma. Uncertainty-aware consistency regularization for cross-domain semantic segmentation. *arXiv preprint arXiv:2004.08878*, 2020. 2.2.2

[191] W. Zhou, Y. Wang, J. Chu, J. Yang, X. Bai, and Y. Xu. Affinity space adaptation for semantic segmentation across domains. *IEEE Transactions on Image Processing*, 30:2549–2561, 2020. 1.1, 5

[192] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2.1.2.3, 2.1.3, 3.1, 4.1, 4.3.1.2

[193] X. J. Zhu. Semi-supervised learning literature survey. University of Wisconsin-Madison Department of Computer Sciences, 2005. 3, 3.1

[194] F. Zhuang, X. Cheng, P. Luo, S. J. Pan, and Q. He. Supervised representation learning: Transfer learning with deep autoencoders. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015. 2.1.2.3

[195] Y. Zou, Z. Yu, X. Liu, B. Kumar, and J. Wang. Confidence regularized self-training. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5982–5991, 2019. 1.1, 4, 4.1, 4.3.2, 4.1, 4.2, 5, 5.1

[196] Y. Zou, Z. Yu, B. Vijaya Kumar, and J. Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pages 289–305, 2018. 1.1, 4, 4.1, 4.2, 4.3.1.2, 4.3.2, 4.1, 4.2, 4.3.2, 4.3.2.1, 5, 5.1, 5.2.3, 5.1, 5.2

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Akkaya, İbrahim Batuhan

Nationality: Turkish (TC)

Date and Place of Birth: 10 August 1990, Yeşilyurt

Marital Status: Married

Phone: +31 0627962879

email: batuhan.akkaya@metu.edu.tr

EDUCATION

Middle East Technical University — *Master's Degree (M.Sc.)*

Electrical and Electronics Engineering (2013 - 2016)

Middle East Technical University — *Bachelor's Degree*

Electrical and Electronics Engineering (2008 - 2013)

Middle East Technical University — *Minor Program*

Physics (2009 - 2014) - Solid State Branch

Samsun Science High School

Science (2004 - 2008)

WORK EXPERIENCE

Navinfo Europe B.V. — *Deep Learning Researcher*

February 2022 - Present

- Responsible for conducting research on neuro inspired deep learning technologies

Aselsan Inc. — *Senior Research Engineer*

March 2019 - February 2022 (2 years 11 months)

- Responsible for conducting research on state-of-the-art computer vision technologies that ASELSAN products need

Aselsan Inc. — *Image Processing and Computer Vision Engineer*

November 2016 - March 2019 (2 years 4 months)

- Responsible for developing real-time computer vision and image processing algorithms for infrared cameras

Aselsan Inc. — *Hardware Design Engineer*

July 2013 - November 2016 (3 years 5 months)

- Responsible for PCB and PLD design for avionics systems.

PUBLICATIONS

- Toraman, Cagri, Furkan Sahinuc, Eyup Halit Yilmaz, Ibrahim Batuhan Akkaya. "Understanding Social Engagements: A Comparative Analysis of User and Text Features in Twitter" Social Network Analysis and Mining.

- Koyun, Onur Can, Reyhan Kevser Keser, Ibrahim Batuhan Akkaya, and Behçet Uğur Töreyin. "Focus-and-Detect: A small object detection framework for aerial images." Signal Processing: Image Communication 104 (2022): 116675

- Altinel, Fazil, and Ibrahim Batuhan Akkaya. "Adversarial Domain Adaptation Enhanced via Self-training." In 2021 29th Signal Processing and Communications Applications Conference (SIU), pp. 1-4. IEEE, 2021.

- Kayabasi, Alper, Kaan Karaman, and Ibrahim Batuhan Akkaya. "Comparison of distance metric learning methods against label noise for fine-grained recognition." In Automatic Target Recognition XXXI, vol. 11729, p. 117290F. International Society for Optics and Photonics, 2021.

- Akkaya, Ibrahim Batuhan, Fazil Altinel, and Ugur Halici. "Self-training Guided Adversarial Domain Adaptation For Thermal Imagery." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 4322-4331. 2021.

- Akkaya, Ibrahim Batuhan, and Ugur Halici. "GAIT: Gradient adjusted unsupervised image-to-image translation." 2020 IEEE International Conference on Image Processing (ICIP). IEEE, 2020.

- Akkaya, Ibrahim Batuhan, and Ugur Halici. "Edge sensitive unsupervised image-to-image translation." 2020 28th Signal Processing and Communications Applications Conference (SIU). IEEE, 2020.

- Karaman, Kaan, Ibrahim Batuhan Akkaya, Berkan Solmaz, and A. Aydın Alatan. "A face recognition technique by representative learning with the quadruplets." 2020 28th Signal Processing and Communications Applications Conference (SIU). IEEE, 2020.

- Karaman, Kaan, Ibrahim Batuhan Akkaya, and A. Aydın Alatan. "Metric learning with Quadruplets on non-hierarchical labeled dataset." 2020 28th Signal Processing and Communications Applications Conference (SIU). IEEE, 2020.

- Akkaya, Ibrahim Batuhan, and Kaan Karaman. "A robust technique for real-time face verification with a generative network." Real-Time Image Processing

and Deep Learning 2020. Vol. 11401. International Society for Optics and Photonics, 2020.

- Karaman, Kaan, and Ibrahim Batuhan Akkaya. “Semi-supervised adversarial training of a lightweight neural network for visual recognition.” Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies III. Vol. 11166. International Society for Optics and Photonics, 2019.

- Akkaya, İbrahim Batuhan, and Ugur Halici. “Mouse face tracking using convolutional neural networks.” IET Computer Vision 12.2 (2017): 153-161.

- Akkaya, Batuhan, et al. “Tracking mice face in video." 2016 20th National Biomedical Engineering Meeting (BIYOMUT). IEEE, 2016.