Explore Epistemic Uncertainty in Domain Adaptive Semantic Segmentation

Kai Yao
University of Liverpool
Liverpool, United Kingdom
Xi'an Jiaotong-Liverpool University
Suzhou, Jiangsu, China
kai.yao@liverpool.ac.uk

Zixian Su
University of Liverpool
Liverpool, United Kingdom
Xi'an Jiaotong-Liverpool University
Suzhou, Jiangsu, China
zixian.su@liverpool.ac.uk

Xi Yang
Xi'an Jiaotong-Liverpool University
Suzhou, Jiangsu, China
xi.yang01@xjtlu.edu.cn

Jie Sun*
Xi'an Jiaotong-liverpool University
Suzhou, Jiangsu, China
jie.sun@xjtlu.edu.cn

Kaizhu Huang*
Duke Kunshan University
Kunshan, Jiangsu, China
kaizhu.huang@dukekunshan.edu.cn

ABSTRACT

In domain adaptive segmentation, domain shift may cause erroneous high-confidence predictions on the target domain, resulting in poor self-training. To alleviate the potential error, most previous works mainly consider aleatoric uncertainty arising from the inherit data noise. This may however lead to overconfidence in incorrect predictions and thus limit the performance. In this paper, we take advantage of Deterministic Uncertainty Methods (DUM) to explore the epistemic uncertainty, which reflects accurately the domain gap depending on the model choice and parameter fitting trained on source domain. The epistemic uncertainty on target domain is evaluated on-the-fly to facilitate online reweighting and correction in the self-training process. Meanwhile, to tackle the class-wise quantity and learning difficulty imbalance problem, we introduce a novel data resampling strategy to promote simultaneous convergence across different categories. This strategy prevents the class-level over-fitting in source domain and further boosts the adaptation performance by better quantifying the uncertainty in target domain. We illustrate the superiority of our method compared with the state-of-the-art methods.

CCS CONCEPTS

• **Computing methodologies → Transfer learning**; *Image segmentation.*

KEYWORDS

domain adaptation, semantic segmentation, uncertainty estimation

*Corresponding authors.

CIKM '23, October 21–25, 2023, Birmingham, United Kingdom.

ACM ISBN 979-8-4007-0124-5/23/10…$15.00
https://doi.org/10.1145/3583780.3614872

ACM Reference Format:
Kai Yao, Zixian Su, Xi Yang, Jie Sun, and Kaizhu Huang. 2023. Explore Epistemic Uncertainty in Domain Adaptive Semantic Segmentation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3583780.3614872

1 INTRODUCTION

Despite the great success in enhancing semantic segmentation, deep-learning-based methods[4, 34] may not work well under the domain shift problem. To tackle this issue, unsupervised domain adaption (UDA)[14, 38] is proposed to transfer the knowledge from the labeled source domain to the unlabeled target images. As predominant solutions in the literature, distance-based metric learning[25, 41] and adversarial learning[14, 38, 43, 47] are usually engaged to align the distribution between source and target domains on image-level or/and feature-level, which implicitly measure the distribution shift. Albeit the usefulness, they either introduce extra artificially designed metrics or require a sizeable computational cost, falling short of delivering a widely adopted practical solution in real-world scenarios. An alternative would be self-training[11, 21, 28, 37, 40, 45, 48], which gradually learns the adaptation in the self-paced learning curriculum by iteratively generating pseudo label on the target domain. Without introducing any extra complex computational pipeline, self-training methods become more popular thanks to their simplicity, stability, and efficiency.

However, most previous attempts fail to consider the reliability/uncertainty of the generated pseudo labels. For instance, the pretrained network may produce pseudo labels with overconfidence in large wrongly-predicted regions, which could result from the softmax output. Unfortunately, one may not address such an issue by simply setting a probability threshold such as maximum class probability (MCP). In another word, methods based on the prediction probability fail to reflect the real reliability/uncertainty of the model in essence [13]. Instead, it may just make the network more confident about existing false predictions by self-training. As explained in Figure 2, in previous self-training paradigm, source-similar target samples have a higher possibility of getting correct pseudo labels. For the source-dissimilar ones, although they may

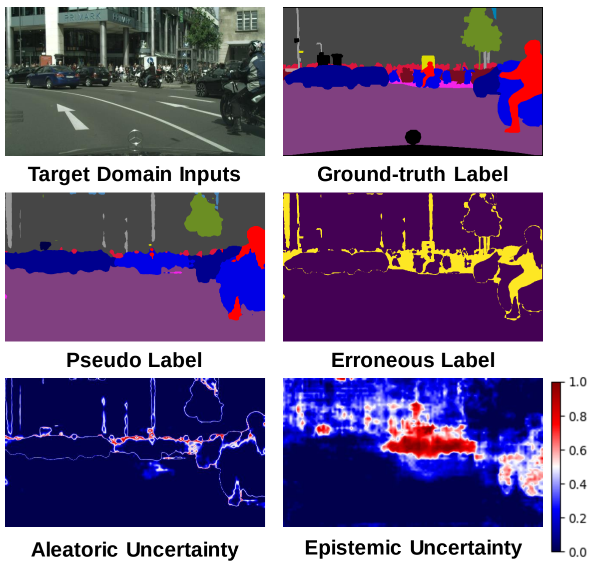


Figure 1: Illustration of aleatoric [13] v.s. epistemic uncertainty (our method) on target images in the early stage of training.

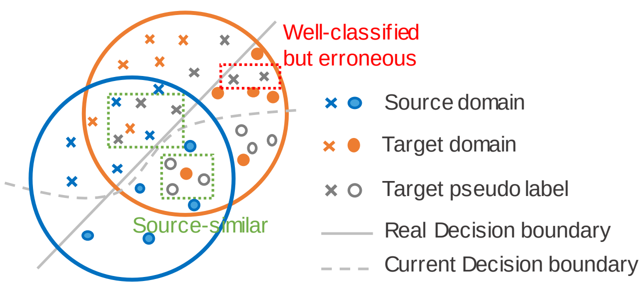


Figure 2: In UDA setting, rather than trusting a well-classified pseudo label, a source-similar pseudo label is much more reliable.

be well-classified with high confidence, the obtained results are entirely wrong and further misguide the model. To this end, how to estimate the real uncertainty in domain adaptation presents a challenging yet crucial problem, which is critically important, especially in practical scenarios like autonomous driving.

Uncertainty in a model's predictions can arise from two different sources: aleatoric and epistemic uncertainty [8, 19]. Aleatoric uncertainty encompasses the noise inherent in data, which is consequently irreducible. Epistemic uncertainty, however, quantifies the uncertainty originating from the model choice and parameter fitting and can be reducible with the increase of the training data. For instance, MCP is more likely to be associated with aleatoric uncertainty [13], focusing more on fitting the data bias. In most cases, it could merely generate limited improvement through self-training. In comparison, epistemic uncertainty can reflect the confidence of the model with regard to the newly introduced distribution. If the epistemic uncertainty is incorporated into the domain adaptation

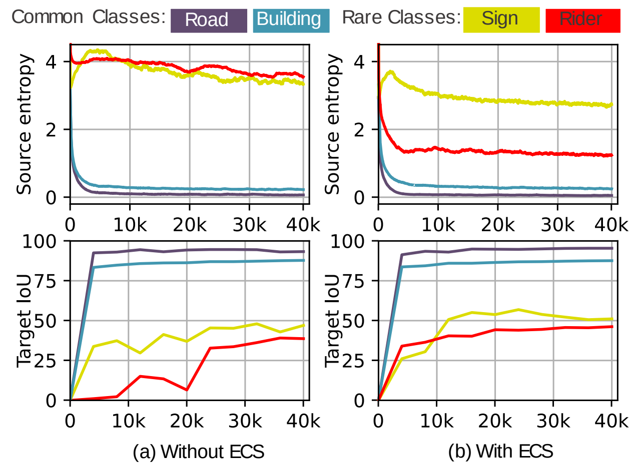


Figure 3: Source class-wise entropy v.s. target IoU for each class (a) without and (b) with ECS. Different classes show a similar trend in entropy decreasing with ECS, and the adaptation performance is further boosted in both common and rare classes.

process, it can explicitly embody the distributional gap and alleviate the cumulative error of the pseudo labels in self-training from another perspective. As shown in Figure 1, compared with aleatoric uncertainty which commonly exists in the edges of different classes, epistemic uncertainty provides richer information of confidence and overlaps more with the erroneous labels.

In addition, although the pseudo labels may be rectified through epistemic uncertainty estimation during training, this process still depends highly on the model's episteme on each class. As observed in Figure 3, the entropy of common classes (Road & Building) decreases quickly at the beginning and remains unchanged afterwards. Based on the interpretation of entropy as uncertainty in information theory [2] and our above definition of uncertainties, we can reasonably infer that irreducible entropy components exist due to inherent noise, which are treated as the aleatoric uncertainty following the above description. Meanwhile, owing to class-imbalanced sampling in the training process, the decreasing speed of the reducible entropy (epistemic uncertainty) is not the same for each class. This results in the epistemic dissonance issue: 1) for common and easy classes, the model will reach the bound of aleatoric uncertainty very fast, thus no further gain can be achieved due to the noise inherent in data; 2) for rare and hard classes, the model cannot possess enough episteme in the early training stage; the resulting poor pseudo labels would further deteriorate the self-training.

In this paper, inspired from the recently emerged new line of uncertainty estimation, termed deterministic uncertainty methods (DUMs) [9, 32], we introduce predictive epistemic uncertainty estimation into the UDA task in a computational efficient manner. To be specific, we replace the last layer of the segmentation model with a distinction maximization layer [27] to generate informative and discriminative representations. Then, an auxiliary uncertainty head is trained on source domain, enabling the measurement of source-dissimilar representations. Last, the pixel-wise epistemic uncertainty prediction on target domain is engaged to rectify the self-training process by filtering the high uncertainty areas, significantly enhancing the reliability of pseudo labels. Meanwhile, we introduce a novel training data resampling strategy, Episteme-based Class Sampling, to prevent degradation in uncertainty estimation, aiming to solve the epistemic dissonance issue. In particular, the sampling strategy is updated dynamically based on the moving average of class entropy and increases the sampling rate of classes according to the reducible entropy, i.e., epistemic uncertainty. This strategy is certified to prevent the class-level over-fitting in source domain and further boosts the adaptation performance by better quantifying the uncertainty in target domain.

Our contributions can be summarized as follows:

- To the best of our knowledge, we are the first to explore epistemic uncertainty in domain adaptive segmentation. Our proposed model can quantify the epistemic uncertainty for improving the reliability of the pseudo labels in the self-training process.
- We propose a novel data sampling strategy, Episteme-based Class Sampling, to dynamically balance the model's episteme on different classes across the training process, benefiting the uncertainty estimation and domain adaptation at the same time.
- Experiments show that our approach outperforms existing state-of-the-art methods by a large margin.

2 RELATED WORK

2.1 Unsupervised Domain Adaptation

Distribution shift hinders the generalization of a pre-trained model in real scenarios. UDA aims to transfer the knowledge learned from the labeled source domain to the target domain in an unsupervised way. This task has been widely studied in image classification[18, 24], semantic segmentation[14, 38], object detection[5, 42], etc., in computer vision. The existing UDA methods can be roughly summarized into three categories:

Metric Learning. An intuitive solution for domain adaptation is to adopt a proper metric to measure the variational distance between two domains and subsequently regularize neural networks to minimize this distance. For example, maximum mean discrepancy[25] measures between the feature embeddings of the source and target domains in a RKHS. Furthermore, higher-order statistics and other well-designed discrepancies are adopted in [3, 30, 41, 46].

Adversarial Learning. Recent research[14, 38, 43, 47] utilized adversarial learning to tackle the domain shift problem from different aspects, including pixel-level alignment, feature-level alignment, and the joint learning methods. A discriminator is leveraged in this setting, where the generative model aims to confuse the discriminator with the synthesized images or extracted features to obtain the domain-invariant representations.

Self-Training. Unlike the previous two approaches that focus on source-target alignment, self-training is more concerned with the specific information in target domain, aiming to lower the uncertainty of the pretrained model. In the past, this approach often takes a two-state pipeline[21, 28, 44, 44], which finetunes the trained model using the pseudo labels on target images given by model prediction. Recently, the online self-training[37, 45] has become popular where pseudo labels are calculated and corrected during training. Less complex setup and quick update enable online self-training a more wide range of application scenarios. Moreover, in [45], pseudo labels obtained online attain high quality within a few steps, showing a distinct advantage over the conventional offline setting.

We argue that the key for self-training is to produce high-accuracy pseudo labels, but prediction errors are inevitable especially when the domain gap is significant. Specifically, pseudo labels with over-confidence in large wrongly-predicted regions cannot be detected through an offline threshold method, making the improvement limited to edge areas. We tackle this problem by introducing Deterministic Uncertainty Methods illustrated below. Benefiting from its capability to quantify epistemic uncertainty, our method can largely improve the quality of pseudo labels.

2.2 Uncertainty Estimation

Uncertainty estimation tries to assign a level of confidence to a model's output. Among approaches that estimate model uncertainty in deep learning, Bayesian models[20, 31] are the predominant one and can predict both epistemic uncertainty and aleatoric uncertainty. While exact Bayesian is intractable, a range of approximate methods have been developed and achieved good results in classification, even though they fail to deliver a practical solution to other application scenarios such as semantic segmentation. An alternative would be Monte-Carlo Dropout[10], which is an easy and simple to implement. Yet, its reliability is not always promising. Previous attempts also exploit Deep Ensembles[22], which trains multiple models from different initializations and averages their predictions as the model output. Though the method is simple and effective, it comes with a price of high computational cost. Recently, Deterministic Uncertainty Methods[1, 9, 23, 27, 32, 39] are leading a trend in quantifying predictive uncertainty. It is specifically designed to quantify epistemic uncertainty from the distribution of the latent representations in a computationally efficient manner. This advanced approach has shown superiority in several computer vision tasks. To the best of our knowledge, we are the first to investigate the epistemic uncertainty in domain adaptive semantic segmentation. The idea that epistemic uncertainty should increase with the distribution shift relates deterministic uncertainty methods to domain adaptation. Following [9, 27], we leverage a set of trainable prototypes to learn a discriminant latent space for accuracy improvement and uncertainty prediction.

3 MAIN METHODOLOGY

3.1 Self-Training Baseline for UDA

We will first give an overview of our baseline self-training UDA framework. Given the paired source domain images with one-hot labels $\mathcal{D}^{\mathcal{S}} = \{(x_n^{\mathcal{S}}, y_n^{\mathcal{S}})\}_{n=1}^{N}$ with C classes and the unlabeled target domain images $\mathcal{D}^{\mathcal{T}} = \{(x_m^{\mathcal{T}})\}_{m=1}^{M}$, we aim to train a segmentation network that achieves promising performance on target domain. Directly training the network f on source data with categorical cross-entropy (CE) loss cannot guarantee good performance on

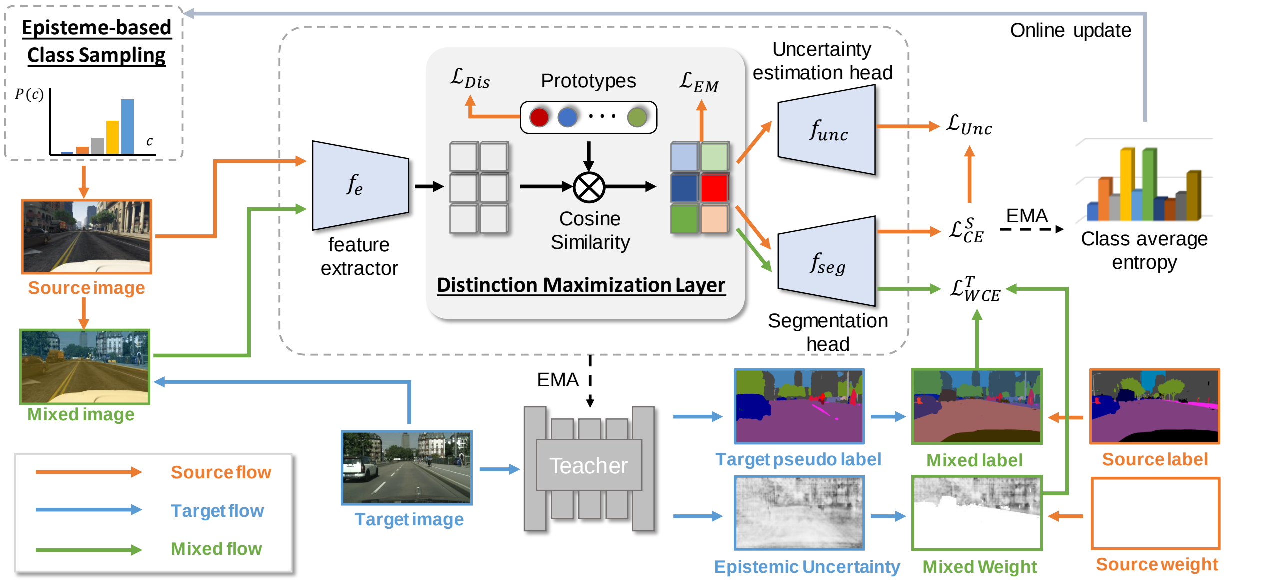


Figure 4: Overview of our self-training UDA framework. We train the uncertainty estimation on source domain upon deterministic representation, which is constrained by the Distinction Maximization Layer. Then we utilize the epistemic uncertainty estimated on target domain images to rectify the self-training process by reweighting the cross entropy loss with pseudo label. In addition, Episteme-based Class Sampling (ECS) is introduced to balance the episteme on source domain.

target domain due to the domain gap:

$$\mathcal{L}^{\mathcal{S}}_{CE} = -\sum_{i=1}^{H\times W}\sum_{c=1}^{C} y^{\mathcal{S}}_{(i,c)} \log(f(x^{\mathcal{S}})_{(i,c)}). \tag{1}$$

To tackle the domain gap, one popular self-training (ST) solution [36] is to utilize a source-trained teacher network f_{te} to generate target domain pseudo labels $\hat{y}^{\mathcal{T}}$ by the maximum probable class:

$$\hat{y}^{\mathcal{T}}_{(i,c)} = \begin{cases} 1, & \text{if } c = \arg\max_{c'} f_{te}(x^{\mathcal{T}})_{(i,c')} \\ 0, & \text{otherwise} \end{cases}.$$

The teacher network f_{te} is not updated by gradient backpropagation but the Exponentially Moving Average (EMA) of the student network weights θ_f after each training step t:

$$\theta^{t+1}_{f_{te}} \leftarrow m\theta^{t}_{f_{te}} + (1-m)\theta_f, \tag{2}$$

where m is the momentum to temporally ensemble the student network.

In this work, to further stabilize the training process, we follow DACS [37] to generate pseudo labels on non-augmented images, and train the student network with domain-mixed images. In each iteration, a pair of source and target images with the corresponding ground truth and pseudo labels are sampled, denoted as $(x^{\mathcal{S}}_n, y^{\mathcal{S}}_n, x^{\mathcal{T}}_m, \hat{y}^{\mathcal{T}}_m)$. Next, a subset of classes [29] is randomly selected from $y^{\mathcal{S}}_n$ to form the binary mask $M \in \{0,1\}^{H\times W}$, where the pixel is 1 if belonging to the subset otherwise 0. The mixed images with their labels are defined as:

$$\begin{cases} x^{mix} = x^{\mathcal{S}}_n \odot M + x^{\mathcal{T}}_m \odot (1-M) \\ y^{mix} = y^{\mathcal{S}}_n \odot M + \hat{y}^{\mathcal{T}}_m \odot (1-M) \end{cases},$$

where $\odot$ denotes the Hadamard product. The student model is then trained with mixed image and its label with weighted cross-entropy (WCE) loss:

$$\mathcal{L}^{\mathcal{T}}_{WCE} = -\sum_{i=1}^{H\times W}\sum_{c=1}^{C} w^{mix}_i y^{mix}_{(i,c)} \log(f(x^{mix})_{(i,c)}). \tag{3}$$

Here, $w^{mix} = \mathbb{1}\odot M + w^{\mathcal{T}}\odot(1-M)$ is the weight map to alleviate the impact of potential erroneous pseudo labels, where $\mathbb{1} \in \mathbb{R}^{H\times W}$ is an all-one map for source domain. Typically, the weight map of target domain $w^{\mathcal{T}}$ is generated by a pre-defined confidence threshold upon the maximum class probability [15, 37, 40], which inevitably degrades the model performance due to the over-confidence nature of CNN. Different from previous works, we introduce epistemic uncertainty estimation to generate more precise reweighting map (described in Section 3.2), significantly boosting the performance of the self-training process.

3.2 Epistemic Uncertainty Estimation

To cope with the epistemic uncertainty quantification, we introduce a distinction maximization (DM) layer [27] that has been recently considered as a replacement of the classification layer (last layer) for uncertainty estimation [9]. To be specific, in a DM layer, the units of the classification layer are seen as class-level representative prototypes and the classification prediction is computed by calculating the similarity between the input representations and the prototypes. This DM layer ensures that similar representations are projected to be close to each other while the dissimilar ones are away. Moreover, it enables the prototypes to be sensitive to source-dissimilar representations. Formally, we denote f_e to be the feature extractor before DM layer, and $z = f_e(x)$ is the input of the

DM layer. Given a set of trainable vectors $\mathcal{P} = \{p_j\}_{j=1}^{n_p}$ as learnable prototypes, the DM layer is defined as follows:

$$DM(z) = [S_c(z, p_1), S_c(z, p_2), ..., S_c(z, p_{n_p})], \tag{4}$$

where $S_c(\cdot, \cdot)$ is the cosine similarity, and n_p is the number of the prototypes. An exponential activation function is followed to sharpen the similarity values, thus facilitating the data embedding alignment to the corresponding prototypes in the latent space. Finally, the segmentation prediction and the uncertainty estimation can be obtained through the segmentation head f_{seg} and the uncertainty estimation head f_{unc} respectively:

$$\begin{aligned} \hat{y} &= Softmax(f_{seg} \circ \exp(-z_{sc})) \\ \hat{u} &= Sigmoid(f_{unc} \circ \exp(-z_{sc})) \end{aligned}, \tag{5}$$

where $z_{sc} = DM(f_e(x))$ is the output of DM layer, exp is the activation function, the symbol $\circ$ denotes function composition.

Uncertainty Estimation Training Loss. To optimize the uncertainty estimation head, three losses are leveraged, including prototype dissimilar loss $\mathcal{L}_{Dis}$, entropy maximization loss $\mathcal{L}_{EM}$, and the uncertainty loss $\mathcal{L}_{Unc}$. These losses are optimized with source images to encourage an unbiased uncertainty estimation on the target domain. Among them, the prototype dissimilar loss constrains the prototypes to be dissimilar and orthogonal:

$$\mathcal{L}_{Dis} = -\sum_{j<k} ||p_j - p_k||. \tag{6}$$

Meanwhile, following the entropy maximization trick described in [27], entropy maximization loss enables the input representations of the DM layer to stay close to different prototypes, thus enabling a discriminative latent space. In practice, we utilize an entropy-like loss:

$$\mathcal{L}_{EM} = \sum_{i=1}^{H\times W} z'_i \cdot \log(z'_i), \quad z' = Softmax(z_{sc}). \tag{7}$$

To associate the prototype to the uncertainty prediction, f_{unc} is trained to predict the error of f_{seg}, i.e., the value of the CE loss. The error is min-max normalized into the range of $[0, 1]$, and binary cross entropy (BCE) is empirically selected for better optimization. The uncertainty loss is shown as below:

$$\mathcal{L}_{Unc} = BCE(\hat{u}^{\mathcal{S}}, Normalize(\mathcal{L}_{CE}^{\mathcal{S}})), \tag{8}$$

where $\hat{u}^{\mathcal{S}}$ is the predicted uncertainty on source domain.

Uncertainty-aware Self-training Rectification. With the uncertainty estimation head f_{unc}, we are able to predict the epistemic uncertainty on target images to rectify the self-training process. Given the target image $x_m^{\mathcal{T}}$, the reweighting map $w^{\mathcal{T}}$ is calculated by:

$$w^{\mathcal{T}} = Sigmoid((f_{unc} \circ \exp(-z_{sc}^{\mathcal{T}}))/T), \tag{9}$$

where T is the temperature to control the sensitivity of uncertainty estimation in target domain reweighting.

Overall Objective. In summary, both source images and mixed images are used to train the segmentation network. Moreover, we additionally optimize the uncertainty estimation in the source domain and use the predicted uncertainty on the target domain to rectify the self-training process by assigning less weight to high uncertainty areas. The overall objective of our method can be written as:

$$\mathcal{L} = \mathcal{L}_{CE}^{\mathcal{S}} + \mathcal{L}_{WCE}^{\mathcal{T}} + \lambda(\mathcal{L}_{Dis} + \mathcal{L}_{EM} + \mathcal{L}_{Unc}). \tag{10}$$

where λ is a hyperparameter to balance the segmentation and uncertainty estimation.

3.3 Episteme-based Class Sampling

We propose Episteme-based Class Sampling (ECS), which samples images based on the class-specific epistemic uncertainty from source domain. Since we update the sampling probability online, we utilize the exponentially moving average to get a stable measurement of overall uncertainty U_c^t, w.r.t. entropy, for each class:

$$U_c^t = \begin{cases} \alpha U_c^{t-1} + (1-\alpha)AE_c^t, & \text{if } AE_c^t \text{ exists} \\ U_c^{t-1}, & \text{otherwise} \end{cases},$$

where $AE_c^t = \frac{1}{N_c}\sum_{i=1}^{N_c} -\hat{y}_i \log(\hat{y}_i)$ denotes the batch average entropy for class c at step t, N_c is the number of pixels for class c, α is the update momentum.

To measure the irreducible entropy Ψ_c, i.e. aleatoric uncertainty, given the model and source data, we train the model without any adaptation until the entropy no longer decreases. Then, the epistemic uncertainty Φ_c^t is calculated by subtracting the irreducible entropy, i.e., $\Phi_c^t = U_c^t - \Psi_c$. The sampling probability $P(c)$ of a certain class c at step t is defined as a function of its epistemic uncertainty:

$$P(c) = \frac{\Phi_c^t + \epsilon}{\sum_{c'=1}^{C}(\Phi_{c'}^t + \epsilon)}, \tag{11}$$

where ϵ is set to 0.05 by default to avoid too small number. As a result, classes with higher epistemic uncertainty will have a higher probability of being sampled. In each training iteration, a class is randomly sampled from the probability distribution, and an image is then selected from the subset of data containing that class. Our whole framework is illustrated in Figure 4 for better understanding.

4 EXPERIMENTS

4.1 Datasets and Implementation Details

Datasets. We evaluate our method on a popular scenario, which transfers the information from a synthesis domain to a real one. For synthesis domains, we use either GTA5 dataset [33], which contains 24,966 images with resolution of 1914×1052, or the SYNTHIA dataset [35], which consists of 9,400 images with resolution of 1280×720. For real domain, Cityscapes street scene dataset [6] is used which contains 2,975 training and 500 validation images with resolution of 2024×1024. Following previous works, we resize the images to 1024×512 pixels for Cityscapes and to 1280×720 for GTA5.

Implementation Details. To be consistent with other comparison methods, we use the Deeplabv2 [4] framework with a ResNet101 [12] backbone as our image encoder. The DM layer is added after ASPP layers, while the segmentation and uncertainty estimation heads are Linear layers. The output map is up-sampled and operated by a softmax layer to match the size of the inputs. The pre-trained model on ImageNet [7] is applied to initialize the backbone. AdamW [26] is used as the optimizer with a learning rate of 6×10^{-5} for the backbone and 10× larger for the rest. For the optimizer, betas are set to {0.9, 0.999}, and weight decay is set to 0.01. Warmup [15]

Table 1: Comparison results of GTA5 → Cityscapes adaptation in terms of mIoU(%). All methods are based on DeepLabv2 with ResNet-101 for a fair comparison.

GTA5 → Cityscapes																				
Method	Road	SW.	Build	Wall	Fence	Pole	Light	Sign	Veg.	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motor.	Bike	**mIoU**
SourceOnly	27.0	20.6	53.9	20.8	19.4	35.3	40.7	23.0	84.6	30.1	73.5	63.9	31.4	65.7	10.5	26.3	2.1	34.1	21.8	36.0
AdaptSegNet$_{\text{CVPR'18}}$ [38]	86.5	36.0	79.9	23.4	23.3	23.9	35.2	14.8	83.4	33.3	75.6	58.5	27.6	73.7	32.5	35.4	3.9	30.1	28.1	42.4
FDA$_{\text{CVPR'20}}$ [43]	92.5	53.3	82.4	26.5	27.6	36.4	40.6	38.9	82.3	39.8	78.0	62.6	34.4	84.9	34.1	53.1	16.9	27.7	46.4	50.4
SegUncer$_{\text{CVPR'21}}$ [48]	90.4	31.2	85.1	36.9	25.6	37.5	48.8	48.5	85.3	34.8	81.1	64.4	36.8	86.3	34.9	52.2	1.7	29.0	44.6	50.3
MetaCorrect$_{\text{CVPR'21}}$ [11]	92.8	58.1	86.2	39.7	33.1	36.3	42.0	38.6	85.5	37.8	87.6	62.8	31.7	84.8	35.7	50.3	2.0	36.8	48.0	52.1
DACS$_{\text{WACV'21}}$ [37]	89.9	39.7	87.9	30.7	39.5	38.5	46.4	52.8	88.0	44.0	88.8	67.2	35.8	84.5	45.7	50.2	0.0	27.3	34.0	52.2
IAST$_{\text{ECCV'20}}$ [28]	94.1	58.8	85.4	39.7	29.2	25.1	43.1	34.2	84.8	34.6	88.7	62.7	30.3	87.6	42.3	50.3	24.7	35.2	40.2	52.2
ProCA$_{\text{ECCV'22}}$ [17]	91.9	48.4	87.4	41.5	31.8	41.9	47.9	36.7	86.5	42.3	84.7	68.4	43.1	88.1	39.6	48.8	40.6	43.6	56.9	56.3
CorDA$_{\text{ICCV'21}}$ [40]	94.7	63.1	87.6	30.7	40.6	40.2	47.8	51.6	87.6	47.0	**89.7**	66.7	35.9	90.2	48.9	57.5	0.0	39.8	56.0	56.6
ProDA$_{\text{CVPR'21}}$ [45]	87.8	56.0	79.7	**46.3**	44.8	45.6	53.5	53.5	88.6	45.2	82.1	70.7	39.2	88.8	45.5	**59.4**	1.0	48.9	56.4	57.5
DecoupleNet$_{\text{ECCV'22}}$ [21]	87.6	49.3	87.2	42.5	41.6	46.6	57.4	44.0	89.0	43.9	90.6	73.0	43.8	88.1	32.9	53.7	**44.3**	49.8	57.2	59.1
DAP$_{\text{CVPR'22}}$ [16]	94.5	63.1	**89.1**	29.8	**47.5**	**50.4**	56.7	**58.7**	**89.5**	**50.2**	87.0	73.6	38.6	**91.3**	50.2	52.9	0.0	50.2	**63.5**	59.8
Ours	**96.2**	**73.4**	88.7	41.1	34.1	46.9	**57.8**	56.0	89.2	49.6	89.5	**75.2**	**49.6**	90.5	**52.7**	54.8	1.0	**54.5**	62.7	**61.2**

Table 2: Comparison results of SYNTHIA → Cityscapes adaptation in terms of mIoU(%). All methods are based on DeepLabv2 with ResNet-101 for a fair comparison. mIOU* denotes the average of 13 classes (computed without the classes marked with *).

SYNTHIA → Cityscapes																	
Method	Road	SW	Build	Wall*	Fence*	Pole*	Light	Sign	Veg.	Sky	Person	Rider	Car	Bus	Motor	Bike	**mIoU**	**mIoU***
SourceOnly	59.9	24.7	57.7	6.3	0.0	32.5	29.7	15.0	72.8	70.8	59.2	17.7	73.0	23.0	11.6	22.6	36.0	41.4
AdaptSegNet$_{\text{CVPR'18}}$ [38]	84.3	42.7	77.5	-	-	-	4.7	7.0	77.9	82.5	54.3	21.0	72.3	32.2	18.9	32.3	-	46.7
FDA$_{\text{CVPR'20}}$ [43]	79.3	35.0	73.2	-	-	-	19.9	24.0	61.7	82.6	61.4	31.1	83.9	40.8	38.4	51.1	-	52.5
SegUncer$_{\text{CVPR'21}}$ [48]	87.6	41.9	83.1	14.7	1.7	36.2	31.3	19.9	81.6	80.6	63.0	21.8	86.2	40.7	23.6	53.1	47.9	55.0
MetaCorrect$_{\text{CVPR'21}}$ [11]	92.6	52.7	81.3	8.9	2.4	28.1	13.0	7.3	83.5	85.0	60.1	19.7	84.8	37.2	21.5	43.9	45.1	52.5
DACS$_{\text{WACV'21}}$ [37]	80.6	25.1	81.9	21.5	2.9	37.2	22.7	24.0	83.7	**90.8**	67.6	38.3	82.9	38.9	28.5	47.6	48.4	54.8
IAST$_{\text{ECCV'20}}$ [28]	81.9	41.5	83.3	17.7	4.6	32.3	31.0	28.8	83.4	85.0	65.6	30.8	86.5	38.2	33.1	52.7	49.8	57.1
ProCA$_{\text{ECCV'22}}$ [17]	90.5	52.1	84.6	29.2	3.3	40.3	37.4	27.3	86.4	85.9	69.8	28.7	88.7	53.7	14.8	54.8	53.0	59.6
CorDA$_{\text{ICCV'21}}$ [40]	**93.3**	**61.6**	**85.3**	19.6	**5.1**	37.8	36.6	42.8	84.9	90.4	69.7	**47.8**	85.6	38.4	32.6	53.9	55.3	63.3
ProDA$_{\text{CVPR'21}}$ [45]	87.8	45.7	84.6	**37.1**	0.6	44.0	**54.6**	37.0	88.1	84.4	74.2	24.3	88.2	51.1	40.5	45.6	55.5	62.0
DecoupleNet$_{\text{ECCV'22}}$ [21]	77.8	48.6	75.6	32.0	1.9	44.4	52.9	38.5	87.8	88.1	71.1	34.3	88.7	58.8	50.2	**61.4**	57.0	64.1
DAP$_{\text{CVPR'22}}$ [16]	84.2	46.5	82.5	35.1	0.2	**46.7**	53.6	45.7	**89.3**	87.5	**75.7**	34.6	**91.7**	**73.5**	49.4	60.5	**59.8**	64.3
Ours	74.6	34.4	82.7	31.2	1.6	41.5	53.4	**57.5**	86.2	89.3	75.2	45.2	87.2	49.0	**54.2**	59.3	57.7	**65.2**

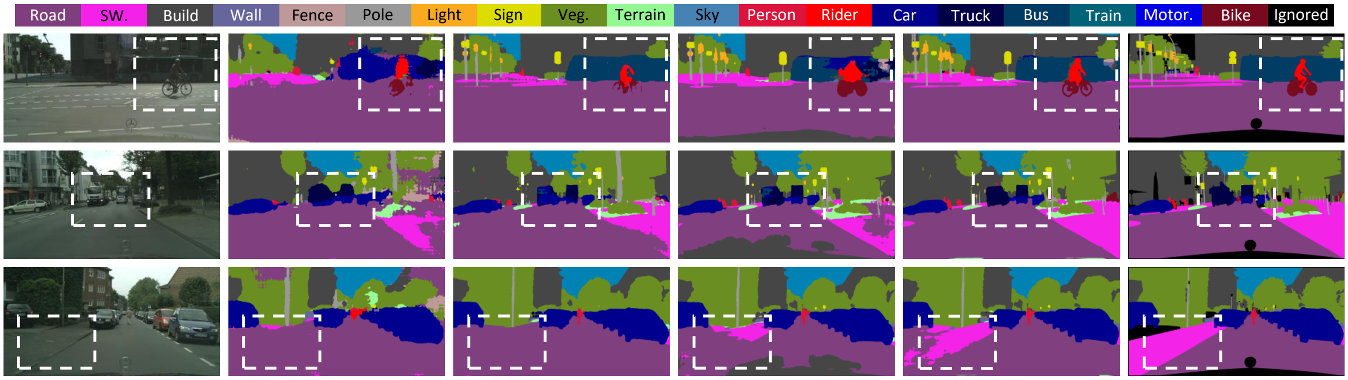
strategy is taken for the learning rate in the first 1,500 iterations. We train the model for 40k iterations on a single NVIDIA RTX 4090 GPU with batch size of 2. Following DACS [37], the same data augmentation strategies are utilized, while the teacher momentum m is set to 0.999. Following the previous works[16, 21, 45], we further distill the trained model to the simCLR initialized student after the training.

4.2 Comparisons with State-of-the-art Methods

We comprehensively compare our proposed method with the recent leading approaches. Among them, AdaptSegNet [38], and FDA [43] employ adversarial learning for domain alignment, while SegUncer [48], MetaCorrect [11], DACS [37], ProCA [17], and CorDA [40] deploy self-training frameworks. The rest of them, including IAST [28], ProDA [45], DecoupleNet [21], and DAP [16] conduct a mixture of adversarial learning and self-training. We also provide the non-adapted results, tagged as SourceOnly.

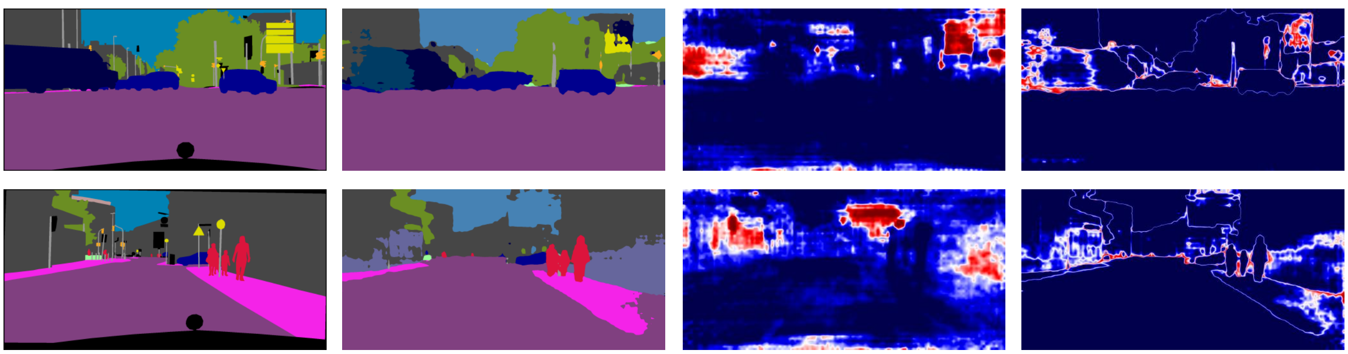
Table 1 illustrates the adaptation results on task GTA5 → Cityscapes. By exploiting the epistemic uncertainty to rectify the self-training process, the proposed method achieves the state-of-the-art mIoU of 61.2%. This yields an improvement of 1.4% compared with the second best method, DAP [16], and is 9% higher than our baseline framework DACS [37]. It is worth noting that, although we inherit the poor performance in class “Train" from DACS, we still outperform the previous works by greatly improving other classes, e.g., “street wall" and “rider".

The comparison on task SYNTHIA → Cityscapes is shown in Table 2. We calculate the mIoU results of 13 categories as well as 16 categories including other three small-scale objectives, i.e., Wall, Fence and Pole. The proposed method achieves 57.7% mIoU of 16 categories and 65.2% mIoU* of 13 categories, demonstrating the effectiveness of the proposed uncertain-aware pseudo label rectification. In particular, compared with the self-training method CorDA [40] that exploits the prior of video to self-supervised depth



(a) Image (b) Source Only (c) DACS (d) ProDA (e) Ours (f) Ground Truth

Figure 5: Qualitative segmentation results on the GTA5 → Cityscapes domain adaptation.



Ground Truth Target Pseudo Label Epistemic Uncertainty Maximum Class Probability

Figure 6: Qualitative results of the discrepancy between the estimated epistemic uncertainty and the maximum class probability. Red regions indicate high uncertainty. It can be observed that the epistemic uncertainty has more overlap between the erroneous prediction in target pseudo label, while maximum class probability often focuses on the edge of the neighboring area of the classes.

estimation, our method improves self-training by its inherent uncertainty without introducing other information. Meanwhile, our method has a simpler pipeline for data-efficient training in contrast to ProDA [45], DecoupleNet [21] and DAP [16], which engage a mixture of self-training and adversarial training.

We present the qualitative segmentation results of our methods in Figure 5. All UDA methods outperform the "Source Only" baseline, and our method achieves even better performance than our UDA baseline DACS [37]. As depicted within the white dashed box in Figure 5, our method exhibits superior performance in regions with high uncertainty, such as blurred foreground/background, unclear object boundaries, or distant and small objects. We attribute this to the incorporation of uncertainty estimation, which enhances the robustness and domain-invariant capabilities of the segmentation model.

4.3 Analytical Experiments

4.3.1 Ablation Study for Main Components. We conduct ablation studies on the three main components in our UDA framework to better demonstrate our contribution, i.e., uncertainty estimation (UE), uncertainty-aware self-training rectifying (USR), and episteme-based class sampling (ECS). Table 3 presents the performance of different variants of our framework in the adaptation

Table 3: Ablation study for each component in the UDA framework. UE: Uncertainty estimation. USR: Uncertainty-aware self-training rectifying. ECS: Episteme-based class sampling.

Methods	UE	USR	ECS	mIoU
baseline	-	-	-	52.2 (+ 0.0)
Variant 1	✓	-	-	52.1 (- 0.1)
Variant 2	✓	✓	-	54.6 (+ 2.4)
Variant 3	-	-	✓	55.1 (+ 2.9)
Variant 4	✓	-	✓	55.0 (+ 2.8)
ours	✓	✓	✓	58.3 (+ 6.1)
ours + simCLR	✓	✓	✓	61.2 (+ 9.0)

Table 4: Ablation study for the distinction maximization layer. The default settings are marked with *.

$\mathcal{L}_{Dis}$	$\mathcal{L}_{EM}$	mIoU	n_p	mIoU	C_p	mIoU
-	-	55.7	16	55.82	64	55.82
-	✓	56.4	32	56.55	128	56.75
✓	-	57.5	64*	58.3	256*	58.3
✓*	✓*	58.3	128	56.7	512	57.6

direction GTA5 → Cityscapes. Comparing Variant 1 with our baseline, replacing the last layer with DM layer cannot benefit the segmentation results but causes a slight degradation in mIoU (about 0.1%). Variant 2 indicates that utilizing uncertainty to reweight the self-training process significantly improves the performance by reducing the negative effect of errors and noise in the pseudo labels (+2.4% mIoU). As shown in Variant 3, solely employing ECS can also improve the baseline by a large margin (+2.9% mIoU), suggesting the importance of the balanced episteme of each class on the source domain. Variant 4 implies that without rectification in self-training, introducing additional uncertainty estimation will not do good to the model performance. In spite of that, it will not harm the model's performance same as in Variant 1. Taken together, our method performs the best, showing that the three components promote mutually and are all indispensable for the superior domain adaptation results. Finally, following the previous works [16, 21, 45], self distillation [45] with simCLR initialized backbone further boosts the performance of our method in the UDA scenarios.

4.3.2 Experiments for Distinction Maximization Layer. To show the necessity of the prototype dissimilar loss $\mathcal{L}_{Dis}$ and entropy maximization loss $\mathcal{L}_{EM}$, we ablate the two losses in the left part of Table 4. Without the constraints of prototypes from $\mathcal{L}_{Dis}$, a large performance degradation can be observed (-1.9% mIoU), which might be caused by inaccurate uncertainty estimation. We conjecture that the dissimilar loss can encourage the prototypes to be distinguishable, improving the source-dissimilar detection. Meanwhile, $\mathcal{L}_{EM}$ contributes an improvement of 0.8% mIoU by additional constraining on the latent space of the DM layer. Without the constraints of the two losses, nearly no gain can be achieved due to the noisy pseudo labels.

In addition, we also study the number of the prototypes n_p and the channels for each prototype C_p. As shown in the right part of Table 4, increasing n_p or C_p can enhance the performance of UDA to a certain degree, but oversize n_p or C_p would not benefit the result. That means too few prototypes and prototype channels would degrade the model by limiting the capability of feature representation, while too much of them shall harm the result of uncertainty estimation as it requires a deterministic representation.

4.3.3 Effectiveness of Uncertainty Estimation. To better illustrate the intuition of uncertainty estimation, we compare previous label refinement methods in Table 5. MCP, also known as aleatoric uncertainty, the most commonly used label refinement method, generates the mIoU of 52.2. Structural Prior Knowledge [49] is not suitable for online self-training but for offline self-training, failing to improve MCP. Prediction Variance Uncertainty [48] is the first method for actively estimating uncertainty in UDA, with a drop in performance without ECS (-1.0) and improvement with ECS (+4.0), suggesting that ECS assists in active uncertainty estimation. Our method focuses on epistemic uncertainty, utilizing the model's abilities to achieve the best results, 54.6 and 58.3 without and with ECS, respectively. Qualitative results compare epistemic uncertainty to maximum class probability in Figure 6, showing the benefit of using epistemic uncertainty to enhance self-training and reduce target pseudo-label errors.

Table 5: Comparison with different pseudo label refinement.

Method	w/o ECS	w ECS
Maximum Class Probability [13]	52.2	54.6
Structural Prior Knowledge [49]	42.6	44.1
Prediction Variance Uncertainty [48]	51.2	56.2
Epistemic Uncertainty	54.6	58.3

5 CONCLUSIONS

This paper investigates epistemic uncertainty in domain adaptive semantic segmentation. Our proposed method quantifies the uncertainty during pseudo-label rectification in self-training, reducing cumulative error and guiding proper adaptation. We also address imbalanced episteme on each class, proposing an episteme-based class sampling strategy that dynamically adjusts sampling probability. Our method is validated with two standard benchmarks, showing its effectiveness both quantitatively and qualitatively.

ACKNOWLEDGMENTS

The work was partially supported by the following: Jiangsu Science and Technology Programme under no.BE2020006-4; Natural Science Foundation of the Jiangsu Higher Education Institutions of China under no.22KJB520039; National Natural Science Foundation of China under no.62206225; Research Development Fund in XJTLU under no.RDF-19-01-21.

REFERENCES

[1] Alexander A Alemi, Ian Fischer, and Joshua V Dillon. 2018. Uncertainty in the variational information bottleneck. *arXiv preprint arXiv:1807.00906* (2018).

[2] Robert B Ash. 2012. *Information theory.* Courier Corporation.

[3] Chao Chen, Zhihang Fu, Zhihong Chen, Sheng Jin, Zhaowei Cheng, Xinyu Jin, and Xian-Sheng Hua. 2020. Homm: Higher-order moment matching for unsupervised domain adaptation. In *Association for the Advancement of Artificial Intelligence*, Vol. 34. 3422–3429.

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

[5] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. 2018. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3339–3348. https://doi.org/10.1109/CVPR.2018.00352

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. 2016. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3213–3223. https://doi.org/10.1109/CVPR.2016.350

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 248–255. https://doi.org/10.1109/CVPR.2009.5206848

[8] Armen Der Kiureghian and Ove Ditlevsen. 2009. Aleatory or epistemic? Does it matter? *Structural safety* 31, 2 (2009), 105–112.

[9] Gianni Franchi, Xuanlong Yu, Andrei Bursuc, Emanuel Aldea, Severine Dubuisson, and David Filliat. 2022. Latent Discriminant deterministic Uncertainty. In *European Conference on Computer Vision*. 243–260. https://doi.org/10.1007/978-3-031-19775-8_15

[10] Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*. 1050–1059.

[11] Xiaoqing Guo, Chen Yang, Baopu Li, and Yixuan Yuan. 2021. Metacorrection: Domain-aware meta loss correction for unsupervised domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3927–3936. https://doi.org/10.1109/CVPR46437.2021.00392

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 770–778. https://doi.org/10.1109/CVPR.2016.90

[13] Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*.

[14] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. 2018. Cycada: Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning*. 1989–1998.

[15] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. 2022. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9924–9935. https://doi.org/10.1109/CVPR52688.2022.00969

[16] Xinyue Huo, Lingxi Xie, Hengtong Hu, Wengang Zhou, Houqiang Li, and Qi Tian. 2022. Domain-agnostic prior for transfer semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7075–7085. https://doi.org/10.1109/CVPR52688.2022.00694

[17] Zhengkai Jiang, Yuxi Li, Ceyuan Yang, Peng Gao, Yabiao Wang, Ying Tai, and Chengjie Wang. 2022. Prototypical contrast adaptation for domain adaptive semantic segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIV*. Springer, 36–54. https://doi.org/10.1007/978-3-031-19830-4_3

[18] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. 2019. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4893–4902. https://doi.org/10.1109/CVPR.2019.00503

[19] Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in Neural Information Processing Systems* 30 (2017).

[20] Durk P Kingma, Tim Salimans, and Max Welling. 2015. Variational dropout and the local reparameterization trick. *Advances in Neural Information Processing Systems* 28 (2015).

[21] Xin Lai, Zhuotao Tian, Xiaogang Xu, Yingcong Chen, Shu Liu, Hengshuang Zhao, Liwei Wang, and Jiaya Jia. 2022. DecoupleNet: Decoupled Network for Domain Adaptive Semantic Segmentation. In *European Conference on Computer Vision*. https://doi.org/10.1007/978-3-031-19827-4_22

[22] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems* 30 (2017).

[23] Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. 2020. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in Neural Information Processing Systems* 33 (2020), 7498–7512.

[24] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. 2018. Conditional adversarial domain adaptation. *Advances in Neural Information Processing Systems* 31 (2018).

[25] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S Yu. 2013. Transfer feature learning with joint distribution adaptation. In *International Conference on Computer Vision*. 2200–2207. https://doi.org/10.1109/ICCV.2013.274

[26] Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

[27] David Macêdo, Tsang Ing Ren, Cleber Zanchettin, Adriano LI Oliveira, and Teresa Ludermir. 2021. Entropic out-of-distribution detection. In *International Joint Conference on Neural Networks*. 1–8. https://doi.org/10.1109/IJCNN52387.2021.9533899

[28] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. 2020. Instance adaptive self-training for unsupervised domain adaptation. In *European Conference on Computer Vision*. 415–430. https://doi.org/10.1007/978-3-030-58574-7_25

[29] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. 2021. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *IEEE/CVF Winter Conference on Applications of Computer Vision*. 1369–1378. https://doi.org/10.1109/WACV48630.2021.00141

[30] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. 2019. Moment matching for multi-source domain adaptation. In *International Conference on Computer Vision*. 1406–1415. https://doi.org/10.1109/ICCV.2019.00149

[31] Janis Postels, Francesco Ferroni, Huseyin Coskun, Nassir Navab, and Federico Tombari. 2019. Sampling-free epistemic uncertainty estimation using approximated variance propagation. In *International Conference on Computer Vision*. 2931–2940. https://doi.org/10.1109/ICCV.2019.00302

[32] Janis Postels, Mattia Segu, Tao Sun, Luc Van Gool, Fisher Yu, and Federico Tombari. 2022. On the practicality of deterministic epistemic uncertainty. In *International Conference on Machine Learning*.

[33] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. 2016. Playing for data: Ground truth from computer games. In *European Conference on Computer Vision*. 102–118. https://doi.org/10.1007/978-3-319-46475-6_7

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

[35] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. 2016. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3234–3243. https://doi.org/10.1109/CVPR.2016.352

[36] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in Neural Information Processing Systems* 30 (2017).

[37] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. 2021. Dacs: Domain adaptation via cross-domain mixed sampling. In *IEEE/CVF Winter Conference on Applications of Computer Vision*. 1379–1389.

[38] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. 2018. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7472–7481. https://doi.org/10.1109/CVPR.2018.00780

[39] Joost van Amersfoort, Lewis Smith, Andrew Jesson, Oscar Key, and Yarin Gal. 2021. On feature collapse and deep kernel learning for single forward pass uncertainty. *arXiv preprint arXiv:2102.11409* (2021).

[40] Qin Wang, Dengxin Dai, Lukas Hoyer, Luc Van Gool, and Olga Fink. 2021. Domain adaptive semantic segmentation with self-supervised depth estimation. In *International Conference on Computer Vision*. 8515–8525. https://doi.org/10.1109/ICCV48922.2021.00840

[41] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. 2017. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2272–2281. https://doi.org/10.1109/CVPR.2017.107

[42] Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. 2021. St3d: Self-training for unsupervised domain adaptation on 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10368–10378. https://doi.org/10.1109/CVPR46437.2021.01023

[43] Yanchao Yang and Stefano Soatto. 2020. Fda: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4085–4095. https://doi.org/10.1109/CVPR42600.2020.00414

[44] Fei Yu, Mo Zhang, Hexin Dong, Sheng Hu, Bin Dong, and Li Zhang. 2021. Dast: Unsupervised domain adaptation in semantic segmentation based on discriminator attention and self-training. In *Association for the Advancement of Artificial Intelligence*, Vol. 35. 10754–10762.

[45] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. 2021. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12414–12424. https://doi.org/10.1109/CVPR46437.2021.01223

[46] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. 2019. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*. 7404–7413.

[47] Yabin Zhang, Hui Tang, Kui Jia, and Mingkui Tan. 2019. Domain-symmetric networks for adversarial domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5031–5040. https://doi.org/10.1109/CVPR.2019.00517

[48] Zhedong Zheng and Yi Yang. 2021. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. In *International Conference on Learning Representations*, Vol. 129. 1106–1120. https://doi.org/10.1007/S11263-020-01395-Y

[49] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. 2018. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *European Conference on Computer Vision*. 289–305. https://doi.org/10.1007/978-3-030-01219-9_18

Received 2 June 2023; revised 18 Aug 2023; accepted 18 Aug 2023