Only a Few Classes Confusing: Pixel-Wise Candidate Labels Disambiguation for Foggy Scene Understanding

Liang Liao[1], Wenyi Chen[2], Zhen Zhang[2], Jing Xiao[2*], Yan Yang[3], Chia-Wen Lin[4], Shin'ichi Satoh[5]
[1]S-lab, School of Computer Science and Engineering, Nanyang Technological University
[2]School of Computer Science, Wuhan University
[3]School of Resource and Environmental Sciences, Wuhan University
[4]Department of Electrical Engineering, National Tsing Hua University
[5]National Institute of Informatics

Abstract

Not all semantics become confusing when deploying a semantic segmentation model for real-world scene understanding under adverse weather. The true semantics of most pixels have a high likelihood of falling in the few top classes ranked by the degree of confidence. In this paper, we replace the one-hot pseudo label with a candidate label set (CLS) that consists of only a few ambiguous classes and exploit its effects on self-training-based unsupervised domain adaptation. Specifically, we cast the problem as a coarse-to-fine process. In the coarse-level process, adaptive CLS selection is proposed to pick a minimal set of confusing candidate labels based on the reliability of label predictions. Then, representation learning and label rectification are iteratively performed to facilitate feature clustering in an embedding space and to disambiguate the confusing semantics. Experimentally, our method outperforms the state-of-the-art methods on three realistic foggy benchmarks.

Introduction

Semantic segmentation assigns pixel-wise semantic labels to a given image, which has found a broad range of applications such as autonomous driving (Liu et al. 2020; Wu et al. 2019; Zhou et al. 2020), medical imaging (Meng, Liao, and Satoh 2022; Taghanaki et al. 2021), and image restoration (Wang et al. 2018; Liao et al. 2020, 2021a,b). Although recent advances in semantic segmentation driven by convolutional neural networks (CNNs) have achieved high accuracy under clear visibility (Cordts et al. 2016; Everingham et al. 2010, 2015; Lin et al. 2014), these models often encounter challenges for real-world scenes, especially for outdoor scenes under "adverse" weather conditions (Erkent and Laugier 2020; Wang et al. 2020; Jiang et al. 2021; Zhong et al. 2022). Even worse, the low visibility in this case makes it difficult for humans to collect or accurately annotate the degraded images. In this paper we focus on unsupervised semantic foggy scene understanding (SFSU) (Dai et al. 2020; Liao et al. 2022; Ma et al. 2022; Sakaridis, Dai, and Van Gool 2018), aiming at adapting models trained on labeled clear images (the source domain) to real-world foggy images (the target domain), involving different scenes without annotations.

Recently, self-training approaches emerge as powerful solutions for transferring knowledge from the source domain to the target domain (Kim and Byun 2020; Mei et al. 2020; Shin et al. 2020; Zhang et al. 2020; Zou et al. 2019b; Xie et al. 2022; Zhu et al. 2022), typically employing the iterative approach between pseudo label prediction and model retraining. Since the discrepancy between the source and target domains leads to noisy predictions, attention has been paid to generating high-quality pseudo labels such as adaptive thresholding for filtering out unreliable predictions (Zou et al. 2018, 2019a), entropy minimization for increasing confidence on unlabeled data (Iqbal, Hafiz, and Ali 2022; Vu et al. 2019; Zheng and Yang 2021), curriculum model adaptation from easy to hard samples (Dai et al. 2020; Lian et al. 2019; Ma et al. 2022; Zhang, David, and Gong 2017), and confident pseudo label diffusion (Liao et al. 2022; Shin et al. 2020). However, two issues related to pseudo labels continue to hinder the performance of domain adaptation: 1) a substantial number of pseudo labels are incorrect, even for some with high confidence; 2) if only pixels with high confidence are utilized, data lying far from the source distribution will never be included in the training stage.

By dissecting the segmentation results predicted in the foggy scenario, we have an interesting observation: although the pseudo label with the highest prediction confidence is not always consistent with the ground-truth (GT), the GT class usually has a high likelihood of falling in the top few classes ranked by the prediction confidence. Taking the *Bus* pixel in Fig. 1 for instance, its predicted probability vector ranked by confidence score is $\{$*Building*(0.742), ***Bus***(0.254), *Road*(0.002), ...$\}$. Based on this observation, we hypothesize that the domain discrepancy will only confuse a GT class with a small set of other classes, but not all of them. Therefore, if we can disambiguate the GT labels from their corresponding confusing labels, the learned model will be able to accurately understand the foggy scene.

To this end, we propose a **Can**didate labels disambiguation-based **D**omain **A**daptation (**CanDA**) framework with the goal of replacing the existing one-hot hard label with a newly introduced candidate label set (CLS). Besides the set of one-hot hard, CLS is intended to include more pseudo labels that their GT labels may not be ranked the first but are within the top ranks so that those samples which lay far from the source

[*]Corresponding author: Jing Xiao (jing@whu.edu.cn)

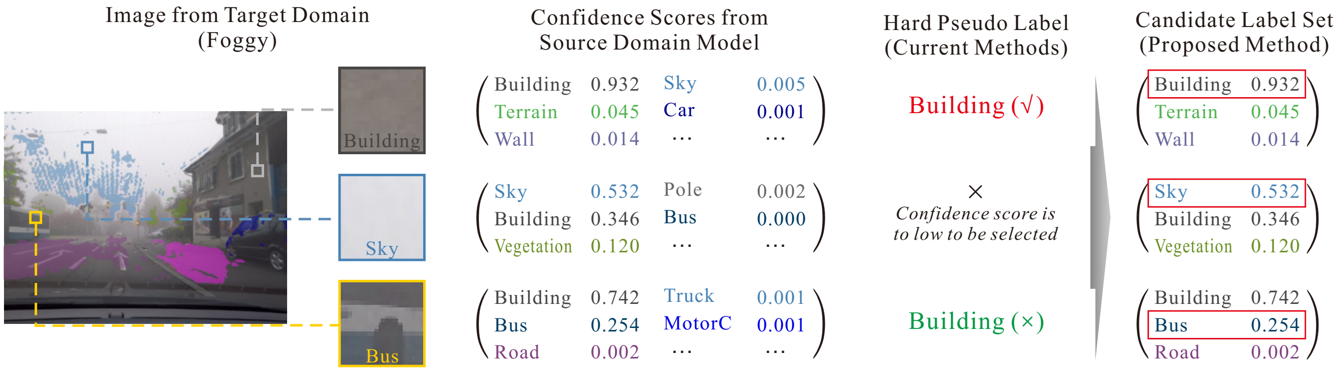


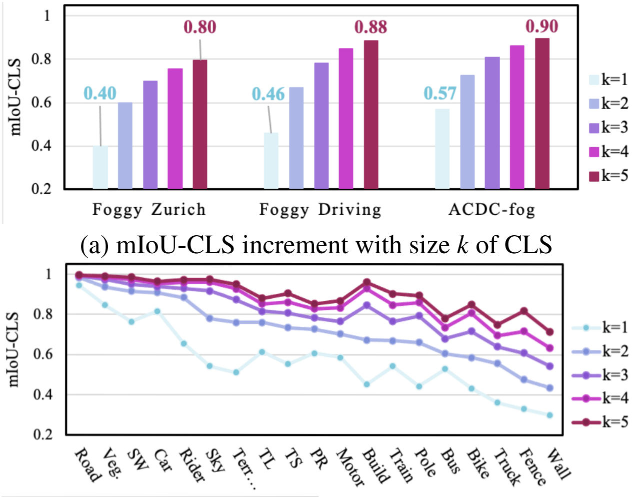
Figure 1: Replacement of hard pseudo-labels in self-training-based SFSU with a candidate label set. The foggy image is overlaid with sparse confident pseudo labels. The middle part shows three samples with their respective sorted confidence scores for all classes. The hard pseudo labels are highly sparse and noisy, but the candidate label set consisting of three candidates is able to cover the ground-truth class and boost it to the top by label disambiguation.

distributions could be included in the adaptive training. Furthermore, to incorporate the GT labels while mitigating the number of confusing labels, we present an adaptive candidate label selection technique based on the reliability of label prediction. Then, we propose a label disambiguation process aiming at bringing the pixel embeddings closer to the prototypes of real semantics, to update the CLS and progressively improve the ranking of the GT label within the set.

We conduct extensive experiments to demonstrate that the proposed label disambiguation progressively enhances the ranking of GT labels in the CLS to improve labeling accuracy, thereby increasing the number of accurate pixels in training, which ultimately contributes to the performance improvement in three realistic foggy benchmarks.

Our main contributions are summarized as follows:

- (**Problem Definition**). Our paper is the first, to the best of our knowledge, to explore candidate pseudo labels for domain-adaptive semantic segmentation. In comparison to the conventional one-hot pseudo-labels, the candidate label set has the potential to find a compact target feature space by involving more training samples even if they are far from the source distribution while simultaneously reducing error information by taking labeling uncertainty into account in the CLS.
- (**Methodology**). To integrate CLS into self-training-based SFSU, we devise a prototypical contrastive learning framework with newly built label set losses and label disambiguation mechanism. By keeping prototypes for all semantics, pixel features are pushed away from the embedding of negative classes excluded from the CLS but are drawn closer to the GT semantics by gradually elevating the GT classes to the highest rank in CLS during training.
- (**Performance**). The proposed **CanDA** substantially outperforms the state-of-the-art for the SFSU tasks on three realistic foggy benchmarks when adapted from **CityScapes** (Cordts et al. 2016), which benefits from the sufficient and categorically balanced training with the CLS setting.



(a) mIoU-CLS increment with size k of CLS

(b) Class-wise mIoU-CLS on ACDC-fog

Figure 2: mIoU-CLS increases as the number k of candidate labels in both scales of datasets and semantic class increases. That is, if we can disambiguate the GT label from the CLS, the mIoU can be significantly improved. mIoU-CLS deems the prediction correct if the GT class falls within the CLS.

Motivation

What makes high-quality pseudo labels for self-training-based domain adaptation? We argue that the labels should not only be highly accurate but also cover most of the target pixels for sufficient and categorically balanced training. To this end, we analyze the impact of domain disparity on the target data using a pre-trained source model (Lin et al. 2020) from a labeled dataset of clear weather, *i.e.*, **CityScapes** (Cordts et al. 2016), and the results are depicted in Fig. 2. It can be observed that using the hard pseudo label (the label with the highest prediction confidence) (Badrinarayanan, Kendall, and Cipolla 2017; Chen et al. 2017; Minaee et al. 2021), achieves an initial mIoU of approximately 0.5. However, after expand-

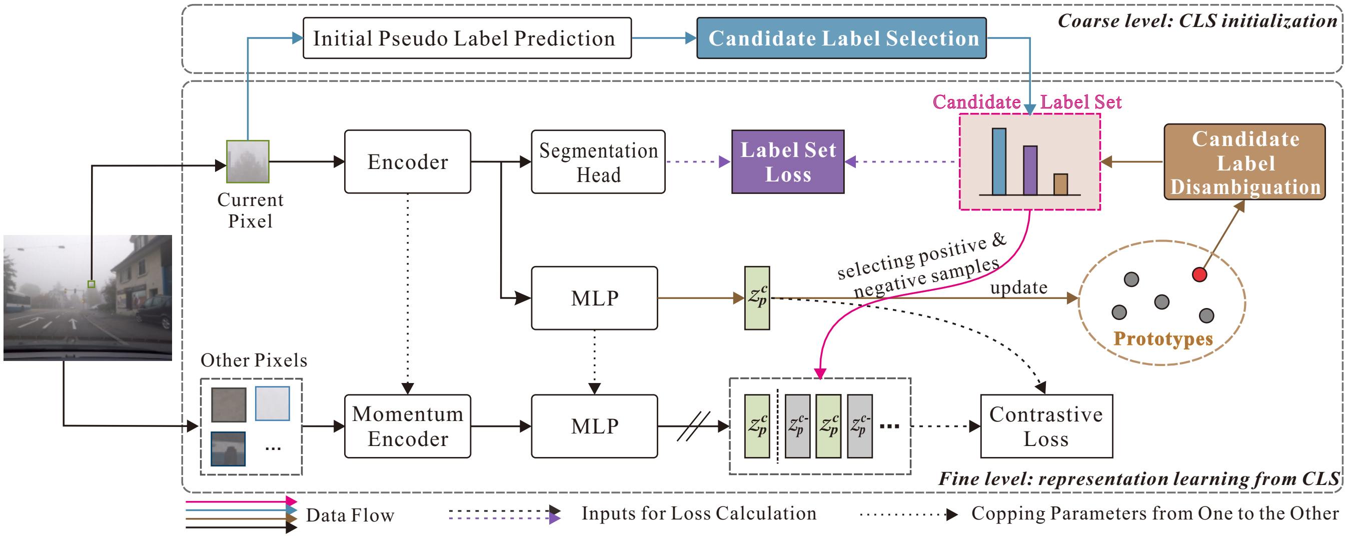

Coarse level: CLS initialization
Initial Pseudo Label Prediction
Candidate Label Selection
Candidate Label Set
Current Pixel
Encoder
Segmentation Head
Label Set Loss
Candidate Label Disambiguation
selecting positive & negative samples
update
MLP
z_p^c
Prototypes
Other Pixels
...
Momentum Encoder
MLP
z_p^c z_p^{c-} z_p^c z_p^{c-} ...
Contrastive Loss
Fine level: representation learning from CLS
Data Flow
Inputs for Loss Calculation
Copping Parameters from One to the Other

Figure 3: Framework of CanDA. The entire learning process is divided into coarse and fine levels. At the coarse level, the CLS of each pixel is initialized. Then, at the fine level, contrastive representation learning is adopted to better cluster the class embedding, whereas the prototype-driven candidate label disambiguation module assists in pushing the GT classes to the top rank. "//" means no back-propagation.

ing the hard pseudo label to a candidate label set (consisting of the k most confident labels), the performance improves significantly under the measurement of ***mIoU-CLS***, which considers the prediction correct if GT class is within the CLS.

This observation motivates us to separate the few most confidently predicted labels from the rest. In particular, the most confident labels for a CLS are the most confusing ones for this pixel, whereas the others can be easily distinguished. On this basis, we propose to decompose the C-class pixel classification problem into two sub-problems: selecting a CLS of size k for each pixel and disambiguating the k confusing labels from the CLS, where $k << C$.

Method

Problem Formulation

In self-training-based SFSU setting, we are given two datasets: a labeled source dataset from clear weather $\mathcal{D}_s = \{(x_s^i, y_s^i)\}_{i=1}^{N_s}$, and an unlabeled target dataset from foggy weather $\mathcal{D}_t = \{x_t^i\}_{i=1}^{N_t}$, where x_s^i and x_t^i denote the images from the source and target domains, N_s and N_t indicate the total number of images in the source and target domains, respectively. $y_s^i(p) \in \{0,1\}^C$ is the one-hot encoded label map corresponding to pixel p in image x_s^i, where C is the total number of classes shared by $\mathcal{D}_s$ and $\mathcal{D}_t$.

Instead of taking a single one-hot pseudo label, we propose a candidate label set (CLS) in the target dataset to facilitate domain adaptation. In particular, each pixel-wise CLS, denoted as $\mathcal{Y}_t^i(p)$, is associated with a vector $\mathbf{y}_t^i(p) \in [0,1]^{k_p}$ representing the probability of each label, where k_p is the number of candidate labels for pixel p in a target image and much less than the total class number C. $\overline{\mathcal{Y}}_t^i(p)$ represents the excluded labels with class number of C-k_p. The entire self-training problem is then converted into two sub-problems: candidate label selection and representation learning with the CLS. The latter sub-problem can be addressed by optimizing this loss function:

$$\mathcal{L}_{seg} = \frac{1}{N_s}\sum_{i=1}^{N_s}\mathcal{L}_{ce}(\Phi(x_s^i), y_s^i) + \alpha_t \frac{1}{N_t}\sum_{i=1}^{N_t}\mathcal{L}_{rl}(\Phi(x_t^i), \mathbf{y}_t^i), \tag{1}$$

where Φ denotes the segmentation model initialized by training on $\mathcal{D}_s$, α_t is a weight that balances the contributions of the two datasets. $\mathcal{L}_{ce}$ is the cross-entropy loss and $\mathcal{L}_{rl}$ denotes our proposed loss functions of representation learning based on CLS. In the remainder, we omit the sample index i when the context is clear.

Framework Overview

The proposed framework is shown in Fig. 3, in which we formulate the self-training-based SFSU as a pixel-wise candidate labels disambiguation problem and solve it with a two-level process.

Coarse-level process. CLS is initialized for each pixel by selecting the top k labels from a sorted label vector in descending order of confidence. With the goal of making CLS cover the GT label while minimizing the labeling uncertainty of CLS, we propose an adaptive CLS generation strategy that adjusts the number of candidate labels according to the reliability of the label predictions, under the assumption that the higher the prediction reliability, the higher the ranking of GT labels in the prediction vector.

Fine-level process. We progressively learn the embedding of the target domain based on the CLSs of pixels and disambiguate GT labels online from CLS. Specifically, we build our framework on the basis of prototypical contrastive learning (He et al. 2020; Khosla et al. 2020) and collaboratively optimize the semantic representation and CLS label ranking.

Contrastive representation learning. This learning process is supervised by optimizing a proposed label set loss for segmentation, and a contrastive loss for clustering embeddings. To accommodate the CLS setting, negative samples for contrastive learning are collected using only the excluded labels in $\overline{\mathcal{Y}}_t$, so that the pixel features are encouraged to stay away from the classes outside of CLS.

Candidate label disambiguation. We disambiguate the CLS and promote the GT label to the first rank, thereby encouraging the clustering of these confusing semantics. In particular, the rectification process estimates class-wise likelihoods according to the relative distance between the pixel embedding and all semantic prototypes, and the candidate label disambiguation module increases the confidence of the label whose prototype is the closest to the pixel. Noting that the prototypes are computed on-the-fly, making the CLS updated incrementally throughout training.

Adaptive Candidate Label Selection

The adaptive candidate label selection aims to select, whenever possible, the minimal labels for the CLS that contains the GT label. Observing that the class confusion issue varies in different contexts, particularly in the foggy scene, where the effect of fog is highly related to the scene depth (Dai et al. 2020), we propose to adaptively assign the number of candidate labels in CLS based on the three categorizations of pixel reliability addressed as follows.

- **Category I - Strongly Reliable (C.I).** As validated by thresholding-based self-training methods (Zou et al. 2018, 2019a), the regions with the highest confidence tend to be reliable and have high recognition accuracy. Thus, we categorize these confident labels as *Category I* with a single label in the CLS.
- **Category II - Reliable (C.II).** Inspired by (Liao et al. 2022) that pixels within a local cluster with a confident pixel tend to have a label co-occurrence, we use the superpixel-based spatial diffusion process to locate the pixels in *Category II* and assign them with soft labels $\hat{\mathbf{y}}_t(p) \in [0,1]^{k_1}$ from the top k_1 highest confidence scores, where $1<k_1<<C$.
- **Category III - Not Very Reliable (C.III).** All pixels not included in the previous categories are grouped into *Category III*. We define the size of CLS as k_0 to balance the size of their CLSs and the expectation that GT label in the CLS, where $k_1 < k_0 << C$.

Correspondingly, the categorization of pixel reliability can be formulated by:

$$\mathbf{m}(p) = \begin{cases} 1 & \text{if } c = \arg\max_c \mathcal{P}_p\left(c \mid x_t, \Phi\right) \\ & \quad \text{and } \mathcal{P}_p\left(c \mid x_t, \Phi\right) > \lambda^c \\ 2 & \text{if } \arg\max_c \mathcal{P}_p\left(c \mid x_t, \Phi\right) \\ & \quad = \arg\max_c \mathcal{P}_q\left(c \mid x_t, \Phi\right) \\ & \quad \text{s.t. } p \text{ and } q \in sp_i \text{ and } \mathbf{m}(q) = 1 \\ 3 & \text{otherwise} \end{cases} \tag{2}$$

where $\mathbf{m}(p)$ and $\mathbf{m}(q)$ denote the category indicators of pixel p and q, respectively, 1, 2 and 3 mean that the pixel belongs to *Category I*, *II* and *III*, respectively. q is the pixel that is in the same super-pixel with p and has been classified as *Category I*. $\mathcal{P}\left(c \mid x_t, \Phi\right)$ is the prediction probability of class c, λ^c denotes the confidence threshold for class c, which is determined by the most confident ρ percentage of the prediction of class c in the entire target set. ρ is typically set to a low value, *e.g.*, 20%, for high accuracy. sp_i denotes the i-th superpixel in a target image.

Contrastive Representation Learning with CLS

The uncertainty of the CLS labeling space posits a unique obstacle to effective representation learning. In this paper, we couple a label set loss for semantic segmentation with a contrastive term to facilitate feature separation of semantically ambiguous embeddings. Notably, based on the proposed candidate labeling setting, some vital prior knowledge is naturally introduced, *e.g.*, GT labels are hidden within positive labels, and conversely, negative labels are highly unlikely.

Label set loss. In general, semantic segmentation tasks employ one-hot cross-entropy loss, but under our CLS settings, it likely penalizes the lower-ranked GT labels excessively. To weaken this penalty, we replace the hard label with a soft probability label when calculating this loss. Additionally, we introduce a ranking loss to balance the significance of all candidate labels in the CLS, emphasizing their ranking relative to other labels while weakening their ranking confusion.

Specifically, the per-sample *soft cross-entropy loss* for the target data is given by:

$$\begin{aligned} \mathcal{L}_{scls} = & \sum_{c=1}^{C} -\hat{\mathbf{y}}_t^c \log(\mathcal{P}(c \mid x_t, \Phi))) \\ & \text{s.t.} \sum_{c \in \mathcal{Y}_t(p)} \hat{\mathbf{y}}_t^c(p) = 1 \text{ and } \hat{\mathbf{y}}_t^c(p) = 0, \forall c \notin \mathcal{Y}_t(p) \end{aligned} \tag{3}$$

A *ranking loss* is formulated to encourage the candidate labels in $\mathcal{Y}_t$ to rank higher than all the excluded labels in $\overline{\mathcal{Y}}_t$:

$$\mathcal{L}_{rank} = 1 - \sum_{p \in x_t} \sum_{c \in \mathcal{Y}_t(p)} \frac{\mathcal{R}^+(c)}{\mathcal{R}(c)}, \tag{4}$$

where $\mathcal{R}^+(c)$ and $\mathcal{R}(c)$ are ranking positions of the c-th class among candidate labels and all labels, respectively. $\mathcal{R}(c)$ is defined using a unit step function $H(\cdot)$ applied on the difference between the probability of class c and the probability of other labels of the same pixel:

$$\mathcal{R}(c) = 1 + \sum_{j \in \mathcal{Y}_t(p), j \neq c} H(h^{cj}) + \sum_{j \in \overline{\mathcal{Y}}_t(p)} H(h^{cj}), \tag{5}$$

where $h^{cj} = -(\mathcal{P}^c - \mathcal{P}^j)$ is positive if $\mathcal{P}^c < \mathcal{P}^j$, and $H(h) = 1$ if $h \geq 0$ and $H(h) = 0$ otherwise. $\mathcal{P}^c$ and $\mathcal{P}^j$ denote the predicted probabilities for classes c and j of pixel p, respectively. $\mathcal{R}^+(c)$ can be defined similarly over $j \in \mathcal{Y}_t(p)$.

Contrastive loss. We leverage the contrastive loss to opti-

Method	Road	SW	Build	Wall	Fence	Pole	TL	TS	Veg.	Terrain	Sky	PR	Rider	Car	Truck	Bus	Motor	Bike	mIoU
SFSU (2018)	64.94	51.25	37.95	28.46	20.87	41.69	$\underline{60.13}$	55.11	34.20	31.22	27.03	3.55	38.04	77.54	0.00	11.08	13.63	4.75	33.41
CMAda (2020)	84.68	$\underline{57.65}$	42.27	27.03	21.28	**47.77**	**61.88**	62.95	62.26	35.73	70.93	$\underline{8.47}$	35.92	85.03	0.00	$\underline{45.32}$	37.65	9.97	44.27
AdSegNet (2018)	21.31	31.53	26.11	14.84	23.45	30.64	48.52	46.75	56.69	22.78	43.52	3.51	20.34	10.86	0.00	4.20	38.49	4.00	24.86
CBST (2018)	**91.64**	57.09	29.92	55.96	31.54	42.32	54.88	60.07	73.35	53.61	52.62	**8.71**	43.06	**87.52**	0.00	16.60	53.90	11.48	45.79
CRST (2019a)	91.16	**57.81**	36.23	54.53	31.19	41.99	51.43	**63.29**	$\underline{75.04}$	54.40	61.21	7.84	40.92	86.89	0.00	25.36	45.01	12.21	46.47
CuDA-Net (2022)	$\underline{91.47}$	51.64	40.07	$\underline{55.99}$	28.37	$\underline{46.38}$	58.22	$\underline{63.07}$	**77.38**	**59.47**	67.90	2.87	**45.74**	86.74	0.00	**54.52**	50.72	3.98	49.14
FIFO (2022)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	48.40
CMDIT (2021)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	41.69
TDo-Dif (2022)	89.52	53.29	**66.69**	**56.65**	$\underline{39.97}$	36.92	53.22	59.20	73.94	$\underline{58.48}$	**90.26**	4.45	33.19	83.79	0.00	42.03	**57.86**	$\underline{18.10}$	$\underline{50.92}$
CanDA (Ours)	90.49	56.79	$\underline{63.43}$	55.31	**44.04**	37.95	50.36	59.85	74.91	58.21	$\underline{88.29}$	4.30	$\underline{45.01}$	$\underline{87.40}$	0.00	44.37	$\underline{57.84}$	**20.75**	**52.18**

Table 1: Quantitative comparison on Foggy Zurich. Notice that we exclude the class *Train* as it is not included in the test set, and we calculate class *Truck* although it cannot be detected by any methods. Numbers in bold and underline represent the best and second-best scores.

Method	Road	SW	Build	Wall	Fence	Pole	TL	TS	Veg.	Terrain	Sky	PR	Rider	Car	Truck	Bus	Train	Motor	Bike	mIoU
SFSU (2018)	90.26	28.83	72.13	25.23	13.41	42.84	$\underline{52.03}$	58.97	64.27	5.78	76.71	57.26	44.02	70.41	13.42	27.73	58.48	19.29	46.48	45.66
CMAda (2020)	91.51	29.24	$\underline{74.77}$	28.37	15.10	$\underline{49.36}$	51.35	59.26	74.76	7.82	$\underline{92.29}$	62.63	$\underline{47.67}$	72.90	19.38	$\underline{32.48}$	52.05	24.62	$\underline{52.81}$	49.39
AdSegNet (2018)	45.82	13.52	43.34	0.63	8.94	25.97	37.57	35.92	54.12	0.53	80.70	30.73	27.08	56.74	0.73	12.58	0.40	11.19	26.47	27.00
CBST (2018)	91.68	31.35	68.63	25.61	15.98	48.14	49.48	60.02	67.85	10.37	82.18	62.22	41.62	73.30	36.96	15.69	31.69	29.90	46.95	46.82
CRST (2019a)	91.82	36.34	70.59	23.93	16.33	46.02	49.66	56.92	70.84	$\underline{12.68}$	86.36	$\underline{64.25}$	42.17	75.07	30.72	13.24	31.32	$\underline{35.06}$	45.70	47.32
CuDA-Net (2022)	90.14	$\underline{45.52}$	71.47	**43.63**	**44.23**	43.83	46.30	52.24	72.63	**36.18**	91.19	59.90	**47.90**	72.04	**48.58**	**40.96**	32.81	33.47	44.09	53.50
FIFO (2022)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	50.70
CMDIT (2021)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	45.35
TDo-Dif (2022)	$\underline{93.03}$	39.26	**76.72**	$\underline{33.35}$	$\underline{18.77}$	48.35	50.17	$\underline{64.41}$	**79.99**	2.32	**92.66**	61.87	46.64	$\underline{78.31}$	44.63	28.22	**70.78**	**41.58**	51.58	$\underline{53.84}$
CanDA (Ours)	**94.05**	**55.94**	72.92	27.84	18.38	**52.08**	**52.33**	**65.78**	$\underline{78.37}$	6.29	90.76	**68.42**	44.94	**78.99**	$\underline{47.29}$	17.63	$\underline{70.24}$	33.87	**53.60**	**54.20**

Table 2: Quantitative comparison on Foggy Driving. Numbers in bold and underline represent the best and second-best scores.

mize the representation during training:

$$\mathcal{L}_{ctrs} = -\frac{1}{C \times P} \sum_{c=1}^{C} \sum_{p=1}^{P} \log \frac{e^{\left\langle z_p^c, z_p^{c^+} \right\rangle}}{e^{\left\langle z_p^c, z_p^{c^+} \right\rangle} + \sum_{n \in \mathcal{N}} e^{\left\langle z_p^c, z_p^{c^-} \right\rangle}}$$
$$\text{s.t. } c, c^+ \in \mathcal{Y}_t(p), c^- \in \overline{\mathcal{Y}}_t(p) \tag{6}$$

where P is the total number of positive anchor pixels and z_p^c denotes the representation of the p-th anchor of class c. Each anchor pixel is followed by a positive sample from the pixel with the same predicted label, while the negative sample set $\mathcal{N}$ is chosen from the corresponding excluded set $\overline{\mathcal{Y}}_t(p)$. They are represented by $z_p^{c^+}$ and $z_p^{c^-}$, respectively. z denotes the output of the representation head, $\langle .,. \rangle$ is the cosine similarity between the features of two distinct pixels.

On-the-fly Disambiguation of CLS

The CLS is rectified by progressively updating the relative ranking of the candidate labels by estimating the class-wise likelihoods, *i.e.*, the feature distances from all class prototypes. The prototypes are iteratively updated with a rectified representation so as to optimize the feature embedding throughout the training process gradually.

Prototype updating. Different from the prototype computation using all predicted labels in (Zhang et al. 2021), we compute the prototypes, denote as $\Psi = \{\psi_1, ..., \psi_c, ..., \psi_C\}$, only adopting the pixel features from *Category I* ($C.I$) of all target images due to their high accuracy. The prototype embedding ψ_c of class c is calculated by:

$$\psi_c = \frac{\sum_{i=1}^{N_t} \sum_{p \in C.I} z_p^i * \mathbb{1}(\mathbf{y}_t^c(p)! = 0)}{\sum_{i=1}^{N_t} \sum_{p \in C.I} \mathbb{1}(\mathbf{y}_t^c(p)! = 0)} \tag{7}$$

where $\mathbb{1}$ is the indicator function. During the progressive updating, we estimate the prototypes as the moving average of the cluster centroids in mini-batches. In each iteration, the prototype is updated as:

$$\psi_c = \beta \psi_c + (1 - \beta) \psi_c' \tag{8}$$

where ψ_c' is the mean feature of class c calculated within the current training batch from the momentum encoder, and β is the momentum coefficient, which is set to 0.9999.

CLS rectification. We propose a simple yet effective method for re-ranking the labels and removing noisy samples in CLS. For each pixel, we update its pseudo label probability $\hat{\mathbf{y}}_t(p)$ by combining the soft pseudo label $\mathbf{y}_t(p)$ with the momentum prototypes Ψ of those classes in the CLS. Concretely, we rectify the soft pseudo labels and weight them progressively by class-wise probabilities, with the update in accordance with the freshly learned knowledge, and the modulation weights ω_p are defined as the softmax over feature distances to the prototypes:

$$\hat{\mathbf{y}}_t^c(p) = \omega_p^c \mathbf{y}_t^c(p) \quad \text{s.t. } c \in \mathcal{Y}_t(p) \tag{9}$$

$$\omega_p^c = \frac{e^{-||z_p^c - \psi_c||}}{\sum_{i \in \mathcal{Y}_t(p)} e^{-||z_p^c - \psi_i||}} \tag{10}$$

where $\mathbf{y}_t(p)$ is initialized by the source model and remains fixed throughout the learning process, thus serving as a boilerplate for the subsequent refinement. ω_p^c approximates the trust confidence of the pixel belonging to the class c. Note that while $\hat{\mathbf{y}}_t(p)$ is updated, it needs to be normalized.

To further reduce the noise caused by these labels which are hard to be re-ranked, we adopt a drop-out operation to discard the labels with low confidence in *Category III*. The drop-out operation can be formulated by:

$$\hat{\mathbf{y}}_t^c(p) = \begin{cases} 0 & \text{if } p \in C.III \text{ and } \max \hat{\mathbf{y}}_t(p) < T \\ \hat{\mathbf{y}}_t^c(p) & \text{otherwise} \end{cases} \tag{11}$$

where T is calculated based on an empirically set confidence threshold for pixels in *Category III*.

Experiments

Datasets and Baselines

We adopt the clear weather dataset **Cityscapes** (Cordts et al. 2016) with fine segmentation labels as the source domain, and validate our method on three real-world foggy datasets, *i.e.*, **Foggy Zurich** (Dai et al. 2020), **Foggy Driving** (Sakaridis, Dai, and Van Gool 2018) and **ACDC-fog** (Sakaridis, Dai, and Van Gool 2021).

We compare our method with the state-of-the-art methods dedicated to domain adaptive foggy scene segmentation. Among them, **SFSU** (Sakaridis, Dai, and Van Gool 2018) is a supervised learning-based approach that employs labeled synthetic data for training. **CMAda** (Dai et al. 2020) retrains the segmentation model with a labeled synthetic source dataset and a noisy pseudo-labeled real foggy weather dataset. **AdSegNet** (Tsai et al. 2018) is a representative adversarial learning strategy for domain adaptive semantic segmentation. **CBST** (Zou et al. 2018) and **CRST** (Zou et al. 2019a) are two representative self-training strategies by selecting a portion of confident pseudo labels. **CuDA-Net** (Ma et al. 2022), **FIFO** (Lee, Taeyoung, and Suha 2022) and **CMDIT** (Vinod et al. 2021) focus on bridging the domain gap between clear images and foggy images to improve the performance of foggy scene segmentation. **TDo-Dif** (Liao et al. 2022) densifies the confident labels by exploiting target domain similarity in both spatial and temporal domains. We use the released models of **SFSU** and **CMAda** and re-train the segmentation models of the three domain adaptive training strategies, *i.e.*, **AdSegNet**, **CBST** and **CRST**, on our dataset. The results of the last four methods are collected from their respective papers. The detailed implementation of our method is presented in the supplementary materials.

Implementation Details

Similar with **CMAda** (Dai et al. 2020) and **TDo-Dif** (Liao et al. 2022), we adopt the RefineNet (Lin et al. 2020) with ResNet-101 as the backbone in all experiments and initialize it with the weights pre-trained by **Cityscapes** (Cordts et al. 2016). We implement our method using the Pytorch toolbox and optimize it using the Adam algorithm with $\beta_1 = 0.5$, $\beta_2 = 0.999$, and a learning rate of 0.0001 following (Sakaridis, Dai, and Van Gool 2018). In all experiments, we use a batch size of 2 and set the self-training round number to 4 and the training epochs in each round to 10.

Similar to **CBST** and **CRST**, we set the hyper-parameter p of the confident portion to 0.2 so as to pick the top 20% of high confidence predictions as pseudo labels. We set the number of superpixels K in each image to 500, the same as in **TDo-Dif**. The number of positive samples of each class used for computing the contrastive loss is set to 20 and the number of negative classes, chosen from classes labels out of the candidate label set, for each positive pixel is set to two, resulting in the number of negative pixels setting to $2 \times 20 = 40$. The number of candidate labels set k_1 and k_0 for *Category II* and *Category III* are set to 3 and 5, respectively. Take $k_1 = 3$ for an example, it means that we adopt the top 3 labels ranked by confidence as the candidate labels and most left labels are excluded. The hyper-parameter T for dropping out the labels with low confidence in *Category III* is set to $T = 0.3$. β is the momentum coefficient, which is set to 0.9999 as in (Zhang et al. 2021).

Components				**mIoU**	**gain**
Initialization				40.02	+0.00
Warm up by **CRST** (2019a)				46.47	+6.45
Contrastive Learning	CLS	CLS Disamb.	Ranking loss	**mIoU**	**gain**
✓	✗	✗	✗	48.27	+1.80
✓	✓	✗	✗	50.41	+2.14
✓	✓	✓	✗	51.62	+1.21
✓	✓	✓	✓	**52.18**	**+0.56**

Table 3: Ablation study on Foggy Zurich dataset

Main Empirical Results

Achieves SOTA results. The comparison results with the baselines on **Foggy Zurich** and **Foggy Driving** are shown in Table 1 and Table 2. For results on **ACDC-fog**, please refer to the **ACDC-fog** benchmark website[1] and our supplementary materials. In general, our proposed **CanDA** with candidate label setting outperforms all baselines on the three datasets. On **Foggy Zurich**, **CanDA** reaches 52.18% mIoU, surpassing both **SFSU** and **CMAda**, which are specifically designed for foggy scene understanding, with significant gains of 18.77% and 7.91%, respectively. This indicates that employing a large number of noisy pseudo labels for model retraining would result in a degradation in performance. Although **CBST** and **CRST** adopt a small proportion of pseudo labels and **TDo-Dif** further diffuses these confident labels to avoid noisy labels for model re-training, our method obtains gains of 6.39%, 5.71%, and 1.26%, respectively. This may be contributed to the fact that by introducing the CLS, more high-quality pixels are involved in model learning. Our method also achieves the best performance when compared to methods that attempt to bridge the gap between the source and target domains, *i.e.,* **CuDA-Net**, **FIFO**, and **CMDIT**. In addition, it is interesting to note that although our method does not achieve the best performance in most individual classes, it does achieve the best average performance, owing to sufficient and categorically balanced training with CLS.

Learns more distinguishable representations. We use t-SNE (van der Maaten and Hinton 2008) to visualize the feature representation of the domain adaptive process of three strategies, *i.e.*, **CMAda** with complete but noisy pseudo labels, **CRST** with confident but sparse pseudo labels and our

[1] https://acdc.vision.ee.ethz.ch/benchmarks

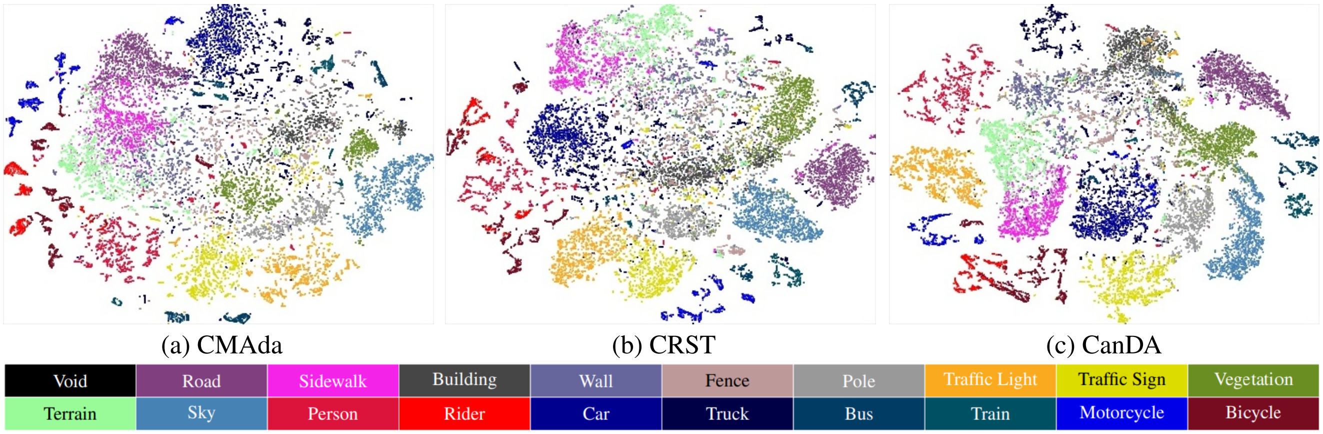

(a) CMAda (b) CRST (c) CanDA

Figure 4: Visualization of embedded features via t-SNE from randomly sampling pixels from ACDC-fog validation set. The features (from left to right) are extracted from the models trained with three types of pseudo labels, *i.e.*, left: entire but noisy pseudo labels (CMAda); middle: confident but sparse pseudo labels (CRST); right: our CLS setting (CanDA). Features are colored according to class labels.

CanDA with CLS settings. As can be observed from Fig. 4, our **CanDA** model exhibits the clearest clustering compared to the other two strategies, revealing the label disambiguation capability of our method. Moreover, our model can well separate the tailed classes, such as *Traffic Light v.s. Traffic Sign* and *Bicycle v.s. Person*, indicating that our method can provide the correct supervision signal for the target data, including the under-represented classes, by introducing CLS.

Model Analysis

Ablation studies. In this section, we conduct a series of ablation studies to validate the contributions of individual components to the foggy scene understanding. The numerical comparisons on **Foggy Zurich** are depicted in Table 3. The non-adapted model **RefineNet** (Lin et al. 2020), which is also the backbone of our **CanDA**, only gives 40.02% mIoU, and increases to 46.47% after warming up with **CRST**. After introducing the CLS and the contrastive learning framework, the performance can have further improvement by 3.94%, showing that involving more pseudo labels with the consideration of label uncertainty can highly boost performance. Moreover, the performance is further improved by the newly designed CLS disambiguation and ranking loss.

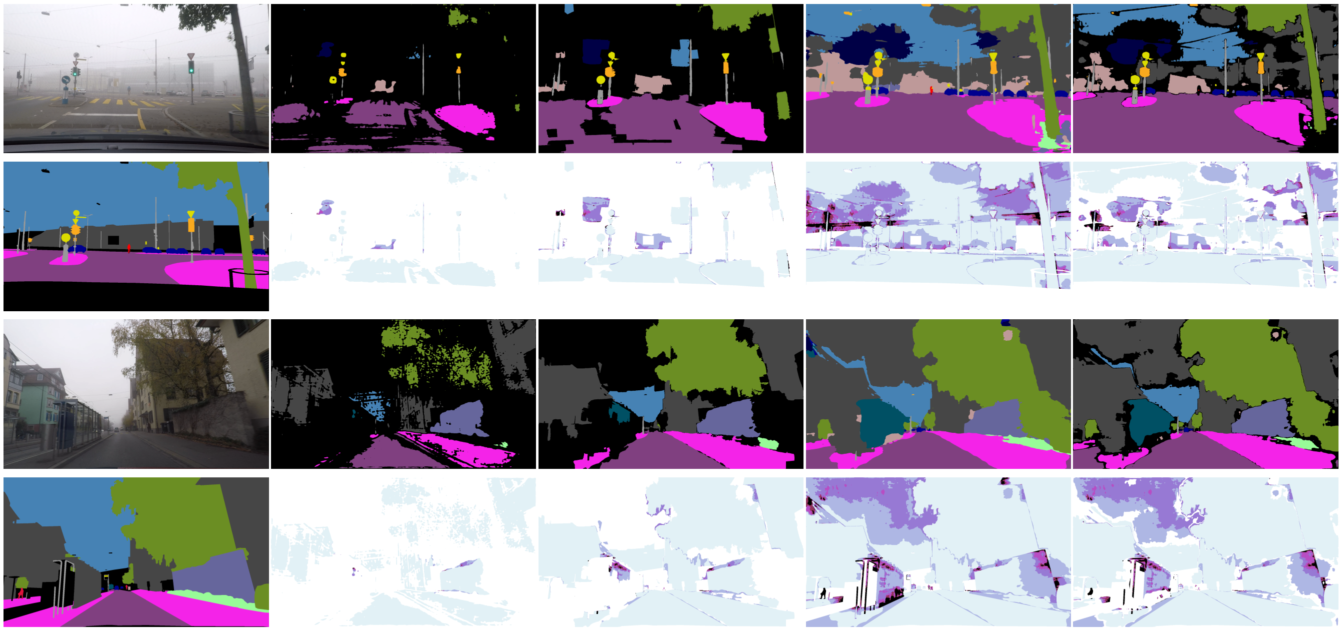
Assessment on pseudo labels settings. We analyze the quantity and quality of pseudo labels by testing four distinct settings: 1) C.I - the highly reliable pseudo labels, 2) C.I+II - both highly reliable and reliable pseudo labels; 3) C.I+II+III - all pseudo labels and 4) Ours - eliminating noisy samples from C.I+II+III. All settings are evaluated using the warm-up model on **CRST** and trained under the same process.

We present the distributions of generated CLS with different settings in Fig. 5. Settings C.I and C.I+II account for a relatively small proportion of the image, while C.III itself accounts for more than half the proportion. It shows that adding labels with lower reliability can well increase the samples in adaptive learning. But to further remove the too noisy samples, our method with CLS rectification can effectively remove those noisy samples (comparing (d) from (c)).

As plotted in Fig. 6, the most reliable C.I pseudo labels achieve the highest accuracy with 83.88% mIoU (the same as mIoU-CLS as the CLS size is 1 for C.I), whereas the model has the lowest mIoU due to its lowest label coverage of only 14%. With spatial diffusion of the pseudo label (C.I+II), the percentage of pseudo labels increases by 20%, along with increments in both mIoU and mIoU-CLS, revealing that the highly improved quality and quantity of pseudo labels greatly boosted performance. But blindly adding all unreliable samples (C.I+II+III) with a great deal of noise degrades the performance even lower than the C.I+II.

Our method is able to select the proper size of CLS and eliminate the noisy labels. Compared with the C.I+II setting, the mIoU of the pseudo labels decreases by about 10% but the value of mIoU-CLS is almost the same, showing that the adaptively selected CLS contains the GT label in the majority of the samples and successfully removes the samples with low-rank GT labels. Consequently, the proposed method achieves the best performance of 52.18% mIoU.

Analysis on label disambiguation of CLS. The success of the CLS largely relies on the label disambiguation of confusion classes. We present the statistics of progressively disambiguated labels, whose GT labels are successfully raised to the first rank in the CLSs (Fig. 7). First, we can observe that in training with CLS, the confusing labels are gradually disambiguated. Pseudo labels belonging to *Category III* are disambiguated greatly more than those belonging to *Category II*, as the latter is far more reliable than the former. In addition, the increasing trends differ between the two categories, *i.e.*, the corrected number in each epoch reaches saturation and even decreases when the training epoch reaches 5 for *Category II*, but the number still increases when the epoch arrives 10 for *Category III*. It motivates us to exploit a mechanism to control the label disambiguation for different categories.



(a) Image /GT (b) C.I (c) C.I+II (d) C.I+II+III (e) Ours

Figure 5: Visualization of various settings of pseudo labels on samples from Foggy Zurich (upper) and ACDC-fog (lower). In each sample, the first row shows the predicted one-hot segmentation labels, while the second row shows the ranking of the GT label in the prediction (the legend is the same as Figure 2). The black color in the second/fourth row of (b)-(e) indicates that the GT label is out of the top five ranked pseudo labels.

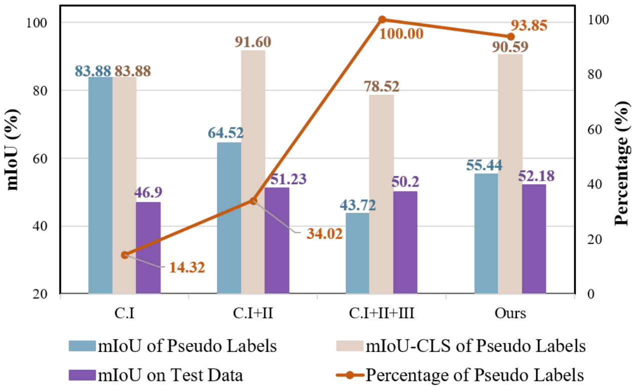


Figure 6: Statistical comparison of the effects of various settings of pseudo labels on the test set of Foggy Zurich.

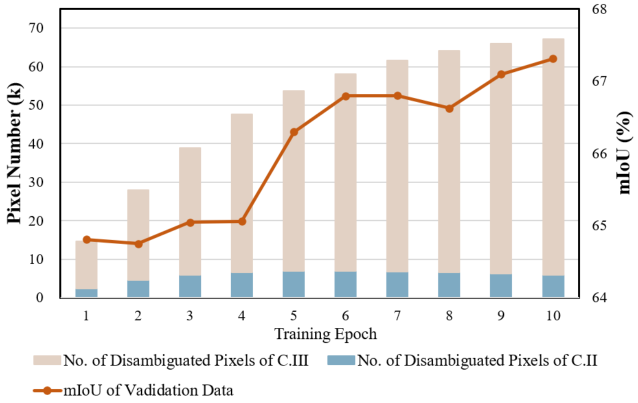


Figure 7: Progressive increment of the disambiguated pixels labels and performance on the validation set of ACDC-fog.

Conclusion

In this paper, we demonstrate the benefits of using candidate label sets in a self-training framework to improve the performance of state-of-the-art domain adaptive semantic segmentation models in real foggy scenes. The overall performance and model analysis shows that the proposed CLS and the prototypical contrastive learning framework can effectively balance the number and quality of pseudo labels, thus allowing more samples away from source distribution to participate in the training, thereby improving the performance of domain adaptation. The limitation of this work lies in the advanced evaluation of CLS rectification, which will further improve the quality of confusing label disambiguation.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62202349, in part by the National Science and Technology Council (NSTC), Taiwan, under grant 111-2634-F-002-023, in part by the Key Research & Development of Hubei Province under Grant 2020BIB006, and in part by the Natural Science Foundation of Hubei Province under Grant 2020CFA001.

References

Badrinarayanan, V.; Kendall, A.; and Cipolla, R. 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(12): 2481–2495.

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4): 834–848.

Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 3213–3223.

Dai, D.; Sakaridis, C.; Hecker, S.; and Van Gool, L. 2020. Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding. *Int. J. Comput. Vis.*, 128(5): 1182–1204.

Erkent, Ö.; and Laugier, C. 2020. Semantic Segmentation With Unsupervised Domain Adaptation Under Varying Weather Conditions for Autonomous Vehicles. *IEEE Robotics Autom. Lett.*, 5(2): 3580–3587.

Everingham, M.; Eslami, S. M. A.; Gool, L. V.; Williams, C. K. I.; Winn, J. M.; and Zisserman, A. 2015. The Pascal Visual Object Classes Challenge: A Retrospective. *Int. J. Comput. Vis.*, 111(1): 98–136.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.*, 88(2): 303–338.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. B. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *CVPR*, 9726–9735.

Iqbal, J.; Hafiz, R.; and Ali, M. 2022. Combining Scale-Invariance and Uncertainty for Self-Supervised Domain Adaptation of Foggy Scenes Segmentation. *arXiv preprint arXiv:2201.02588*.

Jiang, K.; Wang, Z.; Yi, P.; Chen, C.; Wang, Z.; Wang, X.; Jiang, J.; and Lin, C.-W. 2021. Rain-Free and Residue Hand-in-Hand: A Progressive Coupled Network for Real-Time Image Deraining. *IEEE Transactions on Image Processing*, 30: 7404–7418.

Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; and Krishnan, D. 2020. Supervised Contrastive Learning. In *NeurIPS*.

Kim, M.; and Byun, H. 2020. Learning Texture Invariant Representation for Domain Adaptation of Semantic Segmentation. In *CVPR*, 12972–12981.

Lee, S.; Taeyoung, S.; and Suha, K. 2022. FIFO: Learning Fog-invariant Features for Foggy Scene Segmentation. In *CVPR*.

Lian, Q.; Duan, L.; Lv, F.; and Gong, B. 2019. Constructing Self-Motivated Pyramid Curriculums for Cross-Domain Semantic Segmentation: A Non-Adversarial Approach. In *ICCV*, 6757–6766.

Liao, L.; Chen, W.; Xiao, J.; Wang, Z.; Lin, C.-W.; and Satoh, S. 2022. Unsupervised Foggy Scene Understanding via Self Spatial-Temporal Label Diffusion. *IEEE Trans. Image Process.*

Liao, L.; Xiao, J.; Wang, Z.; Lin, C.; and Satoh, S. 2021a. Uncertainty-Aware Semantic Guidance and Estimation for Image Inpainting. *IEEE J. Sel. Top. Signal Process.*, 15(2): 310–323.

Liao, L.; Xiao, J.; Wang, Z.; Lin, C.-W.; and Satoh, S. 2020. Guidance and Evaluation: Semantic-Aware Image Inpainting for Mixed Scenes. In *ECCV*, 683–700.

Liao, L.; Xiao, J.; Wang, Z.; Lin, C.-W.; and Satoh, S. 2021b. Image Inpainting Guided by Coherence Priors of Semantics and Textures. In *CVPR*, 6539–6548.

Lin, G.; Liu, F.; Milan, A.; Shen, C.; and Reid, I. 2020. Refinenet: Multi-path refinement networks for dense prediction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(5): 1228–1242.

Lin, T.; Maire, M.; Belongie, S. J.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common Objects in Context. In *ECCV*, 740–755.

Liu, X.; Han, Y.; Bai, S.; Ge, Y.; Wang, T.; Han, X.; Li, S.; You, J.; and Lu, J. 2020. Importance-aware semantic segmentation in self-driving with discrete wasserstein training. In *AAAI*, 11629–11636.

Ma, X.; Wang, Z.; Zhan, Y.; Zheng, Y.; Wang, Z.; Dai, D.; and Lin, C.-W. 2022. Both Style and Fog Matter: Cumulative Domain Adaptation for Semantic Foggy Scene Understanding. In *CVPR*.

Mei, K.; Zhu, C.; Zou, J.; and Zhang, S. 2020. Instance Adaptive Self-training for Unsupervised Domain Adaptation. In *ECCV*, 415–430.

Meng, Q.; Liao, L.; and Satoh, S. 2022. Weakly-Supervised Learning With Complementary Heatmap for Retinal Disease Detection. *IEEE Transactions on Medical Imaging*, 41(8): 2067–2078.

Minaee, S.; Boykov, Y. Y.; Porikli, F.; Plaza, A. J.; Kehtarnavaz, N.; and Terzopoulos, D. 2021. Image Segmentation Using Deep Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1–1.

Sakaridis, C.; Dai, D.; and Van Gool, L. 2018. Semantic foggy scene understanding with synthetic data. *Int. J. Comput. Vis.*, 126(9): 973–992.

Sakaridis, C.; Dai, D.; and Van Gool, L. 2021. ACDC: The Adverse Conditions Dataset With Correspondences for Semantic Driving Scene Understanding. In *ICCV*, 10765–10775.

Shin, I.; Woo, S.; Pan, F.; and Kweon, I. S. 2020. Two-Phase Pseudo Label Densification for Self-training Based Domain Adaptation. In *ECCV*, 532–548.

Taghanaki, S. A.; Abhishek, K.; Cohen, J. P.; Cohen-Adad, J.; and Hamarneh, G. 2021. Deep semantic segmentation of natural and medical images: a review. *Artif. Intell. Rev.*, 54(1): 137–178.

Tsai, Y.; Hung, W.; Schulter, S.; Sohn, K.; Yang, M.; and Chandraker, M. 2018. Learning to Adapt Structured Output Space for Semantic Segmentation. In *CVPR*, 7472–7481.

van der Maaten, L.; and Hinton, G. 2008. Visualizing Data using t-SNE. *J. Mach. Learn. Res.*, 9(86): 2579–2605.

Vinod, V.; Prabhakar, K. R.; Babu, R. V.; and Chakraborty, A. 2021. Multi-Domain Conditional Image Translation: Translating Driving Datasets from Clear-Weather to Adverse Conditions. In *ICCVW*, 1571–1582.

Vu, T.-H.; Jain, H.; Bucher, M.; Cord, M.; and Pérez, P. 2019. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2517–2526.

Wang, X.; Chen, J.; Wang, Z.; Liu, W.; Satoh, S.; Liang, C.; and Lin, C.-W. 2020. When Pedestrian Detection Meets Nighttime Surveillance: A New Benchmark. In *IJCAI*, 509–515.

Wang, X.; Yu, K.; Dong, C.; and Loy, C. C. 2018. Recovering Realistic Texture in Image Super-Resolution by Deep Spatial Feature Transform. In *CVPR*, 606–615.

Wu, J.; Jiao, J.; Yang, Q.; Zha, Z.; and Chen, X. 2019. Ground-Aware Point Cloud Semantic Segmentation for Autonomous Driving. In *ACM MM*, 971–979.

Xie, P.; Xu, X.; Wang, Z.; and Yamasaki, T. 2022. Sampling and Re-Weighting: Towards Diverse Frame Aware Unsupervised Video Person Re-Identification. *IEEE Transactions on Multimedia*, 24: 4250–4261.

Zhang, P.; Zhang, B.; Zhang, T.; Chen, D.; Wang, Y.; and Wen, F. 2021. Prototypical Pseudo Label Denoising and Target Structure Learning for Domain Adaptive Semantic Segmentation. In *CVPR*, 12414–12424.

Zhang, Y.; David, P.; Foroosh, H.; and Gong, B. 2020. A Curriculum Domain Adaptation Approach to the Semantic Segmentation of Urban Scenes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(8): 1823–1841.

Zhang, Y.; David, P.; and Gong, B. 2017. Curriculum Domain Adaptation for Semantic Segmentation of Urban Scenes. In *ICCV*, 2039–2049.

Zheng, Z.; and Yang, Y. 2021. Rectifying Pseudo Label Learning via Uncertainty Estimation for Domain Adaptive Semantic Segmentation. *Int. J. Comput. Vis.*, 129: 1106–1120.

Zhong, X.; Tu, S.; Ma, X.; Jiang, K.; Huang, W.; and Wang, Z. 2022. Rainy WCity: A Real Rainfall Dataset with Diverse Conditions for Semantic Driving Scene Understanding. In *IJCAI*.

Zhou, W.; Berrio, J. S.; Worrall, S.; and Nebot, E. M. 2020. Automated Evaluation of Semantic Segmentation Robustness for Autonomous Driving. *IEEE Trans. Intell. Transp. Syst.*, 21(5): 1951–1963.

Zhu, H.; Yuan, J.; Yang, Z.; Zhong, X.; and Wang, Z. 2022. Fine-Grained Fragment Diffusion for Cross Domain Crowd Counting. In *ACM MM*, 5659–5668.

Zou, Y.; Yu, Z.; Kumar, B. V. K. V.; and Wang, J. 2018. Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-training. In *ECCV*, 297–313.

Zou, Y.; Yu, Z.; Liu, X.; Kumar, B.; and Wang, J. 2019a. Confidence regularized self-training. In *ICCV*, 5982–5991.

Zou, Y.; Yu, Z.; Liu, X.; Kumar, B. V. K. V.; and Wang, J. 2019b. Confidence Regularized Self-Training. In *ICCV*, 5981–5990.