Refign: Align and Refine for Adaptation of Semantic Segmentation to Adverse Conditions

David Bruggemann   Christos Sakaridis   Prune Truong   Luc Van Gool

ETH Zurich, Switzerland

{brdavid, csakarid, truongp, vangool}@vision.ee.ethz.ch

Abstract

Due to the scarcity of dense pixel-level semantic annotations for images recorded in adverse visual conditions, there has been a keen interest in unsupervised domain adaptation (UDA) for the semantic segmentation of such images. UDA adapts models trained on normal conditions to the target adverse-condition domains. Meanwhile, multiple datasets with driving scenes provide corresponding images of the same scenes across multiple conditions, which can serve as a form of weak supervision for domain adaptation. We propose Refign, a generic extension to self-training-based UDA methods which leverages these cross-domain correspondences. Refign consists of two steps: (1) aligning the normal-condition image to the corresponding adverse-condition image using an uncertainty-aware dense matching network, and (2) refining the adverse prediction with the normal prediction using an adaptive label correction mechanism. We design custom modules to streamline both steps and set the new state of the art for domain-adaptive semantic segmentation on several adverse-condition benchmarks, including ACDC and Dark Zurich. The approach introduces no extra training parameters, minimal computational overhead—during training only—and can be used as a drop-in extension to improve any given self-training-based UDA method. Code is available at https://github.com/brdav/refign.

1. Introduction

Semantic segmentation is arguably one of the most important tasks for scene understanding. Accurate and reliable pixel-level semantics are required for fully autonomous vehicle systems. In such safety-critical applications, robustness of the segmentation model to adverse visual conditions is pivotal. Since state-of-the-art semantic segmentation models are typically trained on clear-weather domains [6], where detailed pixel-level annotations are available, they have proven to be frail [19, 37] with respect to
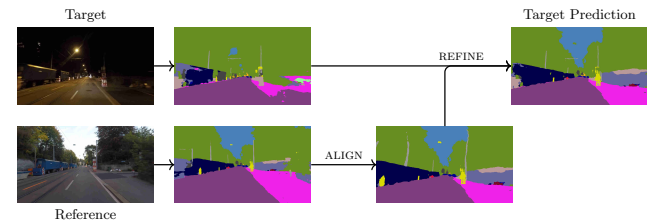

Figure 1. By leveraging a reference image depicting a similar scene as the target, Refign improves target predictions in two steps: (1) The reference predictions are spatially aligned with the target. (2) The target predictions are refined via adaptive label correction.

changes in image quality, illumination, or weather. Accordingly, a large body of research has focused on unsupervised domain adaptation (UDA) to adapt these models to different domains in which no labels are available [14, 35, 52, 54, 65].

In this paper, we propose an extension to UDA approaches, which leverages additional *reference*—or normal-condition—images to improve the predictions of the target-domain images (see Fig. 1). The reference image depicts the same scene as the target image, albeit from a different viewpoint and under favorable conditions (daytime and clear weather). For driving datasets, such image pairs can be collected with minimal extra effort by capturing the same route twice and matching frames via GPS. In recent years, a number of driving datasets have followed this procedure, *e.g.* RobotCar [27], Dark Zurich [38], ACDC [37], and Boreas [4]. When adapting a semantic segmentation model from a normal-condition dataset (*e.g.* Cityscapes [6]) to adverse conditions, the reference frames represent an *intermediate* domain. While they overlap with the target frames in terms of sensor and region characteristics, they share source domain weather and time of day. This can bolster the domain adaptation process by providing complementary, more easily learnable information, even if reference and target images might differ slightly in semantic content.

Current state-of-the-art UDA approaches [15, 61, 70] rely on self-training [66], where the network is trained with its own target domain predictions as self-supervision. The surprising effectiveness of self-training can be largely attributed to clever regularization strategies [57], in the ab-

sence of which it suffers from confirmation bias. These regularization strategies aim at correctly propagating available true labels to neighboring unlabeled samples in an iterative fashion. A critical issue in this procedure is the error propagation of noisy labels, leading to a drift in pseudo-labels if unmitigated. It has been shown that large neural networks easily overfit to label noise, which deteriorates their generalization performance [1, 69]. Our method ameliorates this error propagation issue by incorporating the predictions of two separate *views* to reason about the labels of a given scene. Broadly speaking, it could thus be considered an instance of Multi-View Learning [64]. More specifically, we consider the target prediction as a noisy label to be modified by the complementary reference class-wise probabilities, posing the fusion process as self-label correction [55]. Echoing recent advances in that field, we design an adaptive label refinement mechanism, which allows the learner to ignore or revise noisy labels.

Considering that semantic segmentation requires precise, pixel-level predictions, we hypothesize that the reference-target label fusion benefits greatly from spatial alignment of the two frames. Thus, in a preliminary step to refinement, we warp the reference prediction to align it with the target. In anticipation that such alignment is imperfect—due to dynamic objects, occlusions, and warp inaccuracies—we jointly estimate a confidence for each warped pixel, which serves as guidance in the downstream refinement. To streamline this process, we design a probabilistic extension to the geometric matching framework WarpC [50], and show its effectiveness in terms of accuracy and uncertainty awareness.

Altogether, Refign consists of the alignment module and the refinement module. Both modules introduce limited computational overhead during training and are shown to boost the performance of baseline UDA methods significantly. When added on top of DAFormer [15], Refign achieves 65.6% and 56.2% mIoU for semantic segmentation on ACDC and Dark Zurich respectively, setting the new state of the art on these adverse-condition benchmarks.

2. Related Work

Adaptation to Adverse Domains. Several works on domain adaptation for semantic segmentation have been developed with the synthetic-to-real adaptation setting in mind, focusing on adversarial alignment of the source and target domains [13, 14, 26, 39, 52–54, 56, 71], self-training with pseudo-labels in the target domain [75, 76], and combining self-training with adversarial adaptation [24] or with pixel-level adaptation via explicit transforms from source to target [20, 65]. Significantly fewer methods have been presented for adaptation from *normal to adverse* domains, which is highly relevant for practical scenarios such as automated driving, in which the perception system needs to

be robust to unfavorable conditions such as fog, night, rain, and snow. Generating partially synthetic data with simulated adverse weather from clear-weather counterparts has proven to improve performance on real adverse-condition sets featuring fog [36] and rain [47]. Sequences of domains of progressively increasing adversity have been leveraged in [7, 35, 60] via curriculum-based schemes. Light-weight input adapters [30] and adversarial style transfer [33, 43] are proposed as a generic pre-processing step at test time before predicting with source-domain models. Shared characteristics between different domains, such as sensor and time of day [9] or visibility [22], are leveraged to learn consistent representations across different datasets. The recent introduction of adverse-condition semantic segmentation datasets with image-level cross-condition correspondences, such as Dark Zurich [34] and ACDC [37], has enabled the development of approaches which leverage these correspondences as weak supervision for adapting to adverse conditions. Sparse, pixel-level correspondences are used in [21] to enforce consistency of the predictions across different conditions. DANIA [59] warps daytime predictions into the nighttime image viewpoint and applies a consistency loss for static classes only. Closely related to our work, MGCDA [38] *fuses* cross-time-of-day predictions after two-view-geometry-based alignment. Differently from their work, we directly warp the two corresponding images with an uncertainty-aware dense matching network. The warp uncertainty provides guidance for the downstream fusion, which enables more nuanced refinement, even incorporating dynamic objects. Finally, while most of the aforementioned approaches are tailored to specific conditions, our method can address arbitrary adverse conditions.

Dense Geometric Matching. Dense correspondence estimation aims at finding pixel-wise matches relating a pair of images. Approaches such as [23, 28, 32, 51] predict a 4D correlation volume, from which the dense correspondences are extracted as the `argmax` of the matching scores. Our matching network instead follows a recent line of works [18, 28, 40, 48–50] which directly regress the dense flow field or correspondence map. DGC-Net [28] employs a coarse-to-fine approach where a global cost volume is constructed at the coarsest scale to handle large motions. However, it can only handle input images of a fixed, low resolution, which significantly impacts the accuracy of the predicted flow. To circumvent this issue, GLU-Net [48] integrates both local and global correlation layers. RANSAC-Flow [40] is based on a two-stage refinement strategy, first estimating homographies relating the pair and then refining them by a predicted residual flow. COTR [18] relies on a transformer-based architecture to perform matching. Similarly to us, PDC-Net [49] adopts a probabilistic framework to regress both the flow and its uncertainty. Differently from our work, it requires sparse ground-truth matches obtained
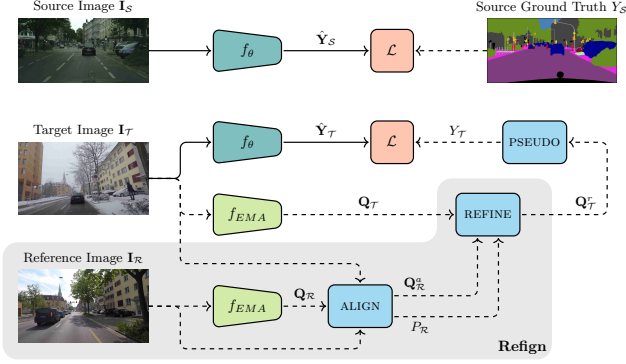
Figure 2. Generic self-training for UDA, complemented by the proposed Refign (gray-shaded area). Refign leverages an additional reference image to improve target pseudo-labels with two modules. The ALIGN module is a pre-trained dense matching network (Sec. 3.1) which (1) warps the reference prediction to align it with the target ($\mathbf{Q}_\mathcal{R}^a$) and (2) estimates a confidence map of the warp ($P_\mathcal{R}$). The REFINE module (Sec. 3.2) combines the above quantities to improve the target pseudo-labels $Y_\mathcal{T}$ for self-training. Only solid arrows backpropagate gradients.

Algorithm 1. Generic UDA self-training, complemented with Refign (shaded in gray).

Require: Samples $\mathcal{D}_\mathcal{S}$, $\mathcal{D}_\mathcal{T}$, $\mathcal{D}_\mathcal{R}$, initialized network f_θ, pretrained alignment module ALIGN
1: **for** $i = 0$ to N **do**
2:   update/initialize teacher network f_{EMA}
3:   sample $(\mathbf{I}_\mathcal{S}, Y_\mathcal{S})$ from $\mathcal{D}_\mathcal{S}$, $\mathbf{I}_\mathcal{T}$ from $\mathcal{D}_\mathcal{T}$, and $\mathbf{I}_\mathcal{R}$ from $\mathcal{D}_\mathcal{R}$
4:   $\hat{\mathbf{Y}}_\mathcal{S} \leftarrow f_\theta(\mathbf{I}_\mathcal{S})$
5:   **if** RAND$(0, 1) < 0.5$ **then**   // Adapt to $\mathcal{T}$
6:     $\mathbf{Q}_\mathcal{T} \leftarrow f_{EMA}(\mathbf{I}_\mathcal{T})$ and $\mathbf{Q}_\mathcal{R} \leftarrow f_{EMA}(\mathbf{I}_\mathcal{R})$
7:     $\mathbf{Q}_\mathcal{R}^a, P_\mathcal{R} \leftarrow$ ALIGN$(\mathbf{Q}_\mathcal{R}, \mathbf{I}_\mathcal{R}, \mathbf{I}_\mathcal{T})$ // Warp $\mathbf{Q}_\mathcal{R}$
8:     $\mathbf{Q}_\mathcal{T}^r \leftarrow$ REFINE$(\mathbf{Q}_\mathcal{T}, \mathbf{Q}_\mathcal{R}^a, P_\mathcal{R})$ // Eq. 6
9:     $Y_\mathcal{T} \leftarrow$ PSEUDO$(\mathbf{Q}_\mathcal{T}^r)$ // *e.g.* with argmax
10:    $\hat{\mathbf{Y}}_\mathcal{T} \leftarrow f_\theta(\mathbf{I}_\mathcal{T})$
11:    $l = \mathcal{L}(\hat{\mathbf{Y}}_\mathcal{S}, Y_\mathcal{S}) + \mathcal{L}(\hat{\mathbf{Y}}_\mathcal{T}, Y_\mathcal{T})$
12: **else**   // Adapt to $\mathcal{R}$
13:    $Y_\mathcal{R} \leftarrow$ PSEUDO$(f_{EMA}(\mathbf{I}_\mathcal{R}))$
14:    $\hat{\mathbf{Y}}_\mathcal{R} \leftarrow f_\theta(\mathbf{I}_\mathcal{R})$
15:    $l = \mathcal{L}(\hat{\mathbf{Y}}_\mathcal{S}, Y_\mathcal{S}) + \mathcal{L}(\hat{\mathbf{Y}}_\mathcal{R}, Y_\mathcal{R})$
16: **end if**
17: apply gradient descent to θ with $\nabla_\theta l$
18: **end for**

---

through structure from motion (SfM) for training. Instead, we present a probabilistic extension of the warp consistency framework [50], which leverages both synthetic flow and real image pairs and requires no training annotations.

Label Correction. Label correction (LC) approaches aim to improve learning from noisy labels by modifying the one-hot labels, *e.g.* through convex combinations with predicted distributions. Response-based knowledge distillation (KD) [12] methods are a prominent example for LC. While the correction is initiated by a separate teacher model in KD, Self-LC methods [31, 42, 44, 55, 68] rely on the learner itself to correct erroneous labels. Similarly to our method, multiple works have used Self-LC to improve self-training for domain-adaptive semantic segmentation [70, 73, 74]. In contrast to these works, our method uses complementary information gained from two different *views* of the same scene to correct the labels.

3. Refign

Given labeled images from a source domain $\mathcal{S}$ (*e.g.* Cityscapes [6]), and unlabeled, corresponding pairs of images from a target domain $\mathcal{T}$ (*e.g.* adverse-condition images of ACDC [37]) *and* a reference domain $\mathcal{R}$ (*e.g.* normal-condition images of ACDC), we aim to learn a model which predicts semantic segmentation maps for target-domain images. Ground-truth semantic segmentation labels are only available for the source-domain images during training. The reference image is assumed to depict the same scene as the target image, but from a different viewpoint and under better visual conditions.

Our method—Refign—is a framework-agnostic extension to self-training-based UDA methods, leveraging the

additional reference image. Underpinning Refign is the hypothesis that $\mathcal{R}$ can serve as an intermediate domain between $\mathcal{S}$ and $\mathcal{T}$. It is a well-established fact that intermediate domains can bolster UDA [10, 11, 35]. In our case, we hypothesize that the higher-quality predictions for $\mathcal{R}$ can be used to guide self-training in $\mathcal{T}$.

Fig. 2 shows a generic self-training UDA setup, with our Refign module shaded in gray. In each training iteration, the model f_θ is trained both with the source ground-truth labels $Y_\mathcal{S}$ and the target pseudo-labels $Y_\mathcal{T}$. Most state-of-the-art UDA approaches [15, 16, 61] generate the pseudo-labels with a Mean Teacher [45] (f_{EMA}), using the exponential moving average (EMA) weights of f_θ. This increases pseudo-label accuracy and mitigates confirmation bias [45]. As depicted in Fig. 2 and summarized in Alg. 1, Refign introduces two additional steps at training time to improve the pseudo-labels: (1) A pre-trained alignment module (Line 7) computes the flow from target to reference image and warps the reference prediction accordingly. The alignment module also estimates a pixel-wise warp confidence map $P_\mathcal{R}$. (2) A non-parametric refinement module (Line 8) fuses the target and warped reference predictions—using $P_\mathcal{R}$ for the fusion weights—to produce refined target predictions. The target predictions are subsequently converted to pseudo-labels according to the base UDA method (Line 9, *e.g.* through argmax and confidence weighting if Refign is built on DACS [46]). Since Refign hinges on high-quality reference predictions, we adapt f_θ to $\mathcal{R}$ in every second training iteration via the employed UDA base method (Lines 12-15, omitted in Fig. 2).

Refign does not introduce any additional training parameters, since the alignment module is pre-trained and frozen, and the refinement module is non-parametric. Con-

sequently, the memory and computation overhead during training is minor since no additional backpropagation is required. During inference, Refign is removed altogether. We describe the main two components of Refign—the alignment and refinement modules—in more detail in Sec. 3.1 and Sec. 3.2 respectively.

3.1. Alignment

Exact spatial alignment of the target and reference images is a crucial preliminary step for precise, pixel-wise semantic label refinement. Our alignment module warps the reference image to align it with the target and estimates a confidence map of the warp, which is an important asset to guide the downstream label refinement. To fulfill these requirements, we extend the warp consistency (WarpC) framework of [50] with uncertainty prediction.

WarpC. We first recap WarpC, referring to the original work [50] for a more in-depth discussion. Given two images $\mathbf{I}, \mathbf{J} \in \mathbb{R}^{h \times w \times 3}$ depicting a similar scene, the goal is to find a dense displacement field $\mathbf{F}_{\mathbf{J} \to \mathbf{I}} \in \mathbb{R}^{h \times w \times 2}$ relating pixels in $\mathbf{J}$ to $\mathbf{I}$. WarpC exploits the consistency graph shown in Fig. 3 to train the flow estimator. $\mathbf{I}$ is augmented heavily—*e.g.* through a randomly sampled homography—to yield $\mathbf{I}'$. The synthetic augmentation warp $\mathbf{W}$ subsequently supervises two objectives: (1) the direct estimate $\hat{\mathbf{F}}_{\mathbf{I}' \to \mathbf{I}}$ of the flow $\mathbf{F}_{\mathbf{I}' \to \mathbf{I}}$,

$$\mathcal{L}_{\mathbf{I}' \to \mathbf{I}} = \left\| \hat{\mathbf{F}}_{\mathbf{I}' \to \mathbf{I}} - \mathbf{W} \right\|^2, \tag{1}$$

and (2) estimation of the composite flow $\mathbf{F}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}}$ formed by chaining $\mathbf{F}_{\mathbf{I}' \to \mathbf{J}}$ and $\mathbf{F}_{\mathbf{J} \to \mathbf{I}}$:

$$\mathcal{L}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}} = \left\| V \cdot \left(\hat{\mathbf{F}}_{\mathbf{I}' \to \mathbf{J}} + \Phi_{\hat{\mathbf{F}}_{\mathbf{I}' \to \mathbf{J}}}(\hat{\mathbf{F}}_{\mathbf{J} \to \mathbf{I}}) - \mathbf{W} \right) \right\|^2$$
$$= \left\| V \cdot \left(\hat{\mathbf{F}}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}} - \mathbf{W} \right) \right\|^2. \tag{2}$$

$\Phi_{\mathbf{F}}(\mathbf{T})$ defines the warp of $\mathbf{T}$ by the flow $\mathbf{F}$ and $V \in \{0, 1\}^{h \times w}$ is the estimated visibility mask. V aims to mask out all pixels in $\mathbf{I}'$ which have no correspondence in $\mathbf{J}$ due to occlusion, image boundary, *etc*. We estimate V analogously to [50] based on the Cauchy–Schwarz inequality (see supplement, Sec. B.1). The two loss terms in (1) and (2) complement each other: $\mathcal{L}_{\mathbf{I}' \to \mathbf{I}}$ promotes convergence and favors smooth solutions, while $\mathcal{L}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}}$ learns realistic motion patterns and appearance changes. The overall network is trained via $\mathcal{L}_{\text{align}} = \mathcal{L}_{\mathbf{I}' \to \mathbf{I}} + \lambda \mathcal{L}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}}$, where λ is a weighting term balancing the individual losses.

UAWarpC. Our extension, Uncertainty-Aware WarpC (UAWarpC) adds predictive uncertainty estimation [29] to WarpC. We model the flow conditioned on the image inputs $\mathbf{I}$, $\mathbf{J}$ via a Gaussian $p(\mathbf{F}_{\mathbf{J} \to \mathbf{I}} | \mathbf{I}, \mathbf{J}) = \mathcal{N}(\mathbf{F}_{\mathbf{J} \to \mathbf{I}}; \hat{\mathbf{F}}_{\mathbf{J} \to \mathbf{I}}, \hat{\Sigma}_{\mathbf{J} \to \mathbf{I}})$, implying that the predicted flow is
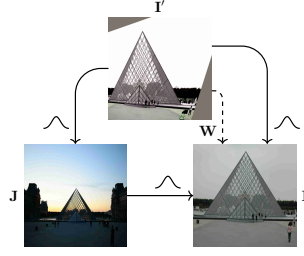

Figure 3. Warp consistency graph of [50]. Synthetic warp $\mathbf{W}$ supervises two flows: (1) the direct flow $\mathbf{I}' \to \mathbf{I}$, and (2) the composite flow $\mathbf{I}' \to \mathbf{J} \to \mathbf{I}$. We propose a probabilistic extension to [50] by predicting parameterized flow distributions (shown as ⌢).

corrupted with additive Gaussian noise. Note that a different Gaussian is predicted for each pixel. To accommodate x and y flow directions, the distributions are bivariate. We assume for simplicity that the variance is equal in both directions. Thus, the network is trained to output mean $\hat{\mathbf{F}}^{ij} \in \mathbb{R}^2$ and log-variance $\log \hat{\Sigma}^{ij} \in \mathbb{R}$ at each spatial location ij.

Gaussianity allows to easily incorporate the composite flow $\mathbf{F}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}}$ into this probabilistic framework. Assuming that $\mathbf{F}_{\mathbf{I}' \to \mathbf{J}}$ and $\mathbf{F}_{\mathbf{J} \to \mathbf{I}}$ are conditionally independent given the images, we can infer[1] that $p(\mathbf{F}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}} | \mathbf{I}, \mathbf{J}, \mathbf{I}') = \mathcal{N}(\mathbf{F}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}}; \hat{\mathbf{F}}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}}, \hat{\Sigma}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}})$ is another Gaussian. Analogously to the composite flow mean in (2), the composite flow variance is calculated through warping.

$$\hat{\Sigma}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}} = \hat{\Sigma}_{\mathbf{I}' \to \mathbf{J}} + \Phi_{\hat{\mathbf{F}}_{\mathbf{I}' \to \mathbf{J}}}(\hat{\Sigma}_{\mathbf{J} \to \mathbf{I}}) \tag{3}$$

We follow the principle of maximum log-likelihood estimation to train our model (derivation in supplement, Sec. A).

$$\begin{aligned} \mathcal{L}_{\mathbf{I}' \to \mathbf{I}}^{\text{prob}} &= -\log p(\mathbf{W} | \mathbf{I}, \mathbf{I}') \\ &\propto \frac{1}{2\hat{\Sigma}_{\mathbf{I}' \to \mathbf{I}}} \mathcal{L}_{\mathbf{I}' \to \mathbf{I}} + \log \hat{\Sigma}_{\mathbf{I}' \to \mathbf{I}} \end{aligned} \tag{4}$$

The formulation for $\mathcal{L}_{\mathbf{I}' \to \mathbf{J} \to \mathbf{I}}^{\text{prob}}$ is obtained simply by replacing the subscripts. Although the negative log-likelihood of a Gaussian corresponds to the squared error loss, in practice we use a Huber loss [17] in (1) and (2) to increase robustness to outliers.

Alignment for Label Refinement. The alignment module is trained separately on the large-scale MegaDepth [25] dataset and it is subsequently frozen during self-training of the segmentation network. During self-training, it estimates the flow $\mathbf{F}_{\mathbf{I}_\mathcal{T} \to \mathbf{I}_\mathcal{R}}$ and accordingly warps the reference class-wise probability map $\mathbf{Q}_\mathcal{R} \in \mathbb{R}^{h \times w \times c}$, yielding $\mathbf{Q}_\mathcal{R}^a$ (see Fig. 2). In addition, it estimates a warp confidence map $P_\mathcal{R} \in [0, 1]^{h \times w}$. To obtain $P_\mathcal{R}$ from our probabilistic model, we compute the probability of the true flow $\mathbf{F}_{\mathbf{I}_\mathcal{T} \to \mathbf{I}_\mathcal{R}}$ being within a radius r of the estimated flow $\hat{\mathbf{F}}_{\mathbf{I}_\mathcal{T} \to \mathbf{I}_\mathcal{R}}$, as in [49] (derivation in supplement, Sec. A).

$$P_\mathcal{R} = p(\|\mathbf{F}_{\mathbf{I}_\mathcal{T} \to \mathbf{I}_\mathcal{R}} - \hat{\mathbf{F}}_{\mathbf{I}_\mathcal{T} \to \mathbf{I}_\mathcal{R}}\| \leq r) = 1 - \exp \frac{-r^2}{2\hat{\Sigma}_{\mathbf{I}_\mathcal{T} \to \mathbf{I}_\mathcal{R}}} \tag{5}$$

---

[1]The sum of independent normally distributed random variables is normally distributed with mean/var. being the sum of the means/vars.

We set $r = 1$. The elements of $P_\mathcal{R}$ corresponding to invalid warp regions are set to zero.

3.2. Refinement

The refinement module aims to improve target class-wise probabilities $\mathbf{Q}_\mathcal{T}$ using the aligned reference class probabilities $\mathbf{Q}_\mathcal{R}^a$ and the matching confidence map $P_\mathcal{R}$. The refined target class weights $\mathbf{Q}_\mathcal{T}^r$ are then converted to pseudo-labels for self-training. The refinement is a convex combination with element-wise weights $\alpha \in \mathbb{R}^{h \times w \times c}$:

$$\mathbf{Q}_\mathcal{T}^r = (1 - \alpha) \odot \mathbf{Q}_\mathcal{T} + \alpha \odot \mathbf{Q}_\mathcal{R}^a, \qquad (6)$$

where $\odot$ denotes element-wise multiplication. Our construction of α builds on principles of self-label correction [55], as we describe in the following.

Confidence. During early training stages, the network's predictions are unreliable, especially in the more challenging adverse domain. In line with the principle of curriculum learning [3], the model should thus rely more heavily on the "easier" reference images first. Even if the reference prediction guidance is inaccurate—*e.g.* due to erroneous warping—degradation during early training is limited, as deep networks tend to learn simple patterns first before memorizing noise [1]. Later in training, on the other hand, the model should be allowed to ignore or revise faulty reference predictions. This progression can be captured via the model confidence, which increases steadily during training. More formally, we gauge the model confidence with the normalized entropy of the target probability map $\hat{H}(\mathbf{Q}_\mathcal{T}) = \frac{H(\mathbf{Q}_\mathcal{T})}{H_{\max}} \in [0,1]^{h \times w}$. We take the mean over all pixels to obtain a global image-level estimate, and introduce a hyperparameter γ as an exponent to allow for tuning. This yields a trust score s:

$$\alpha \propto s(\mathbf{Q}_\mathcal{T}) = \left(\text{mean} \left\{ \hat{H}(\mathbf{Q}_\mathcal{T}) \right\} \right)^\gamma. \qquad (7)$$

Large Static Classes. Based on the average size of connected class segments, we hereafter refer to the three classes *pole*, *traffic light*, and *traffic sign* as *small static classes* (avg. size of 8k pixels on Cityscapes [6]), while the other eight static classes are named *large static classes* (avg. size of 234k pixels). We experimentally observe that large static classes are more accurately matched by the alignment module compared to small static classes (see supplement, Sec. C). In fact, guiding the refinement through $P_\mathcal{R}$ for large static classes might be overly pessimistic. The matching network learns to be uncertain in non-textured regions (*e.g. road*, *sky*), where it is unable to identify distinct matches [49]. However, even though $P_\mathcal{R}$ is low for these regions, the broader semantic class is still matched correctly, due to smooth interpolation learned by the alignment network.

We propose more aggressive refinement for large static classes to compensate for this effect. To rule out unwanted drift towards large classes, we restrict the aggressive mixing both *spatially* and *across channels* via a binary mask $\mathbf{M} \in \{0,1\}^{h \times w \times c}$ with elements m^{ijk}. We define $\mathcal{A}$ as the set of large static classes, $Z_\mathcal{T} = \text{argmax}_c \mathbf{Q}_\mathcal{T}$ as the target predictions, and $Z_\mathcal{R}$ as the reference predictions.

$$m^{ijk} = \begin{cases} 1 & \text{if } k \in \mathcal{A} \text{ and } Z_\mathcal{T}^{ij} \in \mathcal{A} \text{ and } Z_\mathcal{R}^{ij} \in \mathcal{A}, \\ 0 & \text{otherwise}. \end{cases} \qquad (8)$$

$\mathbf{M}$ restricts aggressive mixing only to tensor elements fulfilling two criteria: (1) The element belongs to the channel of a large static class. (2) The corresponding pixel is labeled with a large static class in *both* domains. Aggressive mixing is incorporated as follows:

$$\alpha \propto \max(P_\mathcal{R}, \mathbf{M}). \qquad (9)$$

As slight abuse of notation, we use $\max(\cdot, \cdot)$ to denote the element-wise maximum of two tensors, which are broadcast to the same shape. Here, $P_\mathcal{R}$ is stacked c times along the third dimension to match the shape of $\mathbf{M}$.

Refinement Equation. Combining the two propositions, we obtain adaptive pseudo-label refinement:

$$\alpha = s(\mathbf{Q}_\mathcal{T}) \max(P_\mathcal{R}, \mathbf{M}). \qquad (10)$$

Owing to pixel-wise modulation by $P_\mathcal{R}$, this refinement scheme can ignore dynamic objects and small static objects, which are difficult to align. On the other hand, if *e.g.* two cars are coincidentally present at the same location, information transfer is still possible. Furthermore, the scheme allows for easy tuning of the degree of mixing with a single hyperparameter—the exponent γ of trust score s. Finally, since entries of $P_\mathcal{R}$ corresponding to invalid warp regions are zero, no mixing happens if no match can be found.

4. Experiments

We present extensive experiments for both UDA and geometric matching. Sec. 4.1 provides an overview of the experimental setup. Sec. 4.2 and Sec. 4.3 present comparisons with state-of-the-art methods in UDA and semi-supervised domain adaptation, respectively. Sec. 4.4 discusses ablations and Sec. 4.5 shows geometric matching comparisons. Training settings and implementation details are discussed in Sec. B of the supplement.

4.1. Setup

Datasets. For the source domain we use Cityscapes [6]. For the target and reference domains, we use ACDC [37], Dark Zurich [38], RobotCar Correspondence [21, 27], or CMU Correspondence [2, 21]. Each of these four target-domain datasets contains adverse-normal condition street

Table 1. Comparison to the state of the art in Cityscapes→ACDC domain adaptation on the ACDC test set. Methods above the double line use a DeepLabv2 model. "Ref.": for each adverse input image a reference frame from the same geo-location is used.

Method	Ref.	IoU↑																			mean
		road	sidew.	build.	wall	fence	pole	light	sign	veget.	terrain	sky	person	rider	car	truck	bus	train	motorc.	bicycle	
AdaptSegNet [52]		69.4	34.0	52.8	13.5	18.0	4.3	14.9	9.7	64.0	23.1	38.2	38.6	20.1	59.3	35.6	30.6	53.9	19.8	33.9	33.4
BDL [24]		56.0	32.5	68.1	20.1	17.4	15.8	30.2	28.7	59.9	25.3	37.7	28.7	25.5	70.2	39.6	40.5	52.7	29.2	38.4	37.7
FDA [65]		73.2	34.7	59.0	24.8	29.5	28.6	43.3	44.9	70.1	28.2	54.7	47.0	28.5	74.6	44.8	52.3	63.3	28.3	39.5	45.7
DANNet (DeepLabv2) [58]	✓	82.9	53.1	75.3	32.1	28.2	26.5	39.4	40.3	70.0	39.7	83.5	42.8	28.9	68.0	32.0	31.6	47.0	21.5	36.7	46.3
DANIA (DeepLabv2) [59]	✓	87.8	57.1	80.3	36.2	31.4	28.6	49.5	45.8	76.2	48.8	90.2	47.9	31.1	75.5	36.5	36.5	47.8	32.5	44.1	51.8
DACS [46]		58.5	34.7	76.4	20.9	22.6	31.7	32.7	46.8	58.7	39.0	36.3	43.7	20.5	72.3	39.6	34.8	51.1	24.6	38.2	41.2
Refign-DACS (ours)	✓	49.5	56.7	79.8	31.2	25.7	34.1	48.0	48.7	76.2	42.5	38.5	48.3	24.7	75.3	46.5	43.9	64.3	34.1	43.6	48.0
MGCDA (RefineNet) [38]	✓	73.4	28.7	69.9	19.3	26.3	36.8	53.0	53.3	75.4	32.0	84.6	51.0	26.1	77.6	43.2	45.9	53.9	32.7	41.5	48.7
DANNet (PSPNet) [58]	✓	84.3	54.2	77.6	38.0	30.0	18.9	41.6	35.2	71.3	39.4	86.6	48.7	29.2	76.2	41.6	43.0	58.6	32.6	43.9	50.0
DANIA (PSPNet) [59]	✓	88.4	60.6	81.1	37.1	32.8	28.4	43.2	42.6	77.7	50.5	90.5	51.5	31.1	76.0	37.4	44.9	64.0	31.8	46.3	53.5
DAFormer [15]		58.4	51.3	84.0	42.7	**35.1**	50.7	30.0	57.0	74.8	52.8	51.3	58.3	32.6	82.7	58.3	54.9	82.4	44.1	50.7	55.4
Refign-DAFormer (ours)	✓	**89.5**	**63.4**	**87.3**	**43.6**	34.3	**52.3**	**63.2**	**61.4**	**86.9**	**58.5**	**95.7**	**62.1**	**39.3**	**84.1**	**65.7**	**71.3**	**85.4**	**47.9**	**52.8**	**65.5**

Table 2. Comparison of Cityscapes→Dark Zurich methods on Dark Zurich-test. Trained models are tested for generalization on the Nighttime Driving (ND) and BDD100k-night (Bn) test sets.

Method	mIoU↑		
	Dark Zurich-test [38]	ND [8]	Bn [38,67]
DMAda (RefineNet) [8]	32.1	36.1	28.3
GCMA (RefineNet) [34]	42.0	45.6	33.2
MGCDA (RefineNet) [38]	42.5	49.4	34.9
CDAda (RefineNet) [63]	45.0	50.9	33.8
DANNet (PSPNet) [58]	45.2	47.7	28.0
DANIA (PSPNet) [59]	47.0	48.4	27.0
CCDistill (RefineNet) [9]	47.5	46.2	33.0
DACS (DeepLabv2) [46]	36.7	39.5	25.3
Refign-DACS (DeepLabv2, ours)	41.2	41.5	26.2
DAFormer [15]	53.8	54.1	33.8
Refign-DAFormer (ours)	**56.2**	**56.8**	**35.2**

Table 3. Semi-supervised domain adaptation on Cityscapes→RobotCar and Cityscapes→CMU. "Ref.": a reference frame is used for each adverse input image.

Method	Ref.	mIoU↑	
		RobotCar [21,27]	CMU [2,21]
PSPNet [72]		45.8	73.6
Cross-Season, CE [21]	✓	53.8	79.3
Cross-Season, Hinge$_C$ [21]	✓	50.6	72.4
Cross-Season, Hinge$_F$ [21]	✓	55.4	75.3
DAFormer [15]		51.7	75.6
Refign-DAFormer (ours)	✓	**60.5**	**83.6**

scene image pairs in the training set. ACDC contains 1600 training, 406 validation, and 2000 test images distributed equally among fog, night, rain, and snow. Dark Zurich contains 2416 training, 50 validation, and 151 test images for nighttime. RobotCar (resp. CMU) Correspondence contains 6511 (28766) training, 27 (25) validation, and 27 (33) test images, captured at various conditions. The RobotCar and CMU Correspondence datasets additionally have 40 and 66 coarsely annotated images, enabling semi-supervised domain adaptation. For training the alignment

network, we use MegaDepth [25] and evaluate using the test split of [40]. To test the ability of the alignment module to generalize to road scenes, we additionally evaluate it on the sparse ground-truth matches provided by [21] for the RobotCar and CMU Correspondence datasets.

Architectures. To showcase the flexibility of Refign, we combine it with state-of-the-art UDA methods. We choose DACS [46] (using DeepLabv2 [5]) and DAFormer [15] (based on SegFormer [62]) as base methods. Our alignment network follows almost exactly the same architecture as WarpC [50] (VGG-16 [41] encoder and GLU-Net [48] decoder), complemented with the uncertainty decoder of [49].

Metrics. For evaluating segmentation results, we use mean intersection over union (mIoU). Geometric matching accuracy is evaluated using the percentage of correct keypoints at a given pixel threshold T (PCK-T). The quality of matching uncertainty estimates is evaluated using sparsification error, specifically the area under the sparsification error curve (AUSE) [49] for average end-point error (AEPE).

4.2. Comparison to the State of the Art in UDA

ACDC. We present comparisons to several state-of-the-art methods on the ACDC test set in Table 1. Applying Refign on top of DAFormer [15] results in a mIoU of 65.5%, setting the new state of the art in domain adaptation from Cityscapes to ACDC. Refign boosts the performance of DAFormer by a substantial 10.1%. Besides static classes, we observe substantial improvement for dynamic classes as well, owing to our adaptive refinement method. Among DeepLabv2-based methods, our method Refign-DACS is the second best after DANIA [59]. Note that Refign boosts the mIoU of DACS [46] by 6.8%.

We present qualitative comparisons with FDA, DANIA, and DAFormer in Fig. 4. Our Refign-DAFormer consistently produces more accurate segmentation maps than the other methods. For instance, Refign corrects typical mis-
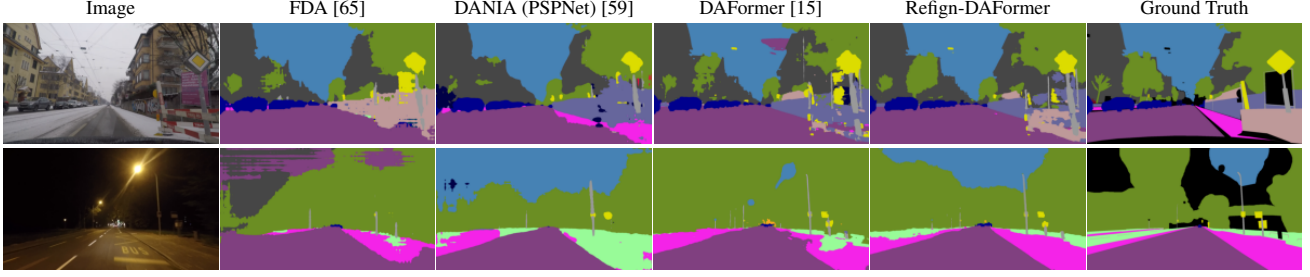
Figure 4. Qualitative segmentation results on two images of the ACDC validation set for models adapted from Cityscapes to ACDC.

Table 4. Ablation study on the ACDC validation set (metric: IoU) for different components of Refign, as detailed in (10). Default values in case of component omission are: $P_\mathcal{R} = \frac{1}{2}$, $\mathbf{M} = 0$, $s = 1$. "$\mathcal{R}$-ad": concurrent adaptation to $\mathcal{R}$, *i.e.*, Lines 12-15 of Alg. 1.

	ALIGN	$P_\mathcal{R}$	$\mathbf{M}$	s	$\mathcal{R}$-ad	road	sidew.	build.	wall	fence	pole	light	sign	veget.	terrain	sky	person	rider	car	truck	bus	train	motorc.	bicycle	**mean**
1						64.3	25.4	63.3	26.5	18.0	5.5	6.4	9.3	63.7	13.3	79.1	6.9	0.8	23.8	24.8	39.5	8.3	6.7	23.7	26.8
2	✓					86.9	60.0	82.7	49.0	32.3	43.9	58.1	40.7	83.3	38.3	94.4	12.8	7.2	52.7	43.4	50.4	15.1	30.0	43.6	48.7
3	✓	✓				67.6	52.0	83.8	47.8	36.7	56.0	69.3	51.7	73.7	37.0	63.7	46.0	27.2	78.9	67.6	75.1	83.2	42.6	48.7	58.3
4	✓	✓	✓			87.8	57.3	84.1	47.7	33.1	55.3	69.8	51.9	84.5	37.8	94.9	45.4	28.5	78.3	68.7	78.7	83.2	45.0	49.2	62.2
5	✓	✓	✓	✓		88.5	58.9	85.0	48.4	34.2	57.2	71.3	54.1	85.2	40.1	95.1	55.4	36.5	82.9	67.6	79.3	83.4	45.5	47.9	64.0
6	✓	✓	✓	✓	✓	89.4	62.4	85.5	48.6	36.6	57.7	71.0	55.0	85.3	41.0	95.1	57.3	33.1	82.9	73.6	82.5	86.0	43.9	48.1	65.0

Table 5. Hyperparameter study of the mean-entropy exponent γ on the ACDC validation set.

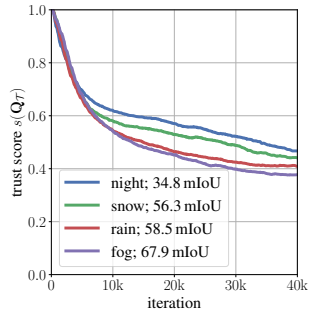|	mIoU↑				
	$\gamma = 1$	$\gamma = \frac{1}{2}$	$\gamma = \frac{1}{4}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{16}$
Refign-DAFormer	59.2	61.8	**65.0**	64.3	63.5

Figure 5. Trust score $s(\mathbf{Q}_\mathcal{T})$ during training for images of the different conditions in ACDC. On average, difficult conditions (night, snow—lower mIoU) exhibit a higher s, meaning their target predictions undergo more intensive correction by reference predictions. Shown are mIoU validation scores of the DAFormer [15] baseline.
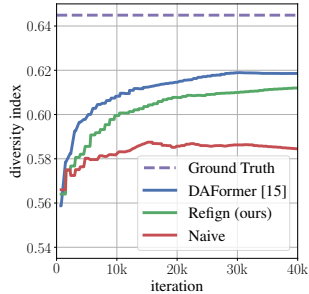

Figure 6. Prediction diversity on the ACDC validation set during training. We use normalized entropy as a diversity index. Refign preserves higher prediction diversity compared to a naive refinement scheme, which consists of simple averaging ($\alpha = 0.5$ in (6)).

classifications of DAFormer, *e.g.* sky as *road*.

Dark Zurich. In Table 2, we benchmark our method on Dark Zurich-test. Following previous works, the trained Dark Zurich models are also tested for generalization on Nighttime Driving [8] and BDD100k-night [38, 67]. Refign

markedly improves over the DACS and DAFormer baselines, both on Dark Zurich-test and the two unseen domains. Notably, Refign-DAFormer achieves a mIoU of 56.2% on Dark Zurich, setting the new state of the art.

4.3. Semi-Supervised Domain Adaptation Results

Table 3 lists semi-supervised domain adaptation results on the RobotCar and CMU Correspondence datasets. We compare to DAFormer [15] and the three PSPNet-based [72] models proposed in [21]. The models in [21] rely on sparse 2D matches obtained via a multi-stage pipeline involving camera pose estimation, 3D point cloud generation and matching, match pruning, and 2D reprojection. In contrast, our alignment module directly establishes correspondences through a dense matching network. Our models achieve the best score on both datasets, demonstrating the generality of our approach.

4.4. Ablation Study and Further Analysis

We perform an extensive analysis of our method on the ACDC validation set. To obtain more reliable performance estimates, all experiments in this section are repeated three times and we report the average performance.

Table 4 shows the ablation study of different components of our refinement scheme (10). The first row lists a naive refinement scheme, where $\alpha = 0.5$ and no alignment is performed. Adding alignment (row 2; +21.9% mIoU) improves performance substantially across all classes. Further adding $P_\mathcal{R}$ (row 3; +9.6% mIoU) sharply improves performance for dynamic classes and small static classes, but some large static classes are insufficiently mixed and their performance decreases due to low warping confidence

Table 6. Comparison to the state of the art in geometric matching. All methods are trained on MegaDepth and evaluated on MegaDepth, RobotCar, and CMU. "w/o SfM": trained without sparse structure from motion matches, "UA": uncertainty-aware matching network.

Method	w/o SfM	UA	MegaDepth [25]				RobotCar [21, 27]				CMU [2, 21]			
			PCK-1↑	PCK-5↑	PCK-10↑	AUSE↓	PCK-1↑	PCK-5↑	PCK-10↑	AUSE↓	PCK-1↑	PCK-5↑	PCK-10↑	AUSE↓
DGC+M-Net [28]	✓	✓	4.10	33.60	49.39	0.320	1.11	19.12	38.92	0.241	1.99	27.15	51.99	0.320
GLU-Net [48]	✓		29.46	55.96	62.39	-	2.21	33.72	55.28	-	21.18	80.95	91.44	-
WarpC [50]	✓		50.86	78.76	83.00	-	2.51	35.93	57.45	-	24.74	86.10	95.65	-
PDC-Net+ [49]		✓	**72.42**	**88.10**	**89.31**	0.293	2.57	36.71	**58.44**	0.186	**27.84**	85.21	92.57	**0.268**
UAWarpC (ours)	✓	✓	53.04	78.52	81.92	**0.217**	**2.59**	**36.79**	58.10	**0.155**	27.44	**88.17**	**95.94**	0.270

Figure 7. Visualizations of warped reference images and the corresponding confidence maps (white ↔ low confidence) from ACDC.

within non-textured regions (Sec. 3.2). Performance for large static classes is fully recovered by incorporating aggressive large static class mixing via $\mathbf{M}$, without sacrificing performance on small static classes or dynamic classes (row 4; +3.9% mIoU). Introducing confidence-adaptive mixing with trust score s increases performance further for 17 of the 19 classes (row 5; +1.8% mIoU). We arrive at the final configuration by adding concurrent adaptation to $\mathcal{R}$—as detailed in Lines 12-15 of Alg. 1—which adds another +1.0% mIoU in performance.

Table 5 shows the sensitivity of our method to different values of its only hyperparameter γ (mean-entropy exponent, see (7)). The method is fairly robust to decreasing γ. In Fig. 5, we compare values of trust score s during training across different conditions. We observe that s remains larger for images of more challenging conditions (night, snow). Thus, the refinement scheme automatically relies more heavily on the reference predictions for these conditions. Finally, we plot in Fig. 6 the diversity of validation set predictions during training, measured by the normalized entropy of the predicted class histogram. Compared to a naive mixing scheme consisting of simple averaging, Refign preserves considerably higher prediction diversity.

4.5. Geometric Matching Results

In Table 6 we compare our alignment strategy UAWarpC to state-of-the-art geometric matching methods. All reported methods are trained on MegaDepth. To estimate their suitability for deployment as an alignment module in Refign, we report generalization performance on two road

datasets, RobotCar and CMU. UAWarpC improves the generalization accuracy compared to the WarpC [50] baseline, demonstrating that our uncertainty modeling increases the robustness of the flow estimator. In terms of uncertainty estimation, our method matches or even outperforms the recent probabilistic PDC-Net+ [49] in AUSE.

Finally, we show qualitative warp results in Fig. 7. Notably, the alignment module successfully manages to identify dynamic objects and assigns them a low confidence.

5. Conclusion

We present Refign, a generic add-on to self-training-based UDA methods that leverages an additional *reference* image for each target-domain image. Refign consists of two steps: (1) uncertainty-aware alignment of the reference prediction with the target prediction, and (2) adaptive refinement of the target predictions with the aligned reference predictions. To enable step (1), we propose UAWarpC, a probabilistic extension to the matching method WarpC [50]. UAWarpC reaches state-of-the-art performance in both flow accuracy and uncertainty estimation. Step (2) consists of a non-parametric label correction scheme. We apply Refign to two existing UDA methods—DACS [46] and DAFormer [15]—and report state-of-the-art normal-to-adverse domain adaptation results for semantic segmentation on multiple driving scene datasets.

References

[1] Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *ICML*, 2017.

[2] Hernán Badino, Daniel Huber, and Takeo Kanade. Visual topometric localization. In *IEEE Intelligent vehicles symposium (IV)*, 2011.

[3] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, 2009.

[4] Keenan Burnett, David J Yoon, Yuchen Wu, Andrew Zou Li, Haowei Zhang, Shichen Lu, Jingxing Qian, Wei-Kang Tseng, Andrew Lambert, Keith YK Leung, et al. Boreas: A multi-season autonomous driving dataset, 2022.

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE TPAMI*, 40(4):834–848, 2017.

[6] 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 *CVPR*, 2016.

[7] Dengxin Dai, Christos Sakaridis, Simon Hecker, and Luc Van Gool. Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding. *IJCV*, 128(5):1182–1204, 2020.

[8] Dengxin Dai and Luc Van Gool. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

[9] Huan Gao, Jichang Guo, Guoli Wang, and Qian Zhang. Cross-domain correlation distillation for unsupervised domain adaptation in nighttime semantic segmentation. In *CVPR*, 2022.

[10] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, 2012.

[11] Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, 2011.

[12] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *IJCV*, 129(6):1789–1819, 2021.

[13] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. CyCADA: Cycle-consistent adversarial domain adaptation. In *ICML*, 2018.

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

[15] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, 2022.

[16] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Hrda: Context-aware high-resolution domain-adaptive semantic segmentation, 2022.

[17] Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pages 492–518. Springer, 1992.

[18] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. COTR: correspondence transformer for matching across images. In *ICCV*, 2021.

[19] Christoph Kamann and Carsten Rother. Benchmarking the robustness of semantic segmentation models. In *CVPR*, 2020.

[20] Myeongjin Kim and Hyeran Byun. Learning texture invariant representation for domain adaptation of semantic segmentation. In *CVPR*, 2020.

[21] Mans Larsson, Erik Stenborg, Lars Hammarstrand, Marc Pollefeys, Torsten Sattler, and Fredrik Kahl. A cross-season correspondence dataset for robust semantic segmentation. In *CVPR*, 2019.

[22] Sohyun Lee, Taeyoung Son, and Suha Kwak. FIFO: Learning fog-invariant features for foggy scene segmentation. In *CVPR*, 2022.

[23] Xinghui Li, Kai Han, Shuda Li, and Victor Prisacariu. Dual-resolution correspondence networks. In *NeurIPS*, 2020.

[24] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. In *CVPR*, 2019.

[25] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018.

[26] 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 *CVPR*, 2019.

[27] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.

[28] Iaroslav Melekhov, Aleksei Tiulpin, Torsten Sattler, Marc Pollefeys, Esa Rahtu, and Juho Kannala. DGC-Net: Dense geometric correspondence network. In *WACV*, 2019.

[29] David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, volume 1, pages 55–60, 1994.

[30] Horia Porav, Tom Bruls, and Paul Newman. Don't worry about the weather: Unsupervised condition-dependent domain adaptation. In *IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019.

[31] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In *ICLR*, 2015.

[32] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. In *NeurIPS*, 2018.

[33] Eduardo Romera, Luis M. Bergasa, Kailun Yang, Jose M. Alvarez, and Rafael Barea. Bridging the day and night domain gap for semantic segmentation. In *IEEE Intelligent Vehicles Symposium (IV)*, 2019.

[34] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *ICCV*, 2019.

[35] Christos Sakaridis, Dengxin Dai, Simon Hecker, and Luc Van Gool. Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In *ECCV*, 2018.

[36] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *IJCV*, 126(9):973–992, 2018.

[37] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. ACDC: The Adverse Conditions Dataset with Correspondences for semantic driving scene understanding. In *ICCV*, 2021.

[38] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *TPAMI*, 44(6):3139–3153, 2022.

[39] Swami Sankaranarayanan, Yogesh Balaji, Arpit Jain, Ser Nam Lim, and Rama Chellappa. Learning from synthetic data: Addressing domain shift for semantic segmentation. In *CVPR*, 2018.

[40] Xi Shen, François Darmon, Alexei A Efros, and Mathieu Aubry. Ransac-flow: generic two-stage image alignment. In *ECCV*, 2020.

[41] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[42] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In *ICML*, 2019.

[43] Lei Sun, Kaiwei Wang, Kailun Yang, and Kaite Xiang. See clearer at night: towards robust nighttime semantic segmentation through day-night image conversion. In *Artificial Intelligence and Machine Learning in Defense Applications*, 2019.

[44] Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *CVPR*, 2018.

[45] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017.

[46] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *WACV*, 2021.

[47] Maxime Tremblay, Shirsendu Sukanta Halder, Raoul De Charette, and Jean-François Lalonde. Rain rendering for evaluating and improving robustness to bad weather. *IJCV*, 129(2):341–360, 2021.

[48] Prune Truong, Martin Danelljan, and Radu Timofte. Glunet: Global-local universal network for dense flow and correspondences. In *CVPR*, 2020.

[49] Prune Truong, Martin Danelljan, Radu Timofte, and Luc Van Gool. Pdc-net+: Enhanced probabilistic dense correspondence network, 2021.

[50] Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Warp consistency for unsupervised learning of dense correspondences. In *ICCV*, 2021.

[51] Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Probabilistic warp consistency for weakly-supervised semantic correspondences. In *CVPR*, 2022.

[52] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *CVPR*, 2018.

[53] Yi-Hsuan Tsai, Kihyuk Sohn, Samuel Schulter, and Manmohan Chandraker. Domain adaptation for structured output via discriminative patch representations. In *ICCV*, 2019.

[54] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019.

[55] Xinshao Wang, Yang Hua, Elyor Kodirov, David A Clifton, and Neil M Robertson. Proselflc: Progressive self label correction for training robust deep neural networks. In *CVPR*, 2021.

[56] Zhonghao Wang, Mo Yu, Yunchao Wei, Rogerio Feris, Jinjun Xiong, Wen-mei Hwu, Thomas S Huang, and Honghui Shi. Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation. In *CVPR*, 2020.

[57] Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data. In *ICLR*, 2021.

[58] Xinyi Wu, Zhenyao Wu, Hao Guo, Lili Ju, and Song Wang. DANNet: A one-stage domain adaptation network for unsupervised nighttime semantic segmentation. In *CVPR*, 2021.

[59] Xinyi Wu, Zhenyao Wu, Lili Ju, and Song Wang. A one-stage domain adaptation network with image alignment for unsupervised nighttime semantic segmentation. *IEEE TPAMI*, (01):1–1, 2021.

[60] Markus Wulfmeier, Alex Bewley, and Ingmar Posner. Incremental adversarial domain adaptation for continually changing environments. In *ICRA*, 2018.

[61] Binhui Xie, Shuang Li, Mingjia Li, Chi Harold Liu, Gao Huang, and Guoren Wang. Sepico: Semantic-guided pixel contrast for domain adaptive semantic segmentation, 2022.

[62] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021.

[63] Qi Xu, Yinan Ma, Jing Wu, Chengnian Long, and Xiaolin Huang. Cdada: A curriculum domain adaptation for nighttime semantic segmentation. In *ICCV*, 2021.

[64] Xiaoqiang Yan, Shizhe Hu, Yiqiao Mao, Yangdong Ye, and Hui Yu. Deep multi-view learning methods: a review. *Neurocomputing*, 448:106–129, 2021.

[65] Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *CVPR*, 2020.

[66] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd annual meeting of the association for computational linguistics*, 1995.

[67] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020.

[68] Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisiting knowledge distillation via label smoothing regularization. In *CVPR*, 2020.

[69] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[70] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *CVPR*, 2021.

[71] Yiheng Zhang, Zhaofan Qiu, Ting Yao, Dong Liu, and Tao Mei. Fully convolutional adaptation networks for semantic segmentation. In *CVPR*, 2018.

[72] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017.

[73] Zhedong Zheng and Yi Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *IJCV*, 129(4):1106–1120, 2021.

[74] Zhedong Zheng and Yi Yang. Unsupervised scene adaptation with memory regularization in vivo. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.

[75] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *ICCV*, 2019.

[76] Yang Zou, Zhiding Yu, B.V.K. Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018.