

# Pixel-by-Pixel Cross-Domain Alignment for Few-Shot Semantic Segmentation

Antonio Tavera<sup>1</sup>, Fabio Cermelli<sup>1</sup>, Carlo Masone<sup>2</sup>, and Barbara Caputo<sup>1</sup>

<sup>1</sup>Polytechnic University of Turin, Turin, Italy

<sup>2</sup>CINI - Consorzio Interuniversitario Nazionale per l'Informatica, Rome, Italy

<sup>1</sup>{antonio.tavera, fabio.cermelli, barbara.caputo}@polito.it

## Abstract

In this paper we consider the task of semantic segmentation in autonomous driving applications. Specifically, we consider the cross-domain few-shot setting where training can use only few real-world annotated images and many annotated synthetic images. In this context, aligning the domains is made more challenging by the pixel-wise class imbalance that is intrinsic in the segmentation and that leads to ignoring the underrepresented classes and overfitting the well represented ones. We address this problem with a novel framework called *Pixel-By-Pixel Cross-Domain Alignment (PixDA)*. We propose a novel pixel-by-pixel domain adversarial loss following three criteria: (i) align the source and the target domain for each pixel, (ii) avoid negative transfer on the correctly represented pixels, and (iii) regularize the training of infrequent classes to avoid overfitting. The pixel-wise adversarial training is assisted by a novel sample selection procedure, that handles the imbalance between source and target data, and a knowledge distillation strategy, that avoids overfitting towards the few target images. We demonstrate on standard synthetic-to-real benchmarks that PixDA outperforms previous state-of-the-art methods in (1-5)-shot settings.<sup>1</sup>

## 1. Introduction

Semantic segmentation is a foundational technology in autonomous driving applications, because it provides the vehicle with information about its surroundings that is critical to safely and reliably navigate the environment. Great strides have been made to improve this technology in the autonomous driving scenario, using supervised deep learning methods trained using great quantities of data with pixel-wise annotations.

However, data collection and annotation is time consuming, expensive and hard-to-scale. A successful strategy to mitigate this issue is to rely on simulators to generate massive amounts of synthetic data [35, 37, 1]. This solution

<sup>1</sup>Code at: <https://github.com/taveraantonio/PixDA>.

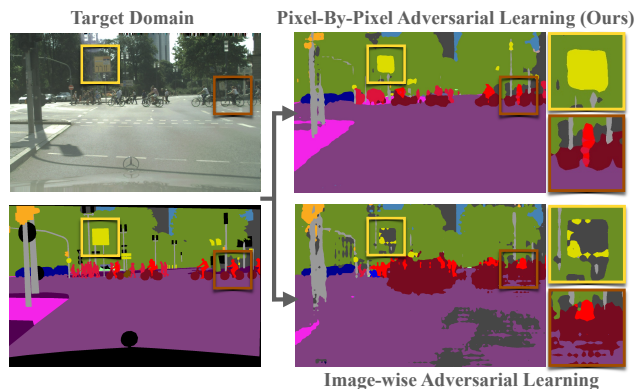


Figure 1. Comparison between the common image-wise adversarial training (bottom) and PixDA that analyzes each pixel individually (top). By prioritizing the pixel alignment according to the imbalance of the classes and the network classification confidence, PixDA achieves better accuracy, particularly for underrepresented semantic classes, e.g. traffic sign, rider and bicycle.

has the benefit that synthetic data is easy and cheap to collect, and the semantic annotations generated automatically by the graphical engines are perfect. The downside is that there is a significant shift between the synthetic domain of the training data and the real domain of the application. There are unsupervised and semi-supervised solutions that address this domain shift [43, 45], however they still need large amounts of images from the real domain thus falling back into the data collection problem. A more viable solution is to consider a few-shot setting where only a few annotated images from the real target are needed, rather than many target images without annotation.

The few-shot learning problem has been studied in several visual learning scenarios (see Sec. 2 for a review of previous works). One of its main challenges is dealing with the intrinsic imbalance between source and target data [41]. When the few-shot learning is considered within the semantic segmentation scenario, this issue is exacerbated by the intrinsic pixel-wise imbalance among segmented classes: some classes are both extremely frequent and spatially extended (e.g., sky, road), while others may appear seldom

and be small in size (*e.g.*, traffic sign). This implies that there can be a great disproportion in the number of pixels-per-class available in the target domain, with some classes that may be scarcely represented or even missing. This imbalance is more pronounced than in other problem settings, causing image-wise adversarial training methods to align large and well-represented classes, resulting in less accurate mapping of those that are under-represented in the target domain (see Fig. 1).

We argue that to address successfully cross-domain few shot learning in semantic segmentation, it is imperative to embed in the solution the intrinsic pixel-wise connotation of spatially segmenting classes. To do this, we introduce the Pixel-By-Pixel Cross-Domain Alignment framework (PixDA), that uses a novel pixel-wise discriminator and modulates the adversarial loss for each pixel to: (i) align pixel-wise source and target domains; (ii) avoid to further align correctly represented pixels and reduce the negative transfer; (iii) regularize the training of underrepresented classes to avoid overfitting. The pixel-wise adversarial training is assisted by a sample selection procedure that handles the imbalance between source and target data by progressively eliminating samples from the source domain. The two mechanisms coexist within an end-to-end training process. Summarizing, the main contributions of this paper are:

- we propose the first algorithm for cross-domain few shot semantic segmentation able to deal with classes scarcely represented in the training data by spatially aligning the domains pixel by pixel;
- we define a new pixel-wise adversarial loss that aligns source and target domains locally while reducing negative transfer and avoiding overfitting the under-represented classes;
- we evaluate our architecture on the two standard synthetic-to-real scenarios, *i.e.*, GTA5→Cityscapes and SYNTHIA→Cityscapes, where it sets new state-of-the-art scores. Additionally, an in-depth ablation study analyzes the influence of all the features introduced by our method.

## 2. Related Works

**Semantic Segmentation.** Over the last few years semantic segmentation has achieved remarkable results thanks to the widespread use of deep learning [27, 7, 55, 24, 54]. The current state-of-the-art methods differentiate themselves in the strategy applied to condition the semantic information on the global context. Methods like RefineNet [24], PSP-Net [55], ExFuse [54] or DeepLab [5, 6, 7] are designed to capture objects as well as image context at multiple scales. Other works model the hierarchical or the spatial dependencies to boost the pixel-level classifier [6, 14]. One problem

with all these methods is that they require a large amount of densely annotated images, which are expensive and time-consuming to obtain. This issue has spurred the creation of synthetic datasets [1, 37, 35] that offer high quality images with automatically generated semantic labels. Despite the clear advantages in terms of data availability and quality of the annotations, models trained using synthetic datasets face a drastic domain gap when tested with real images.

**Domain Adaptation.** Domain Adaptation (DA) refers to the study of solutions to bridge the domain gap that is present when the data used to develop the model (source) and the data the model is applied to (target) come from different distributions. Some of these solutions seek to minimize a measure of the discrepancy across domains, like the MMD in [13, 28]. Other methods exploit generative networks and image-to-image translation algorithms to generate target images conditioned on the source domain or vice-versa [18, 46, 48]. Strategies like [23, 20, 48] combine image-to-image translation with self-learning, using the predictions on a previously pre-trained model as pseudo labels to fine-tune and reinforce the model itself. Finally, the most popular approach for domain adaptation in semantic segmentation is adversarial training [44, 30, 4]. In the Unsupervised DA setting, Luo *et al.* [30] introduces the negative transfer problem caused by the common global-level adversarial alignment strategy, addressing it with a co-training strategy and an alignment performed at a category-level. Conversely, we focus on Few-Shot DA for the autonomous driving scenario and propose a novel training strategy that strengthens domain alignment at pixel level while addressing both negative transfer and overfitting on the target domain.

**Few Shot.** Few-shot learning deals with novel classes given only few images [38, 34, 9, 51, 40, 21, 12, 15, 42]. The problem has been extensively studied in the context of image-classification [40, 21, 12, 15, 42] and, only recently, in the context of semantic segmentation [38, 34, 9, 51]. Differently from few-shot learning, the purpose of few-shot in domain adaptation is to transfer the knowledge from a well-annotated source dataset to a target one containing only few annotated images [9, 29, 32, 52]. FSDA [52] tackles this problem in semantic segmentation with a two-stage method: the first stage implements a static label filtering that guides the learning towards the pixels that are difficult to classify; the second stage performs domain adaptation at image-level via two image-wise domain discriminators and using all the source images, which forces a negative transfer to the target realm. Conversely, our work achieves domain alignment at pixel-level granularity using a new pixel-wise discriminator and a new loss function exploiting the semantic and visual information for each individual pixel. Moreover, we use a novel sample selection strategy to limit the number of source images used and avoid a negative transfer.

**Knowledge Distillation.** Knowledge distillation (KD) [17] is applied to transfer knowledge from a cumbersome model to a lighter model with the aim of improving the performance of the latter by forcing the match between the predictions provided by the two networks. It has been applied first to image classification [2, 36, 50] and object detection problems [22], and only in recent years it has been deployed to the semantic segmentation [47, 26, 39] and the incremental learning tasks [3, 10, 31]. With PixDA we use KD as a regularization term [49] to avoid catastrophic forgetting of the acquired knowledge and to avoid overfitting towards the few number of target images provided by the considered few-shot setting.

### 3. Method

#### 3.1. Problem Setting

We consider the semantic segmentation task in the cross-domain few-shot setting that was formulated by Zhang *et al.* in [52]. The problem setting defines  $\mathcal{K}$ -shot as a task providing  $\mathcal{K}$  real images randomly selected for each of the  $N$  cities of the target dataset. For example, in the 1-shot setting with Cityscapes as target dataset, the whole target data is made of 18 annotated frames because Cityscapes is composed by 18 different cities. This problem setting is tailored for the autonomous driving application, where a single self-driving solution is usually deployed to a finite number of designated cities. Although not all the datasets available in the literature provide meta information regarding the division in different cities, this formulation from [52] gives a precise and well established experimental protocol.

To tackle the problem, let us denote as  $\mathcal{X}$  the set of RGB images composed by the set of pixels  $\mathcal{I}$ , and as  $\mathcal{Y}$  the set of semantic masks associating to each pixel  $i \in \mathcal{I}$  a class from the set of semantic classes  $\mathcal{C}$ . At training time we have available two sets of semantically annotated images:  $X_s = \{(x^s, y^s)\}$  which is a collection of  $N_s$  images, with  $x^s \in \mathcal{X}$  from a synthetic domain (*source*), and  $X_t = \{(x^t, y^t)\}$  which contains a small number of samples  $x^t \in \mathcal{X}$  from the real-world domain (*target*). Similarly to [52], the evaluations discussed in Sec. 4 are carried out in the (1-5)-shot setting. In this notation,  $y^s, y^t \in \mathcal{Y}$  denote the annotation masks associated with the source and target images, respectively. In this problem the goal is to use the datasets  $X_s$  and  $X_t$  to learn a function  $f$ , parameterized by  $\theta$ , from the input space  $\mathcal{X}$  to a pixel-wise probability, *i.e.*,  $f_\theta : \mathcal{X} \rightarrow \mathbb{R}^{|\mathcal{I}| \times |\mathcal{Y}|}$ , 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.*,  $p_i^c(x) = f_\theta(x)[i, c]$ .

Without domain adaptation, the parameters  $\theta$  are opti-

mized to minimize the segmentation loss  $L_{seg}$ :

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

where  $L_{seg}$  corresponds to a focal loss [25] and  $\alpha(1 - p_i^{y_i}(x))^\gamma$  is its modulating factor.

#### 3.2. Pixel-by-Pixel Adversarial Training

Many approaches [44, 30, 4] in domain adaptation deal with the domain shift problem by aligning the features extracted from the source and target domains in an adversarial manner. The common solution, first introduced by [19], is to play a min-max game between the segmentation network and an image-wise domain discriminator, in which the discriminator predicts the domain a feature belongs to, and the segmentation network tries to deceive it by making source and target features indistinguishable.

Since the domains are analyzed and aligned from a global perspective, the discriminator may disregard portions of the scene that expose few pixels of the small classes, focusing mainly on the well-represented ones. As a result, adversarial training would mostly align big and well-represented classes while inducing a negative transfer [30] on the others, which leads to poor adaptation. This problem is amplified in the few shot scenario since there is a discrepancy between the number of images in the source and target domains, and some target semantic classes may be underrepresented or even absent.

**The PixAdv Loss.** To address the imbalance among classes and reduce the negative transfer, we propose a novel adversarial loss that analyzes each pixel individually rather than operating on a global level (see Fig. 2). Our goal is to prioritize and improve pixel alignment using three criteria: (i) align the source and target domain, (ii) avoid to further align correctly represented pixel, limiting negative transfer, and (iii) regularize the training of infrequent classes, forcing the domain alignment to avoid overfitting.

To accomplish this, we use a pixel-wise discriminator whose goal is to discern, for each pixel, what domain it belongs to. The domain discriminator is a computationally less expensive version of the common Fully Convolutional discriminator found in DCGANs [33] (see Sec. 4.2 for more details). The discriminator  $D$  is trained to classify whether the features are coming from the source or the target domain. Formally, we minimize the following loss:

$$L_D(x^s, x^t) = -\sum_{i \in \mathcal{I}} \log D_i(f_\theta(x^s)) + \log(1 - D_i(f_\theta(x^t))), \quad (2)$$

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

However, using a pixel-wise discriminator without considering the class imbalance problem does not prevent a

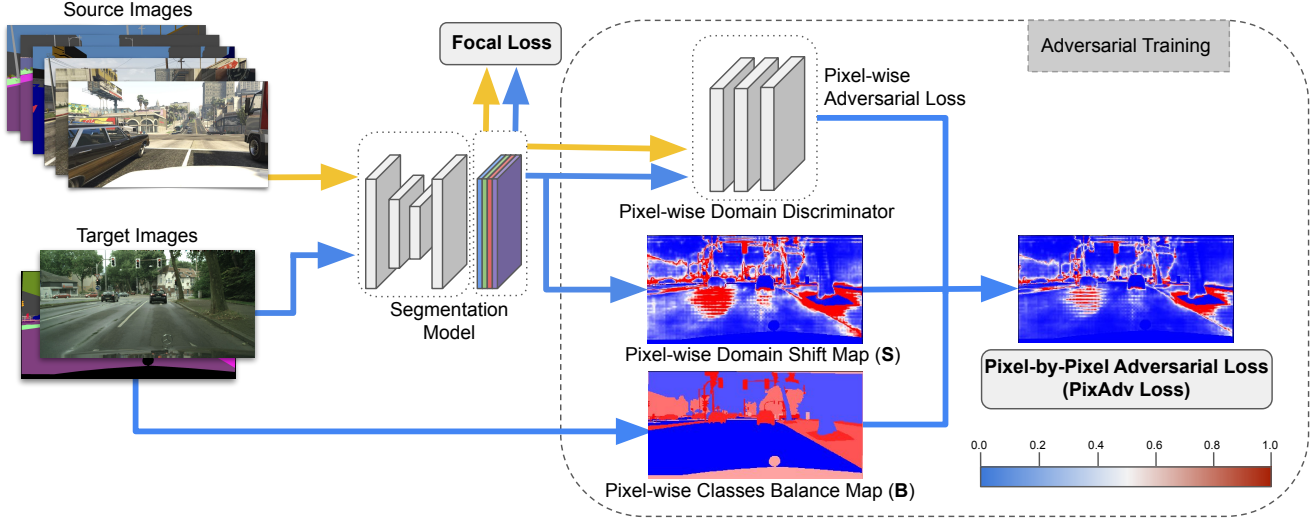


Figure 2. Illustration of the pixel-by-pixel adversarial learning of PixDA. A new pixel-wise discriminator computes the adversarial loss whose contribution at each pixel is weighted by two terms:  $S$ , that considers the ability of the model to correctly represent the pixel, and  $B$ , that weights each pixel based on the frequency of its semantic class. Yellow/blue lines refer to the source/target domain, respectively.

negative transfer effect. Hence, we introduce a novel adversarial loss function (*PixAdv Loss*), denoted as  $L_{\text{PixAdv}}$ , designed to align each pixel according to its importance. In particular, to determine the strength with which each pixel is aligned, the  $L_{\text{PixAdv}}$  modulates the adversarial loss according to a combination of two different terms, each with a specific purpose:

$$L_{\text{PixAdv}}(x^t, y^t) = -\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} S_i(x^t, y^t) B_i(y^t) \log D_i(f_\theta(x^t)). \quad (3)$$

The term  $S$  in eq. (3) is related to the network classification confidence and it is considered as a measure, for each pixel, of the ability of the network to represent it:

$$S_i(x^t, y^t) = -y_i \log p_i^{y_i}(x^t), \quad (4)$$

where  $p_i^{y_i}(x)$  denote the probability for class  $y_i$  at pixel  $i$ . High values of  $S_i$  indicate that the network misrepresents the pixel  $i$ , whereas a small value indicates that the network is able to correctly represent and classify it.

The term  $B$  in eq. (3) represents the imbalance of the pixels and aims to re-balance the classes contribution based on their frequency in the target dataset:

$$B_i(y^t) = 1 - \frac{1}{|\mathcal{I}|} \sum_{j \in \mathcal{I}} \mathbb{1}_{y_j = y_i}, \quad (5)$$

where  $\mathbb{1}$  is the indicator function, being one when  $y_j$  and  $y_i$  are equal, zero otherwise. Values of  $B_i$  which tend to 1 refer to a misrepresented class while values tending to 0 refer to a well-represented class. The term  $B$  is crucial since the target domain exposes many pixels of some classes (e.g.,

road, sidewalk) but very few of others (e.g., train, person). Through  $B$  we are able to balance the classes, resulting in a more heterogeneous and effective adaptation. We would like to point out that the terms  $S$  and  $B$  aren't used in back-propagation, but rather as a pixel-by-pixel map to modulate the adversarial loss.

Summing up, the overall segmentation network training loss function is expressed as follows:

$$\frac{1}{|X_s^k|} \sum_{x^s \in X_s^k} L_{\text{seg}}(x^s, y^s) + \frac{1}{|X_t|} \sum_{x^t \in X_t} L_{\text{seg}}(x^t, y^t) + \lambda L_{\text{PixAdv}}(x^t, y^t), \quad (6)$$

where  $L_{\text{PixAdv}}$  is the proposed adversarial pixel-wise PixAdv loss and  $X_s^k$  is a subset of the source dataset  $X_s$  selected with the sample selection procedure.

### 3.3. Sample Selection

Due to the extent and variety of the synthetic source dataset there will be source samples far away and detached from the target domain (e.g., different perspective or illumination condition). Forcing the alignment to these samples can result in negative transfer in the target dataset, lowering the network's overall performance.

With this in mind, we propose a sample selection procedure that, working side by side with the PixAdv loss, enhances the use of the source data by identifying and selecting source samples that are better aligned with the target semantic distribution. Without affecting the segmentation model and not taking part in the adversarial learning process, we simultaneously train a global image-wise domain



Figure 3. Illustration of the sample selection mechanism (top-left). At each epoch  $k$  the source dataset is subsampled selecting images that would carry worthy information to the target domain. For example, samples with different perspectives of lighting conditions w.r.t. the target data (bottom-right) are discarded.

discriminator  $D_g$  with the following loss:

$$L_{D_g}(x^s, x^t) = -\log D_g(f_\theta(x^s)) - \log(1 - D_g(f_\theta(x^t))). \quad (7)$$

The main reason for using such a discriminator is to distinguish source from target and to capture both semantic and visual domain information. Formally, at each epoch  $k$  we exploit  $D_g$  to predict the likelihood that a source image carries worthy information to the target domain and use this prediction to select a subset  $X_s^k$  of source images to be retained from the previous epoch, *i.e.*,  $|X_s^k| \leq |X_s^{k-1}|$ . Following this intuition, an image  $x^s \in X_s^{k-1}$  is added to  $X_s^k$  if  $D_g(x^s) < \delta$ , where  $\delta$  is a predefined threshold. After each epoch, we raise the threshold consequently to the increasing capacity of the image-wise discriminator to correctly classify the target data as training progresses, selecting an ever decreasing number of relevant samples (see Fig. 3 for a better understanding).

### 3.4. Fine-Tuning and Knowledge Distillation

Once the adversarial training is completed and PixAdv loss has aligned the representation between pixels of the source and target domains, we can further exploit the available semantic information on the target data to enhance the network representation.

However, naïvely fine-tuning on the target data ignores the domain alignment obtained previously and may lead to overfitting the few target images. To avoid this problem we use a Knowledge Distillation (KD) strategy. KD [17] has been designed to regularize the training of a student network using the output of a teacher network. In our framework the student  $f_{\theta_S}$  corresponds to the segmentation model which is fine-tuned, while the teacher, denoted as  $f_{\theta_T}$ , is a frozen copy of the same network after the adversarial learning pro-

cess. Formally, we optimize  $\theta_S$  with the following:

$$\frac{1}{|X_t|} \sum_{x^t \in X_t} L_{seg}(x^t, y^t) + \lambda_{kd} L_{kd}(x^t, f_{\theta_T}, f_{\theta_S}), \quad (8)$$

where  $L_{seg}$  is the segmentation loss from (1),  $\lambda_{kd}$  is a weighting parameter.  $L_{kd}$  is the distillation loss expressed as follow:

$$L_{kd} = -\sigma\left(\frac{f_{\theta_T}(x^t)}{\tau}\right) \log \sigma(f_{\theta_S}(x^t)), \quad (9)$$

where  $\sigma$  indicates the softmax function, and  $\tau$  is a temperature, as in [17].

## 4. Experiments

### 4.1. Datasets and metric

We assess the performance of our method on the two standard synthetic-to-real benchmarks used in the domain adaptation literature: GTA5 [35] to Cityscapes [8], and SYNTHIA [37] to Cityscapes [8].

**GTA5.** It consists of 24966 images synthesized from the homonymous video game. The original images size is  $1914 \times 1052$ . For training and evaluation we used the standard 19 semantic classes in common with Cityscapes.

**SYNTHIA.** We use the "RAND-CITYSCAPES" subset that consists of 9400 images synthesized from a virtual world simulator. The original image resolution is  $1280 \times 760$ . The 19 classes in common with Cityscapes are considered for training while the evaluation, following the standard protocol used in [48] and in [44], is performed on a subset of 13 and 16 classes.

**Cityscapes.** It is a real-world dataset collected across several German cities. It consists of 2975 images but for the experiments we use only a subset according to the standard  $K$ -shot selection ( $K$  images from each of the  $M$  city). We use the whole validation set made of 500 images to test our network. The original resolution is  $2048 \times 1024$ .

We assess the efficiency of PixDA in these two domain adaptation scenarios: GTA5→Cityscapes and SYNTHIA→Cityscapes. In all tests we use the standard Intersection over Union metric [11] to measure performance.

### 4.2. Implementation and training details

**Architecture.** The segmentation module of our method is DeepLab V2 [5] with ResNet101 [16] pre-trained on ImageNet. The pixel-wise discriminator we built is a Fully Convolutional discriminator which has 2 convolutional layers with kernel  $3 \times 3$ , stride 1 and padding 1, followed by a last convolutional layer with kernel  $1 \times 1$ , stride 1 and padding 0. The three layers channel numbers are  $\{64, 128, 1\}$ . The image-wise discriminator is a common Fully Convolutional

discriminator with 5 convolutional layers with kernel  $4 \times 4$ , channel numbers  $\{64, 128, 256, 512, 1\}$  and stride 2. For both discriminators, each layer except the last one is followed by a Leaky ReLU with a negative slope of 0.2.

**Training.** We implement our method in PyTorch and deploy it on two NVIDIA Tesla V100 GPUs with 16GB each. The segmentation model is trained using batch size 4 and SGD with initial learning rate  $2.5 \cdot 10^{-4}$  and adjusted at each iteration with a "poly" learning rate decay with a power of 0.9, momentum 0.9 and weight decay to 0.0005. The discriminators are trained using Adam optimizer, with learning rate  $10^{-5}$  and the same decay schedule of the segmentation model. The momentum for Adam is set to  $\{0.9, 0.99\}$ . To reduce the low-level visual domain shift (*e.g.*, color, brightness, etc.) between the source and target domains (both in the adversarial training and in the sample selection phases) we apply to each source image the FFT style translation algorithm from FDA [48], which is parameterless and computationally light. PixDA training starts with a pre-trained version of the segmentation model on source data and continues until the sample selection module selects relevant source images for the next epoch. The last fine-tuning and knowledge distillation phase lasts 200 iteration. We set  $\lambda$  equal to 0.1 for GTA and 1 for Synthia. The sample selection threshold  $\delta$  is set to 0.4 and doubled at every epoch. Finally,  $\lambda_{kd} = 0.5$  and  $\tau = 0.5$ . Test is performed without any post-processing.

**Baselines.** Our method is compared to several baselines. The first baseline that we consider is the Source Only model, *i.e.*, the network trained only with the source dataset. The Joint Training (JT) baseline, that trains for 4 epochs the model with a concatenation of the source and target images. The Fine-Tuning (FT) baseline, that fine-tunes for 30k iterations the Source Only model on the target domain. Our method, JT and FT exploit the Focal Loss to compute segmentation accuracy. We then report results for three state-of-the-art methods: FDA [48], NAAE [41], and FSDA [52]. FDA [48] and "Not All Areas are Equal" (NAAE) [41] are implemented using the same hyper-parameters proposed in their original papers, replacing only the target train set with the K-shot selection. For FSDA [52] we follow the same results and implementation details reported by the authors. DeepLabV2 with ResNet101 is used as the backbone for all the baselines with the only exception of NAAE that, as provided by its authors, uses a FCN [27] with VGG16 [53].

### 4.3. Results

**GTA5 to Cityscapes.** The results for this scenario are reported in Table 1. At a first glance, we observe that NAAE and Joint Training lead to underwhelming results, with a mIoU below 40% in all tests. FDA is slightly better, but its accuracy does not improve when increasing the number of target images from 1 to 5. Fine-Tuning the model pre-

trained on the source domain leads to comparable accuracy to the current state-of-the-art, FSDA. Finally, our PixDA is the best performer in all (1-5)-shot tests, outperforming the Source Only model by a minimum of +20.44% in the 1-shot setting to a maximum of +24.84% in the 5-shot.

Compared to the next best competitor, *i.e.*, FSDA, PixDA marks an average boost of +3.63% to the mIoU. We also note that in the 1-shot setting, the accuracy of FSDA in few classes (traffic light, motorcycle) drops below the Source Only baseline, which is indicative of a negative transfer. This result confirms that PixDA uses more effectively the information from the domain images depending on the content in the target images.

Finally, we observe that our method not only works well with predominant classes such as "road", "sky" and "building", but on average it improves the recognition of semantic categories that are under-represented, either because containing few pixels (*e.g.*, "traffic light", where we achieve a +9.39% w.r.t. to the Source Only) or because rarely appearing (*e.g.*, "train", where we achieve +15.51% w.r.t. to the Source Only). Overall, on under-represented classes (last column in Tab. 1) we outperform FSDA by +6.58%, demonstrating our ability to correctly align the pixels related to these categories. These results are qualitatively confirmed in Fig. 4, where we show that the PixAdv loss provides a stronger adaptation for small and rare classes, such as "traffic sign" and "bicycle"; hence, these categories are predicted quite accurately even in the 1-shot setting.

**SYNTHIA to Cityscapes.** Results for this scenario are reported in Table 2 and shown in Fig. 4 and confirm what we showed in the first set of experiments. NAAE, FDA and Joint Training lag behind all other methods, confirming that they are not viable solutions to address the cross-domain few-shot problem. Fine-Tuning and FSDA show similar accuracies, although it must be noted that the results from FSDA are only reported for the protocol with 13 classes, where the difficult categories "pole", "fence" and "wall" are excluded from the evaluation. Our method is designed to better handle such categories as well and, on these classes, it improves its performance w.r.t. the Source Only model by +22.88%, +13.60% and +15.16% respectively. Overall, PixDA displays the best mIoU in all (1-5)-shot settings, both with the 13 and 16 classes protocols. It outperforms the Source Only model by a minimum of +25.72% in the 1-shot scenario to a maximum of 31.66% in the 5-shot. Compared to the current state-of-the-art, *i.e.*, FSDA, PixDA scores an average accuracy improvement of +1.69% within the 13 classes protocol and of +1.86% if considering only the rare classes. The closest method to PixDA, *i.e.*, Fine-Tuning, achieves good results on these three categories but it is much less consistent than our solution. In the 16-classes protocol, PixDA achieves an average boost of +2.56% w.r.t. the second best competitor, *i.e.*, 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 <sup>19</sup>	mIoU <sup>Well</sup>	mIoU <sup>Under</sup>
1 shot	Source Only	56.65	73.92	78.8	5.81	33.02	73.72	27.71	19.6	55.47	19.24	11.24	20.22	17.86	5.26	16.01	0	30.73	20.33	18.16	30.72	41.38	20.14
	NAAE [41]	84.54	69.17	71.21	35.8	56.19	66.95	14.55	18.52	28.18	7.55	12.49	17.42	0.07	13.72	13.71	0.1	7.78	0.44	5.1	27.55	42.29	7.29
	Joint Training	66.67	63.67	76.89	17.99	74.92	65.26	17.46	9.44	52.6	5.02	2.92	20.32	26.09	17.33	7.62	0	29.91	13.79	26.66	31.29	41.17	17.72
	FDA [48]	86.6	79.01	83.35	21.26	79.7	66.54	33.29	33.1	58.84	<b>29.07</b>	23.19	24.57	37.5	36.31	14.73	4.8	33.46	<b>28.56</b>	24.54	42.02	54	25.56
	Fine-Tuning	93.44	82.44	85.75	60.53	54.22	<b>86.68</b>	<b>37.24</b>	30.77	57.4	10.64	17.1	<b>41.89</b>	51.77	32.76	34.29	0	31.65	8.21	13.49	45.28	56.02	26.76
	FSDA [52]	<b>94.49</b>	82.76	84.71	<b>62.67</b>	84.87	83.55	32.59	39.81	54.87	19.23	25.03	36.9	48.56	29.39	31.85	0	28.83	25.19	16.8	46.43	<b>60.42</b>	27.19
PixDA	93.33	<b>84.12</b>	<b>85.85</b>	59.19	<b>85.76</b>	86.18	37.15	<b>45.12</b>	<b>63.08</b>	28.05	<b>28.33</b>	32.05	<b>55.02</b>	<b>50.69</b>	<b>35</b>	<b>8.5</b>	<b>38.44</b>	26.79	<b>29.37</b>	<b>51.16</b>	<b>63.29</b>	<b>34.48</b>	
2 shot	NAAE [41]	88.75	70.84	75.9	40.12	62.69	76.46	16.24	25.67	31.87	18.83	12.98	17.65	10.86	24.37	16.23	1.99	11.95	1.12	4.04	32.03	47.3	11.03
	Joint Training	48.12	50.28	81.04	20	77.78	77.1	21.02	9.13	56.15	13.4	11.49	17.11	28.32	8.84	29.07	1.82	30.59	26.84	29.98	33.58	42.32	21.57
	FDA [48]	82.98	74.62	79.32	31.18	64.47	72.2	27.98	34.66	59.13	26.55	15.97	21.75	40.72	38.8	28.86	0.69	28.33	<b>28.88</b>	28.21	41.33	51.73	27.03
	Fine-Tuning	<b>95.2</b>	83.95	85.07	<b>67.9</b>	84.92	<b>86.66</b>	37.76	44.22	57.18	26.66	20.97	<b>43.31</b>	55.02	35.35	11.08	11.2	29.49	5.67	6.19	46.73	62.77	24.66
	FSDA [52]	94.07	84.68	<b>86.41</b>	61.62	84.54	85.07	34.29	43.65	55.97	25.56	<b>35.9</b>	37.83	51.37	36.03	<b>39.74</b>	2.22	34.51	17.91	21.94	49.12	62.89	30.19
	PixDA	94.66	<b>84.98</b>	85.95	64.5	<b>85.06</b>	85.26	<b>38.5</b>	<b>46.21</b>	<b>61.91</b>	<b>30.63</b>	27.48	38.85	<b>56.33</b>	<b>47.12</b>	37.27	<b>25.64</b>	<b>37.69</b>	19.36	<b>32.45</b>	<b>52.78</b>	<b>64.1</b>	<b>36.84</b>
3 shot	NAAE [41]	88.93	74.33	78.49	40.46	67.04	79.53	19.34	26.76	41.02	17.68	15.62	24.99	17.52	13.72	19.32	2.89	15.23	11.27	4.8	34.68	49.93	13.72
	Joint Training	65.01	66.71	80.75	22.75	82.2	74.42	18.46	11.96	55.22	8.05	5.59	21.08	21.14	23.8	29.82	0.28	30.52	10	28.38	34.53	44.65	20.63
	FDA [48]	86.35	77.89	82.92	25.97	76.77	72.85	32.21	32.86	58.92	21.98	19.85	22.08	38.79	39.37	16.91	2	31.75	29.31	25.66	41.81	53.51	25.73
	Fine-Tuning	<b>95.58</b>	84.45	86.17	<b>68.69</b>	86.48	88.11	<b>39.5</b>	46.03	59.66	16.54	18.87	44.08	58.24	18.36	37.68	<b>18.61</b>	37.44	16.75	21.37	45.59	62.7	31.57
	FSDA [52]	94.34	85.25	86.25	64.64	83.39	85.85	36.13	45.17	58.3	<b>27.35</b>	<b>35.63</b>	40.12	56.42	29.28	31.82	11.25	36.93	31.09	29.2	50.97	63.85	33.26
	PixDA	94.26	<b>85.68</b>	<b>86.9</b>	62.22	<b>87.72</b>	<b>88.85</b>	<b>38.55</b>	<b>48.88</b>	<b>64.24</b>	26.57	31.7	<b>44.11</b>	<b>59.78</b>	<b>47.21</b>	<b>42.16</b>	14.96	<b>43.55</b>	<b>35.63</b>	<b>33.37</b>	<b>54.54</b>	<b>65.05</b>	<b>40.09</b>
4 shot	NAAE [41]	90.96	75.84	79.4	45.83	70.38	80.72	19.64	30.68	38.65	19.36	18.44	30.59	20.99	13.92	23.21	1.05	19.21	18.35	9.94	36.9	51.81	16.41
	Joint Training	74.04	67.27	81.5	28.37	77.43	73.88	30.69	12.6	57.5	18.45	13.44	26.2	28.31	25.65	15.3	3.58	32.01	25.91	27.93	37.9	48.65	23.11
	FDA [48]	87.22	78.94	83.48	28.42	73.64	68.45	33.41	33.52	60.29	28.14	22.16	19.28	31.64	39.01	32.21	1.12	29.69	30.74	26.5	42.52	54.33	26.27
	Fine-Tuning	<b>95.49</b>	84.42	86.72	<b>67.84</b>	85.58	89.15	39.81	46.75	62.14	21.36	25.77	45.55	56.62	9.05	29.35	0.01	35.57	33.06	5.38	48.4	64.09	26.82
	FSDA [52]	94.42	85.21	86.73	62.99	85.6	88.83	35.36	45.11	59.18	<b>29.34</b>	<b>38.52</b>	45.07	<b>53.64</b>	29.17	40.65	0	33.16	32.64	21.08	50.88	64.66	31.93
	PixDA	95.24	<b>86.05</b>	<b>86.95</b>	67.28	<b>88.12</b>	<b>89.34</b>	<b>40.14</b>	<b>47.95</b>	<b>64.6</b>	28.87	30.5	<b>47.71</b>	<b>58.19</b>	<b>47.23</b>	<b>47.53</b>	<b>8.54</b>	<b>40.09</b>	<b>41.32</b>	<b>30.03</b>	<b>55.04</b>	<b>65.91</b>	<b>40.08</b>
5 shot	NAAE [41]	91.33	77.23	81.05	47.2	72.52	80.24	21.44	35.96	44.43	19.85	21.15	33.12	25.47	28.02	28.25	12.88	19.97	10.07	9.56	39.98	53.85	20.92
	Joint Training	63.64	71.89	82.14	28.46	79.48	71.3	26.61	7.84	58.32	9.52	6.55	27.27	30.89	6.3	5.81	2.1	34.3	24.35	25.24	34.84	45.98	19.53
	FDA [48]	86.77	78.44	83.72	33.51	78.18	72.39	31.65	33.33	59.57	20.06	20.89	19.83	28.81	29.82	24.33	4.45	29.83	29.35	14.91	41.04	54.41	22.67
	Fine-Tuning	<b>95.55</b>	85.03	<b>87.29</b>	<b>67.72</b>	87.56	88.42	40.24	43.92	61.65	21.3	25.84	<b>53.86</b>	58.41	48.34	<b>47.52</b>	14.32	38.1	23.32	27.39	53.48	64.05	38.91
	FSDA [52]	94.61	85.57	86.79	65.12	85.68	78.56	37.31	47.94	60.51	<b>27.25</b>	<b>33.68</b>	48.63	54.6	45.27	41.18	14.75	36.75	32.12	32.1	53.55	63.91	38.18
	PixDA	95.03	<b>86.33</b>	87.18	66.94	<b>88.4</b>	<b>89.43</b>	<b>40.37</b>	<b>49.25</b>	<b>64.62</b>	24.3	26.64	52.52	<b>60.39</b>	<b>49.01</b>	45.31	<b>19.91</b>	<b>40.84</b>	<b>34.78</b>	<b>34.36</b>	<b>55.56</b>	<b>65.32</b>	<b>42.14</b>

Table 1. GTA5→Cityscapes Experiments. Classes are sorted by decreasing frequency on target domain.

		Well-represented Classes											Under-represented Classes												
Shot	Method	Road	Building	Vegetation	Sidewalk	Car	Sky	Pole	Person	Fence	Wall	TSign	Bicycle	Bus	Truck	Train	TLight	Rider	Motorcycle	mIoU <sup>13</sup>	mIoU <sup>16</sup>	mIoU <sup>Well</sup>	mIoU <sup>Under</sup>		
1 shot	Source Only	27.01	64.23	74.84	16.72	38.42	77.8	14.73	50.86	0.09	3.66	5.36	16.65	6.98	2.22	10.28	1.26	30.2	25.69	36.84	7.13	30.2	25.69	36.84	7.13
	NAAE [41]	85.77	69.89	72.57	36.31	57.54	67.45	14.91	33.9	3.14	1.78	19.52	0.03	13.59	3.56	2.57	1.9	35.73	30.27	44.33	6.86	35.73	30.27	44.33	6.86
	Joint Training	65.88	71.71	76.84	32.22	46.92	83.58	28.9	56.82	0.43	10.91	23.32	25.13	26.4	9.63	16.14	11.76	42.1	36.72	47.42	18.73	42.1	36.72	47.42	18.73
	FDA [48]	30.4	52.51	61.98	15.98	61.77	75.06	27.51	51.01	0.09	6.43	16.14	25.17	10.76	5.83	18.14	7.83	33.51	29.35	38.27	13.98	33.51	29.35	38.27	13.98
	Fine-Tuning	92.08	80.72	84.23	55.1	81.21	<b>86.54</b>	33.77	55.88	<b>5.39</b>	9.18	33.24	51.4	20.77	15.2	2.78	4.22	51.02	44.48	58.41	21.27	51.02	44.48	58.41	21.27
	FSDA [52]	92.74	<b>82.06</b>	<b>84.95</b>	56.54	82.02	84.55	-	56.12	-	-	32.95	52.27	25.59	17.41	<b>23.86</b>	8.36	53.8	-	-	26.74	<b>53.8</b>	-	-	26.74
PixDA	<b>94.23</b>	82	84.05	<b>60.84</b>	<b>84.32</b>	85.04	<b>36.51</b>	<b>59.03</b>	3.85	<b>16.79</b>	<b>37.69</b>	<b>56.46</b>	<b>31.37</b>	<b>22.5</b>	20.63	<b>8.84</b>	<b>55.92</b>	<b>49.01</b>	<b>60.67</b>	<b>29.58</b>	<b>55.92</b>	<b>49.01</b>	<b>60.67</b>	<b>29.58</b>	
2 shot	NAAE [41]	87.34	68.17	75.73	34.24	63.5	75.83	14.14	32.66	16.95	5.89	17.39	6.4	11.79	9.18	3.19	2.69	37.55	32.82	47.45	8.44	37.55	32.82	47.45	8.44
	Joint Training	78.82	77.71	80.72	35.09	72.11	85.98	30.63	54.56	2.56	<b>17.12</b>	21.68	27.3	35.81	12.56	10.46	9.76	46.35	40.8	53.53	19.6	46.35	40.8	53.53	19.6
	FDA [48]	24.54	67.01	68.68	13.16	67.37	77.36	28.81	50.7	0.12	8.3	17.05	23.09	12.11	3.66	16.25	14.66	36.05	30.8	40.61	14.47	36.05	30.8	40.61	14.47
	Fine-Tuning	94.2	82.29	84.13	<b>62.53</b>	83.75	<b>85.26</b>	35.64	58.64	18.54	15.21	34.92	55.37	35.35	21.53	10.91	7.78	55.12	49.12	62.02	27.64	55.12	49.12	62.02	27.64
	FSDA [52]	92.68	82.25	<b>85.08</b>	55.44	79.87	83.09	-	57.01	-	-	35.43	53.02	24.47	<b>24.54</b>	14.95	<b>17.91</b>	54.28	-	-	28.39	54.28	-	-	28.39
	PixDA	<b>94.24</b>	<b>83.14</b>	84.37	61.84	<b>86.23</b>	84.24	<b>36.79</b>	<b>59.43</b>	<b>19.68</b>	16.58	<b>36.94</b>	<b>57.41</b>	<b>36.66</b>	22.9	<b>17.37</b>	10.16	<b>56.53</b>	<b>50.5</b>	<b>62.65</b>	<b>30.24</b>	<b>56.53</b>	<b>50.5</b>	<b>62.65</b>	<b>30.24</b>
3 shot	NAAE [41]	88.44	75.63	79.01	38.59	63.91	80.41	20.2	39.7	13.59	5.98	25.76	13.18	6.81	17.94	10.61	2.54	41.73	36.39	50.55	37.85	41.73	36.39	50.55	37.85
	Joint Training	80.41	75.92	79.23	37.93	71.48	85.81	31.03	56.19	1.52	11.23	22	35.56	24.68	14.92	15.58	16.9	47.43	41.27	53.08					

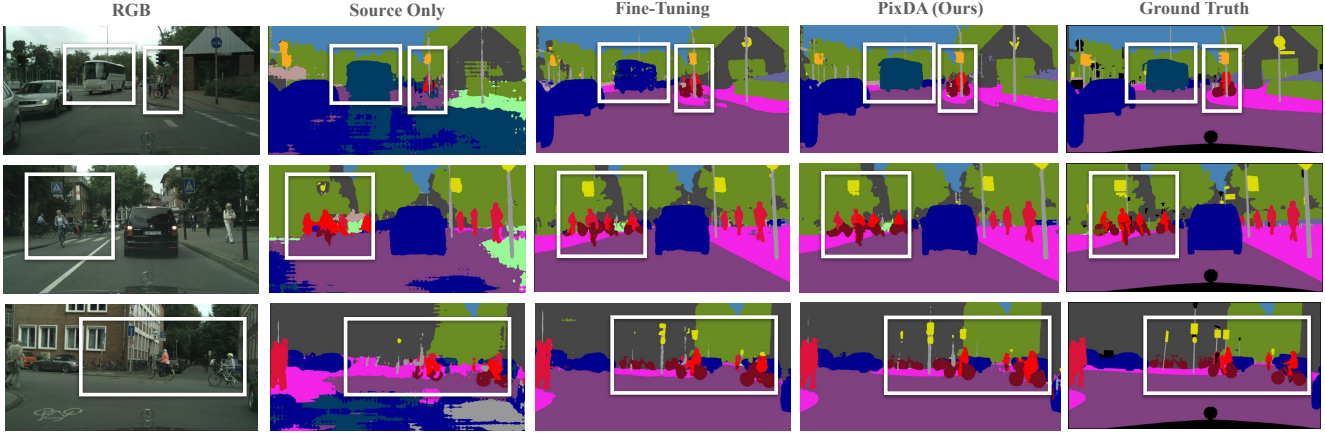


Figure 4. Qualitative results for the GTA→Cityscapes and SYNTHIA→Cityscapes scenarios. The boxes on the images highlight some examples of underrepresented classes. Qualitative comparison to the other methods is left to the Supplementary Material.

Image-wise Adv. Loss	Pixel-wise Adv. Loss	B	S	mIoU
				31.29
✓				42.43
	✓			44.60
	✓	✓		46.06
	✓		✓	47.70
	✓	✓	✓	<b>48.29</b>

Table 3. Ablation study about the choice of the adversarial loss on the GTA→Cityscapes 1-shot scenario.

Indeed, re-weighting the pixels based on their frequency ( $B$  term) yields a further improvement of +1.46%. Furthermore, decreasing the weight of well-represented pixels ( $S$  term) to prevent negative transfer is crucial, providing a +1.64% w.r.t the pixel-wise adversarial loss.

Finally, the  $B$  and  $S$  terms complement each other and boost efficiency even further when used together. The resulting loss (PixAdv) outperforms the image-wise adversarial loss by +5.86% and the pixel-wise adversarial loss by +3.69%, suggesting that weighting each pixel contribution is advantageous to prevent negative transfer and overfitting.

**Contribution of each component in PixDA.** In this section we assess to what extent each component in our framework contributes to the final performance. We move bottom-up examining six different cases: (a) Source Only model; (b) Joint Training; (c) training with the PixAdv loss; (d) with the PixAdv Loss and our Sample Selection mechanism; (e) the fine-tuning step and finally, (f) the knowledge distillation that completes the PixDA framework.

From the results in Tab. 4, it is evident that each component brings an improvement to the overall framework. In particular, the addition of the PixAdv Loss improves the Joint Training by +17%, indicating that domain alignment is necessary to obtain good performance. The Sample Selection provides an additional improvement of +1.45%, indicating that removing samples far from the target distribution is beneficial. Finally, while naïvely fine-

Method	mIoU
Source Only	30.72
Joint Training	31.29
PixAdv	48.29
+ Sample Selection	49.74
+ Fine-Tuning	50.05
+ KD	<b>51.16</b>

Table 4. Ablation study showing the effectiveness of each PixDA component on the GTA→Cityscapes 1-shot scenario.

tuning the network on the target brings a little improvement (+0.31%), using knowledge distillation brings that difference to +1.42%. We remark that only adding the PixAdv Loss already surpasses the state-of-the-art. As a follow-up test, we replaced the Focal Loss with a standard Cross Entropy, yielding a lower but still state-of-the-art result (48.89%) and confirming the effectiveness of our loss. Additional studies to assess the impact of hyperparameters in our framework are included in the supplementary material due to lack of space.

## 5. Conclusion

In this work we address the task of Cross-Domain Few-Shot Semantic Segmentation. We present a pixel-by-pixel adversarial training strategy that uses a new pixel-wise loss and discriminator to better align source and target domain and to reduce the negative transfer problem. We also assist the adversarial training with a sample selection procedure that handles the imbalance between source and target domain. Our framework achieves the state-of-the-art performance in all the 1-to-5 shot settings from the two standard synthetic-to-real benchmarks.

Future work will analyze a modified version of the sample selection strategy that selects only the top  $K$  confident source samples rather than increasing the threshold, as well as the application of the PixAdv Loss to other settings, such as Unsupervised DA, which has a preliminary assessment in the Supplementary Material, and Multi Source DA.



## References

- [1] Emanuele Alberti, Antonio Tavera, Carlo Masone, and Barbara Caputo. IDDA: A large-scale multi-domain dataset for autonomous driving. *IEEE Robot. and Autom. Lett.*, 5(4):5526–5533, 2020.
- [2] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Adv. in Neural Inf. Process. Syst.*, volume 27. Curran Associates, Inc., 2014.
- [3] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulò, Elisa Ricci, and Barbara Caputo. Modeling the background for incremental learning in semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, June 2020.
- [4] 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.
- [5] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848, 2018.
- [6] Liang-Chieh Chen, G. Papandreou, Florian Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *ArXiv*, abs/1706.05587, 2017.
- [7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Eur. Conf. Comput. Vis.*, pages 801–818, 2018.
- [8] 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.
- [9] Nanqing Dong and Eric P. Xing. Few-shot semantic segmentation with prototype learning. In *Brit. Mach. Vis. Conf.*, volume 3, 2018.
- [10] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and M. Cord. Plop: Learning without forgetting for continual semantic segmentation. *ArXiv*, abs/2011.11390, 2020.
- [11] M. Everingham, S. Eslami, L. Gool, C. K. Williams, J. Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.*, 111:98–136, 2014.
- [12] 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.
- [13] Bo Geng, Dacheng Tao, and Chao Xu. DAML: domain adaptation metric learning. *IEEE Trans. Image Process.*, 20(10):2980–2989, October 2011.
- [14] Golnaz Ghiasi and Charless C. Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *Eur. Conf. Comput. Vis.*, pages 519–534, 2016.
- [15] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Int. Conf. Comput. Vis.*, pages 3018–3027, 2017.
- [16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.
- [17] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *Adv. in Neural Inf. Process. Syst.*, 2015.
- [18] 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.
- [19] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation, 2016.
- [20] 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.
- [21] 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.
- [22] Q. Li, S. Jin, and J. Yan. Mimicking very efficient network for object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7341–7349, 2017.
- [23] 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.
- [24] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1925–1934, 2017.
- [25] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *Int. Conf. Comput. Vis.*, pages 2999–3007, 2017.
- [26] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.
- [27] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3431–3440, 2015.
- [28] 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.
- [29] Yawei Luo, Ping Liu, Tao Guan, Junqing Yu, and Yi Yang. Adversarial style mining for one-shot unsupervised domain adaptation, 2020.
- [30] 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.
- [31] Umberto Michieli and Pietro Zanuttigh. Incremental learning techniques for semantic segmentation. In *Inter. Conf. Comput. Vis. Worksh. Transf. Adapt. Source Knowledge in Comput. Vis.*, 2019.

- [32] 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.
- [33] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.
- [34] Kate Rakelly, Evan Shelhamer, Trevor Darrell, Alexei A. Efros, and S. Levine. Few-shot segmentation propagation with guided networks. *ArXiv*, abs/1806.07373, 2018.
- [35] 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.
- [36] A. Romero, Nicolas Ballas, S. Kahou, Antoine Chassang, C. Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *Computing Research Repository (CoRR)*, abs/1412.6550, 2015.
- [37] 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.
- [38] 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.
- [39] Changyong Shu, Yifan Liu, Jianfei Gao, L. Xu, and Chunhua Shen. Channel-wise distillation for semantic segmentation. *ArXiv*, abs/2011.13256, 2020.
- [40] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Adv. in Neural Inf. Process. Syst.*, pages 4077–4087, 2017.
- [41] 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.
- [42] 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.
- [43] Marco Toldo, Andrea Maracani, Umberto Michieli, and Pietro Zanuttigh. Unsupervised domain adaptation in semantic segmentation: A review. *Technologies*, 8(2), Jun 2020.
- [44] 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.
- [45] Garrett Wilson and Diane J. Cook. A survey of unsupervised deep domain adaptation. *ACM Trans. Intell. Syst. Technol.*, 11(5), 2020.
- [46] 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.
- [47] J. Xie, B. Shuai, Jianfang Hu, J. Lin, and W. Zheng. Improving fast segmentation with teacher-student learning. In *Brit. Mach. Vis. Conf.*, 2018.
- [48] Yanchao Yang and Stefano Soatto. FDA: Fourier domain adaptation for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4085–4095, 2020.
- [49] Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisiting knowledge distillation via label smoothing regularization. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3903–3911, 2020.
- [50] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *Int. Conf. Learn. Represent.*, 2017.
- [51] 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.
- [52] 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.
- [53] 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.
- [54] Zhenli Zhang, Xiangyu Zhang, Chao Peng, Xiangyang Xue, and Jian Sun. ExFuse: Enhancing feature fusion for semantic segmentation. In *Eur. Conf. Comput. Vis.*, pages 269–284, 2018.
- [55] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2881–2890, 2017.