Politecnico di Torino
1859

ScuDo
Scuola di Dottorato ~ Doctoral School
WHAT YOU ARE, TAKES YOU FAR

Doctoral Dissertation
Doctoral Program in Computer and Control Engineering (35thcycle)

Learn to Generalize and Adapt across Domains in Semantic Segmentation

By

Antonio Tavera
******

Supervisors
Prof. Barbara Caputo, Supervisor
Prof. Carlo Masone, Co-Supervisor

Referees
Prof. Elisa Ricci, Università di Trento
Prof. Hedvig Kjellström, KTH Royal Institute of Technology

Politecnico di Torino
May 2023

Declaration

I hereby declare that, the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Antonio Tavera
May 2023

* This dissertation is presented in partial fulfillment of the requirements for **Ph.D. degree** in the Graduate School of Politecnico di Torino (ScuDo).

To you, wherever you are

Abstract

Artificial Intelligence (AI) is a rapidly evolving field that has the potential to transform our world in countless ways. A vital part of AI is Computer Vision, which focuses on developing systems and algorithms that can interpret and comprehend visual information from the world around us, such as through Semantic Segmentation - a technique of assigning a distinct class label to each pixel in an image, grouping all pixels that belong to the same object or region under the same label. Semantic Segmentation has many critical applications such as autonomous driving or aerial images understanding. In Autonomous Driving, it is used to accurately recognize and classify different objects and regions in images received from sensors to make informed decisions about safe navigation. In Aerial Images Analysis, it can be useful for a variety of tasks including mapping, land use planning, and disaster response to identify and map damaged infrastructure and impacted areas. Nevertheless, semantic segmentation has several limitations related to data availability and quality, such as a limited diversity in training data, lack of annotation, poor quality of annotation, and imbalanced classes. To address this challenges, the purpose of this thesis was to explore and develop solutions that would make the neural models more robust and capable of generalizing to different domains from the ones they were trained on. One way to overcome these issues is through the use of synthetic datasets, which are computer-generated images that can be generated in large quantities and do not require manual annotation. For this reason we present IDDA, the largest synthetic dataset for autonomous driving, with over 100 different scenarios that allow to assess the domain generalization capability of semantic segmentation models. However, the use of synthetic datasets can present a significant challenge when it comes to generalizing the model to real-world scenarios. Synthetic datasets lack the complexity and diversity and may not include the same types of noise, occlusions, and other factors that are present in a real-world data. To overcome this problem, domain adaptation techniques can be used. In particular, few-shot domain adaptation, which *allows for a more efficient use of real-world annotated data, may be a potential solution. The PixDA technique that we present uses a limited amount of annotated real-world data to prioritize pixel alignment based on class imbalance and network classification confidence, resulting in increased accuracy. Despite its effectiveness in self-driving scenarios, understanding aerial scenes faces additional challenges such as severe camera angle distortions and a lack of reference points. To address these challenges, advanced techniques like the new loss that we present in AIAS can be used. In this context, our HIUDA framework presents a new mixing strategy that is specifically designed for aerial images, taking into account their specific challenges and helping to prevent elements from being placed in unnatural contexts. The effectiveness of the proposed solutions is evaluated using both real-world and synthetic datasets, showing their superiority in comparison to previous methods.*

Contents

List of Figures

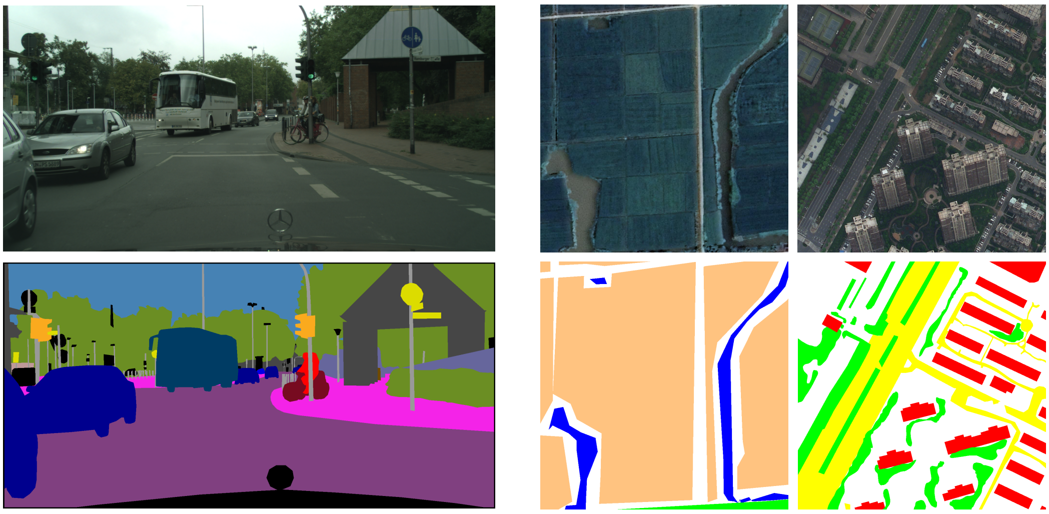
List of Tables

Chapter 1

Introduction

Artificial Intelligence, or AI, is a rapidly evolving field that has the potential to transform our world in countless ways. It refers to the ability of machines and computer programs to perform tasks that would typically require human-like intelligence, such as learning, problem-solving, decision-making, and language understanding. AI has already had a significant impact on a variety of industries, including healthcare, finance, education, and transportation, and is expected to continue to shape the future of technology and society in the years to come. One important subfield of AI is Deep Learning, which involves the use of neural networks to analyze and interpret large and complex datasets. Neural networks are inspired by the structure and function of the human brain, and are able to learn and adapt over time in order to improve their performance. Deep learning has been instrumental in the development of AI systems that are capable of tasks such as image and speech recognition with impressive accuracy.

Computer vision is a field of study within Deep Learning that focuses on the development of algorithms and systems that can understand and interpret visual data from the world around us. This includes tasks such as image recognition, object detection, and semantic segmentation. **Semantic Segmentation**, which is the main focus of this thesis, involves assigning a unique class label to each pixel in an image, such that all pixels belonging to a single object or region are assigned the same label, allowing for a more fine-grained understanding of the objects and structures in the scene (see Figure 1.1).



a) Autonomous Driving

b) Aerial Images Analysis

Fig. 1.1 Semantic segmentation is the process of accurately classify each pixel in an image to its corresponding semantic class. It has a range of applications, such as in Autonomous Driving (a) where it recognizes classes like roads, vehicles, or pedestrians, and in Aerial Image Analysis (b) where for example it differentiates between agricultural patterns, rivers, or buildings.

One important real-world application of semantic segmentation is in the field of **Autonomous Driving** (Figure 1.1a), where it is used to understand and interpret the environment around the vehicle. In order for an autonomous vehicle to safely drive, it must be able to accurately recognize and classify different objects and regions in the images it receives from its sensors. Semantic segmentation can be used to identify and label different objects such as cars, pedestrians, road signs, and traffic lights, which can be used by the vehicle to make informed decisions about how to safely navigate its environment.

Another real-world application of semantic segmentation is in the **Aerial Images Analysis** (Figure 1.1b), such as satellite or drone images. In this context, semantic segmentation can be used to identify and label different features in the images, such as roads, buildings, and natural features like rivers and forests. This can be useful for a variety of applications, including mapping, land use planning that can be used to identify and classify different land cover types in aerial images, such as forests, grasslands, and agricultural areas, and disaster response, to quickly and accurately identify and map damaged infrastructure and impacted areas. This can be useful for tasks such as damage assessment and resource allocation, and can help responders to more effectively and efficiently respond to disasters.

Nevertheless, Semantic Segmentation has several limitations related to data availability and quality, as it requires large amounts of labeled data to train robust and accurate models. More specifically they are related to:

- Limited diversity in training data: If the training data is limited or not representative of the test data, the model may not generalize well to new, unseen images or scenes. For example, if the model is trained on images of a specific city and then deployed to a different city, it may not perform well because the distribution of classes and the appearance of the objects in the new city may be different from the training data;
- Lack of annotation: Annotating large amounts of data for semantic segmentation can be time-consuming and expensive. Manually classifying each picture takes an inordinate amount of time, ranging from 60 to 90 minutes per image, like for the CamVid [1] [2] and Cityscapes [3] datasets. This can make it difficult to obtain sufficient amounts of labeled data to train accurate models.
- Quality of annotation: The bad quality of manual annotation or inconsistencies in the labels can negatively impact the performance of the model.
- Imbalanced classes: Some semantic classes may be under-represented in the dataset, making it difficult for the model to learn to recognize them. This can lead to bias in the model towards the more frequently occurring classes, and poor performance on the under-represented ones.

The main objective of this thesis was to tackle the limitations and difficulties posed by the semantic segmentation problem. The focus was on investigating and creating solutions that would enhance the robustness of the neural models, enabling them to adapt and generalize well to new domains, distinct from the ones they were originally trained on. This research aimed to make the models more versatile, so that they can effectively segment unseen data, without the need for extensive retraining.

One way to overcome the issues is through the use of synthetic datasets [4, 5]. Synthetic datasets are computer-generated images that can be generated in large quantities, allowing for the training of deep neural networks with more data. With synthetic datasets, the researcher has full control over the data, including the number of classes, the size of the objects, and the overall environment (*e.g.* the lighting conditions, the background or the point of view). Additionally, synthetic datasets do not require manual annotation, as the labels can be automatically generated by the simulation software, thus reducing the annotation cost and the annotation errors. With this premise we built **IDDA** [6], which is the largest synthetic dataset for autonomous driving, counting more than 100 different scenarios.

The use of synthetic datasets in semantic segmentation, while convenient and cost-effective, can present a significant challenge when it comes to generalizing the model to real-world scenarios. This is because synthetic datasets, while they may be able to replicate certain aspects of real-world data, often lack the complexity and diversity of real-world data, and may not include the same types of noise, occlusions, and other factors that are present in a real-world scenario. As a result, the model may become overly reliant on the specific characteristics of the synthetic dataset it was trained on, and may not be able to accurately segment real-world images that do not conform to those characteristics. One solution to this problem is to use **Domain Adaptation** techniques. These techniques aim to adapt a model trained on one source dataset (such as synthetic data) to a different but related target dataset (such as real-world data). This can be achieved through various methods such as fine-tuning, transfer learning, and adversarial training. It is important to note that even with the use of these techniques, the model may still not perform perfectly on real-world images, as it is always more diverse and complex than any synthetic dataset can replicate.

Few-shot domain adaptation in semantic segmentation could potentially be a solution to this "dataset bias". In few-shot domain adaptation, the model is trained taking into consideration also a small amount of annotated real-world data. In this case the model is less likely to become overly reliant on the specific characteristics of the synthetic dataset, and may be able to generalize better to real-world scenario, which can significantly improve its performance on unseen images. This is what is presented with the **PixDA** [7] technique that exploits a small number of annotated real-world data to prioritize the pixel alignment based on class imbalance and network classification confidence, resulting in increased accuracy, particularly for semantic classes that are underrepresented.

Although the previous solution proves successful in self-driving scenarios, comprehending aerial scenes brings extra complexities. One of the main challenges is related to the point of view from which the images are captured. Indeed, aerial images are often captured from a bird's-eye view, which can lead to severe distortions in the images due to the angle of the camera or the lack of reference points. Unlike autonomous driving, where the vehicle's perspective is always facing forward, aerial images can be captured from multiple angles and orientations, making it difficult for the model to discern between the four cardinal points. This lack of a fixed perspective can make it challenging for a segmentation model to learn a mapping that is invariant to these changes and to accurately identify and segment objects within the images. These problems require advanced techniques like the ones presented in **AIAS** [8] that introduces a novel loss to let the model be invariant to this particular shifts in perspective.

The same challenges are further intensified when dealing with class mix strategy to adapt from one domain to the other. This strategy is based on overlaying classes from the source domain onto the target image without taking into consideration the semantic hierarchy of visual elements. This can be a problem, especially when applied to the aerial scenario, as these images have lower structural consistency than driving scenes, for which these methods were originally developed. This lack of structural consistency can result in elements being placed in unnatural contexts in the mixed images. For this reason, we present a novel framework called **HIUDA**, which introduces a new mixing strategy that is specifically designed for the aerial images problem. This new strategy takes into account the spatial and contextual information of the aerial images, helping to prevent instances from being placed in unreasonable contexts and resulting in a more precise and adaptable mixing strategy.

1.1 Contribution

To summarize, the main contributions of this work are focused on addressing the Adaptation and Generalization problem in Semantic Segmentation by proposing a novel synthetic dataset and new methods that address both autonomous driving and aerial image analysis tasks. The proposed solutions are evaluated qualitatively and quantitatively to assess their performance and stability. In summary, we propose:

- **the widest synthetic dataset for semantic segmentation** [6], with over 1 million images, over 100 possible scenario combinations, and detailed pixel-by-pixel semantic annotations and depth maps. The situations are well-divided based on three variables: weather, location, and viewpoint. We also give an evaluation of the current state-of-the-art segmentation models and their domain adaptation variations, determining how helpful our dataset is for benchmarking purposes, particularly for a single-source domain adaptation assignment;

- **the first cross-domain few-shot semantic segmentation algorithm capable of learning from limited data** [7], dealing with classes that are sparsely represented in the training data by spatially aligning the domains pixel by pixel, defining a novel pixel-wise adversarial loss that locally aligns source and target domains while minimizing negative transfer and avoiding overfitting of underrepresented classes;

- **an Augmentation Invariance and an Adaptive Sampling method** [8] to improve the domain generalization capability of the state-of-the-art Semantic Segmentation models, the former of which is intended to handle the special issues presented by the perspective in the aerial data and to aid the model in distinguishing semantic information from appearance, and the latter of which is geared to handle the specific challenges presented by the problem of classes imbalance;

- **an original framework for aerial semantic segmentation called HIUDA** [9] which introduces two revolutionary ideas: a **Hierarchical Instance Mixing (HIMix)** to tackle the poor structural consistency for aerial imagery and the severe domain imbalance and, a **Twin-Head architecture** to increase pseudo-label confidence and make the model more resilient and less vulnerable to perturbations across domains.

1.2 Outline

In Chapter 2, we will define the domain adaptation problem in semantic segmentation and provide a comprehensive review of relevant works and literature. We will also introduce the datasets and metric that we will use for the experimental evaluation in this study. This chapter will delve into the unsupervised domain adaptation, few shot learning, and domain generalization problems in detail, offering a formal and unique definition for these specific scenarios. By examining these issues in depth, we aim to provide a clear understanding of the challenges and opportunities in this area of research. Additionally, this chapter will examine the various approaches that have been proposed in the literature to address these problems, and will assess their strengths and limitations.

In Chapter 3, we will delve into the details of the four works proposed in this thesis. Section 3.1 introduces IDDA, a novel, wide synthetic dataset designed to test the domain adaptation and generalization capabilities of state-of-the-art methods through more than 100 different scenarios. Section 3.2 presents PixDA, a novel, end-to-end trainable framework designed to learn from limited data and able to address the overfitting and negative transfer problems that may arise in the few-shot setting. Section 3.3 presents AIAS, which investigates the domain generalization abilities of state-of-the-art methods for autonomous driving when applied to aerial image analysis. AIAS proposes a novel loss to make the model agnostic to changes in perspective and a new training batch selection procedure to address the excessive class imbalance commonly present in this scenario. Finally, in Section 3.4, we present HIUDA, which tackles the problem of unsupervised domain adaptation in an aerial scenario by proposing a novel mixing strategy and a new architecture to provide better refined pseudo labels for the unlabeled target domain.

The conclusion of this thesis, presented in Chapter 4, will summarize the main findings and contributions of the research presented in the preceding chapters. We will discuss the significance of the proposed methods, and the implications of our results for the field of Semantic Segmentation and Domain Adaptation. We will also reflect on the limitations of our work and the challenges that remain to be addressed in future research. Finally, we will outline some potential directions for future work, including possible extensions and improvements to the proposed methods, as well as potential applications in other domains.

1.3 Publications

The following list provides a chronological summary of the author's publications; please note that some of these articles (marked with a *) are not included in this thesis, while the * means equal contribution:

- Arnaudo* E., Tavera* A., Dominici F., Masone C., Caputo B., **Hierarchical Instance Mixing across Domains in Aerial Segmentation**, IEEE Access vol.11, 2023 (Journal Paper) [9]

- Shenaj D., Fanì E., Toldo M., Caldarola D., Tavera A., Michieli U., Ciccone M., Zanuttigh P., Caputo B., **Learning Across Domains and Devices: Style-Driven Source-Free Domain Adaptation in Clustered Federated Learning**, IEEE Winter Conference on Applications of Computer Vision (WACV), 2023. (Conference Paper) [10]*

- Fantauzzo L., Fanì E., Caldarola D., Tavera A., Cermelli F., Ciccone M., Caputo B., **FedDrive: Generalizing Federated Learning to Semantic Segmentation in Autonomous Driving**, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2022. (Conference Paper) [11]*

- Tavera* A., Arnaudo* E., Masone C., Caputo B., **Augmentation Invariance and Adaptive Sampling in Semantic Segmentation of Agricultural Aerial Images**, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2022. (Conference Paper) [8]

- Cermelli* F., Fontanel* D., Tavera* A., Cicone M., Caputo B., **Incremental Learning in Semantic Segmentation from Image Labels**, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. (Conference Paper) [12]*

- Paolicelli V., Tavera A., Masone C., Berton G., Caputo B., **Learning Semantics for Visual Place Recognition through Multi-Scale Attention**, 21st International Conference on Image Analysis and Processing (ICIAP), 2021. (Conference Paper) [13]*

- Arnaudo E., Cermelli F., Tavera A., Rossi C., Caputo B., **A Contrastive Distillation Approach for Incremental Semantic Segmentation in Aerial**

Images, 21st International Conference on Image Analysis and Processing (ICIAP), 2021. (Conference Paper) [14]*

- Tavera A., Cermelli F., Masone C., Caputo B., **Pixel-by-Pixel Cross-Domain Alignment for Few-Shot Semantic Segmentation**, IEEE Winter Conference on Applications of Computer Vision (WACV), 2022. (Conference Paper) [7]

- Tavera A., Masone C., Caputo B., **Reimagine BiSeNet for Real-Time Domain Adaptation in Semantic Segmentation**, Italian Institute of Robotics and Intelligent Machines (I-RIM), 2021. (Conference Paper) [15]*

- Alberti* E., Tavera* A., Masone C., Caputo B., **IDDA: A Large-Scale Multi-Domain Dataset for Autonomous Driving**, IEEE Robotics and Automation Letters (RA-L), 2020. (Journal Paper) [6]

- Alberti* E., Tavera* A., Masone C., Caputo B., **IDDA: A Large-Scale Multi-Domain Dataset for Autonomous Driving**, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020. (Conference Paper) [6]

Chapter 2

Background and Related Work

This second chapter deals with the Domain Adaptation problem considered in the context of the Semantic Segmentation scenario. The chapter begins with a brief introduction and the definition of the problem and settings, then moves on to a review of relevant works before presenting the metric and datasets on which the proposed algorithms are evaluated.

2.1 Problem Setting and Definition

The task of semantic segmentation [16] involves assigning a semantic label to each pixel in an image, with the goal of accurately recognizing different semantic elements such as people, vehicles, or buildings. Mathematically, let us denote as $\mathscr{X}$ the set of RGB images composed by the set of pixels $\mathscr{I}$, and as $\mathscr{Y}$ the set of semantic masks associating to each pixel $i \in \mathscr{I}$ a class from the set of semantic classes $\mathscr{C}$. Each pixel in the semantic masks is assigned a class c from a pre-defined set of semantic categories. The goal is to predict a label map L, where each element L_i in the map corresponds to a specific semantic class, c_j, such that $L_i = c_j$. During training we have available a set of data $X = \{(x,y)\}$, with $x \in \mathscr{X}$ and $y \in \mathscr{Y}$.

Despite the importance of semantic segmentation, obtaining large amounts of real-world labeled data is a challenging task. As a result, models trained on one data distribution often perform poorly when deployed in a new and different distribution, leading to the phenomenon of domain shift.

To tackle this issue, Domain Adaptation has emerged as a particular subfield of Transfer Learning, focusing on bridging the gap between the *source* domain where the model was trained, and the *target* domain where it will be deployed. In this thesis, we will delve into three main settings of domain adaptation, including unsupervised domain adaptation, few-shot domain adaptation, and domain generalization.

Unsupervised domain adaptation aims to improve the performance of a model trained on a source domain with ample labeled data, when applied to a target domain with no labeled data. In this setting, the model must learn to leverage the knowledge obtained from the source domain to accurately predict labels for the target domain.

Few-shot domain adaptation focuses on adapting a model to a new domain with limited labeled data. In this scenario, the model must learn to quickly and effectively generalize to new domains using a small number of labeled examples.

Domain generalization, on the other hand, involves training a model on one or multiple domains to be able to perform well when applied to any unseen domain. In this setting, the model must learn to identify and extract common features across domains that are transferable to new domains.

Overall, domain adaptation is a crucial area of study for the field of semantic segmentation, as it allows models to generalize better to new and different distributions, improving their overall robustness and practicality.

2.1.1 Experimental Settings

Unsupervised Domain Adaptation

The Unsupervised Domain Adaptation [17] is a particular setting of domain adaptation that attempts to adapt the model to the target scenario using only the available labeled data from the source domain and the unlabeled data from the target domain. This can be done by aligning the distributions of the source and target domains, either by learning domain-invariant features or by reducing the discrepancy between the two distributions.

To tackle the Unsupervised Domain Adaptation problem in Semantic Segmentation we expand the previous definition; in this particular case, at training time we have available two sets of images: $X_s = \{(x_s, y_s)\}$ which is a collection of N_s images, with $x_s \in \mathscr{X}$ from a synthetic domain (*source*), and $X_t = \{(x_t)\}$ which contains a N_t number of samples $x_t \in \mathscr{X}$ from the real-world domain (*target*). In this notation, $y_s \in \mathscr{Y}$ denote the annotation masks associated with the source images.

The goal is to use the datasets X_s and X_t to learn a function f, parameterized by θ, from the input space $\mathscr{X}$ to a pixel-wise probability, *i.e.*, $f_\theta : \mathscr{X} \to \mathbb{R}^{|\mathscr{I}| \times |\mathscr{Y}|}$, and evaluating it on unseen images from the target domain. In the following, we indicate the model output in a pixel i for the class c as p_i^c, *i.e.*, $p_i^c(x) = f_\theta(x)[i, c]$.

Few-Shot Domain Adaptation

Few-shot Domain Adaptation, as the name suggests, involves transferring knowledge from a well-annotated source dataset to a limited target dataset with only a few labeled examples per target domain. This problem setting has more flexible constraints with respect to Unsupervised Domain Adaptation setting and is especially suited for use in the field of autonomous driving, where a single self-driving solution is typically deployed over a finite number of designated cities. In such cases, the unsupervised domain adaptation problem setting is modified to consider the cross-domain few-shot setting as defined by Zhang et al. in their work [18].

In this setting, $\mathscr{K}$-shot is defined as a task that provides $\mathscr{K}$ real images randomly selected for each of the $\mathscr{N}$ cities of the target dataset. For instance, in the 1-shot setting with Cityscapes as the target dataset, the entire target data comprises 18 annotated frames, since Cityscapes consists of 18 different cities. During training, we have two sets of images available: $X_s = (x_s, y_s)$, which is a collection of N_s images with $x_s \in \mathscr{X}$ from a synthetic source domain, and $X_t = (x_t, y_t)$, which contains a small number of samples $x_t \in \mathscr{X}$ with their corresponding labels $y_t \in \mathscr{Y}$ from the real-world target domain.

Similar to what defined in the paragraph above, the aim is to use the large dataset X_s and the limited dataset X_t to learn a function f, parameterized by θ, from the input space $\mathscr{X}$ to a pixel-wise probability, i.e., $f_\theta : \mathscr{X} \rightarrow \mathbb{R}^{|\mathscr{I}| \times |\mathscr{Y}|}$. Once the function is trained, it is evaluated on unseen images from the target domain. In particular, we indicate the model output in a pixel i for the class c as p_i^c, that is, $p_i^c(x) = f_\theta(x)[i, c]$.

Domain Generalization

Lastly, Domain Generalization is a more challenging variant of Domain Adaptation [19]. In this particular setting, the training phase operates on a set of source data $X_s = (x_s, y_s)$, consisting of N_s images with corresponding y_s labels. Unlike Domain Adaptation, there is no target data X_t available during model training. To address this issue, Domain Generalization approaches aim to train models that are highly generic and resilient so that they perform well on previously unseen domains.

During training, the goal is to learn a function f that maps inputs from X_s to a pixel-wise probability distribution. Mathematically, $f_\theta : X_s \rightarrow \mathbb{R}^{|\mathscr{I}| \times |\mathscr{Y}|}$, where f_θ is parametrized by θ. The output of the function in pixel position i for class c is denoted as p_i^c. Mathematically, $p_i^c(x) = f_\theta(x)[i, c]$.

By training on the source domain X_s, models are expected to capture the salient features of the data and generalize across previously unseen domains. The efficacy of these models is evaluated on unseen data from the target domain X_t.

2.2 Related Work

2.2.1 Semantic Segmentation

One of the earliest approaches was the use of Fully Convolutional Networks (FCN) [20], which only use convolutional layers and skip connections to incorporate semantic and appearance information from different layers of a network. Most common segmentation models, such as U-Net [21], HRNet [22] and HRNetV2 [23], employ an encoder-decoder structure to extract objects and image context at different scales. Multi-scale approaches are also used in solutions such as Feature Pyramid Networks (FPN) [24], UperNet [25] and Pyramid Scene Parsing Networks (PSPNet) [26] to improve global scene context modeling. DeepLab V2 [27] and V3 [28] use the dilation parameter of convolutional layers and introduce the Atrous Spatial Pyramid Pooling (ASPP) module to robustly segment objects at multiple scales. DeepLab V3+ [29] extends the DeepLab family by adopting an encoder-decoder structure.

Recently, the Vision Transformer (ViT) [30] has presented a transformer-based architecture for image classification that bypasses the typical convolutional neural networks. This approach processes input images as sequences of patch tokens, resulting in a novel and effective means of image classification that has since served as a foundation for further work. Following the success of ViT, researchers have explored the use of transformers for semantic segmentation as well [31, 23, 32–36]. For instance, the Segmenter [32] method proposed a transformer encoder-decoder architecture for semantic image segmentation in which the ViT backbone played a critical role. To enable the model to generate masks, Segmenter introduced a mask decoder inspired by DETR [36]. Similarly, the SegFormer [33] method presented a hierarchically structured Transformer encoder, which output multiscale features, and a lightweight decoder. MaskFormer [34] instead, addressed semantic segmentation as a mask classification problem. Most recently, OneFormer [37] emerged as the first multi-task universal image segmentation framework built upon transformer-based architecture. This particular approach requires only one training phase with a single universal architecture, making it a unique and efficient solution for multi-task image segmentation. Overall, the continued evolution and exploration of transformers-based architectures have led to novel and effective approaches for image segmentation, especially when applied to the autonomous driving task.

Semantic segmentation in aerial and remote sensing applications can be applied to a variety of environments, including urban areas [38–40], land cover [41–43], and agricultural scenarios [44–46]. These different environments often have specific challenges and requirements. For instance, urban monitoring typically involves identifying infrastructure elements like roads [47] and buildings [48], which often requires high-resolution imagery and the consideration of temporal changes [49]. Land cover mapping presents challenges such as the extreme size and visual variability of semantic categories, which can be addressed using multi-level or multi-scale feature aggregation [50] and domain adaptation techniques [42, 51]. In agricultural scenarios, traditional segmentation solutions often rely on vegetation indices such as the Normalized Difference Vegetation Index (NDVI) [52], but there is a trend towards more robust computer vision techniques such as the automated fusion of multi-spectral data [53]. Agricultural aerial images often include bands beyond the visible spectrum, such as Near-Infrared (NIR), and common deep learning approaches for jointly exploiting RGB and NIR images include duplicating input weights [46, 54] or using multi-modal fusion [50, 45].

2.2.2 Unsupervised Domain Adaptation

Some Domain Adaptation approaches in Semantic Segmentation aim to minimize the discrepancy between the source and target domains, such as by using the Maximum Mean Discrepancy (MMD) measure, that quantifies the difference between two probability distributions [55, 56]. Others exploit generative networks and image-to-image translation algorithms to generate target images conditioned on the source domain or vice versa [57–59]. The CyCADA solution [57] is based on the idea of using generative networks and adversarial training to bridge the gap between the source and target domains in a way that is both cycle-consistent and adversarially correct. In other words, the method seeks to learn a mapping from the source domain to the target domain and back again, such that the resulting images are both realistic and semantically similar to their counterparts in the opposite domain. To achieve this, the CyCADA method uses an adversarial loss to encourage the model to generate images that are indistinguishable from real images in the target domain, and a cycle-consistency loss to enforce the idea that images should be similar after being mapped from one domain to another and back again. Similarly, the DCAN [58] uses dual-convolutional layers and a global and local adaptation layer. The global adaptation layer uses a multi-class adversarial loss to align the source and target distributions, while the local adaptation layer uses a pixel-wise adversarial loss to reduce the domain shift at the pixel level. In particular, the Fourier Domain Adaptation approach [59] transform the input images from the spatial domain (i.e., the domain in which the pixels are arranged in a grid) to the Fourier domain (i.e., the frequency domain) before feeding them into the model. This transformation allows the model to learn more general, frequency-based features that are less sensitive to differences between the source and target domains, and thus better able to generalize to new data.

Some methods combine image-to-image translation with self-learning, using the model's own predictions as pseudo-labels to fine-tune and improve the model [60, 61]. Adversarial training is a popular approach for domain adaptation in semantic segmentation [62–64]. For example, the ADVENT paper [62] suggests using an entropy loss to reduce the uncertainty of predictions made on the target domain, and proposes a new adversarial training method that focuses on both entropy minimization and adapting the structure of the model from the source domain to the target domain. CLAN [63] follows a similar approach and examines the category-level joint distribution in detail, aligning each class with an adaptive adversarial loss. The weight of the adversarial loss is reduced for well-aligned features, while the adversarial force is increased for poorly aligned features.

Other methods [65–67] use self-learning techniques to generate fine pseudo-labels on the target data to fine-tune the model. CBST [66] presents a new unsupervised domain adaptation approach that involves repeatedly training a model on target data using generated pseudo labels. The author of the paper called this solution iterative self-training, which involves minimizing a latent variable loss and involves re-training the model using the generated labels. In addition they propose a class-balanced self-training method to prevent large classes from dominating the pseudo-label generation process and incorporate spatial information to improve the accuracy of the generated labels. Similarly, the IAST method proposed in [67] uses an instance adaptive self-training approach, while [65] combines self-training with curriculum domain adaptation techniques.

Several recent techniques have been developed to improve the quality of pseudo-labels in self-training approaches, including combining self-training with class mixing to mitigate the effect of domain shifts on the labels [68–70]. Augmentation through mixing has shown remarkable performance in classification as well as semantic segmentation tasks, where two training images are combined to develop a modified sample by merging their pixels. This technique can either interpolate pixel values from both images or selectively use pixels via a binary mask. CutMix [69] enhances this strategy by cutting out a rectangular area from one image and pasting it onto another image, preserving a binary mask. On the other hand, ClassMix [70] goes a step ahead of CutMix by dynamically generating a binary mask based on the network predictions, with the model selecting certain classes for an image and cutting/pasting relevant pixels accordingly. Similarly, DACS [71] follows the ClassMix method by producing augmented samples from images of different domains, combining them with their respective labels and pseudo-labels. DACS combines source domain labels with pseudo-labels from target domain images to generate pseudo-labels for the new image. This approach allows replacing components of the pseudo-labels with components from the ground-truth semantic maps, ensuring that all classes are surrounded by pixels from the other domain, enhancing training efficacy. DAFormer [72] is a new method for unsupervised domain adaptation that uses a Transformer encoder and a multi-level context-aware feature fusion decoder. It is based on the same idea as DACS. In addition, to address the challenges of adaptation instability and overfitting to the source domain, the paper proposes three training strategies: sampling images with rare classes, distilling knowledge from expressive ImageNet features, and using a learning rate warm-up.

2.2.3 Few-Shot Learning

Few Shot Learning has been widely studied in the context of image classification [73–77]. [78] presents a model called "Matching Networks" that is capable of learning to perform one-shot classification tasks, such as classifying images based on a small number of examples. The model uses a Siamese network architecture and a matching function to compare the input example to a set of labeled examples. [73] introduces a model called "Prototypical Networks" that learns to classify examples based on a small number of labeled examples by learning a metric space in which examples with the same label are close together. The model is trained on a large number of tasks and learns to classify new examples by finding the nearest prototype in the metric space. [79] presents a few-shot learning model that uses graph neural networks to represent the relationships between examples and labels. The model is trained on a large number of tasks and learns to classify new examples by updating the graph representation based on the labels of the new examples. [80] presents a few-shot learning approach for text classification that uses pre-trained word embeddings and a human in the loop to classify examples based on a small number of labeled examples. The model is able to learn from a small number of examples and achieve good performance on a variety of text classification task.

Only recently, the Few-Shot learning task has also been applied to semantic segmentation [81–84]. One approach is to use meta-learning, which involves learning a model that can adapt quickly to new tasks with only a few examples. For example, in [85], the authors propose a meta-learning approach that trains a segmentation model on a set of related tasks, allowing it to adapt quickly to new tasks with only a few examples.

Other approaches to few-shot learning in semantic segmentation include the use of generative models, such as GANs (generative adversarial networks), to synthesize additional training examples [81] and the use of attention mechanisms to weight the importance of different features in the input image.

A solution to solve the few shot problem is to use domain adaptation, which involves adapting a pre-trained model on a large dataset to a new task with a limited amount of data [83, 86, 87, 18]. FSDA [18] is a two-stage method that addresses this problem in semantic segmentation. A data pairing method for data enhancement is proposed to ensure stable training and reduce over-fitting. It can be difficult to effectively train a whole network using very few target data, especially for networks that are far from the output. Therefore, a two-stage structure is designed, with the first stage being a shallow auxiliary network and the second stage being a deep network. The first stage not only enhances the adaptation of low-level features, but also provides an auxiliary prediction mask to guide the learning of the second-stage network. The label filtering method, which is aided by the auxiliary prediction mask, helps to strengthen the network's learning of difficult-to-classify pixels. Finally, spectral weight normalization is used in the discriminators and a strategy of training alternately is proposed to further improve the stability and effectiveness of the training.

2.2.4 Domain Generalization

Several approaches have been suggested for domain generalization in image classification tasks, including meta-learning [88–91] that involves learning a model that can adapt to new tasks quickly, using a small amount of task-specific data, adversarial training [92–94], autoencoders [95, 93] that are a type of neural network that can learn a compact representation of an input, which can be used for domain generalization by encouraging the model to learn domain-invariant features in the latent space, metric learning [96, 97] which involves learning a distance function between data points, which can be used to identify similar and dissimilar examples, and data augmentation [98, 99] that generates new training examples from existing ones, which can help the model generalize to new domains.

There have been only a few approaches proposed for semantic segmentation [100–105], as research in this area is still in its infancy. These approaches typically focus on two main strategies: domain randomization and normalization. Domain randomization involves generating images with various styles in order to improve the model's ability to generalize to new domains. This can be done through methods such as image-to-image translation, which transfers a source domain image to multiple styles [100], or by synthesizing synthetic images with styles of unrealistic paintings [101]. Normalization techniques, on the other hand, aim to extract style-invariant features from the images while preserving content-related information. This can be achieved using techniques such as instance normalization [106] or whitening [103].

2.3 Datasets and Metric

2.3.1 Datasets

As part of our evaluation, we will explain in the following paragraphs a diverse set of datasets to assess the effectiveness of the domain adaptation and domain generalization methods presented in Chapter 3. These datasets include both real-world and synthetic data, and are designed to challenge the proposed methods in terms of domain shift, task complexity, and sample size. Figures 2.1, 2.2 and 2.3 present a visual comparison of the real-world and synthetic datasets listed below.

Cityscapes

Cityscapes [3] is a real-world dataset for self-driving cars gathered in numerous German cities. It comprises of 2975 photos with detailed annotations spread over 33 annotated classes; the data collection and annotation was aimed to capture the great variety of the outside street scene. The original resolution of pictures is 2048x1024, although they are frequently scaled to 1024x512 during training to maintain the original proportion. The validation set consists of 500 frames, and training and validation are done on 19 classes that establish the benchmark for semantic segmentation in self-driving cars. One of the associated challenges with the Cityscapes dataset is the large size and complexity of the dataset. The high resolution of each image requires extensive processing power and time to train models.

BDD100K

BDD100K [107] is an extensive real-world dataset containing 100,000 high-resolution driving videos covering 10 evaluation tasks to assess image recognition algorithms for autonomous driving. The dataset includes over 100 million frames and GPU/IMU data for trajectory information, representing over 1,000 hours of driving experience. The dataset's diversity in geographic, environmental, and weather conditions makes it an excellent resource for training models to handle various conditions. However, this variation in urban landscapes makes it challenging to train models that can effectively generalize across different environments.

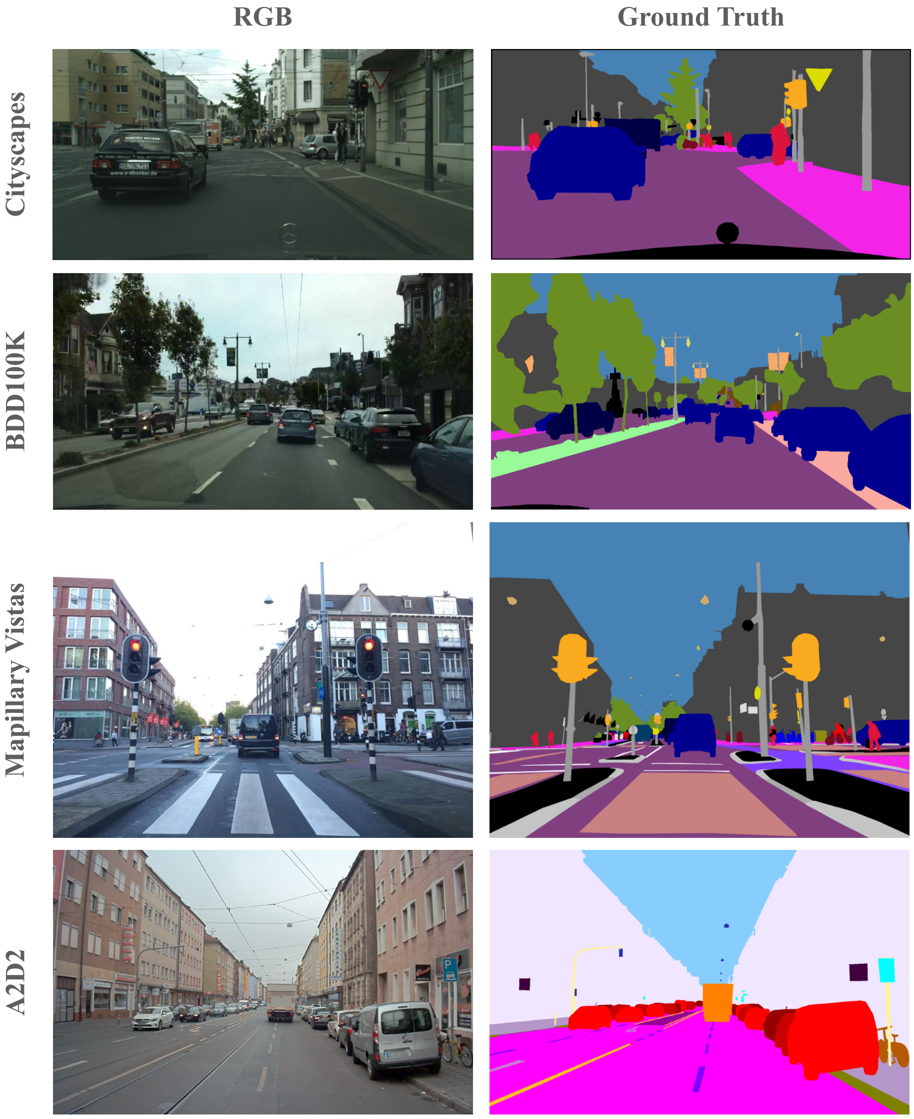


Fig. 2.1 Samples that compare the real-world datasets used in this work when dealing with the autonomous driving task.

Mapillary Vistas

Mapillary Vistas [108] is a real-world datasets, that comprises 25,000 high-resolution images, with annotations for 66/124 object categories. Images are sourced from different locations globally, captured under various weather, seasonal, and daytime conditions, and using diverse imaging devices such as mobile phones, tablets, action cameras, and professional capturing rigs. This broad spectrum of variables presents an excellent opportunity for training models to tackle a range of real-world scenarios.

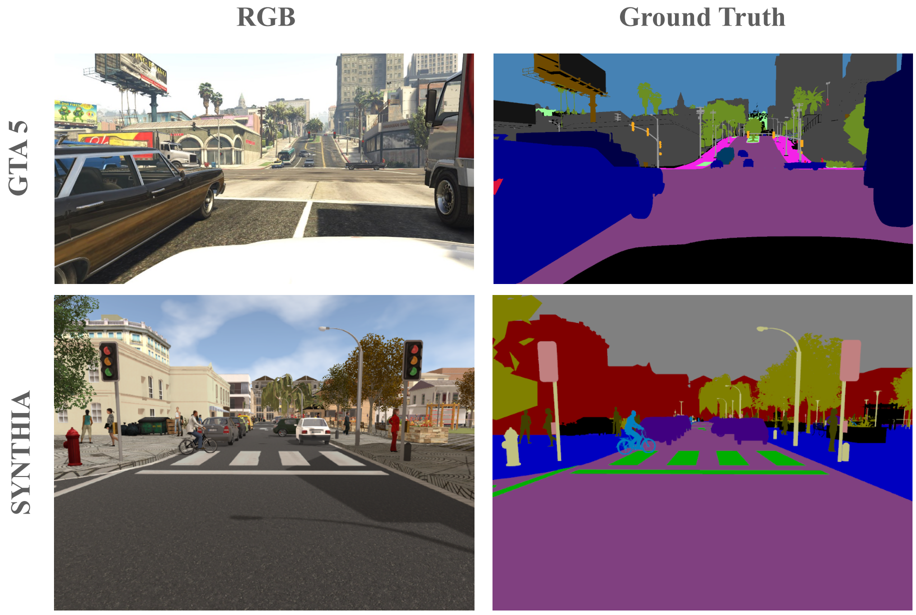


Fig. 2.2 Samples that compare the synthetic datasets used in this work when dealing with the autonomous driving task.

Nevertheless, due to the variance in lighting and weather conditions across different cities in the world and times of day, the dataset requires careful pre-processing and augmentation to achieve robust performance.

A2D2

A2D2 [109] comprises 41,280 frames and provides semantic segmentation annotations for 38 categories. It encompasses data captured in the south of Germany across highways, country roads, and cities, in varying weather conditions ranging from cloudy to rainy and sunny.

GTA 5

GTA 5 [4] includes a total of 24966 synthetic images with pixel-level semantic annotations. These images were created by rendering the open-world video game Grand Theft Auto 5 from a car perspective in virtual American cities. The dataset, thought for autonomous driving application, includes 19 semantic classes, which are compatible with the classes used in the Cityscapes dataset. The original images size corresponds to 1914 × 1052 resolution. The GTA5 dataset does not capture all aspects of the real world, thus creating challenges when used to adapt to real-world scenarios. Moreover, GTA5 has raised ethical concerns as it is based on a violent video game and may be seen as promoting violence. This can make it difficult for researchers to justify using the dataset in their work.

SYNTHIA

SYNTHIA [5] is distinguished by more than 200k photo-realistic frames produced from a virtual city with accurate pixel-level semantic annotations. All of the images are targeted for use in autonomous driving and exhibit a high degree of unpredictability owing to scene diversity, a range of dynamic objects, camera views, numerous seasons, and varying lighting and weather conditions. The "RAND-CITYSCAPES" subset, which includes 9400 images, is often used for semantic segmentation tasks. The original picture is 1280x760 in size. The 19 classes shared by Cityscapes are considered for training, while the assessment is done on a subset of 13 and 16 classes, following the standard technique used in [110] and [111], respectively. The datasets contain scenes from a limited set of locations, which limits the diversity of the data.

Agriculture Vision

Agriculture Vision [46] consists of aerial images of farmlands that have been labeled with 9 different categories, such as "Nutrient Deficiency," "Water," and "Weed Clusters." This type of information is valuable for farmers and agricultural researchers, as it can help them to better understand the patterns and anomalies in their fields and make informed decisions about how to optimize yields. The dataset is designed for aerial image analysis, with over 50,000 training images and almost 20,000 validation images, and it includes both RGB and NIR (near-infrared) channels. The images are already provided in tiled format, which means that they are divided into smaller, 512 x 512 pixel images for easier processing. One of the difficulties linked with the Agriculture vision dataset is related to the variations in lighting that can lead to shadows and lighting discrepancies, hindering the precision of the segmentation models. Also, the low resolution due to the elevated altitude of image capture makes it challenging to discern finer particulars of objects illustrated in the images. Ad-

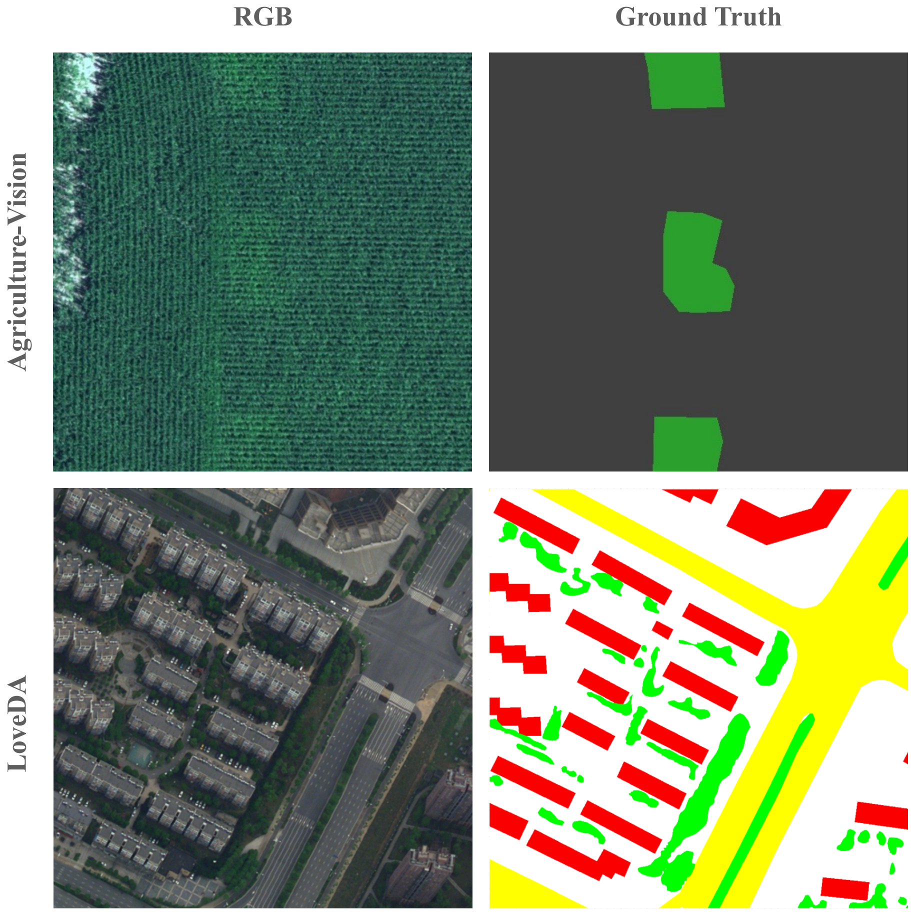


Fig. 2.3 Samples that compare the real-world datasets used in this work when dealing with the aerial image analysis task.

ditionally, the aerial images may exhibit partially or completely obscured objects, making it a struggle for models to adequately recognize and categorize them.

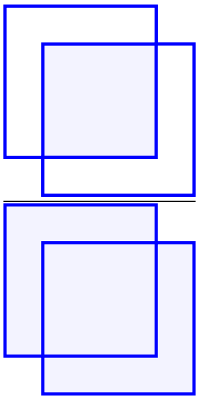
LoveDA

LoveDA [42] is a collection of images for land cover semantic segmentation in remote sensing, specifically designed for unsupervised domain adaptation. The dataset includes both urban and rural areas, allowing researchers to train and evaluate machine learning models that can adapt to different types of environments. The dataset was collected from 18 different administrative districts in China, and it includes a total of 2522 images. The urban training set contains 1156 images, while the rural training set contains 1366 images. Each image is supplied in a tiled format of 1024 x 1024 pixels, and they are annotated with seven different categories, such as "building," "road," and "waterways.". The LoveDA dataset presents similar challenges as mentioned for Agriculture Vision.

2.3.2 Metric

All the experiments presented in Chapter 3 are evaluated using the standard mean Intersection over Union [112] metric. The Intersection over Union (IoU) is a widely used metric in computer vision for evaluating the performance of image segmentation algorithms. It measures the overlap between the predicted output and the ground truth annotations, and is defined as the ratio of the area of their intersection to the area of their union:

$$IoU = \frac{\text{area of overlap}}{\text{area of union}} =$$



Formally, let A and B be two sets representing the ground truth and prediction respectively. The IoU can be defined as:

$$IoU(A,B) = \frac{Intersection(A,B)}{Union(A,B)} = \frac{|A \cap B|}{|A \cup B|} \tag{2.1}$$

Where $|\cdot|$ represents the cardinality of the set.

The mean Intersection over Union (mIoU) extends the IoU metric by considering multiple classes. Given N classes, for each class, the IoU between the predictions and the ground truth is calculated and then the mean of these values is taken over all classes. This provides a more comprehensive evaluation of the performance of the segmentation algorithm, especially when dealing with datasets that have a large number of semantic classes.

Chapter 3

Learning to see across domains

This chapter presents a variety of strategies to let the models more robust and be able to learn and see across different domains. To begin with, we introduce a new synthetic dataset called IDDA [6] specifically designed for autonomous driving, which can be used to evaluate the model's adaptation and generalization capabilities. Next, we present a framework called PixDA [7] that enables learning across domains with limited annotated target data. Additionally, we demonstrate that traditional Semantic Segmentation networks may not be able to generalize and not suitable for use in an aerial setting and we propose a solution called AIAS [8] to address this issue. Finally, we leverage these techniques to develop a new model and domain adaptation algorithm (HIUDA) [9] that are particularly robust and well-suited for use with aerial images.

3.1 The demand for data: IDDA

In this section we focus on the difficulty in creating large, annotated datasets for semantic segmentation when applied to the autonomous driving task. One solution to this problem has been the use of synthetic datasets, which are designed to help develop algorithms that can handle the "visual domain shift" between training and test data. However, the domain shift is a significant challenge even when using real data, as the appearance of cities and the weather conditions can vary greatly between different vehicles and even at test time for a single vehicle. Domain adaptation and the effective use of multiple data distributions (source domains) are active areas of research in this field. To support this research, in the following we present a new, large-scale synthetic dataset for semantic segmentation featuring more than 100 different source visual domains. The dataset was created to explicitly address the challenges of domain shift in various weather and viewpoint conditions, in seven different city types. Through extensive benchmark experiments, we demonstrate the open challenges for the current state of the art in this field.

Obtaining large amounts of labeled data for training and evaluating algorithms for Semantic Segmentation can be a challenging task. Collecting images from a wide range of driving conditions and manually classifying each image is both time-consuming and costly [1–3], and the accuracy of the labels produced in this way may vary. Synthetic datasets [4, 5], which are created using 3D graphics engines and have perfect labeling, offer an alternative solution. However, models trained on these virtual datasets often perform poorly when applied to real-world scenarios due to the so-called "domain shift." To address this issue, techniques such as domain adaptation and generalization [17] have been developed to improve the ability of the Semantic Segmentation algorithms to handle changes in driving conditions. However, it remains important to have diverse and large datasets with labeled data to support the training and evaluation of these techniques.

The increasing interest in using Semantic Segmentation for autonomous driving has led to the development of several datasets for this purpose. More datails in Table 3.1. Early datasets, such as CamVid [1, 2] and KITTI [113], contained relatively few labeled images (less than 1k) in low resolution and with limited variability. The release of Cityscapes [3], with 5k finely annotated and 20k coarsely annotated images, established the first benchmark for testing Semantic Segmentation in autonomous driving. This was followed by the development of larger datasets by academic

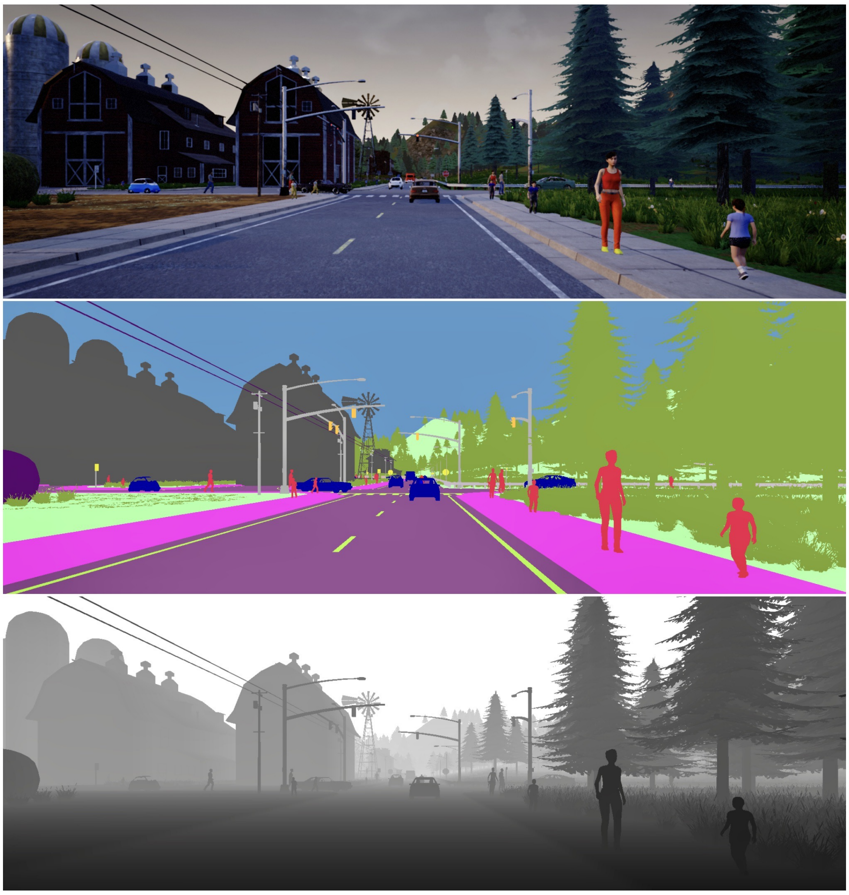


Fig. 3.1 The IDDA dataset. An example with an RGB image and its corresponding semantic and depth map.

researchers (BDD100K [107]), image providers (Mapillary Vistas [108]), and the automotive industry (Apolloscape [114], A2D2 [109]).

Despite the availability of these datasets, none of them provides a good benchmark for evaluating the performance of a Semantic Segmentation network on a different domain. Some datasets, such as CamVid, KITTI, and Apolloscape, lack variability as they only contain images from a single city or viewpoint. Others, such as Mapillary Vistas and BDD100K, offer scene diversity but do not provide a way to easily select scenarios from different domains, or Virtual KITTI [115], which only includes a small number of images per scenario, are difficult to use for evaluating domain adaptation approaches.

The challenge of collecting and labeling large quantities of images with a wide range of conditions has led to the creation of datasets based on 3D game engines

Table 3.1 Summary of the most popular datasets for Semantic Segmentation

Dataset	**Year**	**Size**	**Depth**	**Resolution (pixels)**	**FoV**	**#Cls**	**Annotation Time (min)**	**#Annotated Pixels (10^9)**	**#Domains**	**#images (avg*scene)**
Real-World Dataset										
CamVid	2008	701	No	920×720	-	32	60	0.62	1	-
KITTI	2012	400	**Yes**	1392×512	-	33	-	0.07	1	-
Cityscapes	2016	5k fine 20k coarse	No	2048×1024	90°	33	90 7	9.43 26.0	50	160
Mapillary Vistas	2017	25k	No	≥1920×1080	-	**66**	94	-	1	-
BDD100K	2018	10k	No	1280×720	-	40	-	-	1	-
ApolloScape	2018	144k	**Yes**	**3384×2170**	-	25	-	-	3	**29k**
A2D2	2019	41k	No	1920×1280	**120°**	38	-	-	23	1.7k
Synthetic Dataset										
Virtual KITTI	2016	21260	Yes	1242×375	29°	14	-	-	50	426
Synthia-Rand Synthia-Seqs	2016	13,400 200k	**Yes**	960×720	100°	13	**Instant**	147.5	1 51	- 8k
GTA 5	2016	25k	No	1914×1052	-	19	7	50.15	1	-
IDDA	**2020**	**1M**	**Yes**	1920×1080	90°	24	**Instant**	**2087.70**	**105**	16k

such as SYNTHIA [5] and GTA 5 [4]. These datasets also allow for the creation of finely annotated images without the cost of manual labeling. However, even these datasets have limitations when it comes to evaluating domain adaptation, as GTA 5 does not currently offer the option of selecting scenes from different domains, and SYNTHIA-Seqs only contains low-resolution images and a limited number of labels.

That is the reason why we introduced the ItalDesign DAtaset (IDDA)[1], which is a large synthetic dataset comprising over one million labeled images across more than 100 different scenarios, including 5 viewpoints, 7 towns, and 3 weather conditions. This diversity allows for a thorough analysis and benchmarking of the performance of current and future state-of-the-art Semantic Segmentation architectures, particularly in the context of Domain Adaptation tasks.

In comparison to these prior datasets, IDDA offers multiple, easily and separately selectable domains. Along with each RGB image, the dataset also includes its corresponding depth map and high-quality semantic annotation for a total of 24 semantic classes, as shown in Figure 3.1 and Figure 3.2. We decided not to record LIDAR data for memory consumption constraints since the dataset weights around 4,7TB.

[1] Download at: https://idda-dataset.github.io/home/

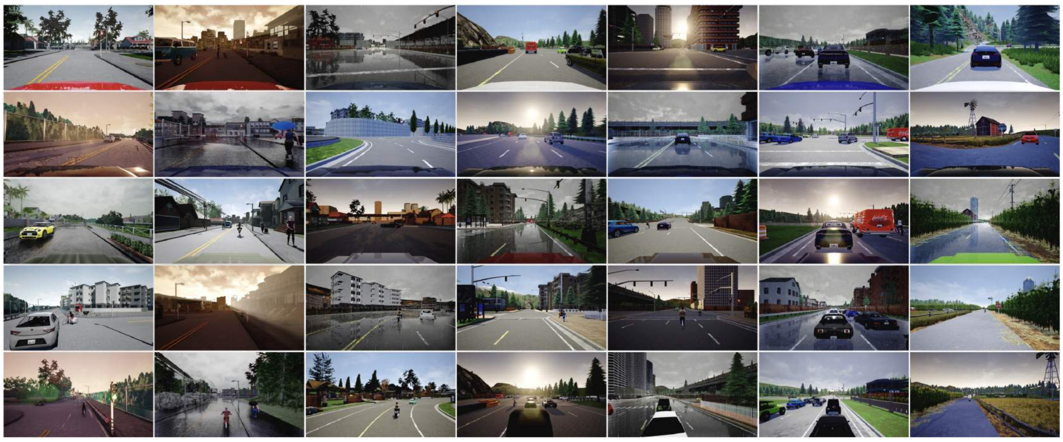


Fig. 3.2 Samples for any instance of variety provided by the IDDA dataset. On the row the 5 viewpoints (Audi, Mustang, Jeep, Volkswagen T2 and Bus), on the column the 7 environments (from Town1 to Town7). Images iterate over the 3 weather conditions (Clear Noon, Clear Sunset and Hard Rain Noon).

3.1.1 The dataset

Data Creation

We generate IDDA by using the CARLA simulator [116], respectively the versions 0.8.4 and 0.9.6. CARLA is an open-source project that is designed to support the development and testing of autonomous driving systems. One reason we chose CARLA is because it offers a high level of customization, including the ability to set the number of pedestrians and vehicles, the environment conditions, the map, and the speed of the simulation. Additionally, CARLA utilizes the Unreal Engine 4, which is a leading technology in computer graphics. CARLA has a client-server architecture, with the client controlling a specific agent, that is called player, and the server simulating the rest of the world and the other agents.

Our client is able to start new simulations (called "episodes"), setting automatically the parameters and meta-parameters each time. The number of frames captured by the player in each episode is determined by the size of the town, with smaller towns resulting in fewer images. To create a variety of traffic scenarios, each episode is initialized with a random number of vehicles and pedestrians ranging from 20 to 150 and 0 to 100, respectively. Additionally, players are spawned in different locations, and the distribution of vehicle models and colors changes from episode to episode. These choices were made to reduce the occurrence of deadlocks and ensure that the player is not stationary for extended periods of time. Overall, these factors help ensure that the collected data is diverse and rich. The client also specifies which sensors are equipped on the player vehicle. For the creation of the dataset, we used an RGB camera, a semantic segmentation sensor, and a depth sensor, all with a field of view (FoV) of 90 degrees. The semantic segmentation sensor produces pixel-wise labeled images based on the object blueprints in the Unreal Engine, and the depth sensor provides images that encode depth in the three channels of the RGB color space (R > G > B). The actual distance in meters is calculated using the following formula:

$$\text{distance} = 1000 \times \frac{R + G \times 256 + B \times 256^2}{256^3 - 1}$$

The sensors are mounted on the player's windshield, approximately at the height of the rear-view mirror. To collect the data, we used five different player vehicle models (two sport cars, a jeep, a minivan, and a bus), which resulted in a range of camera heights from 1.2 to 2.5 meters. The portion of the image occupied by the player's hood also varies depending on the vehicle model, ranging from 11.08% to 13.99% when the hood is visible (on sedans and jeeps) and being 0% in the other cases. All of the sensors are synchronized to capture a frame every 3 seconds, resulting in episodes that last from 3 to 4 minutes of simulation time. When a frame is captured, six frames are stored simultaneously: one RGB frame, three depth frames (raw, grayscale, and log-grayscale), and two semantic frames (raw and colored using the Cityscapes color palette). The RGB camera also has post-processing effects applied, such as bloom, lens flare, and motion blur, to increase the realism of the images. For the acquisition of the dataset we chose not to stop the capture when the vehicle was waiting at traffic lights stop due to the limited amount of traffic lights across the environments.

Data Description

The IDDA dataset consists of 1,006,800 frames and took approximately two weeks to be created using two workstations, each equipped with a single NVIDIA Quadro P5000 GPU with 16GB of memory. The total of 1 million images is the result of the fact that our dataset is composed of several domains and each of these has a dimension of about 16K images. We wanted to have enough data for each individual scenario having a reasonable number of images for train/val/test splits. In terms of

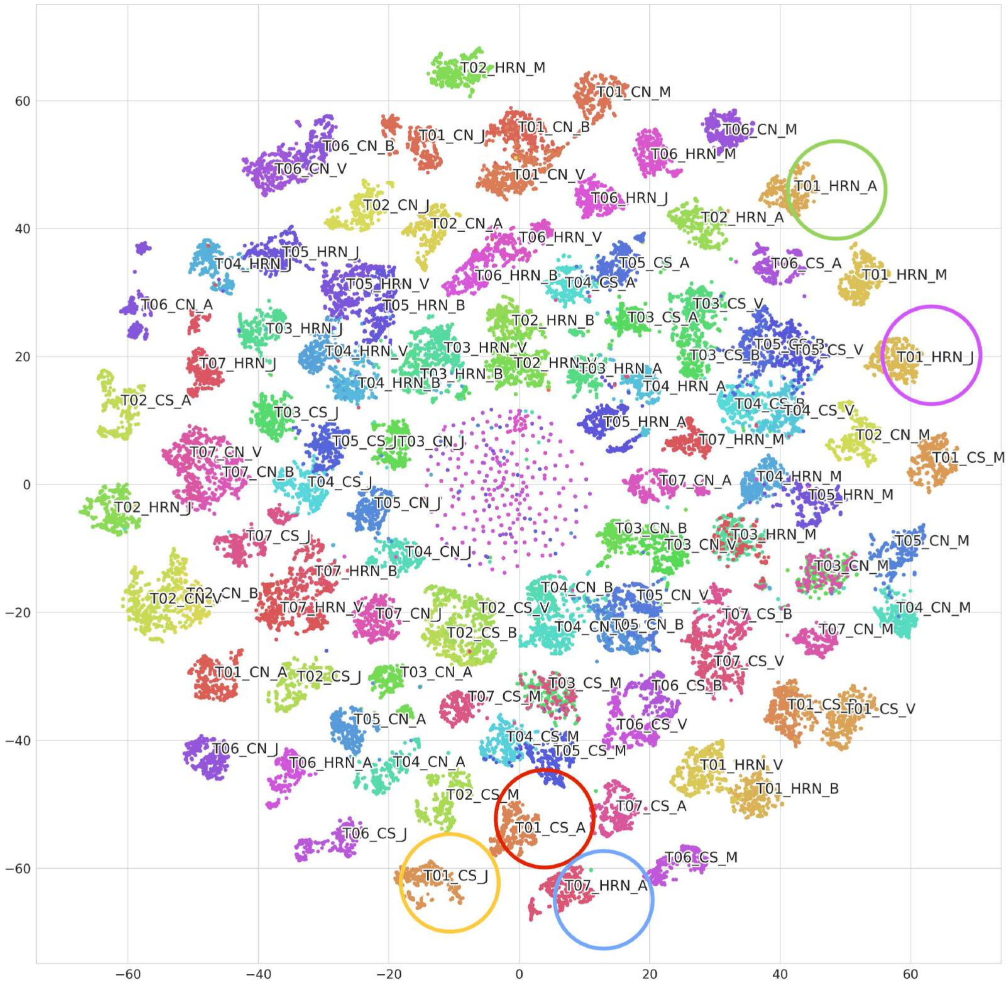


Fig. 3.3 The tSNE representation of the 105 different IDDA's scenarios. Circled the subdomains that are used for the experiments.

the number of frames, IDDA is significantly larger than other datasets such as GTA 5 [4] and SYNTHIA [5], and it is more than five times larger than the semantically annotated images in KITTI [113]. IDDA features 105 scenarios, as shown in Figure 3.2, created by varying three aspects of the simulation:

- **Town**: The frames in the dataset are collected from seven different towns. Towns 1 and 2 (T01 and T02) are characterized by 2.9 km and 1.4 km of drivable roads with buildings, bridges, vegetation, terrain, traffic signs, and various kinds of infrastructure. Towns 3, 4, 5, and 6 (T03, T04, T05, and T06) are characterized by complex urban scenes with multi-lane roads, tunnels, roundabouts, freeways, and connection ramps. Finally, Town 7 (T07) is different from the others because it depicts a rural countryside with narrow roads, fewer traffic lights, and many non-signalized crossings. We believe that this entirely different domain is an important novelty provided by our dataset for the autonomous driving task. All seven towns are populated by vehicles and pedestrians.

- **Weather**: We considered three weather settings that are significantly different from each other: Clear Noon (CN), characterized by bright daylight; Clear Sunset (CS), with the sun low above the horizon and pink/orange hues; and Hard Rain Noon (HRN), with a cloudy sky, intense rain, and puddles that cause reflections on the ground.
- **Viewpoint**: The third parameter that we varied to create the scenarios is the player vehicle. For each vehicle, we positioned the sensor system approximately at the height of the rear-view mirror. We used five player vehicles that differ significantly in their height and shape, including an Audi TT (A), a Ford Mustang (M), a Jeep Wrangler (J), a Volkswagen T2 (V), and a bus (B). This choice not only results in images with distinct perspectives, but also ensures that the hood of the player vehicle, if visible[2], is different in both shape and color. To the best of our knowledge, the inclusion of images from the perspective of not only cars but also jeeps, vans, and buses is a unique feature of IDDA and adds a new dimension of variability.

One of the main objectives in the development of IDDA was to create a dataset that was competitive in terms of the variety of recognizable items within a scene. Specifically, we aimed to increase the number of semantic classes provided by the simulator to be as close as possible to those found in Cityscapes or GTA 5. To achieve this, we made modifications to the 3D maps and the source code of the simulator so that each static and dynamic element would be labeled and tagged before being spawned in the virtual world. This approach allowed us to increase the number of tags provided by the simulator from the original 13 to a total of 24 semantic classes. The distribution of these classes in the IDDA dataset is shown in Figure 3.4, where it can be seen that the most prevalent classes are building, road, vehicle, vegetation, terrain, and sky. Additional useful statistics are summarized in Table 3.1.

[2] The hood is not visible in the case of the bus and the Volkswagen T2.

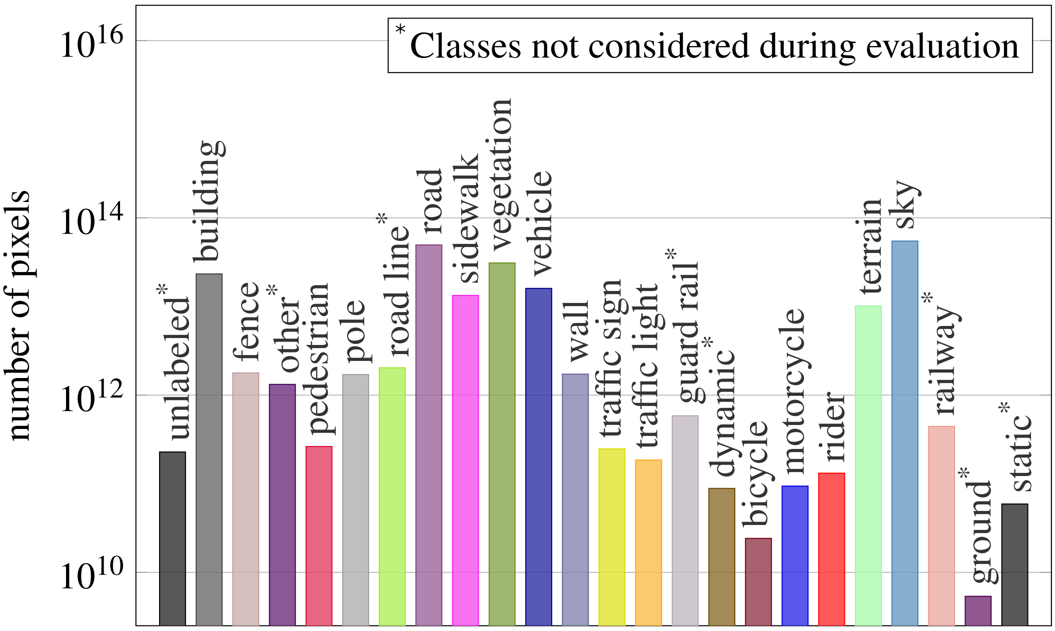


Fig. 3.4 Number of high-quality annotated pixels (y-axis) per class (x-axis).

3.1.2 Experiments

To showcase the key features and potential applications of IDDA, we conducted two experiments. In the first one, we aimed to verify that the scenarios available in IDDA can be used to effectively evaluate and benchmark the ability of the Semantic Segmentation methods to adapt to domain shifts in driving applications. To do this, we selected several state-of-the-art networks, both with and without domain adaptation, and we measured the performance degradation when the train and test sets were taken from different scenarios. In the second experiment, we used the scenarios available in IDDA to investigate how different data distributions in the synthetic source domain can impact the performance of a network on real target domains, such as Cityscapes, BDD100K, Mapillary Vistas, and A2D2. For this purpose, we used the same networks from the first experiment but tested them on these real-world datasets.

Compared Methods

We conducted experiments using four state-of-the-art Semantic Segmentation architectures:

- PSPNet [26]: it expands the capabilities of Feature Pyramid Networks (FPNs) by employing dilated fully convolutional networks and global pyramid pooling to capture pixel-level features;
- PSANet [117]: it incorporates long-range contextual information;
- DeepLab V3+ [29]: with respect to the DeepLab V3, which introduces modules that use atrous convolution in cascade or parallel and an augmented version of the Atrous Spatial Pyramid Pooling, it also uses the Xception model and depthwise separable convolution to enhance its capabilities;
- DeepLab V2 [27]: it introduces atrous spatial pyramid pooling (ASPP) to segment objects at multiple scales and improve the localization of object boundaries by combining techniques from deep convolutional neural networks (DCNNs) and probabilistic graphical models through the implementation of a fully connected Conditional Random Field (CRF).

All of these networks have a ResNet-101 [118] as a backbone. The DeepLab V2 serves as the main building block for the remaining four Domain Adaptation models included in our experiments:

- ADVENT [111]: it solves the unsupervised domain adaptation task adversarially by introducing losses based on the entropy of pixel-wise predictions;
- DISE [64]: it separates images into domain-invariant structure and domain-specific texture representations for label transferring;
- CLAN [63]: it focuses on preserving local semantic consistency while aligning global distributions to reduce the negative transfer effect of misaligned features;
- DADA [119]: it proposes an adversarial training method that utilizes the same entropy minimization approach introduced by ADVENT and leverages the knowledge of dense depth map in the source domain.

Table 3.2 IDDA Scenarios Distances

	Case		
Distance Function	Viewpoint Change	Weather Change	City Change
Euclidean	2.7604	5.6555	6.4551
Cosine	0.2590	1.2633	1.0586
Bhattacharyya	0.0149	0.0337	0.0426

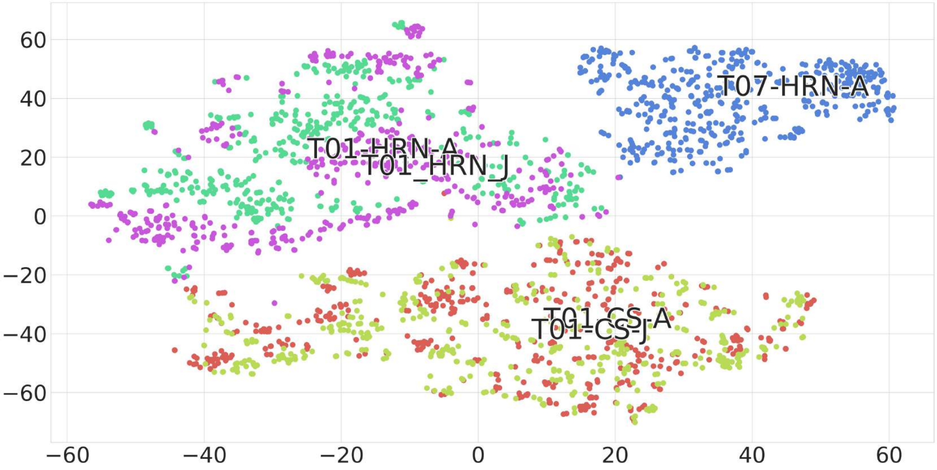


Fig. 3.5 The tSNE representation of the 5 chosen scenarios to assess IDDA.

Implementation and Training Details

For our experiments, we used the hyperparameters reported in the original papers for each network to ensure a fair evaluation of their performances. For the Domain Adaptation architectures, we used the official implementation provided by the authors, while for the Semantic Segmentation part of the experiments, we used PyTorch re-implementations. To better compare with Cityscapes, which is the main real dataset for evaluating Semantic Segmentation in autonomous driving, we excluded ambiguous classes (dynamic, static, other) and those not present in the reference dataset (road line) from our experiments, resulting in a total of 16 labels as shown in Figure 3.4. To quantify the distance between the source and target domains (similar to 3.1 in [120]), we extracted features using a ResNet-101 [118] model pretrained on ImageNet from the first 500 samples of each domain and reduced the dimensionality using PCA on the first 50 principal components. We then calculated the mean-feature

Table 3.3 IDDA vs IDDA Experiment Results

Semantic Segmentation Networks		Scenarios (% mIoU)		
		Viewpoint Change	Weather Change	City Change
	Source: Target:	T01 CS A T01 CS J	T01 CS J T01 HRN J	T01 HRN A T07 HRN A
w/o DA	DeepLab V2	62.6	40.2	21.7
	DeepLab V3+	64.9	33.9	14.3
	PSPNet	67.3	29.7	14.6
	PSANet	66.9	33.6	15.5
	DeepLab V2 (source=target)	79.1	78.3	78.0
w/ DA	DADA	66.4	55.9	36.5
	ADVENT	68.4	61.1	39.3
	CLAN	70.3	65.5	41.2
	DISE	73.6	71.9	46.7

vector for each domain and measured the Euclidean and Cosine distances, as well as the feature-wise Bhattacharyya distance.

IDDA vs IDDA Results

We evaluated the ability of the selected networks to adapt to a new domain by testing them in three cases that covered three different types of variability: viewpoint change (from source A to target J), weather change (from source CS to target HRN), and background change (from source T01 to target T07). We used the method described above to measure the distance and difficulty of these cases (see Table 3.2). As a visual confirmation, we used tSNE to project the features extracted with ResNet-101 into a 2D space (see Figure 3.5).

The results of the experiments are shown in Table 3.3. As expected, the shift across cities, made even more challenging by the rainy weather condition, resulted in a higher degradation in performance than the other two experiments. In the town shift, the Semantic Segmentation networks struggled to correctly classify the scene and their accuracy dropped as low as 22.0%. In this case, domain adaptation (DA) produced a considerable boost, with an average accuracy of around 41.0%. This trend was also observed in the shift across weather conditions, but since the gap

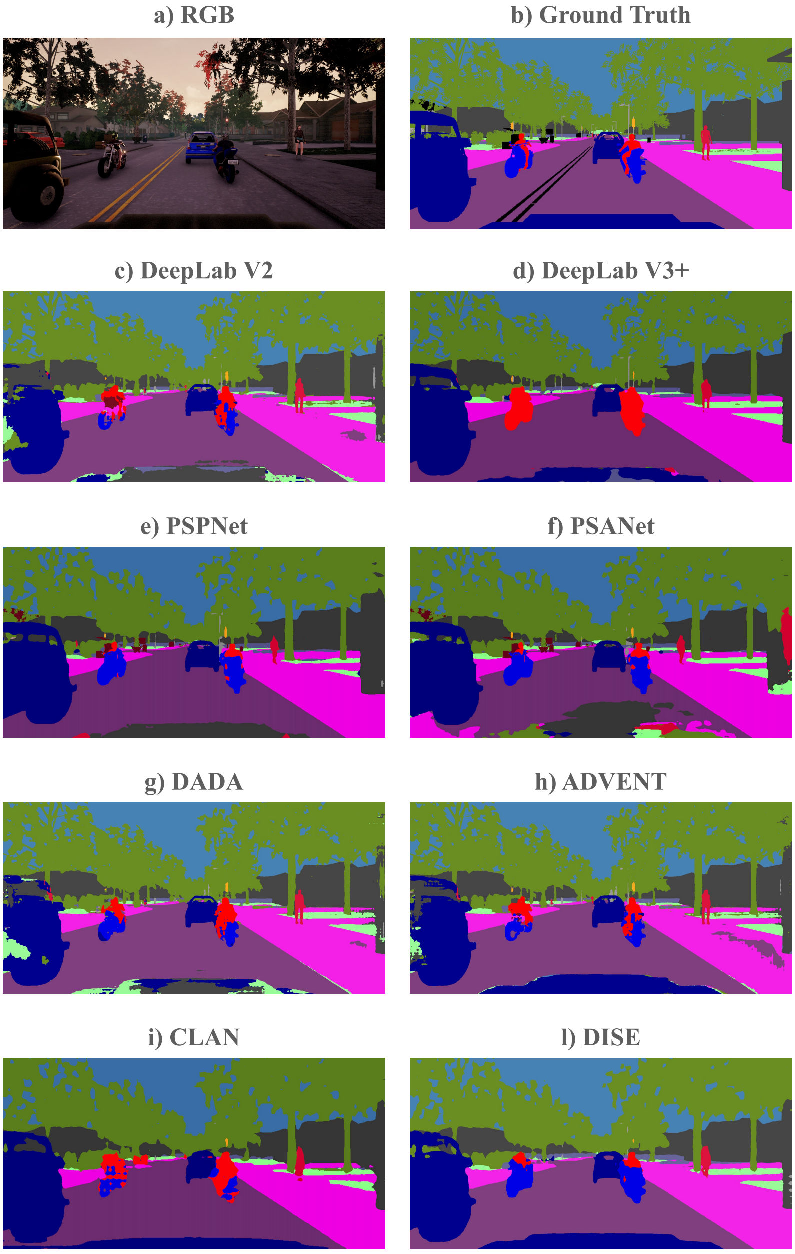
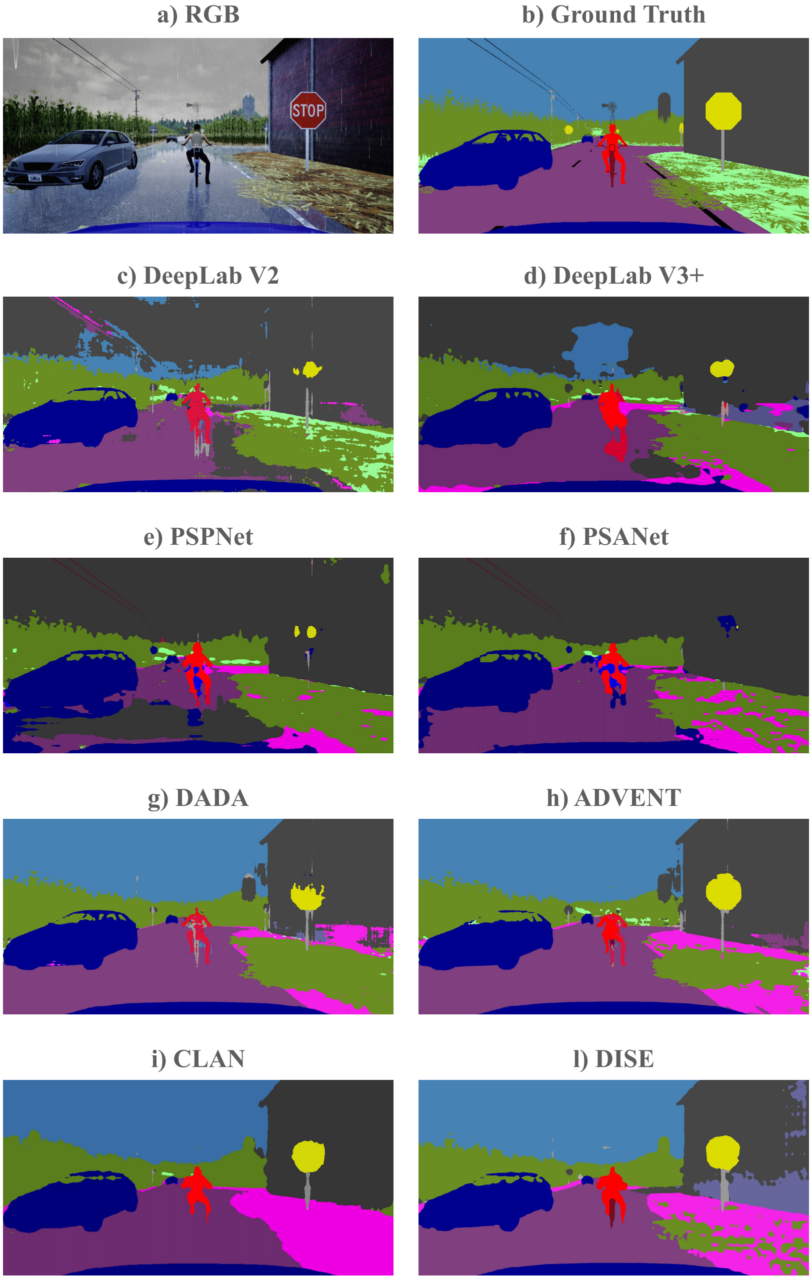


Fig. 3.6 Qualitative results for the viewpoint shift experiment.

between the source and target domains was smaller, the resulting average mIoU was 34.4% (without DA). In this case, Domain Adaptation performed well, resulting in an average accuracy of 63.6%. The viewpoint change proved to be the best performing set of experiments, with the addition of Domain Adaptation increasing the average accuracy by only 4.3%. Among all of the Domain Adaptation networks, DISE



a) RGB

b) Ground Truth

c) DeepLab V2

d) DeepLab V3+

e) PSPNet

f) PSANet

g) DADA

h) ADVENT

i) CLAN

l) DISE

Fig. 3.7 Qualitative results for the background shift experiment.

proved to be the most capable, while the depth information exploited by DADA did not seem to improve the performance.

The Figures 3.6 and 3.7 illustrates some qualitative results of our experiments. Looking at the output produced, we can identify two interesting problems that seem

Table 3.4 Distances between IDDA and Real-world datasets

	Distance Function	Dataset			
		Cityscapes	BDD100K	Mapillary	A2D2
Best Case	Euclidean	7.4419	7.6177	5.4493	6.3874
	Cosine	1.3582	1.6209	1.2924	1.0589
	Bhattacharyya	0.0552	0.0502	0.0106	0.0447
Worst Case	Euclidean	8.2360	7.7618	4.9548	7.0150
	Cosine	1.5465	1.5526	0.9147	1.1849
	Bhattacharyya	0.0498	0.0381	0.0267	0.0387

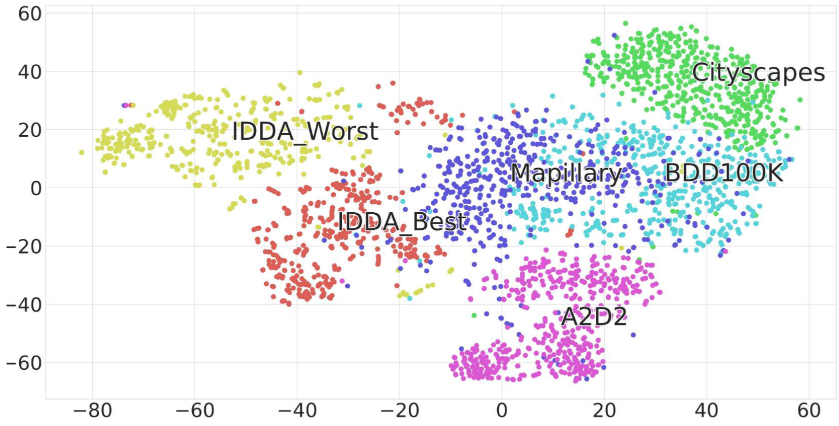


Fig. 3.8 The tSNE representation of the distributions of synthetic and real datasets.

to affect the Semantic Segmentation networks and their generalization capability. In the viewpoint change, all the Semantic Segmentation models without Domain Adaptation struggle to classify the portion of the image occupied by the hood of the vehicle, improperly classifying it as a building. Additionally, when changing the scenario and moving to a countryside scene with vegetation in place of roadside and sidewalks (the "city change" case), we observe that all the networks (with the exception of DeepLab V2) learned and memorized the pattern "building-sidewalk-road" of the source scenario during training. As a result, when transitioning to a new environment, especially a rural one, they struggle to adjust and often mistake the terrain for a sidewalk.

Table 3.5 Synthetic vs Real Experiment Results
*considering only 13 labels

Source	Networks	Target			
		Cityscapes	BDD100K	Mapillary Vistas	A2D2*
Same as target (baseline)	DeepLab V2	62.9	52.7	67.6	65.4
Best case	DeepLab V2	32.7	24.2	36.1	32.1
	DADA	33.1	29.6	37.3	38.6
	ADVENT	35.3	33.2	37.0	42.6
	CLAN	39.3	33.5	39.4	44.3
	DISE	42.1	40.1	41.7	46.7
Worst case	DeepLab V2	16.8	17.5	27.1	29.8
	DADA	23.7	23.5	32.6	36.2
	ADVENT	23.8	27.0	30.1	38.6
	CLAN	25.8	30.7	30.9	42.7
	DISE	31.3	31.4	33.7	45.5

IDDA vs Real-World Datasets Results

With the previous experiments we have demonstrated the limitations of the current state-of-the-art Semantic Segmentation networks and how IDDA could be a powerful tool for validating the adaptation performance to a domain shift in driving applications.

We then move on to evaluate the performance of the networks that were trained on IDDA as synthetic data on a real datasets. Specifically, we consider two cases: a "best case" and a "worst case." The "best case" consists of 29,952 samples that are selected from urban environments, have a car-like point-of-view and clear weather conditions, and are stratified to ensure similar environmental conditions to the target domains. The "worst case" consists of 40,128 samples that are taken from a countryside town, have a hooded and a non-hooded point of view, and are taken in rainy conditions. These samples are intended to have a higher visual discrepancy with respect to the target samples.

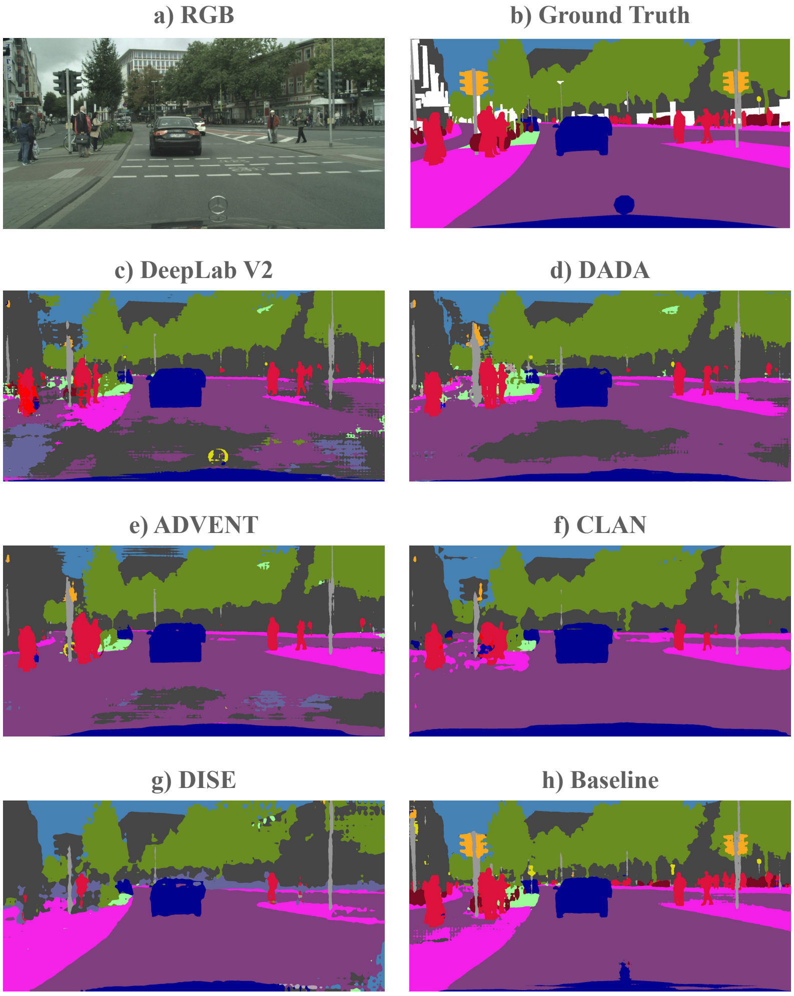
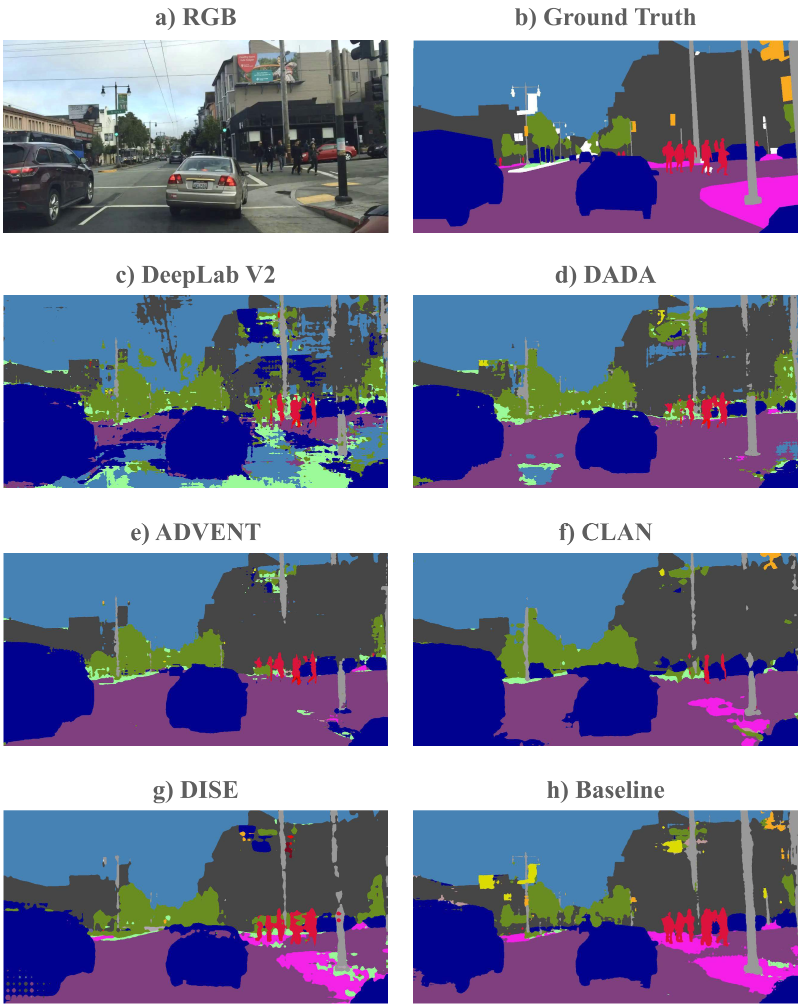


Fig. 3.9 Examples of the results when testing on real datasets in the best case scenario.

Table 3.4 and Figure 3.8 demonstrate the distance between the distributions of the synthetic and real datasets, both numerically and visually. When evaluating the performance on the target datasets, we only considered labels that are shown in Figure 3.4 and labeled all four-wheeled vehicles as our semantic class "vehicle." For the A2D2 dataset, we only considered 13 labels due to labeling inconsistencies with IDDA, such as the absence of the "rider" and "wall" classes and the combination of the "vegetation" and "terrain" classes.

The results of our comparison between the synthetic and real datasets are presented in Table 3.5. When using Semantic Segmentation-only architecture (DeepLab V2), we observe an average drop in performance of 30.1% in the best case (excluding



a) RGB

b) Ground Truth

c) DeepLab V2

d) DADA

e) ADVENT

f) CLAN

g) DISE

h) Baseline

Fig. 3.10 Examples of the results when testing on real datasets in the worst case scenario.

the A2D2 experiments due to differences in evaluation setup). As shown in Figure 3.9c (best case), the network has difficulty distinguishing between buildings, roads, and sidewalks, although it performs acceptably in recognizing pedestrians. Among the domain adaptation approaches, DISE proves to be the most effective. However, the gap with the baseline is still significant, and the improvements introduced by domain adaptation are not sufficient to achieve acceptable performance.

Interestingly, it appears that the additional depth information exploited by DADA is only helpful in Mapillary. In the worst case, the domain shift is much more severe, with a maximum drop of 46.1% when tested on Cityscapes. In this case, the Semantic Segmentation-only network fails to even identify the road, instead confusing it with "terrain" (see Figure 3.10c, worst case). This may be due to the significant textural differences between the source and target domains. The impact of domain adaptation is visually significant, but numerically we see that even the best-performing architecture does not approach the baseline. We also note that in both Cityscapes and BDD100K, the best-performing domain adaptation approach (DISE) almost doubles the performance of the Semantic Segmentation-only architecture, but has a much smaller increase in performance for Mapillary. This suggests that the higher the performance of the Semantic Segmentation-only networks, the lower the impact of domain adaptation.

Looking at the A2D2 results, the Semantic Segmentation-only architecture shows a drop in performance of an average of 34.4% when considering the best and worst cases. The domain shift in the worst case is slightly less severe than in Cityscapes and BDD100K, and more similar to Mapillary. This may be due to the higher presence of roads outside of urban areas, reducing the difference between the A2D2 and worst case distributions.

3.1.3 Findings

In this section we introduced IDDA, a synthetic database designed specifically for supporting research on domain adaptive semantic segmentation for autonomous driving. It is the largest existing dataset for this purpose, with 105 different domains. As demonstrated in the experimental section, IDDA is well suited for benchmarking a wide range of domain adaptation cases, due to the domain gap that exists both within IDDA and with respect to a real dataset.

3.2 Learn from limited data: PixDA

In this section we focus on the cross-domain few-shot scenario applied to the autonomous driving task, where we have limited access to real-world annotated images but a larger number of annotated synthetic images. This setting presents a challenge when it comes to aligning the domains, as there is often a class imbalance in the segmentation data that can lead to overfitting for well-represented classes and neglect of underrepresented ones. To address this issue, we introduce a new approach called Pixel-By-Pixel Cross-Domain Alignment (PixDA). This framework includes a pixel-by-pixel domain adversarial loss that aims to (1) align pixel-wise the source and target domains, (2) prevent negative transfer on those pixels that are correctly classified from the model, and (3) regularize the training of underrepresented classes to avoid overfitting. Additionally, we use a sample selection method that helps balance the source and target data and a knowledge distillation strategy to prevent overfitting to the few available target images. Through experiments on synthetic-to-real benchmarks, we show that PixDA outperforms previous state-of-the-art methods in 1-5-shot settings.

There have been efforts to address the issue of domain shift through unsupervised and semi-supervised approaches, but these still require a large number of images from the real domain, which can be a significant obstacle [121, 122]. An alternative solution is to consider a few-shot setting, where only a small number of annotated images from the real target domain are needed, rather than relying on a larger number of unlabeled images. Few-shot learning has been studied in a variety of visual learning contexts as depicted in Chapter 2.2.3. One of the main challenges of this approach is dealing with the inherent imbalance between source and target data [123]. In the case of semantic segmentation, this challenge is exacerbated by the pixel-wise imbalance among segmented classes. Some classes may be very frequent and take up a large amount of space (e.g., sky, road), while others may be rare and occupy a small area (e.g., traffic signs). This means that the number of pixels per class in the target domain can vary significantly, with some classes being poorly represented or even absent. This imbalance is more pronounced in semantic segmentation than in other problem settings, which can cause image-wise adversarial training methods to focus on aligning well-represented classes, leading to less accurate mapping of underrepresented classes in the target domain, as you can see from the Figure 3.11.

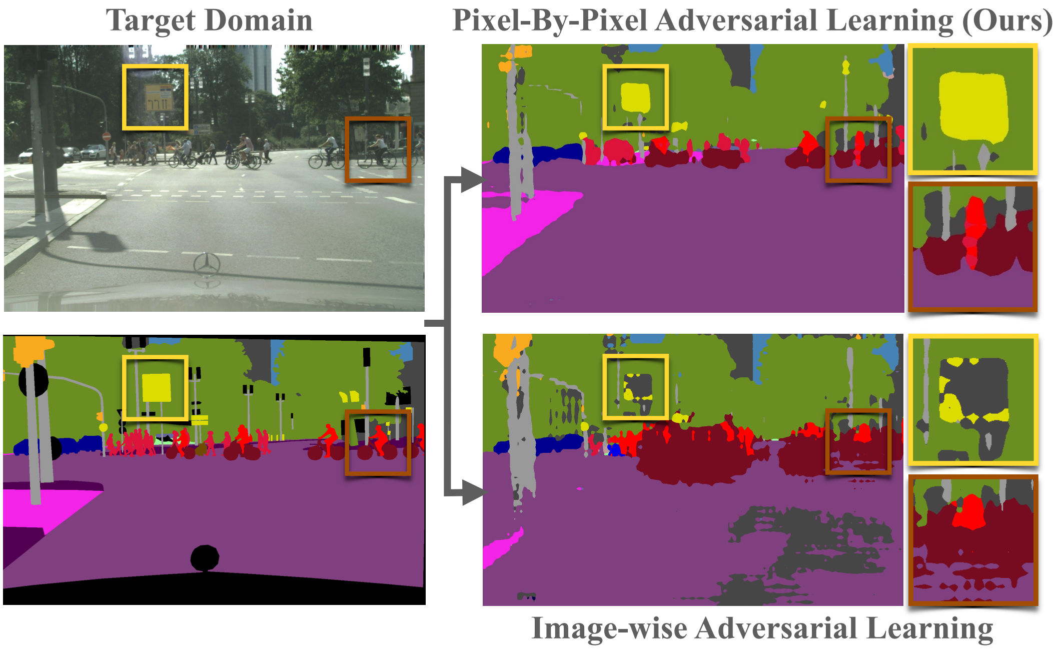

Target Domain
Pixel-By-Pixel Adversarial Learning (Ours)
Image-wise Adversarial Learning

Fig. 3.11 A comparison between traditional image-wise adversarial training (bottom) and our approach, which analyzes each pixel individually (top). By prioritizing pixel alignment based on class imbalance and network classification confidence, our method PixDA achieves higher accuracy, particularly for underrepresented semantic classes such as traffic signs, riders, and bicycles. This is in contrast to common image-wise adversarial training, which may not effectively address class imbalances and may result in less accurate mapping of these types of classes.

We believe that addressing the challenge of cross-domain few-shot learning in semantic segmentation requires considering the intrinsic pixel-wise nature of class segmentation. In this context we present the Pixel-By-Pixel Cross-Domain Alignment framework that we named PixDA[3], which utilizes a novel pixel-wise discriminator and modulates the adversarial loss for each pixel to:

- align the pixel-wise source and target domains;
- prevent further alignment of correctly represented pixels and reduce negative transfer;
- regularize the training of underrepresented classes to avoid overfitting.

The pixel-wise adversarial training is aided by a sample selection procedure that helps balance the source and target data by gradually eliminating samples from the

[3] Code can be found at: https://github.com/taveraantonio/PixDA

source domain. These two mechanisms work together in an end-to-end training process. We evaluate our architecture on the standard synthetic-to-real benchmarks, GTA 5→Cityscapes and SYNTHIA→Cityscapes, where it sets new state-of-the-art scores.

3.2.1 Methodology

The Pixel-Wise Adversarial Loss

Several techniques for domain adaptation in semantic segmentation have been developed to address the issue of domain shift, including those described in [62–64]. These approaches aim to resolve the discrepancy between source and target domains by using adversarial alignment of the features extracted from both domains. The standard approach, first proposed in [124], involves a min-max game between the segmentation network and an image-wise domain discriminator. In the context of this game, the role of the discriminator is to determine the domain that a particular feature belongs to. On the other hand, the segmentation network endeavors to render the source and target features indistinguishable, making it challenging for the discriminator to make accurate predictions.

Since the domains are analyzed and aligned from a global perspective, the discriminator may not consider parts of the scene that only contain a few pixels of small classes, and instead focus mainly on well-represented classes. As a result, adversarial training may primarily align large, well-represented classes, causing negative transfer [63] for other classes and leading to poor adaptation. This problem is exacerbated in the few-shot scenario due to the mismatch in the number of images between the source and target domains, as well as the possibility of some target semantic classes being underrepresented or even absent.

To mitigate the issue of class imbalances and minimize the negative transfer effect, we present a novel adversarial loss that focuses on each pixel individually, rather than working at a global level. This is demonstrated in Figure 3.12. Our goal is to prioritize and improve pixel alignment using three criteria:

- align the source and target domains;

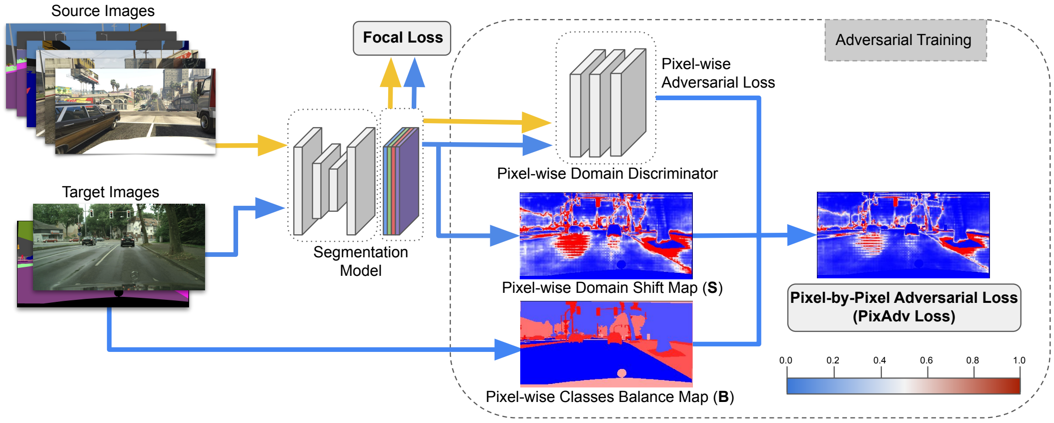

Source Images
Target Images
Segmentation Model
Focal Loss
Adversarial Training
Pixel-wise Adversarial Loss
Pixel-wise Domain Discriminator
Pixel-wise Domain Shift Map (S)
Pixel-wise Classes Balance Map (B)
Pixel-by-Pixel Adversarial Loss (PixAdv Loss)
0.0
0.2
0.4
0.6
0.8
1.0

Fig. 3.12 The PixDA method uses adversarial learning to improve pixel-level accuracy in image segmentation. This is achieved by introducing a new pixel-wise discriminator that computes the adversarial loss. The adversarial loss at each pixel is influenced by two factors: S and B. S evaluates the model's capability to accurately depict the pixel in question, while B assigns a weight to each pixel based on the prevalence of its semantic class in the dataset. These two terms work together to weight the contribution of the adversarial loss at each pixel. The source domain is represented by yellow lines, while the target domain is represented by blue lines. This approach helps to improve the generalization of the model to the target domain by considering the specific characteristics of each pixel and the class it belongs to.

- prevent further alignment of correctly represented pixels to limit negative transfer;
- regularize the training of infrequent classes to avoid overfitting.

To achieve this, we exploit a pixel-wise discriminator whose aim is to determine, for each pixel, which domain it belongs to. The domain discriminator is a modified version of the DCGANs [125] in order to be lighter and more efficient. The discriminator D is trained to distinguish whether the features are sourced from the source domain or the target domain. Formally, we minimize the following loss:

$$L_D(x_s, x_t) = -\sum_{i \in \mathscr{I}} \log \, D_i(f_\theta(x_s)) + \log(1 - D_i(f_\theta(x_t))), \tag{3.1}$$

where D is the discriminator, and $D_i(x)$ indicates the probability that pixel i belongs to the source domain.

We propose a novel adversarial loss function, denoted as L_{PixAdv}, which aims to address the negative transfer effect that can occur when using a pixel-wise discriminator without considering the class imbalance problem. The L_{PixAdv} loss function is designed to align each pixel according to its importance, and is given by equation (3.2):

$$L_{\text{PixAdv}}(x_t, y_t) = -\frac{1}{|\mathscr{I}|} \sum_{i \in \mathscr{I}} S_i(x_t, y_t) B_i(y_t) \log D_i(f_\theta(x_t)). \tag{3.2}$$

In this equation, $S_i(x_t, y_t)$ is a measure of the network's ability to represent each pixel i and is given by equation (3.3):

$$S_i(x_t, y_t) = -y_i \log p^{y_i} i(x_t), \tag{3.3}$$

where $p^{y_i} i(x)$ is the probability for class y_i at pixel i. Large values of S_i indicate that the network is misrepresenting pixel i, while small values indicate that the network is able to correctly represent and classify it.

On the other hand, the term $B_i(y_t)$ in equation (3.2) aims to re-balance the contribution of each class based on their frequency in the target dataset and is given by equation (3.4):

$$B_i(y_t) = 1 - \frac{1}{|\mathscr{I}|} \sum_{j \in \mathscr{I}} \mathbb{1} y_j = y_i, \tag{3.4}$$

where $\mathbb{1}$ is the indicator function, equal to 1 when y_j and y_i are equal and 0 otherwise. Values of Bi close to 1 correspond to a class that is misrepresented, while values near to 0 correspond to a well-represented semantic class. The term B is essential because the target domain usually has numerous pixels of some classes (e.g., road, sidewalk, sky) but very few of other classes (e.g., train, person). By using the B term, we are able to balance the classes and achieve a more effective adaptation.

Intuitively, the domain alignment for each pixel is modulated by the combination of S and B, measuring respectively how well a pixel classification is accurate and its frequency and thus providing more (↑) or less (↓) strength to the adversarial loss according to the following scenarios:

- ↑↑: for high values of S_i (poorly represented) and high values of B_i (infrequent class);
- ↑: for high values of S_i (poorly represented) but low values of B_i (frequent class);

- $\downarrow$: for low values of S_i (correctly represented) but high values of B_i (infrequent class);
- $\downarrow\downarrow$: for low values of S_i (correctly represented) and low values of B_i (frequent class).

The total loss function for training the segmentation network can be expressed as:

$$\frac{1}{|X_s^k|}\sum_{x_s\in X_s^k} L_{\text{seg}}(x_s,y_s)+\frac{1}{|X_t|}\sum_{x_t\in X_t} L_{\text{seg}}(x_t,y_t)+\lambda L_{\text{PixAdv}}(x_t,y_t), \tag{3.5}$$

where λ is a weighting hyperparameter, X_s^k is a subset of the source dataset X_s that has been selected using our sample selection procedure, L_{PixAdv} is the proposed adversarial pixel-wise PixAdv loss term, while L_{seg} represents the segmentation loss (focal loss) that is defined as:

$$L_{\text{seg}}(x,y)=-\frac{1}{|\mathscr{I}|}\sum_{i\in\mathscr{I}}(\alpha(1-p_i^{y_i}(x))^{\gamma}\log(p_i^{y_i}(x)) \tag{3.6}$$

with $\alpha(1-p_i^{y_i}(x))^{\gamma}$ is its modulating factor.

The Sample Selection procedure

The synthetic source dataset used in our method may contain samples that are significantly different from the target domain, such as those with different perspectives or illumination conditions. Using these samples to train the model could lead to negative transfer and negatively affect the model's performance on the target dataset. To address this issue, we propose a sample selection approach that works in collaboration with the PixAdv loss to enhance the utilization of the source data by pinpointing and selecting source samples that are more analogous to the target domain.

To achieve this, we train a global image-wise domain discriminator D_g simultaneously with the segmentation model. The discriminator is used to distinguish between source and target images and to capture both semantic and visual information about the domains. The loss function for the discriminator is defined as:

$$L_{D_g}(x_s,x_t)=-\log\ D_g(f_\theta(x_s))-\log(1-D_g(f_\theta(x_t))). \tag{3.7}$$

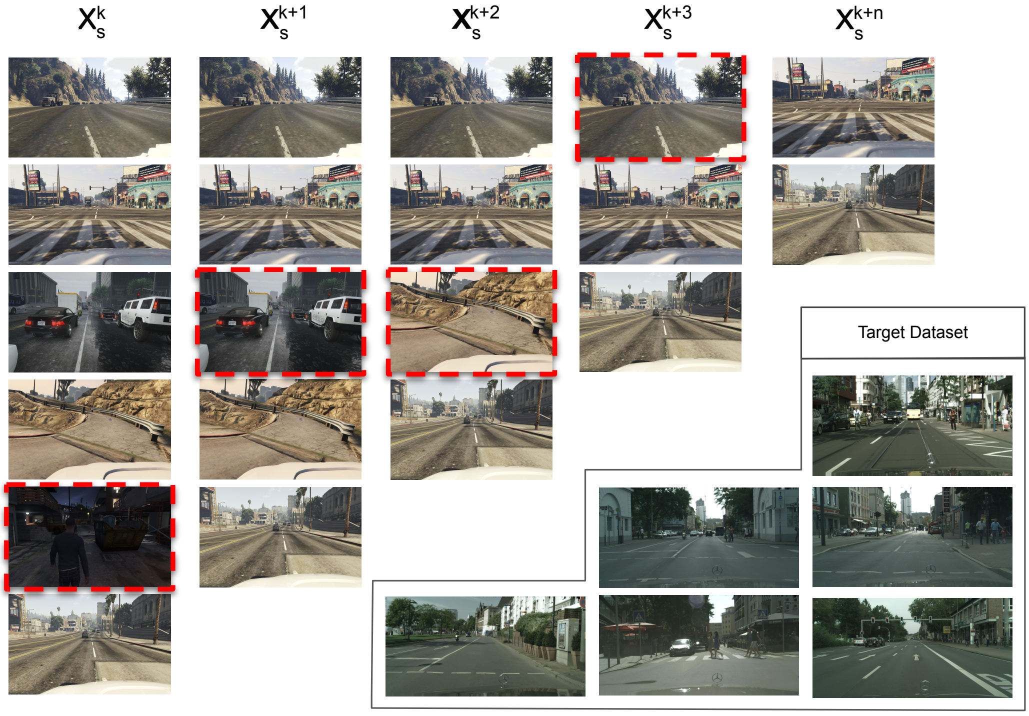


Fig. 3.13 Our method includes a sample selection mechanism that helps to ensure that the model is only trained on source samples that are most similar to the target domain. This is achieved by subsampling the source dataset at each epoch k and selecting only those images that are deemed to contain information relevant to the target domain. For example, images with different perspectives or lighting conditions relative to the target data may be discarded, as they are less likely to be useful for improving the model's performance on the target dataset. The illustration in the top-left corner of the figure shows the sample selection process, while the bottom-right corner shows examples of source samples that might be discarded due to their differences from the target data.

At each epoch k, we use the domain discriminator to predict the likelihood that a source image contains relevant knowledge to be transferred in the target domain, and use this prediction to select a subset X_s^k of source images to retain from the previous epoch. Specifically, we include an image $x_s \in X_s^{k-1}$ in X_s^k if $D_g(x_s) < \delta$, where δ is a predefined threshold. As training progresses, we gradually increase the threshold to select an ever-decreasing number of relevant samples, as shown in Figure 3.13.This helps to ensure that the model is only trained on source samples that are most similar to the target domain, which can improve the model's generalization to the target dataset.

The last fine-tuning

When the adversarial training process is completed, we can further take advantage of the available semantic information in the target data to improve the network's representation and confidence. However, simply fine-tuning the model on the target data may lead to overfitting to the few target images available. To avoid this issue, we use a knowledge distillation strategy to regularize the training of the segmentation model, which we refer to as the student network, using the output of a frozen copy of the same network, which we refer to as the teacher network.

The student network is trained to minimize the following loss function:

$$\frac{1}{|X_t|}\sum_{x_t \in X_t} L_{\text{seg}}(x_t, y_t) + \lambda_{kd} L_{\text{kd}}(x_t, f_{\theta_T}, f_{\theta_S}), \tag{3.8}$$

where L_{seg} is the focal loss function, λ_{kd} is a weighting parameter, and L_{kd} is the distillation loss defined as:

$$L_{kd} = -\sigma(\frac{f_{\theta_T}(x_t)}{\tau}) \log \sigma(f_{\theta_S}(x_t)), \tag{3.9}$$

where σ represents the softmax function, and τ is a temperature parameter, as in the original knowledge distillation method [126]. The knowledge distillation loss helps to regularize the training of the student network by encouraging it to produce outputs that are similar to those of the teacher network, which has already been trained using the adversarial learning process and is better aligned with the target domain. This can help to prevent overfitting to the target data and improve the generalization of the model.

3.2.2 Experiments

To evaluate the effectiveness of our approach, we conducted experiments on two widely used benchmarks for synthetic-to-real domain adaptation in the literature: GTA 5→Cityscapes (as described in [4]) and SYNTHIA→Cityscapes. (as described in [5]). Both of these benchmarks utilize the Cityscapes dataset [3].

Implementation and Training Details

Our main segmentation module is DeepLab V2 [27] with a ResNet101 [118] model that has been pre-trained on the ImageNet dataset. We have also developed a pixel-wise discriminator, which is a fully convolutional model with two 3x3 kernel convolutional layers, each with a stride of 1 and padding of 1, followed by a final 1x1 kernel convolutional layer with a stride of 1 and padding of 0. The channel numbers for the three layers are 64, 128, and 1, respectively. The image-wise discriminator is a fully convolutional model with five 4x4 kernel convolutional layers, each with a stride of 2 and channel numbers of 64, 128, 256, 512, and 1. Both discriminators include a Leaky ReLU activation function with a negative slope of 0.2 after each layer except the final one. We implemented our method using PyTorch and ran it on two NVIDIA Tesla V100 GPUs with 16GB of memory each. The batch size for training the segmentation model was 4, and we used SGD with an initial learning rate of $2.5 \cdot 10^{-4}$ and a "poly" learning rate decay with 0.9 set as power, momentum of 0.9, and weight decay of 0.0005. The discriminators were trained using the Adam optimizer with a learning rate of 10^{-5} and the same decay schedule as the segmentation model, with momentum values of 0.9 and 0.99. To address the low-level visual domain shift between the source and target domains during both the adversarial training and sample selection phases, we applied the parameterless FFT style translation algorithm from FDA [59] to each source image. Our training process started with a pre-trained model of the segmentation on source data and proceeded until the sample selection module identified pertinent source images for the next epoch. The final fine-tuning section lasted for 200 iterations. We set the value of λ to 0.1 for GTA 5 and 1 for Synthia. The sample selection threshold δ was set to 0.4 and doubled at every epoch, and we set $\lambda_{kd} = 0.5$ and $\tau = 0.5$. The test was conducted without any post-processing.

Compared Methods

In our comparison, we consider several baselines including a Source Only (SO) model, which is a network trained only using the source dataset, a Joint Training (JT) baseline that trains the model for 4 epochs using a combination of source and target images, and a Fine-Tuning (FT) baseline that fine-tunes the Source Only model for 30k iterations on the target domain. All of our method, JT, and FT make use of the

Focal Loss to compute segmentation accuracy. Additionally, we report results for three state-of-the-art methods:

- FDA [59]: it introduces a technique to reduce the distributional difference between the source and target domains by exchanging their low-frequency spectra;
- NAAE [123]: it employs weighting networks to evaluate the similarity between synthetic and real pixels and selectively learn from both data sources;
- FSDA [18]: it utilizes a two-stage adversarial network that includes a scene parser and two discriminators.

The FDA [59] and NAAE [123] methods are implemented using the same hyperparameters as in their original papers, with the exception of replacing the target train set with a K-shot selection. For FSDA [18], we follow the same implementation details and results as reported by the authors. All of the baselines use DeepLabV2 with a ResNet101 backbone, with the exception of NAAE, which uses a FCN [20] with a VGG16 [127] model as specified by the authors.

GTA 5→Cityscapes Results

The results for the GTA 5→Cityscapes experiments are presented in Table 3.6 above. At first glance, we can observe that NAAE and Joint Training both produce unsatisfactory results, with a mIoU lower than 40.0% in all tests. FDA performs a bit better, but its accuracy doesn't improve when the number of target images is augmented from 1 to 5. Fine-Tuning the model pre-trained on the source domain leads to accuracy levels comparable to the current state-of-the-art, FSDA. Our PixDA method outperforms all other approaches in all 1-5 shot tests, with a minimum improvement of +20.5% in the 1-shot setting and a maximum of +24.9% in the 5-shot setting compared to the Source Only model. In comparison to the next best performer, FSDA, PixDA achieves an average improvement of +3.6% in mIoU. We also observe that in the 1-shot setting, FSDA's accuracy in certain classes (traffic light, motorcycle) falls below the Source Only baseline, indicating a negative transfer. This suggests that PixDA is more effective in utilizing the information from domain images based on the content of the target images.

Table 3.6 GTA 5→Cityscapes experiments. Semantic classes are in a decreasing order according to their frequency on target domain. SO stands for Source Only, JT for Joint Training while FT for Fine-Tuning.

		Well-represented Classes											*Under*-represented classes										
Shot	Method	Road	Building	Vegetation	Sidewalk	Car	Sky	Pole	Terrain	Person	Fence	Wall	TSign	Bicycle	Bus	Truck	Train	TLight	Rider	Motorcycle	mIoU^{19}	mIoU^{Well}	mIoU^{Under}
	SO	56.7	73.9	78.8	5.8	33.0	73.7	27.7	19.6	55.5	19.2	11.2	20.2	17.9	5.3	16.0	0.0	30.7	20.3	18.2	30.7	41.4	20.1
1	NAAE	84.5	69.2	71.2	35.8	56.2	67.0	14.6	18.5	28.2	7.6	12.5	17.4	0.1	13.7	13.7	0.1	7.8	0.4	5.1	27.6	42.3	7.3
	JT	66.7	63.7	76.9	18.0	74.9	65.3	17.5	9.4	52.6	5.0	2.9	20.3	26.1	17.3	7.6	0	29.9	13.8	26.7	31.3	41.2	17.7
	FDA	86.6	79.0	83.4	21.3	79.7	66.5	33.3	33.1	58.8	**29.1**	23.2	24.6	37.5	36.3	14.7	4.8	33.5	**28.6**	24.5	42.0	54.0	25.6
	FT	93.4	82.4	85.8	60.5	54.2	**86.7**	37.2	30.8	57.4	10.6	17.1	**41.9**	51.8	32.8	34.3	0.0	31.7	8.2	13.5	45.3	56.0	26.8
	FSDA	**94.5**	82.8	84.7	**62.7**	84.9	83.6	32.6	39.8	54.9	19.2	25.0	36.9	48.6	29.4	31.9	0.0	28.8	25.2	16.8	46.4	60.4	27.2
	PixDA	93.3	**84.1**	**85.9**	59.2	**85.8**	86.2	**37.2**	**45.1**	**63.1**	28.1	**28.3**	32.1	**55.0**	**50.7**	**35.0**	**8.5**	**38.4**	26.8	**29.4**	**51.2**	**63.3**	**34.5**
2	NAAE	88.8	70.8	75.9	40.1	62.7	76.5	16.2	25.7	31.9	18.8	13.0	17.7	10.9	24.4	16.2	2.0	12.0	1.1	4.0	32.0	47.3	11.0
	JT	48.1	50.3	81.0	20.0	77.8	77.1	21.0	9.1	56.2	13.4	11.5	17.1	28.3	8.8	29.1	1.8	30.6	26.8	30.0	33.6	42.3	21.6
	FDA	83.0	74.6	79.3	31.2	64.5	72.2	28.0	34.7	59.1	26.6	16.0	21.8	40.7	38.8	28.9	0.7	28.3	**28.9**	28.2	41.3	51.7	27.0
	FT	**95.2**	84.0	85.1	**67.9**	84.9	**86.7**	37.8	44.2	57.2	26.7	21.0	**43.3**	55.0	35.4	11.1	11.2	29.5	5.7	6.2	46.7	62.8	24.7
	FSDA	94.1	84.7	**86.4**	61.6	84.5	85.1	34.3	43.7	56.0	25.6	**35.9**	37.8	51.4	36.0	**39.7**	2.2	34.5	17.9	21.9	49.1	62.9	30.2
	PixDA	94.7	**85.0**	86.0	64.5	**85.1**	85.3	**38.5**	**46.2**	**61.9**	**30.6**	27.5	38.9	**56.3**	**47.1**	37.3	**25.6**	**37.7**	19.4	**32.5**	**52.8**	**64.1**	**36.8**
3	NAAE	88.9	74.3	78.5	40.5	67.0	79.5	19.3	26.8	41.0	17.7	15.6	25.0	17.5	13.7	19.3	2.9	15.2	11.3	4.8	34.7	49.9	13.7
	JT	65.0	66.7	80.8	22.8	82.2	74.4	18.5	12.0	55.2	8.1	5.6	21.1	21.1	23.8	29.8	0.3	30.5	10.0	28.4	34.5	44.7	20.6
	FDA	86.4	77.9	82.9	26.0	76.8	72.9	32.2	32.9	58.9	22.0	19.9	22.1	38.8	39.4	16.9	2.0	31.8	29.3	25.7	41.8	53.5	25.7
	FT	**95.6**	84.1	86.2	**68.7**	86.5	88.1	**39.5**	46.0	59.7	16.5	18.9	44.1	58.2	18.4	37.7	**18.6**	37.4	16.8	21.4	45.6	62.7	31.6
	FSDA	94.3	85.3	86.3	64.6	83.4	85.9	36.1	45.2	58.3	**27.4**	**35.6**	40.1	56.4	29.3	31.8	11.3	36.9	31.1	29.2	51.0	63.9	33.3
	PixDA	94.3	**85.7**	**86.9**	62.2	**87.7**	**88.9**	38.6	**48.9**	**64.2**	26.6	31.7	**44.1**	**59.8**	**47.2**	**42.2**	15.0	**43.6**	**35.6**	**33.4**	**54.5**	**65.1**	**40.1**
4	NAAE	91.0	75.8	79.4	45.8	70.4	80.7	19.6	30.7	38.7	19.4	18.4	30.6	21.0	13.9	23.2	1.1	19.2	18.4	3.9	36.9	51.8	16.4
	JT	74.0	67.3	81.5	28.4	77.4	73.9	30.7	12.6	57.5	18.5	13.4	26.2	28.3	25.7	15.3	3.6	32.0	25.9	27.9	37.9	48.7	23.1
	FDA	87.2	78.9	83.5	28.4	73.6	68.5	33.4	33.5	60.3	28.1	22.2	19.3	31.6	39.0	32.2	1.1	29.7	30.7	26.5	42.5	54.3	26.3
	FT	**95.5**	84.4	86.7	**67.8**	85.6	89.1	39.8	46.8	62.1	21.4	25.8	45.6	56.6	9.1	29.4	0.0	35.6	33.1	5.4	48.4	64.1	26.8
	FSDA	94.4	85.2	86.7	63.0	85.6	88.8	35.4	45.1	59.2	**29.3**	**38.5**	45.1	53.6	29.2	40.7	0	33.2	32.6	21.1	50.9	64.7	31.9
	PixDA	95.2	**86.1**	**87.0**	67.3	**88.1**	**89.3**	**40.1**	**48.0**	**64.6**	28.9	30.5	**47.7**	**58.2**	**47.2**	**47.5**	**8.5**	**40.1**	**41.3**	**30.0**	**55.0**	**65.9**	**40.1**
5	NAAE	91.3	77.2	81.1	47.2	72.5	80.2	21.4	36.0	44.4	19.9	21.2	33.1	25.5	28.0	28.3	12.9	20.0	10.1	9.6	40.0	53.9	20.9
	JT	63.6	71.9	82.1	28.5	79.5	71.3	26.6	7.8	58.3	9.5	6.6	27.3	30.9	6.3	5.8	2.1	34.3	24.4	25.2	34.8	46.0	19.5
	FDA	86.8	78.4	83.7	33.5	78.2	72.4	31.7	33.3	59.6	20.1	20.9	19.8	28.8	29.8	24.3	4.5	29.8	29.4	14.9	41.0	54.4	22.7
	FT	**95.6**	85.0	**87.3**	**67.7**	87.6	88.4	40.2	43.9	61.7	21.3	25.8	**53.9**	58.4	48.3	**47.5**	14.3	38.1	23.3	27.4	53.5	64.1	38.9
	FSDA	94.6	85.6	86.8	65.1	85.7	78.6	37.3	47.9	60.5	**27.3**	**33.7**	48.6	54.6	45.3	41.2	14.8	36.8	32.1	32.1	53.6	63.9	38.2
	PixDA	95.0	**86.3**	87.2	66.9	**88.4**	**89.4**	**40.4**	**49.3**	**64.6**	24.3	26.6	52.5	**60.4**	**49.0**	45.3	**19.9**	**40.8**	**34.8**	**34.4**	**55.6**	**65.3**	**42.1**

Finally, we can see that our method not only performs well on prevalent classes like "road", "sky", and "building", but also on average improves the recognition of underrepresented classes such as "traffic light" (+9.4% compared to the Source Only model) and "train" (+15.5% compared to the Source Only model). In fact, on underrepresented classes (last column in 3.6), we outperform FSDA by +6.6%.

Figure 3.14 provide more qualitative details and confirms that there is a relatively low visual domain shift between GTA 5 and Cityscapes for some major classes, such as "road", "sky", and "car", as the source-only model is able to correctly classify these classes without domain adaptation. However, this is not the case for classes that are underrepresented in the target domain, such as "bus", "train", "rider", "truck", "traffic sign", and "traffic light". Notably, the source-only model even struggles to classify the "sidewalk" category, which is relatively frequent, due to a high domain shift, as evident in the 1, 4, and 5 shot settings where both the source-only model and the joint training baseline struggle to classify sidewalks.

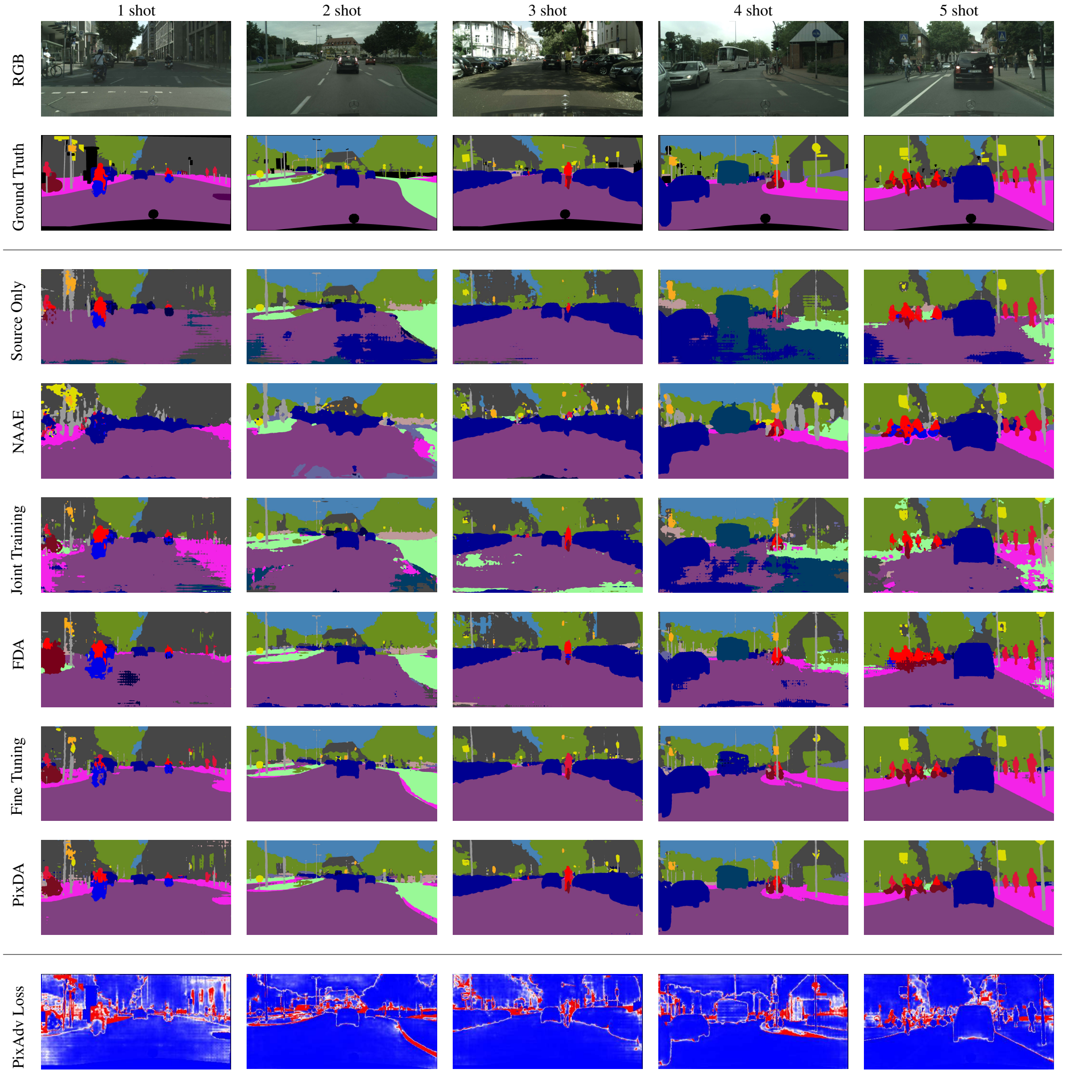


Fig. 3.14 Here the qualitative results for the GTA 5→Cityscapes 1-5 shot scenarios: the columns represent different few-shot settings, from 1-shot (left) to 5-shot (right). The first row shows the RGB image and the second row displays the ground truth. All other rows show the predictions produced by different methods, except for the last row which visualizes the normalized levels of the PixAdv loss L_{PixAdv} used by PixDA: blue indicates a low value while a darker red indicates a higher value. It's important to note that the loss image level is captured at a specific point in time and varies with each iteration of training.

When considering other baselines, including unsupervised domain adaptation with FDA [59] and transfer learning with NAAE [123], we see that although there are improvements compared to the source-only model, these methods still struggle with finer details such as bicycles, motorcycles, and traffic signs. Fine-tuning with a

Table 3.7 SYNTHIA→Cityscapes. experiments. Semantic classes are in a decreasing order according to their frequency on target domain. SO stands for Source Only, JT for Joint Training while FT for Fine-Tuning.

		Well-represented Classes										*Under*-represented Classes									
Shot	Method	Road	Building	Vegetation	Sidewalk	Car	Sky	Pole	Person	Fence	Wall	TSign	Bicycle	Bus	TLight	Rider	Motorcycle	mIoU^{13}	mIoU^{16}	mIoU^{Well}	mIoU^{Under}
	SO	27.0	64.2	74.8	16.7	38.4	77.8	14.7	50.9	0.1	3.7	5.4	16.7	7.0	2.2	10.3	1.3	30.2	25.7	36.8	7.1
1	NAAE	85.8	69.9	72.6	36.3	57.5	67.4	14.9	33.9	3.1	1.8	19.5	0.0	13.6	3.6	2.6	1.9	35.7	30.3	44.3	6.9
	JT	65.9	71.7	76.8	32.2	46.9	83.6	28.9	56.8	0.4	10.9	23.3	25.1	26.4	9.6	16.1	11.8	42.1	36.7	47.4	18.7
	FDA	30.4	52.5	62.0	16.0	61.8	75.1	27.5	51.0	0.1	6.4	16.1	25.2	10.8	5.8	18.1	7.8	33.5	29.3	38.3	14.0
	FT	92.1	80.7	84.2	55.1	81.2	**86.5**	33.8	55.9	**5.4**	9.2	33.2	51.4	20.8	15.2	2.8	4.2	51.0	44.5	58.4	21.3
	FSDA	92.7	**82.1**	**85.0**	56.5	82.0	84.6	-	56.1	-	-	33.0	52.3	25.6	17.4	**23.9**	8.4	53.8	-	-	26.7
	PixDA	**94.2**	82.0	84.1	**60.8**	**84.3**	85.0	**36.5**	**59.0**	3.9	**16.8**	**37.7**	**56.5**	**31.4**	**22.5**	20.6	**8.8**	**55.9**	**49.0**	**60.7**	**29.6**
2	NAAE	87.3	68.2	75.7	34.2	63.5	75.8	14.1	32.7	17.0	5.9	17.4	6.4	11.8	9.1	3.2	2.7	37.6	32.8	47.5	8.4
	JT	78.8	77.7	80.7	35.1	72.1	86.0	30.6	54.6	2.6	**17.1**	21.7	27.3	35.8	12.6	10.5	9.8	46.4	40.8	53.5	19.6
	FDA	24.5	67.0	68.7	13.2	67.4	77.4	28.8	50.7	0.1	8.3	17.1	23.1	12.1	3.7	16.2	14.7	36.1	30.8	40.6	14.5
	FT	94.2	82.3	84.1	**62.5**	83.8	**85.3**	35.6	58.6	18.5	15.2	34.9	55.4	35.4	21.5	10.9	7.8	55.1	49.1	62.0	27.6
	FSDA	92.7	82.3	**85.1**	55.4	79.9	83.1	-	57.0	-	-	35.4	53.0	24.5	**24.5**	15.0	**17.9**	54.3	-	-	28.4
	PixDA	**94.2**	**83.1**	84.4	61.8	**86.2**	84.2	**36.8**	**59.4**	**19.7**	16.6	**36.9**	**57.4**	**36.7**	22.9	**17.4**	10.2	**56.5**	**50.5**	**62.7**	**30.2**
3	NAAE	88.4	75.6	79.0	38.6	63.9	80.4	20.2	39.7	13.6	6.0	25.8	13.2	6.8	17.9	10.6	2.5	41.7	36.4	50.6	37.9
	JT	80.4	75.9	79.2	37.9	71.5	85.8	31.0	56.2	1.5	11.2	22.0	35.6	24.7	14.9	15.6	16.9	47.4	41.3	53.1	21.6
	FDA	15.5	57.0	61.2	12.8	66.0	76.5	26.9	53.0	0.1	11.0	15.9	21.7	20.9	4.1	16.6	9.2	33.1	30.0	38.0	14.7
	FT	**94.8**	82.2	85.0	**64.0**	84.3	88.0	37.0	59.0	11.8	8.4	37.7	57.1	7.1	28.2	22.8	6.0	55.1	48.3	61.4	26.5
	FSDA	94.0	83.8	**85.8**	61.0	84.3	**88.3**	-	59.4	-	-	**40.5**	59.0	11.6	28.0	28.6	**15.4**	56.9	-	-	30.4
	PixDA	94.4	**84.1**	85.4	62.6	**85.7**	88.2	**37.7**	**61.5**	**13.7**	**15.6**	38.9	**61.3**	**23.7**	**32.4**	**29.3**	12.3	**58.4**	**51.7**	**62.9**	**33.0**
4	NAAE	90.8	74.8	79.9	47.8	67.6	81.5	20.3	43.4	14.5	10.9	28.7	22.7	13.8	17.6	19.8	1.5	45.4	39.7	53.1	17.3
	JT	67.4	75.8	78.1	35.6	48.1	86.3	31.0	55.9	1.2	11.3	27.6	30.5	24.0	17.2	17.0	9.5	44.1	38.5	49.1	21.0
	FDA	23.4	62.9	69.5	13.6	67.3	80.2	30.3	53.2	0.3	12.7	15.9	26.0	22.5	4.5	18.1	13.2	36.2	32.1	41.3	16.7
	FT	95.3	83.5	85.8	66.5	84.5	**89.3**	37.6	**63.1**	16.5	20.6	43.2	56.8	9.7	27.5	31.1	2.4	56.8	50.8	64.3	28.5
	FSDA	94.4	84.0	**85.9**	62.7	83.1	87.7	-	59.2	-	-	42.0	58.0	10.6	29.7	**35.8**	**20.0**	57.9	-	-	32.7
	PixDA	**95.4**	**84.5**	85.6	**66.6**	**86.2**	88.4	**38.1**	62.7	**18.1**	**21.6**	**45.3**	**58.1**	**26.0**	**33.2**	34.7	6.9	**59.5**	**53.2**	**64.7**	**34.0**
5	NAAE	91.3	75.9	80.9	48.3	75.0	82.7	20.3	45.3	16.0	10.3	32.3	30.9	19.6	22.3	19.1	10.2	48.8	42.5	54.6	22.4
	JT	65.2	75.7	79.1	27.5	64.9	85.5	31.8	56.7	2.1	13.5	22.9	34.1	29.9	14.7	20.0	15.4	45.5	39.9	50.2	22.8
	FDA	18.6	62.1	64.5	13.2	71.9	79.8	31.0	50.6	0.2	11.7	21.2	16.2	20.4	5.8	17.4	5.6	34.4	30.6	40.4	14.4
	FT	95.0	83.7	85.9	65.3	86.3	87.7	38.1	62.2	**13.6**	19.5	50.0	60.3	**42.7**	29.6	25.0	16.0	60.7	53.8	63.7	37.3
	FSDA	94.4	84.5	**86.1**	63.3	86.3	**88.2**	-	61.3	-	-	50.9	58.6	35.5	30.9	28.2	**24.2**	60.9	-	-	38.0
	PixDA	**95.5**	**84.6**	86.0	**66.7**	**87.2**	88.1	**39.0**	**63.5**	13.1	**23.3**	**50.9**	**60.3**	32.8	**33.8**	**33.3**	21.6	**61.9**	**55.0**	**64.7**	**38.8**

few images also suffers from similar issues and in some cases produces completely incorrect predictions (e.g., the bus in the 4-shot setting).

In contrast, our method, consistently performs well in all settings, providing good predictions across all classes and being overall the closest to the ground truth. Analysis of the PixAdv loss reveals that it places greater weight (dark blue) on those areas that are challenging for the other methods, such as classes with a high domain shift ("sidewalk") or with fewer pixels in the target domain ("bicycle", "motorcycle", or "signs"). It is worth noting that PixDA is able to achieve excellent results even in the 1-shot setting, where all other methods struggle with underrepresented classes ("bicycle", "motorcycle", "traffic sign", "traffic light") and some of them even struggle with the predominant ones.

SYNTHIA→Cityscapes Results

The results for the SYNTHIA→Cityscapes scenario are shown in Table 3.7 and confirm the findings from the first set of experiments. NAAE, FDA, and Joint Training all perform inadequately, further affirming the deduction that they are not suitable solutions for tackling the cross-domain few-shot task. Fine-Tuning and FSDA show similar accuracies, although it should be noted that the results for FSDA are only reported for the protocol with 13 classes, which excludes the difficult categories "pole", "fence", and "wall" from evaluation. Our method is designed to handle these categories better and improves its performance compared to the Source Only model by +22.9%, +13.6%, and +15.1%, respectively, for these classes. Overall, PixDA shows the best mIoU in all the 1-5 shot settings for both the 13 and 16 class protocols. It outperforms the Source Only model by a minimum of +25.7% in the 1-shot scenario and a maximum of +31.7% in the 5-shot. Compared to the current state-of-the-art, FSDA, PixDA achieves an average accuracy improvement of +1.7% within the 13 class protocol and +1.9% when considering only the rare classes. The next best performer, Fine-Tuning, performs well on these three categories but is less consistent than our solution. In the 16 class protocol, PixDA achieves an average improvement of +2.6% compared to Fine-Tuning.

One significant difference compared to the GTA 5 experiment is the performance on the "road" and "sidewalk" classes, as shown in Figure 3.15. All baselines, including FDA and NAAE, show poorer results in classifying these semantic categories due to a higher domain shift between SYNTHIA [5] and Cityscapes. The increased difficulty in predicting the "road" and "sidewalk" classes is supported by the PixAdv loss levels, which show a higher emphasis on parts of the road and sidewalk compared to the GTA 5 experiment. As a result, the prediction of these classes using PixDA is the best across all settings. Once again, we observe that PixDA exhibits very consistent results, providing good predictions even in the 1-shot setting where all the other methods struggle.

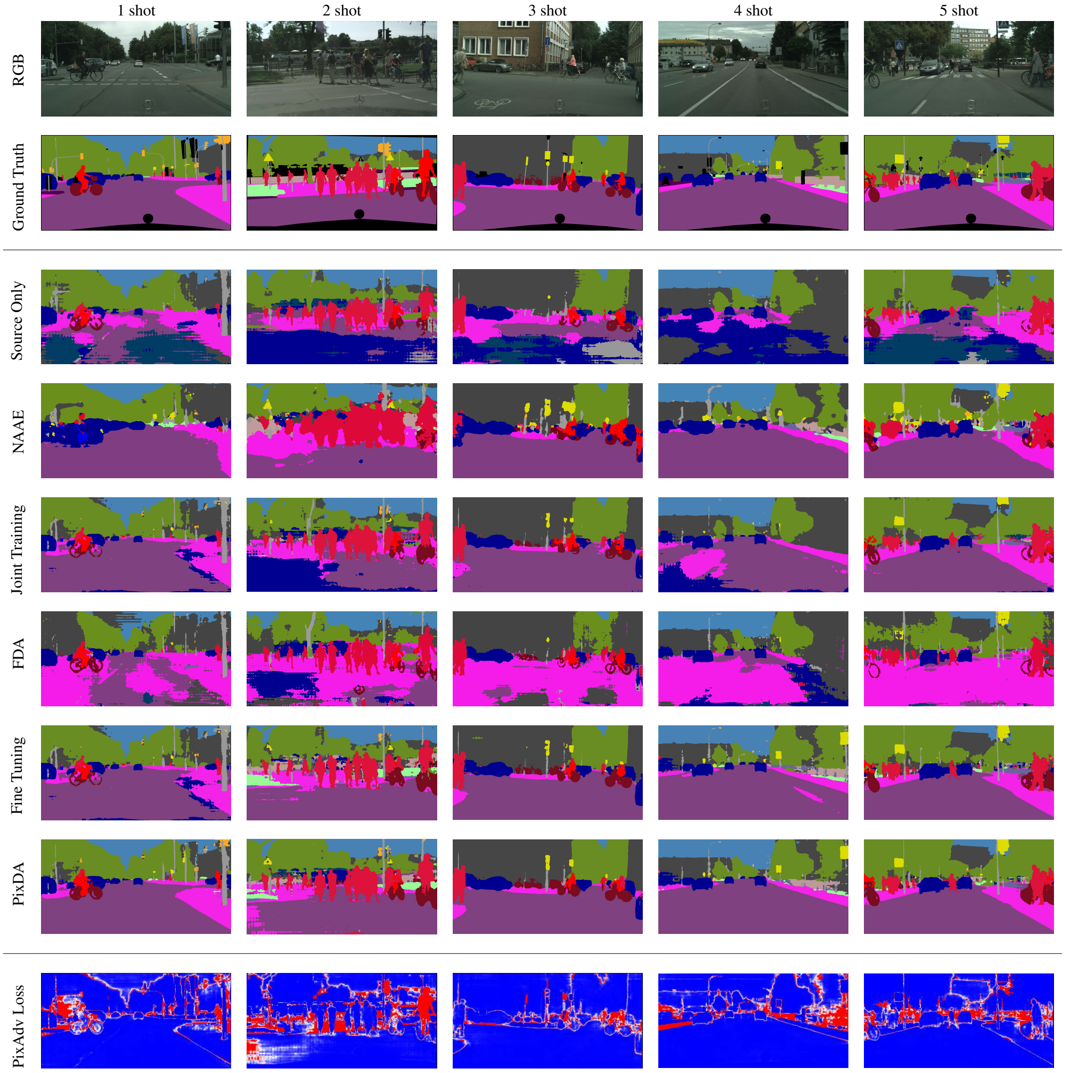


Fig. 3.15 Here the qualitative results for the SYNTHIA→Cityscapes. 1-5 shot scenarios: the columns represent different few-shot settings, from 1-shot (left) to 5-shot (right). The first row shows the RGB image and the second row displays the ground truth. All other rows show the predictions produced by different methods, except for the last row which visualizes the normalized levels of the PixAdv loss L_{PixAdv} used by PixDA: blue indicates a low value while a darker red indicates a higher value. It's important to note that the loss image level is captured at a specific point in time and varies with each iteration of training.

Table 3.8 Ablation study about the choice of the adversarial loss on the GTA 5→Cityscapes 1-shot scenario.

Image-wise Adv. Loss	Pixel-wise Adv. Loss	B	S	mIoU
				31.3
✓				42.4
	✓			44.6
	✓	✓		46.1
	✓		✓	47.7
	✓	✓	✓	**48.3**

3.2.3 Ablation Studies

PixAdv Loss Components Contribution

In Table 3.8, we present a thorough ablation study to demonstrate the effectiveness of aligning domains at the pixel level. These results were obtained with sample selection, knowledge distillation and fine-tuning switched off. The table shows that aligning the source and target domains using an image-wise discriminator leads to a significant improvement (+11.1%) compared to joint training (no adversarial loss). However, this approach aligns well-represented classes effectively while ignoring others due to its global nature.

On the other hand, the pixel-level adversarial loss, by aligning each pixel individually, further improves performance by +2.2%. Despite this improvement, merely aligning the pixels does not prevent negative transfer and overfitting to few-shot images. Re-weighting the pixels based on their frequency (*B* term) leads to an additional improvement of +1.5%. Additionally, decreasing the weight of well-represented pixels (*S* term) to prevent negative transfer is crucial, resulting in a +1.6% improvement over the pixel-wise adversarial loss.

In conclusion, the combination of the *B* and *S* terms further grants a surge in the performance of the model. The ensuing loss (PixAdv) surpasses the image-wise adversarial loss by +5.9% and the pixel-wise adversarial loss by +3.7%, demonstrating that weighting each pixel's contribution is advantageous to evade negative transfer and overfitting.

Table 3.9 Ablation study showing the effectiveness of each PixDA component on the GTA 5→Cityscapes 1-shot scenario.

Method	mIoU
Source Only	30.7
Joint Training	31.3
PixAdv	48.3
+ Sample Selection	49.7
+ Fine-Tuning	50.1
+ KD	**51.2**

PixDA Components Contribution

In this paragraph, we investigate the contribution of each component in our framework to the final performance. To do so, we consider six different cases: (a) a model that uses only the source data; (b) joint training with both source and target data; (c) training with the PixAdv loss function; (d) training with the PixAdv loss and our sample selection mechanism; (e) fine-tuning the model on the target data; and finally, (f) using knowledge distillation to complete the PixDa framework.

As shown in Table 3.9, each component brings a noticeable improvement to the overall performance of the framework. In particular, the use of the PixAdv loss function leads to a $+17.0\%$ improvement in the joint training case, suggesting that domain alignment is crucial for obtaining good performance. The sample selection mechanism provides an additional $+1.4\%$ improvement, indicating that removing samples that are far from the target distribution can be beneficial. While simply fine-tuning the network on the target data leads to a small improvement ($+0.4\%$), using knowledge distillation results in a larger improvement of $+1.1\%$. It is worth noting that even just adding the PixAdv loss function alone outperforms the state-of-the-art. We also conducted a follow-up test where we replaced the Focal Loss with a standard Cross Entropy, resulting in a lower but still state-of-the-art performance of 48.9%, which confirms the effectiveness of our loss function. Due to space constraints, additional studies on the impact of hyperparameters on the PixDA framework can be found in the supplementary material.

Table 3.10 Ablation on the GTA 5→Cityscapes 1-shot scenario to show the effect of lambda on the PixAdv loss.

λ	mIoU	
	GTA 5 to Cityscapes	SYNTHIA to Cityscapes
1	49.2	**55.9**
0.1	**51.2**	54.5
0.01	50.6	53.4
0.001	49.1	53.7

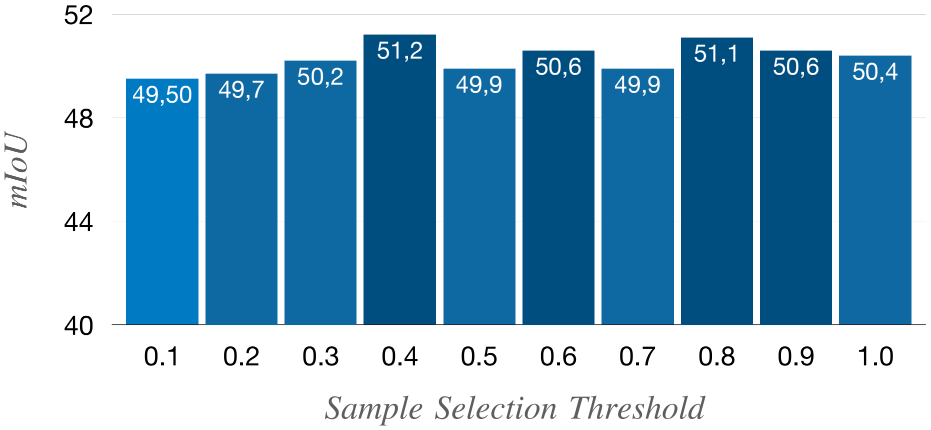


Fig. 3.16 Ablation study about the choice of the sample selection threshold δ, performed on the GTA 5→Cityscapes 1-shot scenario.

Lambda

The hyper-parameter λ is adjusted in our experiment, with four different values tested: 1, 0.1, 0.01, and 0.001. As indicated in 3.10, the best results for the 1-shot setting in the GTA 5→Cityscapes task are obtained with $\lambda = 0.1$, while for SYNTHIA→Cityscapes., the optimal value is $\lambda = 1.0$. However, the difference in performance between $\lambda = 1$ and $\lambda = 0.1$ is only 2.0% and 1.4% for GTA 5 and SYNTHIA, respectively. Despite this, our method, PixDA, still outperforms all the baselines by a significant margin even with a sub-optimal choice of λ.

Sample Selection Threshold

We conduct an experiment in the 1-shot GTA 5→Cityscapes scenario, where the source data is sampled with different thresholds (δ). The results depicted in Figure 3.16 demonstrate that iteratively subsampling the source dataset and choosing source samples that align more closely with the target semantic distribution leads to an increase in accuracy. The highest result is achieved for $\delta = 0.4$, which is doubled at each epoch. It is worth noting that using values ranging from 0.1 to 1 results in a change of less than 1.6% in the final performance, yet still maintains state-of-the-art results, indicating a strong robustness.

3.2.4 Findings

In this work, we tackle the challenge of cross-domain few-shot semantic segmentation by proposing a pixel-by-pixel adversarial training approach that utilizes a novel pixel-wise loss and discriminator to more effectively align the source and target domains and mitigate negative transfer. We also incorporate a sample selection method that addresses the imbalance between the source and target domains into our adversarial training framework. Our approach achieves state-of-the-art performance in all 1- to 5-shot settings across two standard synthetic-to-real benchmarks.

3.3 Learn to generalize: AIAS

In this section, we examine the generalization problem of Semantic Segmentation when applied for aerial imagery analysis in agriculture. We have found that the state-of-the-art approaches in Semantic Segmentation currently used for autonomous driving do not take into account two key features of the aerial data: (i) the top-down perspective implies that the model has to adapt to a flexible semantic arrangement of the scene, as the same scene can be observed from various sensor perspectives; (ii) there may be a significant imbalance in the distribution of semantic classes, as the relevant objects in the scene can appear at vastly different scales (e.g., a field of crops versus a small building). To address these issues, we propose a solution based on two ideas: (i) we employ a strategic blend of appropriate data augmentation and a consistency loss to direct the model in acquiring semantic representations that can withstand the photometric and geometric variations commonly encountered in a top-down perspective (Augmentation Invariance); (ii) similar to the Sample Selection introduced with the PixDA framework, we employ a sampling method (Adaptive Sampling) that selects training images based on a measure of the pixel-wise distribution of classes and the actual network confidence. We present convincing evidence of the efficacy of our proposed strategies through a thorough series of experiments conducted on the Agriculture-Vision dataset, showcasing a remarkable improvement in performance compared to the current state-of-the-art method.

The use of remote aerial images for environmental monitoring has seen significant growth in recent years, with applications including land cover categorization [43, 42], wildfire delineation [128], and identification of deforested regions [129]. Deep learning models have been particularly successful in this field, aided by the availability of open datasets and large collections of aerial images [41, 42]. However, many of these deep learning models were initially designed for other purposes, such as self-driving vehicles [16] or medical imaging [21], and were subsequently applied to aerial images without considering their specific characteristics, thus letting the model not be able to generalize well. In particular, aerial image segmentation has two unique features: a top-down perspective and an extreme class imbalance. More specifically:

- **top-down perspective**: in remote sensing the images are obtained from a camera mounted on an aircraft that point towards the ground. This perspective

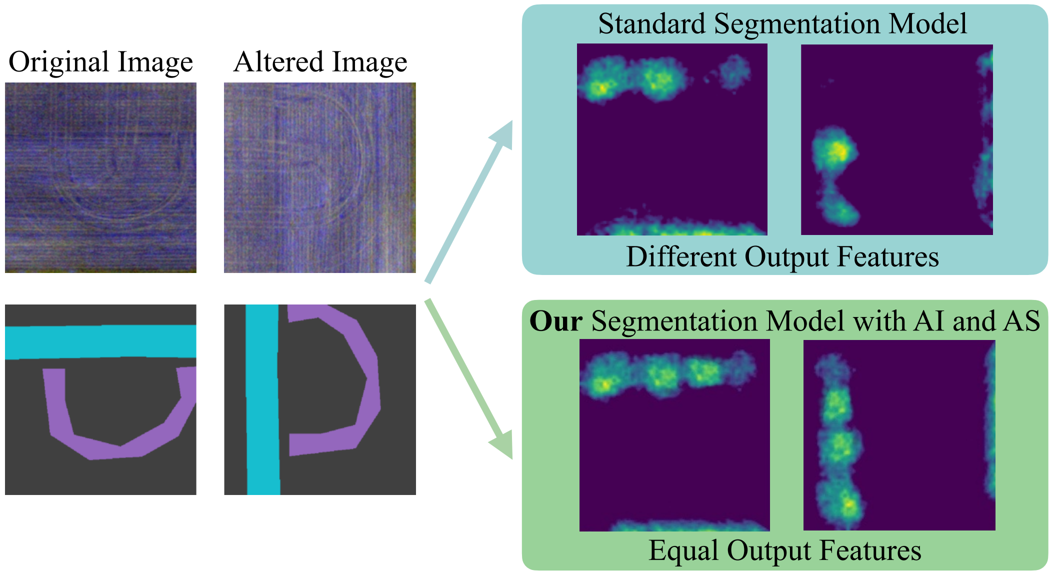

Original Image
Altered Image
Standard Segmentation Model
Different Output Features
Our Segmentation Model with AI and AS
Equal Output Features

Fig. 3.17 It happens that a semantic segmentation model, which is not programmed to anticipate variations in perspective, may generate distinct output features for the same image depending on the angle from which it is viewed. In contrast, our technique ensures that the model is insensitive to these shifts in viewpoint, thereby incentivizing it to acquire more durable representations.

can lead to a lack of depth and reference points in the images and allows for the same scene to be captured with arbitrary rotations around the vertical axis. In contrast to autonomous driving datasets [3, 6], where the model is accustomed to a structured organization of semantic elements in the scene (e.g., the road is typically found at the bottom of the image and the sky at the top), this is not the case in aerial imagery;

- **class imbalance**: in which some classes have significantly more pixels than others, is a common issue in semantic segmentation. However, in aerial images this issue is exacerbated because the entities to be recognized can vary greatly in size, from small vehicles to large natural scenes.

We believe that a semantic segmentation model that is specifically designed to handle the unique characteristics of aerial images will be more effective and able to generalize also in this particular task. To this end, we propose a solution comprising two components:

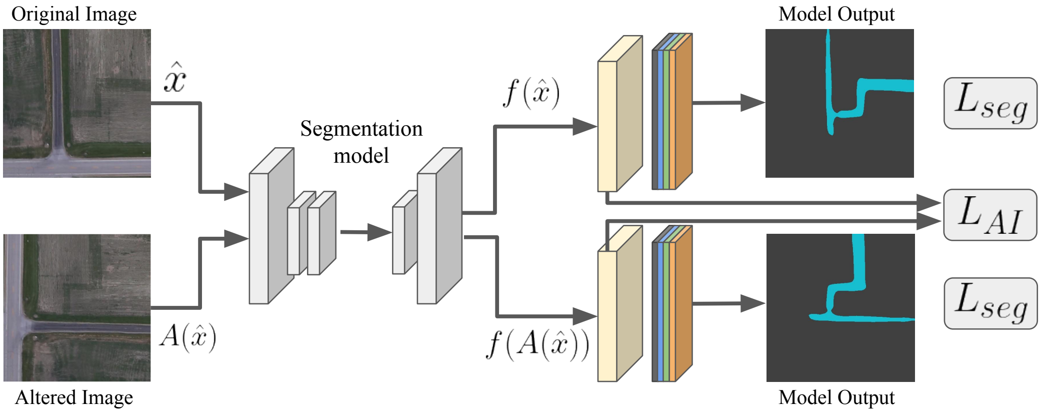


Fig. 3.18 The overall framework is illustrated in the following way: the Adaptive Sampling module selects a sample and generates an augmented version of it. Both images are then input to the segmentation model, which calculates the segmentation loss L_{seg}. Conversely, the L_{AI} loss compels the model to extract consistent features from both the original image and its transformed counterpart.

- **Augmentation Invariance (AI)**: that employs augmentations to teach the model to learn representations that are invariant to changes in appearance and perspective, such as rotations around the vertical axis;
- **Adaptive Sampling (AS)**: that aims to regularize the training of underrepresented classes by adaptively sampling training images based on the distribution of pixels and the actual network confidence.

These two components work together in an end-to-end training process[4]. We carried out an extensive series of experiments on the Agriculture Vision dataset [46], the only available dataset for aerial agriculture that boasts a diversity of semantic categories and complexity. In these experiments, we explored the use of only RGB images for training as well we introduce a new training protocol which comprises the incorporation of NIR data in the training process. Additionally, we conducted an extensive ablation study to evaluate the individual contributions of each solution we introduced.

3.3.1 Methodology

Augmentation Invariance

Our approach, illustrated in Figure 3.18, builds upon the SegFormer architecture [33]. To be more precise, we improve this architecture by incorporating a loss term that harmonizes the pixel embeddings generated by the Transformer network for both the original image and its augmented form. This encourages the model to learn semantic representations that are invariant to photometric distortions and perspective changes, which are common in aerial images.

At present, state-of-the-art frameworks for autonomous driving often struggle to deliver good performance when applied to aerial data. We have identified several reasons for this:

- Aerial imagery does not demand a rigid viewing perspective, and in particular, the camera orientation around the vertical axis is unrestricted and unconfined;
- Aerial images may exhibit significant distortions due to the angle of the camera;
- There can be significant photometric variations across different fields.

To address these issues, we propose a mechanism called Augmentation Invariance (AI) that uses augmentations to guide the model in learning a mapping that is invariant to shifts in perspective and appearance.

We begin by designating $\mathscr{Z}$ as the set of images captured in the Near-Infrared (NIR) spectrum. Subsequently, we introduce $\hat{x} \in \hat{\mathscr{X}}$ as a four-channel image that is created by combining channel-wise the input image x, as defined in Chapter 2, and the corresponding NIR data z.

To implement Augmentation Invariance, we first input an image $\hat{x}$ and extract its pixel-wise features $f_i(\hat{x})$ from the second-to-last layer of the SegFormer architecture, skipping the last layer used for pixel-wise segmentation. We then create a transformed copy of $\hat{x}$ using a combination of geometric augmentations A_g (such as horizontal flipping, vertical flipping, and random rotation) and a photometric augmentation A_p (such as color jitter), denoted as $A_p \circ A_g = A$. The transformed image $A(\hat{x})$ is also passed through the model to extract its features $f_i(A(\hat{x}))$. In order

[4] Code can be found at: https://github.com/taveraantonio/AIAS

to motivate the model to remain constant in the face of shifts in perspective and appearance, we impose the condition that the features extracted from the original image $\hat{x}$ align with the features obtained from the transformed image $A(x)$, after undoing the geometric augmentation.

To encourage the model to learn invariant representations, we use a pixel-wise mean squared error loss L_{AI} defined as:

$$L_{\text{AI}}(\hat{x}, A(\hat{x})) = \frac{1}{\mathscr{I}} \sum_{i \in \mathscr{I}} (f_i(\hat{x}) - A_g^{-1}(f_i(A(\hat{x}))))^2 \tag{3.10}$$

where A_g^{-1} denotes the inverse of the geometric augmentations applied to $\hat{x}$, which guarantees that we are comparing the original and augmented features associated with the same pixel. We also maintain the ground truth annotations for the augmented images, so that the same segmentation loss can be applied to them. The total training loss, L_{tot}, is given by:

$$L_{\text{tot}} = L_{\text{seg}}(\hat{x}, y) + L_{\text{seg}}(A(\hat{x}), A_g(y)) + \lambda L_{\text{AI}}(\hat{x}, A(\hat{x})), \tag{3.11}$$

where $A_g(y)$ denotes the same geometric transformation applied to the ground truth annotation y, λ is a scaling factor and L_{seg} is a standard cross-entropy loss:

$$L_{\text{seg}}(\hat{x}, y) = -\frac{1}{|\mathscr{I}|} \sum_{i \in \mathscr{I}} \sum_{c \in \mathscr{C}} y_i^c \log(p_i^c(\hat{x})), \tag{3.12}$$

It is worth noting that the Augmentation Invariance mechanism employed here differs from conventional data augmentation techniques, as it does not simply rely on photometric and geometric transformations to increase the size of the training dataset by incorporating examples outside the original data distribution. Instead, through the loss L_{AI}, we also use the original and transformed images paired together to provide stronger guidance to the training process.

Adaptive Sampling

The issue of class imbalance in semantic aerial data, where some classes are rarely present and others are extremely common, can be addressed using an Adaptive Sampling (AS) approach that is combined with Augmentation Invariance. This approach involves selecting a sample of images to train the network based on both the global, pixel-wise distribution of classes and the class-wise confidence of the network. The data sampler will prioritize images with low-frequency categories and those for which the network has low confidence.

The Adaptive Sampling probability for a class c, denoted as AS_c, is defined as follows:

$$AS_c = \sigma((1 - dist * conf)^{\gamma}), \tag{3.13}$$

where $dist$ is an array representing the class distribution, $conf$ represents the network's confidence in each class, σ is a min-max normalization function, and γ is a relaxation parameter. Once a semantic category has been selected using this probability, an image is chosen randomly from a subset X_c containing that class. The definitions of $dist$ and $conf$ are used to compute AS_c in Equation 3.13:

- ***dist***: Since we are working in a supervised learning environment, we can calculate a static, fixed estimate of the distribution of pixels for each semantic class $c \in \mathscr{C}$ as a preprocessing step. This array, referred to as "dist," reflects the distribution of the classes and is normalized to the range $[0,1]$ using min-max normalization. Additionally, for each class c, we maintain a subset of images X_c in which that category is present.

- ***conf***: During training, we compute the confidence of the network for each class and store the results in an array called $conf$ with size $|C|$. At each iteration stage t, we compute the pixel-wise Softmax probabilities from the prediction logits for the current batch. The mean confidence value for each class c is then determined by averaging the pixels that belong to that category, using the available ground truth labels. Finally, the actual network confidence at step t is calculated as an exponential moving average of the previous confidence at step $t-1$:

  $$conf_t = \alpha conf_{t-1} + (1 - \alpha) conf_t, \tag{3.14}$$

  where α is a smoothing factor.

3.3.2 Experiments

To evaluate the effectiveness of our approach, we follow the protocol described in the Agriculture-Vision paper [46]. As the test set is not available, we measure performance on the provided validation set. We conduct two sets of experiments:

the first experiment only employs RGB images for both training and testing, while the second experiment utilizes both RGB and NIR data.

Implementation and Training Details

Our method is based on the SegFormer architecture [33], which uses a MiT-B5 encoder pretrained on ImageNet-1k as the backbone for the segmentation module. We use the mmsegmentation framework [130], which is based on PyTorch, to develop our framework and reproduce all of the baselines. We train all configurations on two NVIDIA Tesla v100 GPUs with 16GB of RAM each. During training, we use dataset augmentation techniques such as random resizing with a ratio in the range (1.0, 2.0), random horizontal and vertical flipping, and random crops resized to 512x512.

For evaluation, we perform inferences on raw data without any further preprocessing. We train all of the baselines and our model for 80,000 iterations using the AdamW optimizer. The learning rate is set to 6×10^{-5}, the weight decay to 0.01, and the betas are set to (0.9, 0.999). We use a "poly" learning rate decay with a factor of 1.0 and an initial linear warm-up for 1,500 iterations. We do not use class-balanced loss or OHEM approaches as in SegFormer [33]. When training using NIR data, we expand the network input to four channels by doubling the input weights of the red channel.

For the Augmentation Invariance variants, we further modify the available images using horizontal and vertical flipping, random rotation from 0° to 360° with a step of 90°, photometric and perspective distortion with a strength of 0.1. The probability of applying each transform is set to 0.5. We set the value of λ in equation (3) to 0.75. Through a hyperparameter search that compared the values of $\gamma = 1, 2, 4, 6$ and $\alpha = 0.75, 0.85, 0.90, 0.968, 0.99$ on both settings, we set $\gamma = 4$ in equation 3.13 and $\alpha = 0.968$ in equation 3.14.

Compared Methods

We evaluate our method against a variety of baselines, including state-of-the-art semantic segmentation methods from the existing literature:

- FCN [20]: it combines a deep, coarse layer and a shallow, fine layer to merge semantic and appearance information, resulting in highly accurate and detailed segmentations;

- DeepLab V3 [28]: it employs modules that use atrous convolution in cascade or parallel to capture multi-scale context, utilizing multiple atrous rates. Additionally, it features an augmented version of the Atrous Spatial Pyramid Pooling module, which not only examines convolutional features at various scales but also includes image-level features that encode global context to further enhance performance;

- DeepLab V3+ [29]: it extends DeepLab V3 by incorporating a decoder module and leveraging the Xception model and depthwise separable convolution in both the Atrous Spatial Pyramid Pooling and decoder modules to further enhance its capabilities;

- FPN [24]: it constructs a feature pyramid by taking advantage of the inherent multi-scale of deep convolutional networks, utilizing a top-down architecture with lateral connections to create high-level semantic feature maps at all scales;

- UperNet [25]: it is a multi-task framework that leverages the hierarchical structure of features within a single network;

- PSPNet [26]: it expands the FPN capabilities by not only employing traditional dilated fully convolutional networks, but also integrating a specially designed global pyramid pooling approach to capture pixel-level features;

- HRNetV2 [23]: it maintains high-resolution representations throughout training and progressively adds high-to-low resolution convolution streams one after the other, while simultaneously connecting the multi-resolution streams in parallel. The V2 version of this network combines representations from all high-to-low resolution parallel streams to improve its overall performance;

- HRNetV2+OCR [31]: it incorporates the object-contextual representation (OCR) scheme to HRNetV2;

- SegFormer [33]: it features a hierarchically structured Transformer encoder that generates multiscale features and does not rely on positional encoding. It includes a simple yet effective decoder.

Table 3.11 In these experiments, we use only RGB images for both training and testing on the Agriculture Vision dataset.

Method	Background	Double Plant	Drydown	Endrow	Nutrient Deficiency	Planter Skip	Water	Waterways	Weed Cluster	mIoU
FCN	70.0	16.9	45.6	0.2	13.7	6.6	42.3	0.5	8.5	22.9
DeepLab V3	66.3	17.0	40.6	9.5	16.4	10.0	17.1	12.3	10.0	22.1
DeepLab V3+	68.6	16.3	46.4	6.5	16.1	4.6	16.6	19.1	13.9	23.1
UperNet	65.8	15.8	38.0	10.1	17.3	11.1	4.5	15.5	16.9	21.7
SFPN	69.7	10.6	49.5	2.7	11.5	4.8	35.7	9.9	11.2	22.8
PSPNet	68.1	16.9	45.8	4.9	19.0	8.5	11.3	17.6	17.2	23.3
HRNetV2	71.2	16.8	55.1	5.2	18.6	13.3	13.0	21.2	14.1	25.4
HRNetV2+OCR	72.4	19.5	56.8	12.3	17.3	21.3	28.4	24.6	18.1	30.1
SegFormer	74.9	33.2	**59.7**	18.3	**31.6**	39.2	78.0	**41.5**	28.3	45.0
Ours	**75.5**	**37.0**	58.5	**22.7**	31.3	**41.4**	**80.2**	40.1	**30.4**	**46.4**

The backbone for the first six models is the ResNet-50. HRNetV2 and HRNetV2+OCR use HRNetV2-W18. The SegFormer architecture use the MiT-B5 encoder. All of these backbones are pretrained on ImageNet.

RGB Experiments Results

The results of the experiments using only RGB data are shown in Table 3.11. These results demonstrate the difficulty of the task, as the average mIoU is only 23.0% when all of the baseline approaches are considered, except for the transformer-based architectures. The average mIoU increases to 32.7% when the OCR and SegFormer approaches are included. The UperNet approach has the lowest mIoU at 21.7%, but it performs well in segmenting underrepresented classes such as "double plant," "waterways," and "weed cluster," as it is designed to capture multi-scale information. When transformer architectures are used, significantly better results are obtained, with an mIoU of 30.1% using the HRNetV2+OCR technique and 45.0% using SegFormer. This represents an improvement of 8.4% and 23.3%, respectively, over UperNet.

Most of these strategies were developed for the autonomous driving domain and do not account for the unique challenges that are inherent to aerial data. The intro-

Table 3.12 In these experiments, we use the NIR data in combination with the RGB images for both training and testing on the Agriculture Vision dataset.

Method	Background	Double Plant	Drydown	Endrow	Nutrient Deficiency	Planter Skip	Water	Waterways	Weed Cluster	mIoU
FCN	68.4	9.4	47.6	0.5	15.2	10.1	53.7	0.5	10.2	23.9
DeepLab V3	69.0	20.0	43.9	5.9	24.0	17.9	46.7	29.0	11.4	29.8
DeepLab V3+	68.3	17.2	48.1	7.5	24.2	19.6	19.4	24.6	13.2	26.9
UperNet	67.4	15.6	36.4	10.7	20.4	14.6	34.2	25.3	14.5	26.6
SFPN	68.7	6.0	48.7	0.2	22.7	17.2	44.5	18.3	12.8	26.6
PSPNet	66.9	17.7	29.9	10.2	28.0	18.7	13.9	29.8	12.0	25.2
HRNetV2	71.3	17.0	54.3	4.5	27.9	15.7	21.7	25.5	17.9	28.4
HRNetV2+OCR	72.6	18.0	56.7	12.0	27.9	23.8	49.0	27.7	22.1	34.4
SegFormer	76.2	33.6	59.0	18.9	40.6	38.9	80.6	42.9	27.9	46.5
Ours	**76.2**	**37.3**	**61.8**	**24.6**	**42.8**	**42.0**	**81.3**	**43.7**	**31.8**	**49.0**

duction of our Augmentation Invariance and Adaptive Sampling approaches leads to a significant increase in performance for almost all semantic classes, particularly the underrepresented ones like "double plant" and "endrow." This results in a total mIoU of 46.4%, an improvement of 24.7% over the least performing UperNet approach and of 1.4% over the SegFormer architecture.

NIR-RGB Experiments Results

As expected, the use of NIR (near-infrared) data in addition to RGB data leads to improved performance for all of the baselines in the Agriculture-Vision dataset. The results of these experiments, shown in Table 3.12, indicate that the average mIoU for all baselines without transformer-based architectures is 26.7%, while the average increases to 35.9% when transformer-based architectures are included. This represents an improvement of 3.7% and 3.2%, respectively, compared to the results using only RGB data. This demonstrates the value of adding NIR data, which enhances the overall training method by providing additional knowledge, as has been previously demonstrated in the literature [45]. Among the baselines, the FCN architecture has the lowest mIoU at 23.9%, while the SegFormer architecture again

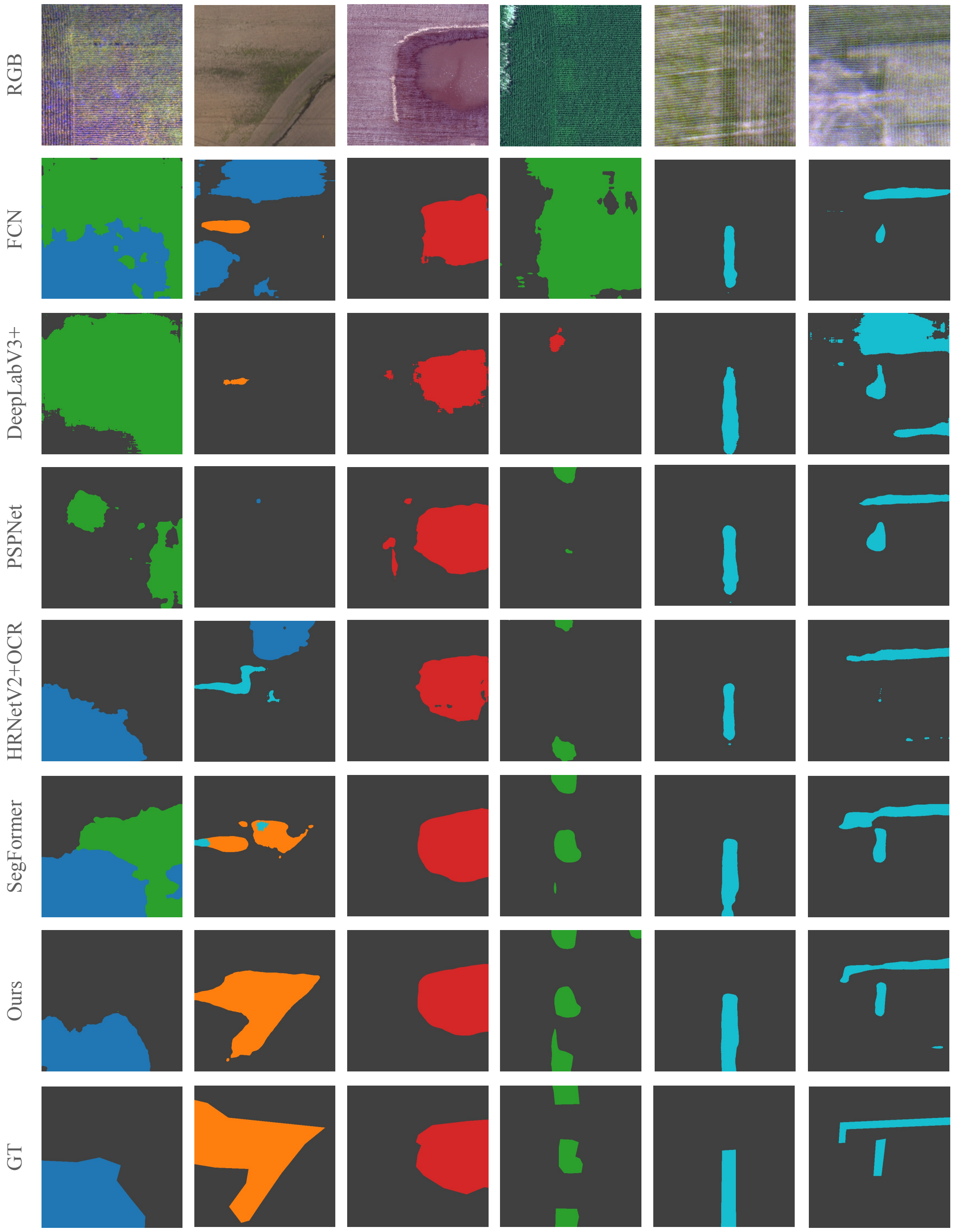


Fig. 3.19 Qualitative results computed on some validation samples of the Agriculture-Vision dataset.

Table 3.13 Ablation study that demonstrate the effectiveness of each component, thus Augmentation Invariance (AI) and Adaptive Sampling (AS), computed considering the NIR-RGB setting.

Components	Background	Double Plant	Drydown	Endrow	Nutrient Deficiency	Planter Skip	Water	Waterways	Weed Cluster	mIoU
SegFormer	76.2	33.6	59.0	18.9	40.6	38.9	80.6	42.9	27.9	46.5
SegFormer + AI	76.6	35.3	61.2	20.7	43.5	43.5	80.4	45.1	33.1	48.8
SegFormer + AS	75.9	35.9	59.2	22.5	41.3	40.7	78.0	40.9	31.0	47.3
SegFormer + AI + AS	76.2	37.3	61.8	24.6	42.8	42.0	81.3	43.7	31.8	**49.0**

performs the best with an mIoU of 46.5%. The overall improvement over FCN is 22.6%.

In these experiments, our solution performs the best among all approaches, achieving an mIoU of 49.0%. The Augmentation Invariance and Adaptive Sampling approaches improve performance for all semantic classes, with particularly notable improvements for underrepresented classes such as "double plant," which gains 27.9%, "endrow," which gains 24.1%, "planter skip," which gains 31.9%, and "waterways," which gains 43.2% compared to the lowest performing FCN. The overall improvement over FCN that Augmentation Invariance and Adaptive Sampling allow us to achieve is 25.1%. These results, as well as the qualitative examples shown in Figure 3.19, demonstrate the effectiveness of our solution in addressing the primary challenges of this task.

3.3.3 Ablation Studies

AIAS Components Contribution

We examine the effect of each proposed component on the overall performance of our method. We consider four cases: (a) the SegFormer framework, (b) the inclusion of our Augmentation Invariance technique, (c) the incorporation of our Adaptive Sampling (AS) approach, and (d) the full framework, which includes both AI and AS. The results of these experiments are shown in Table 3.13. This table demonstrates the significance of the Augmentation Invariance in improving the overall performance

Table 3.14 Ablation study that measure the influence of λ paramenter, computed considering the NIR-RGB setting.

λ	Background	Double Plant	Drydown	Endrow	Nutrient Deficiency	Planter Skip	Water	Waterways	Weed Cluster	mIoU
0.1	76.6	33.9	60.2	18.8	41.9	41.3	82.2	42.5	31.7	47.7
0.25	76.5	35.3	60.7	20.6	42.2	43.8	80.6	43.2	33.3	48.5
0.50	76.5	35.8	59.7	20.3	42.7	40.0	81.1	44.5	32.0	48.1
0.75	76.6	35.3	61.2	20.7	43.5	43.5	80.4	45.1	33.1	**48.8**
1	76.6	34.4	60.3	20.3	42.0	40.0	82.1	43.5	32.2	47.9

of the framework, supporting our conjecture about the specific challenges posed by aerial imagery in agriculture. The addition of Augmentation Invariance leads to an improvement of 2.3% compared to the baseline SegFormer architecture. The inclusion of Adaptive Sampling also enhances the simple SegFormer design, resulting in state-of-the-art performance and emphasizing the importance of addressing the imbalance of semantic classes. The combination of Augmentation Invariance and Adaptive Sampling further improves the results, particularly for underrepresented classes such as "double plant," which improves by 3.7% and "endrow," which improves by 5.7% compared to SegFormer, and 2.0% and 3.9%, respectively, compared to Augmentation Invariance alone.

Lambda

The hyperparameter λ controls the intensity of the Augmentation Invariance loss. In these experiments, we compare five different values of λ: 0.1, 0.25, 0.50, 0.75, and 1.0. We perform these experiments using the NIR-RGB protocol without applying Adaptive Sampling, and the results are shown in Table 3.14. The best performance is achieved with $\lambda = 0.75$. Although the lowest performance is obtained with $\lambda = 0.1$, with a difference of 1.1% compared to $\lambda = 0.75$, the resulting score is still considered state of the art. Overall, our Augmentation Invariance approach outperforms all the baselines, even when using sub-optimal hyperparameters, highlighting its effectiveness.

3.3.4 Findings

In this work, we tackle the task of semantic segmentation for agricultural aerial images, which presents several challenges beyond those typical of semantic segmentation, such as how to effectively use the additional multi-modal data from the visible spectrum, how to handle the imbalance in class-wise pixel distribution, and how to cope with changes in point of view. In order to tackle these challenges, we introduce an end-to-end trainable framework that encompasses two innovative strategies: Augmentation Invariance, which compels the model to learn semantically consistent representations that can withstand the point-of-view changes commonly encountered in aerial imagery, and Adaptive Sampling, which addresses class imbalance by proactively choosing training images based on their class-wise pixel distribution and the network's current level of confidence. We conduct a thorough set of experiments and ablation studies on the Agriculture-Vision dataset, demonstrating the effectiveness of our methods in significantly improving the generalization performance of state-of-the-art models, particularly for underrepresented classes.

3.4 Learn to be robust: HIUDA

In this last section, we continue analyzing the issues presented in the section before and we demonstrate how they are further intensified when dealing with the problem of unsupervised domain adaptation. Starting from this, we identify several challenges with the class mixing techniques often used for this task: (1) they do not account for the large variations in the size and extent of semantic categories present in aerial images, leading to imbalanced domains in the mixed images; (2) they do not consider the lower structural consistency of aerial scenes compared to the driving scenes for which these methods were originally developed, resulting in elements being placed in unnatural contexts in the mixed images; and (3) the source models used to generate pseudo-labels may be affected by domain shifts, causing inconsistent predictions on target images and potentially undermining the mixing strategy. To address these issues, we propose HIUDA, a novel framework for aerial semantic segmentation in unsupervised domain adaptation settings. HIUDA introduces two key innovations: (1) a Hierarchical Instance Mixing (HIMix) technique that merges connected components from each semantic mask according to a semantic hierarchy, and (2) a twin-head architecture in which two separate segmentation heads are fed variations of the same images in a contrastive manner to produce more accurate segmentation maps. We thoroughly evaluate our approach on the LoveDA benchmark and show that it outperforms the current state-of-the-art.

In the field of aerial semantic segmentation, deep learning models have achieved impressive performance when trained on large, annotated datasets [39, 40, 131, 38, 8] such as Chiu et al. [46] and Wang et al. [42]. However, these models often do not generalize well to images from new domains that differ from the domain in which they were trained. In the absence of large amounts of labeled data from the new domain, fine-tuning the model is not a practical solution due to the high cost of generating pixel-level annotations. To address this problem, we consider the task of unsupervised domain adaptation for aerial semantic segmentation. A common approach to unsupervised domain adaptation is to combine self-supervised learning with domain mixing, in which the source model is used to generate semantic predictions (or "pseudo-labels") on the unlabeled target data, and then the labeled source images and pseudo-labeled target images are mixed to create synthetic images containing elements from both domains. This encourages the model to learn domain-agnostic features. Two recent state-of-the-art methods, DACS [71] and DAFormer

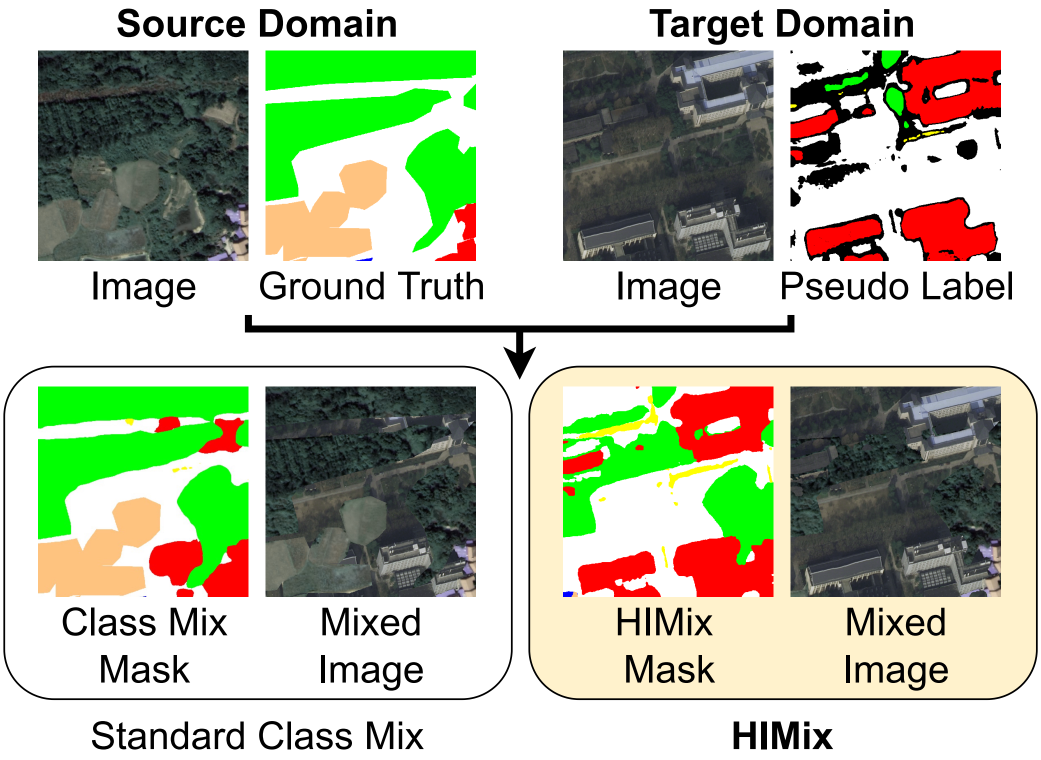


Fig. 3.20 ClassMix simply overlays classes from the source domain onto the target image without considering the semantic hierarchy of visual elements. This may result in the creation of incorrect or deceptive images that have a detrimental impact on Unsupervised Domain Adaptation training in the context of aerial imagery. In contrast, our proposed HIMix method extracts instances from each semantic label and then constructs the mixing mask by sorting the extracted instances based on their pixel count. This helps to reduce artifacts (e.g. partial buildings) and improve the balance of the two domains.

[72], both rely on ClassMix [70], a mixing strategy originally developed for driving scenes that creates composite images by randomly selecting half of the semantic classes from the source image and pasting them onto the target image (see Figure 3.20 left).

However, we argue that this self-supervised mixing approach has several shortcomings when applied to aerial semantic segmentation:

- **Domain imbalance:** Aerial segmentation datasets often contain categories with very different extents, with some occupying only a few pixels (e.g., *cars*) and others taking up large portions of the image (e.g., *forest*). This imbalance in raw pixel counts between classes can be problematic for effective domain adaptation through class mixing because the composition may be skewed towards one of the two domains, depending on how the classes are sampled (see Figure 3.20 left), which can lead to poor feature alignment.

- **Out-of-context instances:** Mixing strategies used in aerial segmentation, such as ClassMix [70], were originally developed for driving scene applications. On the other hand, the environments captured by a front-facing camera on a car maintain a stable composition, with the road at the bottom, the sky at the top, sidewalks and structures along the sides, etc. This structure is also preserved across domains, as in the classic Synthia [5] $\rightarrow$ Cityscapes [3] setting. As a result, when copying objects from a driving scene to another one, they are likely to end up in a reasonable context. This is not the case for aerial images, which lack a consistent semantic structure (see Figure 3.20 left).

- **Pseudo-labels:** The success of semi-supervised mixing is greatly influenced by the precision of the faux labels produced for the target images during the learning process. However, the source model used to generate these pseudo-labels may be susceptible to domain shifts, leading to inconsistent predictions on the target images and potentially undermining the domain mixing strategy.

To solve these issues we propose a new framework for unsupervised domain adaptation in aerial semantic segmentation called **Hierarchical Instance Mixing for Unsupervised Domain Adaptation** (**HIUDA**). HIUDA addresses the limitations of current domain mixing strategies by introducing two technical innovations:

- A new mixing strategy for aerial segmentation across domains called **Hierarchical Instance Mixing** (HIMix). HIMix extracts connected components, or "instances," from each semantic mask. These instances represent individual objects or areas in the image, such as a specific tree in a forested area or a single car on a road. HIMix randomly selects a set of these instances to use as layers in a binary mixing mask, which helps to balance the pixel counts of the source and target domains in the synthetic image. HIMix then arranges these layers according to a semantic hierarchy based on the pixel count of the instances, with smaller instances placed on top of larger ones. This hierarchical composition helps to prevent instances from being placed in unreasonable contexts (e.g. cars in the water) and reduces bias towards categories with larger surface areas in terms of pixels, as they are placed below other layers in the mask (see Figure 3.20 right).

- A new **twin-head Unsupervised Domain Adaptation architecture** in which two separate segmentation heads are fed with contrastive variations of the same images to improve pseudo-label confidence and make the model more robust to domain shifts, leading to more augmentation-consistent representations.

We evaluate HIUDA on the LoveDA benchmark [42], the only dataset specifically designed for evaluating unsupervised domain adaptation in aerial segmentation, and show that it outperforms the current state-of-the-art. We also conduct a thorough ablation study to assess the impact of each of our proposed solutions.

3.4.1 Methodology

We present HIUDA, a novel end-to-end trainable Unsupervised Domain Adaptation framework that utilizes target pseudo-labels. To better align the domains, we use our HIMix strategy to construct artificial images by mixing instances from both the source ground truth and the target pseudo-labels. Instead of using a secondary teacher network derived from the student network, as in previous methods such as DACS [71] and DAFormer [72], we propose a twin-head architecture with two separate decoders that are trained in a contrastive fashion to produce more accurate target pseudo-labels.

Hierarchical Instance Mixing Strategy

Given the pairs (x_s, y_s) and $(x_t, \hat{y}_t)$, where $\hat{y}_t = f_\theta(x_t)$ are the pseudo-labels computed by the model on the target domain, the purpose of the mixing strategy is to create a third pair, (x_m, y_m), using a binary mask M that combines elements from both the source and target domains. This allows the model to learn domain-agnostic features that are applicable to both domains.

However, current class mixing strategies, such as ClassMix, may not be suitable for aerial semantic segmentation because they do not consider the semantic hierarchy of the visual elements in the source and target domains. This can lead to unrealistic

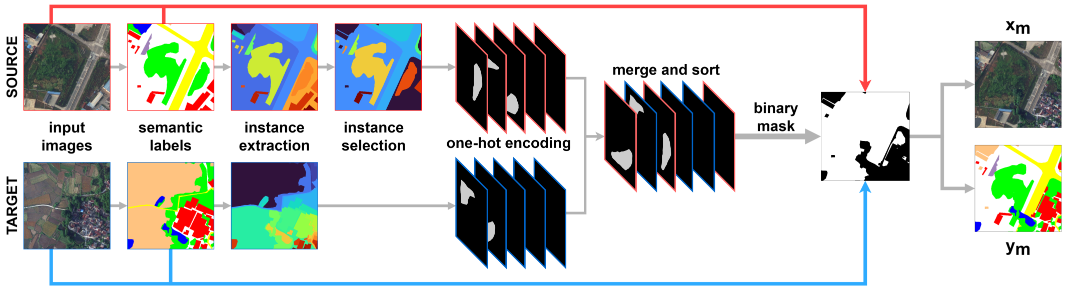


Fig. 3.21 Hierarchical Instance Mixing (HIMix) generates mixed images by performing the following steps: (i) extracting the connected components, or "instances," from the source label and target pseudo-label, (ii) uniformly selecting which instances from the source domain should be mixed, (iii) merging the source and target instances hierarchically based on the size of the instances (with smaller ones on top), and (iv) producing a binary mask M to construct the final blended image x_m and its label y_m.

mixed images with objects placed in unreasonable contexts, such as cars appearing on top of roads.

To address this issue, we propose a new mixing strategy called **Hierarchical Instance Mixing** (HIMix). HIMix consists of two steps: (i) *instance extraction* and (ii) *hierarchical mixing*. In the first step, HIMix extracts connected components, or "instances," from each semantic mask. These instances represent individual objects.

Instance Extraction. Aerial images often contain large areas of uniform land cover, with multiple instances of the same category within a single image. Without the presence of real instance labels, we can utilize this feature by dividing the semantic annotations into interconnected segments. A connected component is a set of pixels that have the same semantic label and are connected to each other by paths entirely contained within the set. By dividing the semantic annotations into connected components, we increase the number of regions that can be randomly selected for the mixing phase. This helps to mitigate the pixel imbalance between the source and target domains in the final mixed sample.

For example, consider the case of a forest that is separated into two instances by a road (see Figure 3.21). Without instance separation, the entire forest would be treated as a single entity in the mixing process. However, by dividing the forest into connected components, we can randomly select one instance from the source and one from the target to be mixed together. This helps to balance the pixel counts of the source and target domains in the mixed image and avoids the situation where one domain dominates the other. Note that this process is applied to the concatenation of the source and target labels.

Hierarchical Mixing. The hierarchy of semantic categories in aerial imagery is reflected in the inherent structure of the instances present in the image. For example, categories such as "barren" or "agricultural" land cover, which tend to encompass a larger area, will often appear in the background relative to smaller instances like "roads" or "buildings". This hierarchy can be observed in Figure 3.21, which illustrates the mixing process used to combine instances from the source and target images.

To facilitate the mixing of these instances, both sets of instance labels are first converted into a one-hot representation[5]. This representation maps each pixel to the index of the class it belongs to, resulting in separate mask layers for each component.

These layers are then merged and sorted based on the number of pixels, with the larger layers at the bottom. Finally, a reduction process is applied from top to bottom, projecting the resulting 3D tensor into a 2D binary mask M. In this mask, positive values represent "source" pixels, while null values represent "target" pixels.

Overall, this process allows for the combination of instances from the source and target images in a way that maintains the inherent hierarchy present in the aerial imagery. This can be particularly useful for tasks such as image segmentation, where the relative size and position of different instances can provide valuable contextual information.

Twin-Head Architecture

In the field of unsupervised domain adaptation, state-of-the-art self-training approaches often utilize "teacher-student" networks to improve the consistency of the generated pseudo-labels. These networks work by training a "teacher" model on the

[5] The one-hot encoding is carried out on ground truth labels and pseudo-labels, rather than raw features or the image itself. Assuming a set of classes C with cardinality N and images with dimension HxW, the one-hot encoding procedure refers to the transformation of the available label from a single-channel representation, where each pixel is assigned a single value indicating the class, to a multi-channel representation, where each pixel is assigned to a vector of length N. Each position i of the vector contains either a 0, if the pixel does not belong to the class with index i, or 1, if the pixel belongs to that category. As standard practice in semantic segmentation frameworks, ground truth labels and pseudo-labels are provided (or transformed) in the form of an index map, where each pixel indicates the index of the class it belongs to.

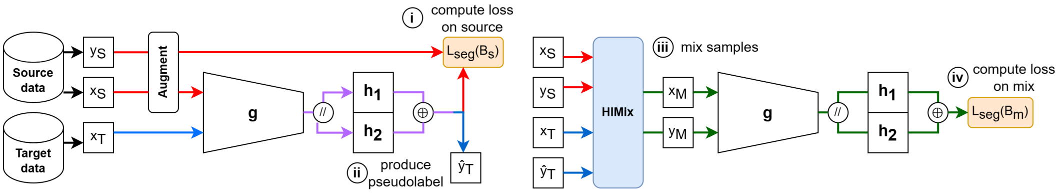


Fig. 3.22 During training, our framework follows the following steps: (i) Standard training is conducted on the source domain using samples composed of a "source image" (x_S) and its corresponding "target image" (y_S). The segmentation loss $L_{seg}(B_s)$ is computed for these samples; (ii) Pseudo-labels $\hat{y}_T$ are generated for the target domain by encoding the "target image" (x_T) with the shared backbone g and applying majority voting to the outputs of each segmentation head (h_1, h_2); (iii) The source and target samples are combined using the HIMix technique, resulting in a new pair of mixed samples (x_M, y_M); (iv) The segmentation loss $L_{seg}(B_m)$ is computed on the mixed samples. This process allows for the effective training of the model using both annotated data from the source domain and pseudo-labels generated for the target domain.

source domain, and using its predictions to guide the training of a "student" model on the target domain. While effective at ensuring consistency over time, teacher-based approaches do not address issues of geometric or stylistic consistency.

To address this issue, we propose a twin-head segmentation framework that directly addresses the problem of consistency in pseudo-labels, resulting in improved performance compared to standard methods. Our architecture, depicted in Figure 3.22, consists of a shared encoder g followed by two parallel and lightweight segmentation decoders, h_1 and h_2. Training is conducted in an end-to-end fashion, leveraging annotated data from the source domain and generating pseudo-labels for the target domain online during the training process.

Our approach differs from traditional teacher-based methods in that it directly addresses the issue of consistency in pseudo-labels, rather than just ensuring consistency over time. This is achieved through the use of a twin-head architecture, which allows for the direct comparison of predictions from the two decoders and the generation of more consistent pseudo-labels as a result.

Source Images Training. To guide the model towards representations that are compatible with multiple augmentations, we input different variations of the same source image into the two heads in a contrastive manner. Specifically, at each iteration we apply a random sequence of geometric and photometric augmentations (T_g, T_p) to the source image x_s and its ground-truth label y_s, resulting in augmented versions $\tilde{x}_s$

and $\tilde{y}_s$. These augmented pairs are then concatenated, resulting in a full augmented pair $B_s = (\text{concat}(x_s, \tilde{x}_s), \text{concat}(y_s, \tilde{y}_s))$, which is forwarded through the shared encoder module g to produce a set of features. These features, containing information from both the original and augmented images, are then split and forwarded to the two parallel heads, resulting in two comparable outputs, $h_1(g(x_s))$ and $h_2(g(\tilde{x}_s))$.

A standard categorical cross-entropy loss defined as:

$$L_{\text{seg}}(x,y) = -\frac{1}{|\mathscr{I}|} \sum_{i \in \mathscr{I}} \sum_{c \in \mathscr{C}} y_i^c \log(p_i^c(x)), \tag{3.15}$$

is computed on both segmentation outputs. By working independently on different variations of the same images, the two heads are able to evolve in different ways while trying to minimize the same objective function. Using the same encoder for both heads yields a more robust and contrastive-like feature extraction that is less susceptible to perturbations, which is essential for producing more stable and precise pseudo-labels.

HIMixed Images Training. The twin-head architecture is specifically designed to generate more refined pseudo-labels for use in the mixing strategy. Given an unlabeled target image x_t, the probabilities $\sigma(h_1(g(x_t)))$ and $\sigma\,(h_2(g(x_t)))$ are obtained by forwarding the image to both heads and passing the outputs through a Softmax function. The maximum value between the two probabilities is then selected, and used to generate the pseudo-label $\hat{y}_t^{(i,c)}$ for each head output using:

$$\hat{y}_t^{(i,c)} = [c = \arg\max_{c}\; p_i^c(x_t)] \tag{3.16}$$

With the pseudo-labels in hand, the mixed pairs of inputs can be computed using the HIMix technique, resulting in (x_m, y_m) as a combination of the source and target samples. Similar to the training process for the source domain, an augmented pair $B_m = (\text{concat}(x_m, \tilde{x}_m), \text{concat}(y_m, \tilde{y}m))$ is created through the application of geometric and photometric transformations, and fed to the model to compute $L_{seg}(B_m)$.

To reduce the impact of low-confidence areas on the training process, a pixel-wise weight map w_m is generated. Similar to previous approaches [71, 72], this weight map is computed as the percentage of valid points above a threshold. Formally, for each pixel i, w_m^i is set to 1 for regions derived from the source domain, or by a factor obtained as the number of pixels above a confidence threshold, normalized by the total number of pixels in the pseudo-label. Formally:

$$w_m^i = \begin{cases} 1, & i \in y_s \\ \\ \frac{m_\tau}{|\mathscr{I}|}, & i \in \hat{y}_t \end{cases} \tag{3.17}$$

This threshold, m_τ, is computed as the number of pixels with a maximum probability above a specified threshold:

$$m_\tau^i = \mathbb{1}_{[argmax_c p_i^c(x_t) > \tau]}, \tag{3.18}$$

It is important to note that gradients are not propagated during these computations. This allows for the generation of more stable and accurate pseudo-labels without affecting the training process.

The overall training procedure is detailed in the Algorithm 1.

3.4.2 Experiments

We gauged the efficacy of our HIUDA framework on the LoveDA dataset [42] through two sets of unsupervised domain adaptation evaluations: *rural→urban* and *urban→rural*. As per the standard protocol for Unsupervised Domain Adaptation in Semantic Segmentation, we used labeled samples from the source domain to adapt our model to the unlabeled samples in the target domain during the training phase. For evaluation, we tested the model on a separate set of images from the target domain for which ground truth data was available.

Implementation and Training Details

To implement HIUDA we used the *mmsegmentation* framework based on PyTorch. All experiments were trained on an NVIDIA Titan GPU with 24 GB of RAM. The architecture and configuration of hyperparameters for HIUDA were based on the approach proposed in DAFormer [72]. Specifically, we utilized the MiT-B5 model [33] as the encoder for HIUDA and the SegFormer head [33] as the segmentation decoder module.

Algorithm 1: HIUDA Training Procedure

```
Initialize:
Model $f_\theta : \mathscr{X} \to \mathbf{R}^{|\mathscr{I}|\times|\mathscr{Y}|}$ with encoder $g$ and twin heads $h_1, h_2$;
Input: $\mathscr{X}_s$ source domain with $N_s$ pairs $(x_s, y_s)$, $x_s \in \mathscr{X}, y_s \in \mathscr{Y}$ and
  semantic classes $\mathscr{C}$;
$\mathscr{X}_t$ target domain with $N_t$ images $x_t$, lacking ground truth labels;
Output: $y = \{\mathrm{argmax}_{c\in\mathscr{Y}} p_i^c\}_{i=1}^N$, where $p_i^c$ the model prediction of pixel $i$
  for class $c$ and $\mathscr{Y}$ the label space;
while epoch in max_epochs do
    while $x_s, y_s, x_t$ in $\mathscr{X}_s \times \mathscr{X}_t$ do
        Train on source $\mathscr{X}_s$
            // Compute augmented source batch
              $B_s = (\mathrm{concat}(x_s, \tilde{x}_s), \mathrm{concat}(y_s, \tilde{y}_s))$;
            // Train $f_\theta$ on source labels with $L_{seg}(B_s)$
        end
        Mix source and target pairs
            // Compute pseudo-labels via majority
            // voting $\hat{y}_t = max(h_1(g(x_t)), (h_2(g(x_t))))$;
            // Extract source instance labels $i_s = CCL(y_s)$ with instances
              $\in K_s$;
            // Extract target instance pseudo-labels $i_t = CCL(\hat{y}_t)$ with
              instances $\in K_t$;
            // Compute one-hot encoded labels,
            // sorted by pixel size as: $1_m = sorted(\mathrm{concat}(1_{K_s}(i_s), 1_{K_t}(i_t)))$;
            // Reduce $z$ axis to 2D indexed mask $m = argmax_z 1_m(i,j,z)$;
            // Binarize mask $\forall i,j \in m,\ M = \begin{cases} 1 & if\ m(i,j) \in K_s \\ 0 & if\ m(i,j) \in K_t \end{cases}$;
            // Compute mixed image and labels as:
              $x_m = M \odot x_s + (1 - M) \odot x_t$;
            $y_m = M \odot y_s + (1 - M) \odot \hat{y}_t$;
            // Compute $w_m$ as in Eq. 3.17
        end
        Train on mixed $\mathscr{X}_m$ pairs
            // Compute augmented mixed batch
            $B_m = (\mathrm{concat}(x_m, \tilde{x}_m), \mathrm{concat}(y_m, \tilde{y}_m))$;
            // Train $f_\theta$ on mixed samples with
            // $L_{seg}(B_m)$, weighted by $w_m$
        end
    end
end
```

During training, we applied data augmentation to the input images to improve the generalization of the model. This included randomly resizing the images in the range $[0.5, 2.0]$, flipping them horizontally and vertically, rotating them by 90 degrees with probability $p = 0.5$, and applying random photometric distortions (i.e., changes to brightness, saturation, contrast, and hue). In addition, we trained the model on random crops of the images. We used AdamW as the optimizer, with a learning rate of $6x10^{-5}$, weight decay of 0.01, and betas set to $(0.9, 0.99)$. We also applied a polynomial decay with a factor of 1.0 and warm-up for 1,500 iterations. We trained all experiments for 40,000 iterations and obtained the results for every experiment as the average over three different random seeds $0, 1, 2$ to account for variations.

As in previous work such as DACS [71] and DAFormer [72], we set $\tau = 0.968$ in 3.18 to determine the confidence threshold for using the model's own predictions on the target domain as pseudo-labels. For final evaluation on the test set, we used raw images without any further transformations. This allowed us to assess the performance of HIUDA on the target domain in a realistic setting.

Compared Methods

In this study, we conducted a comprehensive comparison of the performance of our proposed unsupervised domain adaptation method, HIUDA, to various state-of-the-art Unsupervised Domain Adaptation approaches. Our comparison included both a baseline model and several alternative Unsupervised Domain Adaptation methods.

The baseline model we considered was the Source Only model, which is a network trained solely on the source dataset without any adaptation to the target domain. This model serves as a reference for comparison and allows us to assess the improvement gained through domain adaptation.

We evaluate the original metric-based approach and several well-known adversarial training methods:

- MMD [132]: it adopts the Maximum Mean Discrepancy (MMD) loss in combination with the standard classification loss on the source to learn a representation that is discriminative and domain invariant;

- AdaptSegNet [133]: it presents one of the earliest proposals for domain adaptation of semantic segmentation via adversarial learning;
- FADA [134]: it introduces a fine-grained adversarial learning framework for cross-domain semantic segmentation that incorporates class-level information and enables class-level feature alignment;
- CLAN [63]: it emphasizes maintaining local semantic consistency while aligning global distributions to mitigate the negative transfer effect of misaligned features;
- TransNorm [135]: it is a trainable layer that enables DNNs to be more transferable across domains and is designed to be trained end-to-end.

We finally compared HIUDA to several self-training Unsupervised Domain Adaptation methods:

- CBST [66]: it represents an early foray into self-training methods as they apply to domain adaptation in the realm of semantic segmentation. Additionally, a novel approach to self-training that prioritizes balanced representation of classes is introduced to mitigate the potential for large classes to dominate the training process;
- PyCDA [65]: it proposes a new approach for domain adaptation of semantic segmentation networks by connecting self-training and curriculum adaptation methods;
- IAST [67]: it suggests an adaptive self-training approach that operates at the instance level;
- DACS [71]: it presents an algorithm that combines pictures from the source and target domains to produce new, modified samples;
- DAFormer [72]: it incorporates the DACS mixing strategy, while also featuring a Transformer encoder and a multi-level context-aware feature fusion decoder.

Overall, our comparison of HIUDA to these various Unsupervised Domain Adaptation methods allowed us to assess the effectiveness of our proposed method and understand its strengths and limitations in comparison to the state of the art.

Table 3.15 Urban→Rural experiments. Experiments marked with an asterisk (*) were repeated using the original method.

Method	Backg.	Building	Road	Water	Barren	Forest	Agric.	mIoU
Source Only	24.2	37.0	32.6	49.4	14.0	29.3	35.7	31.7
MCD [132]	25.6	44.3	31.3	44.8	13.7	33.8	26.0	31.4
AdaptSeg [133]	26.9	40.5	30.7	50.1	17.1	32.5	28.3	32.3
FADA [134]	24.4	33.0	25.6	47.6	15.3	34.4	20.3	28.7
CLAN [63]	22.9	44.8	26.0	46.8	10.5	37.2	24.5	30.4
TransNorm [135]	19.4	36.3	22.0	36.7	14.0	40.6	03.3	24.6
PyCDA [65]	12.4	38.1	20.5	57.2	18.3	36.7	41.9	32.1
CBST [66]	25.1	44.0	23.8	50.5	08.3	39.7	49.7	34.4
IAST [67]	30.0	49.5	28.3	64.5	02.1	33.4	61.4	38.4
DACS* [71]	20.1	50.5	35.9	60.6	09.9	35.4	17.5	32.9
DAFormer* [72]	29.5	57.9	41.8	67.1	07.6	35.3	48.1	41.0
HIUDA	31.5	59.6	51.5	68.1	08.2	37.4	53.9	**44.3**

Urban→Rural Results

The outcomes of the Urban→Rural tests, displayed in Table 3.15, highlight the difficulties of the task, caused by the stark and inconsistent class distribution in the source domain. This domain is characterized by urban scenes with a mix of buildings and highways, but relatively few natural items. This uneven distribution of classes has negative consequences for the transfer of knowledge to the target domain, as both adversarial training and self-training approaches yield overall performance that is comparable to, or worse than, the Source Only model.

Specifically, the best-performing adversarial training technique, CLAN, only manages to achieve a $+1.8\%$ improvement over the Source Only model. On the other hand, self-training approaches have shown to be more effective in this case. DACS, which utilizes a class mixing strategy, leads to a $+1.2\%$ improvement in the performance of the Source Only model. DAFormer, utilizing a Transformer backbone and the same class mixing approach as DACS, surpasses the Source Only model by a margin of $+9.3\%$.

However, the most successful self-training approach in this set of experiments is HIUDA, which combines a twin-head architecture with an innovative class mix. This

Table 3.16 Rural→Urban experiments. Experiments marked with an asterisk (*) were repeated using the original method.

Method	**Backg.**	**Building**	**Road**	**Water**	**Barren**	**Forest**	**Agric.**	**mIoU**
Source Only	43.3	25.6	12.7	76.2	12.5	23.3	25.1	31.3
MCD [132]	43.6	15.4	12.0	79.1	14.3	33.1	23.5	31.5
AdaptSeg [133]	42.4	23.7	15.6	82.0	13.6	28.7	22.1	32.6
FADA [134]	43.9	12.6	12.8	80.4	12.7	32.8	24.8	31.4
CLAN [63]	43.4	25.4	13.8	79.3	13.7	30.4	25.8	33.1
TransNorm [135]	33.4	05.0	03.8	80.8	14.2	34.0	17.9	27.7
PyCDA [65]	38.0	35.9	45.5	74.9	07.7	40.4	11.4	36.3
CBST [66]	48.4	46.1	35.8	80.1	19.2	29.7	30.1	41.3
IAST [67]	48.6	31.5	28.7	86.0	20.3	31.8	36.5	40.5
DACS* [71]	46.0	31.6	33.8	76.4	16.4	29.3	27.7	37.3
DaFormer* [72]	49.2	47.7	55.2	86.6	16.5	39.5	30.8	46.5
HIUDA	49.3	55.0	55.4	86.0	17.1	41.2	36.9	**48.7**

approach outperforms the Source Only model by a wide margin of +12.6%, and also surpasses its closest competitor, DAFormer, by +3.3%. The success of HIUDA can be attributed to its ability to boost the performance of rural and underrepresented classes, such as agriculture, as demonstrated by the qualitative results presented in Figure 3.23. Additionally, HIUDA demonstrates exceptional results in identifying and categorizing contours and classes such as water, despite their scarcity in the source domain. This holds true for frequently encountered categories that exhibit different visual characteristics, like road, which can be encountered in both paved and unpaved forms.

Rural→Urban Results

The results of the Rural→Urban experiments are summarized in Table 3.16. The source domain in this case is dominated by large-scale natural objects, with only a few man-made samples present. Despite this imbalance, the models being evaluated are able to effectively transfer knowledge to these underrepresented categories. Self-learning approaches outperform adversarial methods, with an average boost

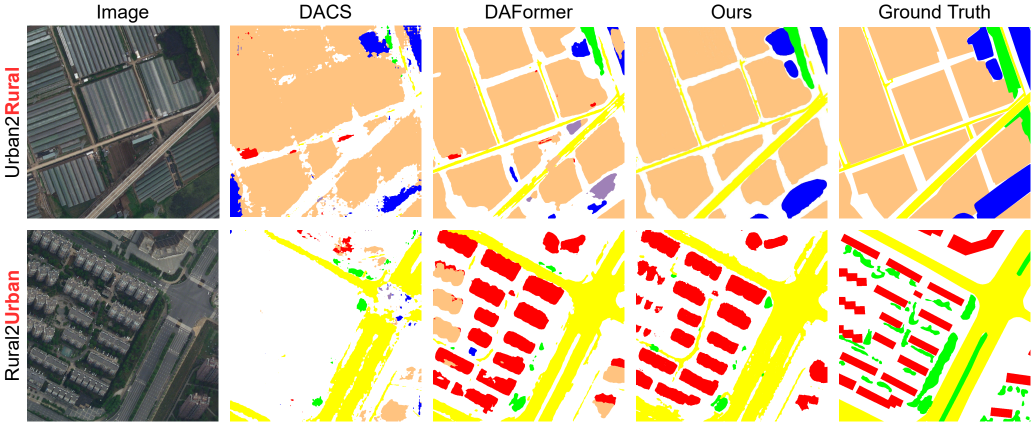


Fig. 3.23 Qualitative results from the two settings, Urban to Rural and Rural to Urban, after testing on the **target** domain.

of +9.1% over the Source Only model. Adversarial training methods, on the other hand, achieve accuracy that is comparable to the Source Only model.

In terms of mIoU, the two best-performing self-training models and our closest competitor all achieve improvements over the Source Only model, with gains of +6.0%, +15.2%, and +17.4%, respectively. Among these, HIUDA stands out as the most successful approach, outperforming DACS and DAFormer by +11.4% and +2.2%, respectively. The qualitative results presented in Figure 3.23 support the superior ability of HIUDA to distinguish between rural and urban classes. While DACS fails to identify buildings and DAFormer wrongly categorizes some of them as agricultural land, our model effectively reduces bias towards categories with larger surfaces, resulting in performance that is more aligned with the actual truth.

3.4.3 Ablation Studies

Twin-Head and HIMix Components

To assess the efficacy of the twin-head architecture, we compare it against a conventional single-head design that creates pseudo-labels using a secondary teacher network, which is obtained from the student model as an exponential moving average. In addition, this study investigates the potential of the HIMix when paired with traditional single-head training. To this end, we carry out an extensive ablation study using MiT-B5 [33] as the backbone, and we report the results in Table 3.17.

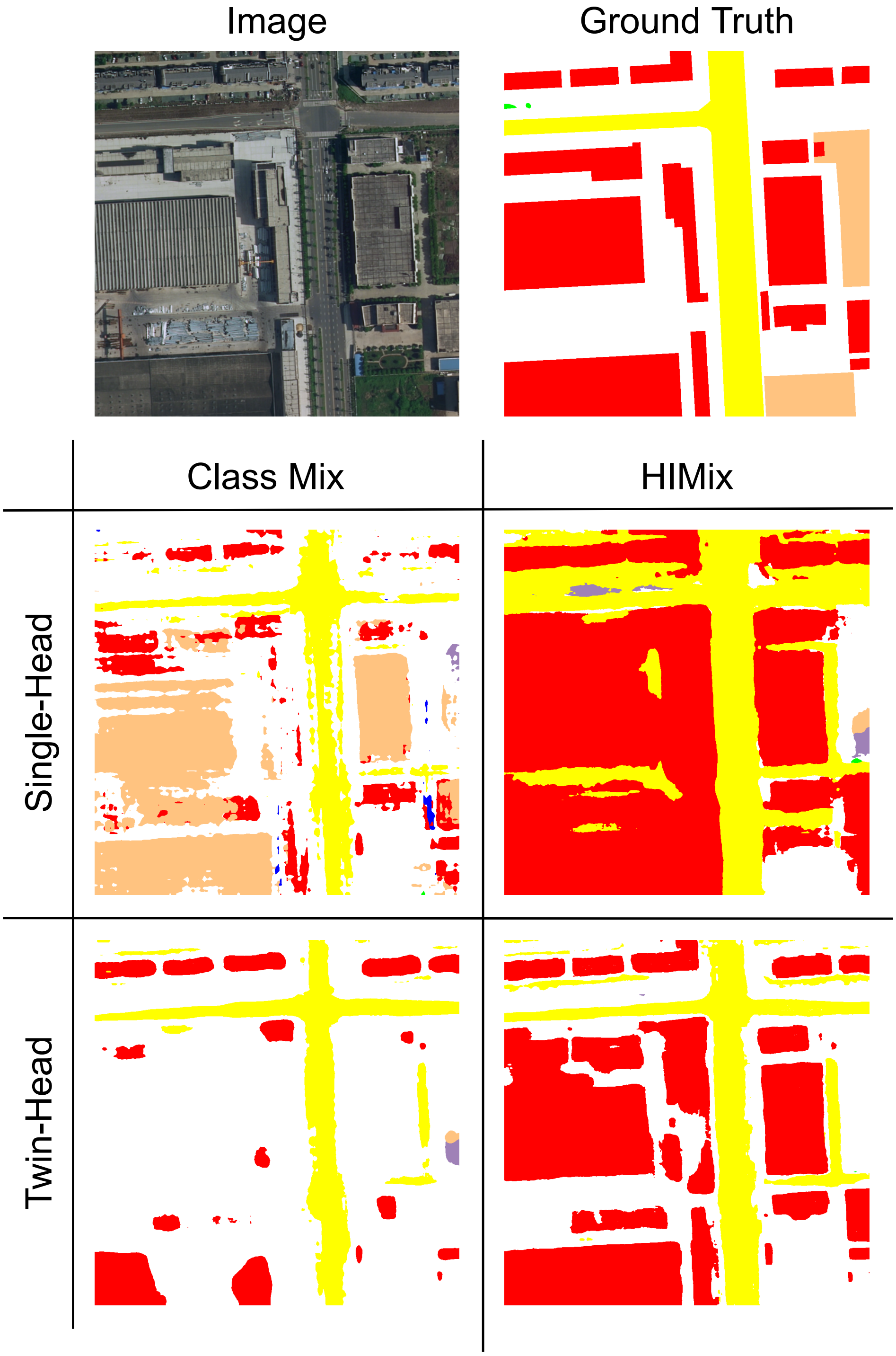


Fig. 3.24 A qualitative comparison between Single or Twin-Head architectures using the Standard ClassMix or our HIMix strategy.

Table 3.17 An ablation study was conducted on our HIUDA framework to evaluate the contribution of its twin-head architecture and HIMix strategy

ID	Twin Head	Class Mix	Instance Mix	Hierarchical. Mix	mIoU U2R	mIoU R2U
1		✓			41.0 ± 0.33	46.5 ± 0.41
2			✓	✓	43.4 ± 0.76	47.6 ± 0.10
3	✓	✓			42.9 ± 0.35	47.1 ± 0.34
4	✓		✓		43.2 ± 0.35	47.4 ± 0.16
5	✓		✓	✓	**44.3** ± 0.39	**48.7** ± 0.06

The results reveal that the twin-head design paired with the Standard ClassMix (line 3) performs better than the single-head architecture (line 1). This implies that our solution is more effective at producing finer pseudo-labels with correct class segmentation, as also depicted in the first column of Figure 3.24.

Even when used with a single-head architecture, the HIMix still leads to an improvement in recognition performance (line 2). This is particularly evident for categories with a smaller surface area in pixels, which are ranked below those with larger surfaces when using the Standard ClassMix. This is why, in the top-left image of Figure 3.24, the model is unable to effectively grasp the semantics and wrongly classifies the building as agricultural land. Contrastingly, HIMix can accurately identify buildings (as seen in the top-right image of Figure 3.24) even when the prediction has indistinct contours.

The highest performance is achieved when the twin-head's ability to provide an enhanced segmentation map is combined with the HIMix's ability to maintain a correct semantic structure (line 5). This combination results in the highest accuracy and the most detailed segmentation map, as shown in the bottom-right image of Figure 3.24.

To assess the contribution of our HIMix to overall performance, we conduct also an ablation study on each components (lines 4-5). The results show that the Hierarchical Mixing consistently improves Instance Extraction by $+1.1\%$ and $+1.3\%$ in the Urban to Rural and Rural to Urban scenarios, respectively. These findings highlight the value of the Hierarchical Mixing in enhancing the performance of the HIMix in both settings.

3.4.4 Findings

In conclusion, we have explored the challenge of Unsupervised Domain Adaptation in the field of aerial Semantic Segmentation and discovered that the distinctive features of aerial imagery, including the absence of structural consistency and the marked discrepancy in semantic class coverage, must be considered. In order to address these issues, we have developed HIUDA, a fully trainable Unsupervised Domain Adaptation framework that includes two main contributions. The first solution introduced is an innovative domain mixing technique that is comprised of two components: a connected component extraction component that selects instances from each semantic map, and a hierarchical mixing component that categorizes and integrates the instances based on their pixel count. The second contribution is a twin-head architecture that generates finer pseudo labels for the target domain, thereby improving the effectiveness of the domain mixing. We have demonstrated the success of HIUDA through a comprehensive set of experiments on the LoveDA benchmark, showing that it is capable of effectively adapting knowledge from one domain to another, even in the challenging conditions presented by aerial imagery.

Chapter 4

Conclusion

In this concluding chapter, we will summarize the main results and contributions presented throughout this thesis, highlighting the significance of the proposed methods and datasets and the implications of our findings for the field of Semantic Segmentation and Domain Adaptation. We will also discuss the open issues and challenges that remain to be addressed in this research area, and will outline some potential directions for future work. This may include extensions and improvements to the proposed methods, as well as potential applications in other domains. Overall, the goal of this chapter is to provide a comprehensive summary of the research presented in this thesis, and to offer insights and guidance for future work in this important and rapidly-evolving field.

4.1 Summary

The extensive analysis of semantic segmentation models has shed light on a significant issue that these models face: poor generalization performance to unseen domains. This issue is primarily attributed to the limited data availability and quality. Collecting and annotating real-world data is a time-consuming and costly process, which results in models that struggle to generalize to new, unseen scenarios.

To address this challenge, the purpose of this thesis was to explore and develop solutions that would make the neural models more robust and capable of generalizing to different domains from the ones they were trained on. One such solution is to use synthetic datasets, such as IDDA, a large-scale synthetic dataset designed specifically for the autonomous driving task. Using such datasets can help reduce the domain gap with real-world scenarios by providing diverse examples of different weather, illumination, perspectives, and environmental conditions.

However, it is worth noting that even synthetic datasets have their limitations. It is not possible to fully replicate the nuances of real-world data in a simulated environment, which can result in models that are not able to accurately segment real-world scenes. To overcome this challenge, one approach is to use a combination of synthetic data and a limited amount of annotated real-world data. This is what our PixDA framework does when applied to self-driving task - it learns from both synthetic and real-world domain, reducing the likelihood of the model becoming overly reliant on the specific characteristics of the synthetic dataset.

These solutions may not yield the same results when applied to particular tasks such as aerial image analysis, which presents additional challenges. One of these challenges is the lack of reference points, leading to the same scene being viewed from different rotations around the vertical axis, making the model unable to handle these changes in perspective. Even using domain adaptation solutions, such as the newly introduced class mix strategy which combines images from different domains, lead to unnatural mixed images and introduce negative bias during model training.

To overcome these challenges, we introduce a novel augmentation invariance loss to make the model invariant to changes in perspective, and a new domain adaptation framework that considers the lack of structural consistency in aerial images. This results in a more precise and adaptable mixing strategy that leads to improved performance for aerial image analysis.

In conclusion, our study highlights the importance of utilizing the available data in an efficient manner to train effective and robust deep learning models. Despite the challenges posed by scenarios with limited data or complexity, such as autonomous driving or aerial image analysis, our work demonstrates that it is possible to achieve good results through proper exploitation of the available information. This emphasizes the requirement for cutting-edge approaches and structures, resulting in enhanced performance and precision in challenging scenarios.

4.2 Limitations and Future Work

There are still several aspects of these works that could be improved and remain relevant for future research.

Regarding the use of synthetic datasets for semantic segmentation, the primary challenge lies in the lack of realism. As these datasets are generated from 3D models, they often miss the intricate details, variability, and diversity found in real-world scenes, leading to models that struggle to adapt to real-world variations. This is particularly evident in challenging domains such as aerial imaging and autonomous driving, where the appearance and texture of real-world scenes can vary greatly. In response to this limitation, future works aim to enhance the realism of synthetic datasets by utilizing cutting-edge computer graphics techniques to generate more varied and representative scenes.

Another crucial aspect is to integrate some real-world annotated data into training, similar to what was done with PixDA, however this approach in some cases lead to low segmentation accuracy due to overfitting or underfitting, especially if there is imbalance in the semantic content causing complete misclassification of some classes. Additionally, the model's complexity can be high, resulting in high computational needs and longer training durations. Future efforts will explore meta-learning techniques where the model can rapidly adapt to new domains, making the adaptation process more efficient and effective. Different types of data such as depth maps and point clouds can also be utilized to enhance the adaptation process and address the challenge of data scarcity. Moreover, it could be intriguing to investigate the application of our pixel-wise adversarial loss in other settings such as unsupervised or multi-source domain adaptation.

The same limitations persist in the aerial scenario with the AIAS approach, which has only been tested on one dataset and utilizes a transformer as its backbone, providing strong feature extraction but also requiring a substantial amount of computational power. This is similarly evident in the HIUDA framework, which boasts exceptional performance but necessitates a larger number of parameters, slowing down the training process. Future work will examine the efficiency of our solution when applied to various complex datasets and examine the potential of using a simpler network architecture. Furthermore, it will be fascinating to assess the performance when incorporating diverse data types besides the near infrared into the training phase, in addition to relying solely on the RGB image.

In conclusion, this thesis has provided an in-depth examination of domain adaptation in semantic segmentation, emphasizing the importance of robust models that can handle varying data distributions. Further research is needed to develop models that are more robust to domain shift and can be applied to a wide variety of applications, from robotics to medical imaging. By driving advances in this areas, such models can not only benefit the wider research community, but also lead to tangible improvements in the lives of people all over the world.

References

[1] Gabriel J. Brostow, Jamie Shotton, Julien Fauqueur, and Roberto Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, pages 44–57, 2008.

[2] Gabriel J. Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *PRL*, 2008.

[3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[4] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Eur. Conf. Comput. Vis.*, volume 9906 of *LNCS*, pages 102–118. Springer International Publishing, 2016.

[5] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3234–3243, June 2016.

[6] E. Alberti, A. Tavera, C. Masone, and B. Caputo. Idda: A large-scale multi-domain dataset for autonomous driving. *IEEE Robotics and Automation Letters*, 5(4):5526–5533, 2020.

[7] Antonio Tavera, Fabio Cermelli, Carlo Masone, and Barbara Caputo. Pixel-by-pixel cross-domain alignment for few-shot semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1626–1635, January 2022.

[8] Antonio Tavera, Edoardo Arnaudo, Carlo Masone, and Barbara Caputo. Augmentation invariance and adaptive sampling in semantic segmentation of agricultural aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1656–1665, June 2022.

[9] Edoardo Arnaudo, Antonio Tavera, Carlo Masone, Fabrizio Dominici, and Barbara Caputo. Hierarchical instance mixing across domains in aerial segmentation. *IEEE Access*, 11:13324–13333, 2023.

[10] Donald Shenaj, Eros Fanì, Marco Toldo, Debora Caldarola, Antonio Tavera, Umberto Michieli, Marco Ciccone, Pietro Zanuttigh, and Barbara Caputo. "learning across domains and devices: Style-driven source-free domain adaptation in clustered federated learning. In *Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2023.

[11] Lidia Fantauzzo, Eros Fanì, Debora Caldarola, Antonio Tavera, Fabio Cermelli, Marco Ciccone, and Barbara Caputo. Feddrive: Generalizing federated learning to semantic segmentation in autonomous driving. In *Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2022.

[12] Fabio Cermelli, Dario Fontanel, Antonio Tavera, Marco Ciccone, and Barbara Caputo. Incremental learning in semantic segmentation from image labels. In *Proceedings of the IEEE/CVF Computer Vision and Patter Recognition Conference (CVPR)*, June 2022.

[13] Valerio Paolicelli, Antonio Tavera, Carlo Masone, Gabriele Berton, and Barbara Caputo. Learning semantics for visual place recognition through multi-scale attention. In Stan Sclaroff, Cosimo Distante, Marco Leo, Giovanni M. Farinella, and Federico Tombari, editors, *Image Analysis and Processing – ICIAP 2022*, pages 454–466, Cham, 2022. Springer International Publishing.

[14] Edoardo Arnaudo, Fabio Cermelli, Antonio Tavera, Claudio Rossi, and Barbara Caputo. A contrastive distillation approach for incremental semantic segmentation in aerial images. In Stan Sclaroff, Cosimo Distante, Marco Leo, Giovanni M. Farinella, and Federico Tombari, editors, *Image Analysis and Processing – ICIAP 2022*, pages 742–754, Cham, 2022. Springer International Publishing.

[15] Barbara Caputo Antonio Tavera, Carlo Masone. Reimagine bisenet for real-time domain adaptation in semantic segmentation. In *2021 I-RIM Conference*, pages 33–37. I-RIM, 2021.

[16] Irem Ulku and Erdem Akagunduz. A survey on deep learning-based architectures for semantic segmentation on 2d images, 2019.

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

[18] Junyi Zhang, Ziliang Chen, Junying Huang, Liang Lin, and Dongyu Zhang. Few-shot structured domain adaptation for virtual-to-real scene parsing. In *Int. Conf. Comput. Vis. Worksh.*, 2019.

[19] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, and Tao Qin. Generalizing to unseen domains: A survey on domain generalization. In *International Joint Conference on Artificial Intelligence*, 2021.

[20] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2015.

[21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, 2015.

[22] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. *CoRR*, abs/1904.04514, 2019.

[23] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *TPAMI*, 2019.

[24] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017.

[25] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 418–434, 2018.

[26] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6230–6239, 2017.

[27] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin P. Murphy, and Alan Loddon Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:834–848, 2018.

[28] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *ArXiv*, abs/1706.05587, 2017.

[29] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[30] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[31] Yuhui Yuan, Xiaokang Chen, Xilin Chen, and Jingdong Wang. Segmentation transformer: Object-contextual representations for semantic segmentation. *arXiv preprint arXiv:1909.11065*, 2019.

[32] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Int. Conf. Comput. Vis.*, 2021.

[33] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34, 2021.

[34] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *NeurIPS*, 2021.

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[36] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with Transformers. In *ECCV*, 2020.

[37] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. OneFormer: One Transformer to Rule Universal Image Segmentation. In *CVPR*, 2023.

[38] Nicolas Audebert, Bertrand Le Saux, and Sébastien Lefèvre. Beyond rgb: Very high resolution urban remote sensing with multimodal deep networks. *ISPRS Journ. Phot. Rem. Sens.*, 140:20–32, 2018.

[39] Foivos I. Diakogiannis, François Waldner, Peter Caccetta, and Chen Wu. Resunet-a: A deep learning framework for semantic segmentation of remotely sensed data. *ISPRS Journ. Phot. Rem. Sens.*, 162:94–114, 2020.

[40] Keiller Nogueira, Mauro Dalla Mura, Jocelyn Chanussot, William Robson Schwartz, and Jefersson A. dos Santos. Learning to semantically segment high-resolution remote sensing images. In *Int. Conf. Pattern Recog.*, pages 3566–3571, 2016.

[41] Adrian Boguszewski, Dominik Batorski, Natalia Ziemba-Jankowska, Tomasz Dziedzic, and Anna Zambrzycka. Landcover. ai: Dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1102–1110, 2021.

[42] Junjue Wang, Zhuo Zheng, Ailong Ma, Xiaoyan Lu, and Yanfei Zhong. Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation. *arXiv preprint arXiv:2110.08733*, 2021.

[43] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 172–181, 2018.

[44] Andres Milioto, Philipp Lottes, and Cyrill Stachniss. Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 2229–2235. IEEE, 2018.

[45] S. Yang, S. Yu, B. Zhao, and Y. Wang. Reducing the feature divergence of rgb and near-infrared images using switchable normalization. In *IEEE Conf. Comput. Vis. Pattern Recog. Work.*, pages 206–211, jun 2020.

[46] Mang Tik Chiu, Xingqian Xu, Yunchao Wei, Zilong Huang, Alexander G Schwing, Robert Brunner, Hrant Khachatrian, Hovnatan Karapetyan, Ivan Dozier, Greg Rose, et al. Agriculture-vision: A large aerial image database for agricultural pattern analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2828–2838, 2020.

[47] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753, 2018.

[48] Dominic Cheng, Renjie Liao, Sanja Fidler, and Raquel Urtasun. Darnet: Deep active ray network for building segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7431–7439, 2019.

[49] Ning Lv, Chen Chen, Tie Qiu, and Arun Kumar Sangaiah. Deep learning and superpixel feature extraction based on contractive autoencoder for change detection in sar images. *IEEE transactions on industrial informatics*, 14(12):5530–5538, 2018.

[50] Qinglie Yuan, Helmi Zulhaidi Mohd Shafri, Aidi Hizami Alias, and Shaiful Jahari bin Hashim. Multiscale semantic feature optimization and fusion network for building extraction using high-resolution aerial images and lidar data. *Rem. Sens.*, 13(13), 2021.

[51] Nadir Bengana and Janne Heikkilä. Improving land cover segmentation across satellites using domain adaptation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:1399–1410, 2020.

[52] John Weier and David Herring. Measuring vegetation (ndvi & evi). *NASA Earth Observatory*, 20, 2000.

[53] Hao Sheng, Xiao Chen, Jingyi Su, Ram Rajagopal, and Andrew Ng. Effective data fusion with generalized vegetation index: Evidence from land cover segmentation in agriculture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 60–61, 2020.

[54] Bin Pan, Zhenwei Shi, Xia Xu, Tianyang Shi, Ning Zhang, and Xinzhong Zhu. Coinnet: Copy initialization network for multispectral imagery semantic segmentation. *IEEE Geos. Rem. Sens. Lett.*, 16(5):816–820, 2019.

[55] Bo Geng, Dacheng Tao, and Chao Xu. DAML: domain adaptation metric learning. *IEEE Trans. Image Process.*, 20(10):2980—2989, October 2011.

[56] Mingsheng Long, Y. Cao, J. Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. In *Int. Conf. Mach. Learn.*, pages 97–105, 2015.

[57] Judy Hoffman, E. Tzeng, T. Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A. Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *Int. Conf. Mach. Learn.*, pages 1989–1998, 2018.

[58] Zuxuan Wu, Xintong Han, Yen-Liang Lin, Mustafa Gokhan Uzunbas, Tom Goldstein, Ser Nam Lim, and Larry S. Davis. DCAN: Dual channel-wise alignment networks for unsupervised scene adaptation. In *Eur. Conf. Comput. Vis.*, pages 518–534, 2018.

[59] Yanchao Yang and Stefano Soatto. FDA: Fourier domain adaptation for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4085–4095, 2020.

[60] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6936–6945, 2019.

[61] Myeongjin Kim and Hyeran Byun. Learning texture invariant representation for domain adaptation of semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12975–12984, 2020.

[62] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. ADVENT: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2517–2526, 2019.

[63] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2507–2516, 2019.

[64] Wei-Lun Chang, Hui-Po Wang, Wen-Hsiao Peng, and Wei-Chen Chiu. All about structure: Adapting structural information across domains for boosting semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1900–1909, 2019.

[65] Qing Lian, Fengmao Lv, Lixin Duan, and Boqing Gong. Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach. In *Int. Conf. Comput. Vis.*, October 2019.

[66] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Eur. Conf. Comput. Vis.*, pages 289–305, 2018.

[67] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. In *Eur. Conf. Comput. Vis.*, 2020.

[68] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[69] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Young Joon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. *Int. Conf. Comput. Vis.*, pages 6022–6031, 2019.

[70] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *IEEE Winter Conf. App. Comput. Vis.*, pages 1369–1378, 2021.

[71] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *IEEE Winter Conf. App. Comput. Vis.*, pages 1379–1389, January 2021.

[72] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9924–9935, June 2022.

[73] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4080–4090, Red Hook, NY, USA, 2017. Curran Associates Inc.

[74] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *Int. Conf. Mach. Learn. Worksh.*, volume 2. Lille, 2015.

[75] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Int. Conf. Mach. Learn.*, pages 1126—-1135, 2017.

[76] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Int. Conf. Comput. Vis.*, pages 3018–3027, 2017.

[77] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1199–1208, 2018.

[78] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[79] Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043*, 2017.

[80] Katherine Bailey and Sunny Chopra. Few-shot text classification with pre-trained word embeddings and a human in the loop, 2018.

[81] Amirreza Shabana, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. In Gabriel Brostow Tae-Kyun Kim, Stefanos Zafeiriou and Krystian Mikolajczyk, editors, *Brit. Mach. Vis. Conf.*, pages 167.1–167.13. BMVA Press, September 2017.

[82] Kate Rakelly, Evan Shelhamer, Trevor Darrell, Alexei A. Efros, and S. Levine. Few-shot segmentation propagation with guided networks. *ArXiv*, abs/1806.07373, 2018.

[83] Nanqing Dong and Eric P. Xing. Few-shot semantic segmentation with prototype learning. In *Brit. Mach. Vis. Conf.*, volume 3, 2018.

[84] C. Zhang, G. Lin, F. Liu, R. Yao, and C. Shen. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5212–5221, 2019.

[85] Zhiying Cao, Tengfei Zhang, Wenhui Diao, Yue Zhang, Xiaode Lyu, Kun Fu, and Xian Sun. Meta-seg: A generalized meta-learning framework for multi-class few-shot semantic segmentation. *IEEE Access*, 7:166109–166121, 2019.

[86] Yawei Luo, Ping Liu, Tao Guan, Junqing Yu, and Yi Yang. Adversarial style mining for one-shot unsupervised domain adaptation, 2020.

[87] Saeid Motiian, Quinn Jones, Seyed Iranmanesh, and Gianfranco Doretto. Few-shot adversarial domain adaptation. In *Adv. in Neural Inf. Process. Syst.*, pages 6670–6680, 2017.

[88] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[89] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 31:998–1008, 2018.

[90] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M Hospedales. Episodic training for domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1446–1455, 2019.

[91] Yiying Li, Yongxin Yang, Wei Zhou, and Timothy Hospedales. Feature-critic networks for heterogeneous domain generalization. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 3915–3924, 2019.

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

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

[94] Mohammad Mahfujur Rahman, Clinton Fookes, Mahsa Baktashmotlagh, and Sridha Sridharan. Correlation-aware adversarial domain adaptation and generalization. *Pattern Recognition (PR)*, 100:107124, 2020.

[95] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2551–2559, 2015.

[96] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 32:6450–6461, 2019.

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

[98] Rui Gong, Wen Li, Yuhua Chen, and Luc Van Gool. Dlow: Domain flow for adaptation and generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2477–2486, 2019.

[99] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 561–578, 2020.

[100] Xiangyu Yue, Yang Zhang, Sicheng Zhao, Alberto Sangiovanni-Vincentelli, Kurt Keutzer, and Boqing Gong. Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2100–2110, 2019.

[101] Duo Peng, Yinjie Lei, Lingqiao Liu, Pingping Zhang, and Jun Liu. Global and local texture randomization for synthetic-to-real semantic segmentation. *IEEE Transactions on Image Processing (TIP)*, 30:6594–6608, 2021.

[102] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 464–479, 2018.

[103] Xingang Pan, Xiaohang Zhan, Jianping Shi, Xiaoou Tang, and Ping Luo. Switchable whitening for deep representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1863–1871, 2019.

[104] Sungha Choi, Sanghun Jung, Huiwon Yun, Joanne T Kim, Seungryong Kim, and Jaegul Choo. Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11580–11590, 2021.

[105] Duo Peng, Yinjie Lei, Munawar Hayat, Yulan Guo, and Wen Li. Semantic-aware domain generalized segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2594–2605, June 2022.

[106] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6924–6932, 2017.

[107] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving video database with scalable annotation tooling. *CoRR*, abs/1805.04687, 2018.

[108] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulò, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *ICCV*, 2017.

[109] Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S. Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, Tiffany Fernandez, Martin Jänicke, Sudesh Mirashi,

Chiragkumar Savani, Martin Sturm, Oleksandr Vorobiov, Martin Oelker, Sebastian Garreis, and Peter Schuberth. A2d2: Audi autonomous driving dataset, 2020.

[110] Yanchao Yang and Stefano Soatto. FDA: Fourier domain adaptation for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4085–4095, 2020.

[111] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. ADVENT: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2517–2526, 2019.

[112] Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111:98–136, 2014.

[113] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, June 2012.

[114] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang. The apolloscape dataset for autonomous driving. In *2018 CVPRW)*, pages 1067–10676, June 2018.

[115] A Gaidon, Q Wang, Y Cabon, and E Vig. Virtual worlds as proxy for multi-object tracking analysis. In *CVPR*, 2016.

[116] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *CoRL*, pages 1–16, 2017.

[117] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. PSANet: Point-wise spatial attention network for scene parsing. In *ECCV*, 2018.

[118] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE CVPR)*, pages 770–778, June 2016.

[119] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Dada: Depth-aware domain adaptation in semantic segmentation. In *Proceedings of the IEEE ICCV*, pages 7364–7373, 2019.

[120] Brian Chu, Vashisht Madhavan, Oscar Beijbom, Judy Hoffman, and Trevor Darrell. Best practices for fine-tuning visual classifiers to new domains. In *ECCV Workshops*, 2016.

[121] Marco Toldo, Andrea Maracani, Umberto Michieli, and Pietro Zanuttigh. Unsupervised domain adaptation in semantic segmentation: A review. *Technologies*, 8(2), Jun 2020.

[122] Garrett Wilson and Diane J. Cook. A survey of unsupervised deep domain adaptation. *ACM Trans. Intell. Syst. Technol.*, 11(5), 2020.

[123] Ruoqi Sun, Xinge Zhu, Chongruo Wu, Chen Huang, Jianping Shi, and Lizhuang Ma. Not all areas are equal: Transfer learning for semantic segmentation via hierarchical region selection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4360–4369, 2019.

[124] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation, 2016.

[125] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

[126] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *Adv. in Neural Inf. Process. Syst.*, 2015.

[127] X. Zhang, J. Zou, K. He, and J. Sun. Accelerating very deep convolutional networks for classification and detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(10):1943–1955, 2016.

[128] Alessandro Farasin, Luca Colomba, and Paolo Garza. Double-step u-net: A deep learning-based approach for the estimation of wildfire damage severity through sentinel-2 satellite data. *Applied Sciences*, 10(12):4332, 2020.

[129] RB Andrade, GAOP Costa, GLA Mota, MX Ortega, RQ Feitosa, PJ Soto, and Christian Heipke. Evaluation of semantic segmentation methods for deforestation detection in the amazon. *ISPRS Archives; 43, B3*, 43(B3):1497–1505, 2020.

[130] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020.

[131] Lei Ding, Jing Zhang, and Lorenzo Bruzzone. Semantic segmentation of large-size vhr remote sensing images using a two-stage multiscale training architecture. *IEEE Trans. Geo. Rem. Sens.*, 58(8):5367–5376, 2020.

[132] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance, 2014.

[133] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[134] Haoran Wang, Tong Shen, Wei Zhang, Lingyu Duan, and Tao Mei. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In *Eur. Conf. Comput. Vis.*, August 2020.

[135] Ximei Wang, Ying Jin, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Transferable normalization: Towards improving transferability of deep neural networks. In *Adv. Neural Inform. Process. Syst.*, 2019.