Re-distributing Biased Pseudo Labels for Semi-supervised Semantic Segmentation: A Baseline Investigation

Ruifei He[1,2]*, Jihan Yang[1]*, Xiaojuan Qi[1]†
[1]The University of Hong Kong [2]Zhejiang University
rfhe@zju.edu.cn, {jhyang, xjqi}@eee.hku.hk

Abstract

While self-training has advanced semi-supervised semantic segmentation, it severely suffers from the long-tailed class distribution on real-world semantic segmentation datasets that make the pseudo-labeled data bias toward majority classes. In this paper, we present a simple and yet effective Distribution Alignment and Random Sampling (DARS) method to produce unbiased pseudo labels that match the true class distribution estimated from the labeled data. Besides, we also contribute a progressive data augmentation and labeling strategy to facilitate model training with pseudo-labeled data. Experiments on both Cityscapes and PASCAL VOC 2012 datasets demonstrate the effectiveness of our approach. Albeit simple, our method performs favorably in comparison with state-of-the-art approaches. Code will be available at https: //github.com/CVMI-Lab/DARS.

1. Introduction

Recent years have witnessed the great success of deep convolutional neural network (DCNNs) in semantic segmentation [46, 37, 8, 36, 57]. The success, however, heavily relies on a large number of training data with accurate pixel-level human annotations, which are prohibitively expensive and time-consuming to collect.

Semi-supervised learning (SSL) provides a promising path [51, 49, 29, 52, 47] to ease human annotation burden by using a small amount of labeled data in conjunction with a large amount of unlabeled data to obtain an accurate model. In this regard, self-training, alternating between generating pseudo labels for unlabeled data using model predictions and training the model with pseudo-labeled data, is a classic and effective approach for semi-supervised learning and has obtained state-of-the-art results [17, 5, 60] in semi-supervised semantic segmentation with DCNNs.

Motivations. Despite the encouraging results, most of the previous self-training approaches [60, 59, 31, 56, 47] as-
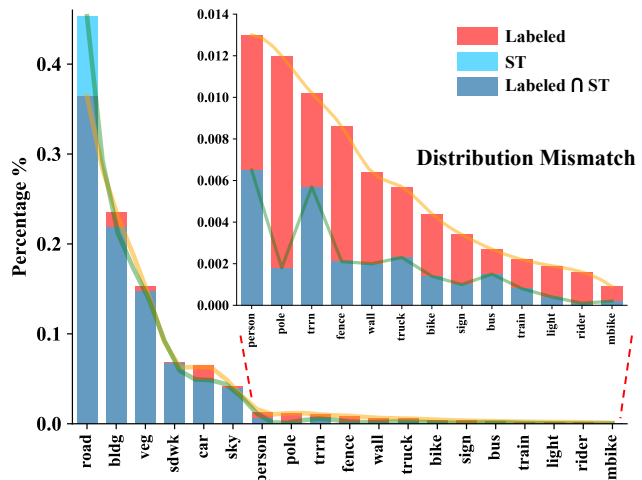

Figure 1. Class distribution mismatch on the Cityscapes dataset [10]. 'Labeled' and 'ST' denote the class distribution of true labels in the labeled set and pseudo labels produced by ST. We line up percentages of each class for better visualization.

sume a class-balanced data distribution and hence adopt a single confidence thresholding **(ST)** scheme to produce the pseudo labels (*i.e.* pixels with prediction confidence score exceeding a pre-defined threshold are pseudo-labeled) to guarantee pseudo label qualities. However, most real-world semantic segmentation datasets [35, 10, 15, 58] have long tail class distributions with few categories occupying the majority of pixels as illustrated in Fig. 1. And, it is well known that DCNNs trained with such long-tailed data distribution will produce predictions biased toward the dominant categories [12]. This can be even more problematic for self-training, since pseudo labels are generated based on these biased model predictions. There exists a severe distribution mismatch between true and pseudo labels, especially for tail categories (see Fig. 1), which will harm self-training.

Recently, only very few works [61, 17] attempt to address the class distribution issue in pseudo labels via sampling the same percentage of pixels for each category based on the predicted results instead of using a single confidence threshold. However, as the class distribution for the predictions has already deviated from the true distribution, the

produced pseudo labels will undoubtedly still suffer from the bias. Here, we argue that this distribution mismatch issue is a largely overlooked problem, hindering further improvements in semi-supervised semantic segmentation.

Our Contributions. In this work, we present a simple yet effective baseline method to re-distribute the biased pseudo labels, aligning their distribution with the true distribution for improving semi-supervised semantic segmentation.

First, we highlight the distribution mismatch issue in semi-supervised semantic segmentation, formulate the task as an optimization problem, and further design a Distribution Alignment and Random Sampling (DARS) method to obtain unbiased pseudo-labeled data, matching the true class distribution. We point out that many pixels share the same confidence value (confidence overlapping) due to the over-confident issue in DCNNs [20], which makes it not viable to achieve distribution matching only by thresholding. Therefore, we propose distribution alignment with class-wise thresholding and random sampling to achieve perfect distribution alignment.

Second, inspired by [53], during the self-training process, we contribute a progressive strategy tailored to semantic segmentation which gradually increases the strength of data augmentation (*e.g.* the range of random scaling) and enlarges the labeling ratio. This strategy prevents the model from being overwhelmed by noisy data from an inaccurate model or strongly augmented examples at the initial stage, and avoids overfitting to high-confident pseudo-labeled easy examples through leveraging diversified augmented data and an increased number of pseudo-labeled data from an improved model.

Third, our proposed method is generic, simple and efficient, which can be seamlessly incorporated into other self-training pipelines for semi-supervised semantic segmentation by adding only a few lines of code. Albeit simple, our approach achieves surprisingly good performance compared with state-of-the-art approaches. For Cityscapes dataset, our model gains a significant amount of performance boost of 8.89% mIoU in the $\frac{1}{8}$ split setting, approaching the fully supervised results. Moreover, we also verify our method on PASCAL VOC 2012, where ours outperforms previous state-of-the-art by 4.49% mIoU.

Finally, we further explore the performance gain in semi-supervised semantic segmentation with the growth of unlabeled data and find that the performance gain gradually saturates in the high-data regime. Further, we analyze the potential bottlenecks for this issue and suggest future directions, hoping to inspire more works in this direction.

2. Related Work

Supervised Semantic Segmentation. The introduction of fully convolutional neural networks (FCN) [46, 37] is a remarkable milestone in semantic segmentation. Most following works build upon it and either take advantage of multi-scale inputs [8, 11, 16, 32, 33, 44], or use feature pyramid spatial pooling [36, 57], or dilated convolutions [6, 7, 9, 30, 50, 54] to improve the model, and encoder-decoder models [1, 9, 30, 45] have also been proved effective. We choose PSPNet [57] in our main experiments for its simplicity and compelling performance, and Deeplabv2 [6] for a fair comparison with previous works.

Semi-Supervised Learning. Recently, noticeable progress has been made in the literature of semi-supervised learning, and successful examples usually fall onto two lines of work. One is consistency training, assuming the model's predictions to be invariant when various perturbations are applied, such as UDA [51], MT [49], VAT [40], Temporal Ensemble [29], Dual Student [27]. The other line of work is self-training [52], closely related to entropy minimization [19] and pseudo labeling [3, 47, 13].

In this work, we mainly focus on self-training, which often utilizes prediction confidence to assign pseudo labels to confident predictions assuming that high confidence corresponds to good accuracy. To do so, a confidence threshold is often used to filter out low confidence unreliable predictions, and the remaining are constructed as pseudo-labels. Berthelot *et al*. [3] average the predictions of different augmented versions of an unlabeled sample, and applies sharpening and mixup to generate pseudo labels. Sohn *et al*.[47] use a confidence threshold to generate pseudo-labels for weakly-augmented versions of unlabeled images and then train a model in the fully supervised way with obtained pseudo labels and stronger data augmentation. Xie *et al*. [52] iteratively generate pseudo labels and train models with them. While these methods achieve impressive results, little attention has been paid to the structure and quality of pseudo-labeled data. Concurrently, [2] and [28] propose to refine pseudo labels using distribution information, but their methods may not be well extended to pixel-level tasks like semantic segmentation due to the over-confident predictions and their high computational complexities for optimization. In contrast, our proposed method is simple yet efficient to handle bias in pseudo-labeling for segmentation.

Semi-Supervised Semantic Segmentation. Inspired by the recent development of SSL methods in the image classification domain, a few works explore semi-supervised learning in semantic segmentation and show promising results. Hung *et al*. [24] and Mittal *et al*. [39] turn to adversarial learning, and a discriminator or a multi-label mean teacher (MLMT) branch is added to select reliable predictions as pseudo labels. Mendel *et al*. [38] extend the GAN-Framework and add a secondary model as a corrector to correct the predictions from the segmentation model.

Consistency based methods are also frequently revisited. French *et al*. [18] build upon [55] and enforce the mixed predictions and predictions of mixed inputs to be consistent with each other. Ouali *et al*. [42] apply perturbations in the
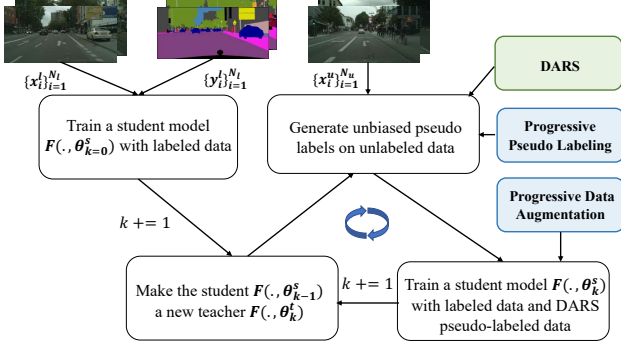
Figure 2. An overview of our semi-supervised semantic segmentation framework. k denotes the round index, where the round $k=0$ is the pre-training round, and rounds $k>0$ are self-training rounds.

feature space and enforce consistency between predictions of different perturbation versions. Ke *et al.* [26] propose a flaw detector and apply dynamic consistency constraint.

Our proposed method is more closely related to self-training or pseudo labeling based methods. Concurrently, [60] and [5] extend the self-training strategy of [52] from image classification to semantic segmentation. Feng *et al.* [17] propose a class-balanced curriculum for semi-supervised semantic segmentation, which can be viewed as the most related work to ours. However, these works do not exploit the bias in pseudo-labeling and either using a single confidence threshold for all classes or confining the number of samples in each class with respect to the biased prediction. In contrast, our method explicitly process the bias in pseudo-labeling and prevent it from harming self-training.

3. Method

3.1. Overview

In SSL, we are given a small set of labeled examples and a large set of unlabeled examples. Let $\mathcal{D}_l = \{(x_i^l, y_i^l)\}_{i=1}^{N_l}$ represent the N_l labeled examples, and $\mathcal{D}_u = \{x_i^u\}_{i=1}^{N_u}$ represent the N_u unlabeled examples, where x_i^l is the i-th labeled image with spatial dimensions $H \times W$, $y_i^l \in \{0, 1\}^{C \times H \times W}$ is its corresponding one-hot encoded pixel-wise label map with C as the number of categories, and x_i^u is the i-th unlabeled image.

Given $\mathcal{D}_l$ and $\mathcal{D}_u$, our goal is to train a semantic segmentation network $F(\theta)$ with parameters θ to achieve satisfactory results on the test set with the same distribution as the training data. An overview of our framework is shown in Fig. 2, which consists of several steps explained as follows.
Step 1: At round $k=0$, we learn a student model $F(., \theta_0^s)$ only on the N_l labeled examples by minimizing

$$\frac{1}{N_l} \sum_{i=1}^{N_l} \mathcal{L}(y_i^l, F(x_i^l, \theta_0^s)), \tag{1}$$

where $\mathcal{L}(.,.)$ denotes the cross entropy loss.
Step 2: At round k, we use the learned student model $F(., \theta_{k-1}^s)$ to be the teacher model $F(., \theta_k^t)$, producing pre-

dictions $\{p_i^u\}_{i=1}^{N_u}$ for the N_u unlabeled examples. Here, $p_i^u \in [0, 1]^{C \times H \times W}$ are the network outputs after the softmax operation. Given $\{p_i^u\}_{i=1}^{N_u}$ and $\{y_i^l\}_{i=1}^{N_l}$, we generate the pseudo labels $\{\tilde{y}_i^u\}_{i=1}^{N_u}$ with our DARS method.
Step 3: At round k, equipped with $\widetilde{\mathcal{D}}_u = \{(x_i^u, \tilde{y}_i^u)\}_{i=1}^{N_u}$, we use both $\mathcal{D}_l, \widetilde{\mathcal{D}}_u$ to train a student model $F(., \theta_k^s)$. The student model resumes from the teacher model and is optimized by minimizing

$$\frac{1}{N_l} \sum_{i=1}^{N_l} \mathcal{L}(y_i^l, F(x_i^l, \theta_k^s)) + \frac{1}{N_u} \sum_{i=1}^{N_u} \mathcal{L}(\tilde{y}_i^u, F(x_i^u, \theta_k^s)). \tag{2}$$

Self-training iterates between **Step 2** and **Step 3** until no more performance gain can be achieved. During iterative training, a progressive augmentation and labeling strategy is designed to further enhance performance.

In the following, we first elaborate on our unbiased pseudo-labeled data generation with our **DARS** method in Sec. 3.2 . Then, we explain our designed strategies in data augmentation and labeling for effective training and pseudo label generation in Sec. 3.3 during self-training.

3.2. Unbiased Pseudo Label Generation

To reduce noise in pseudo-labeled data and enhance their qualities, previous works either adopt a single confidence threshold [60, 59, 31, 56, 47] for all categories or a labeling ratio controlling the percentage of labeled pixels [61, 17]. However, both criteria suffer from the long-tail data distribution which biases pseudo-labeled data toward the dominant categories and causes a severe distribution mismatch between true and pseudo labels, thereby harming effective learning for tail categories.

In this section, we will present a very simple yet effective technique to produce unbiased high-quality pseudo-labeled data whose class distribution matches the true distribution. Here, we use the class distribution of the labeled data as the true distribution, since it should be representative of the real-world data under unbiased random sampling.

Problem Formulation. Before diving into the details, we present our formulation of this problem as follows. Given the labeled data $\{(x_i^l, y_i^l)\}_{i=1}^{N_l}$ and predictions $\{p_i^u\}_{i=1}^{N_u}$ from the teacher model, we aim to obtain the pseudo labels $\{\tilde{y}_i^u\}_{i=1}^{N_u}$ that occupy $\alpha\%$ of all the pixels, where α is the labeling ratio to control the quality of pseudo-labeled pixels. To ensure the distribution matching and encourage pixels with high prediction confidence to have a larger possibility to be pseudo-labeled, we adopt category-specific confidence thresholds $T = \{t_i\}_{i=1}^{C}$ to derive the pseudo-labeled data $\{\tilde{y}_i^u\}_{i=1}^{N_u}$, where pixels with confidence scores not smaller than the corresponding class threshold are pseudo-labeled. T is derived by solving the following optimization problem,

$$\underset{T}{\arg\min} \ D_{\text{KL}}(R(\{y_i^l\}_{i=1}^{N_l}), R(\{\tilde{y}_i^u\}_{i=1}^{N_u})),$$

$$\text{subject to } \tilde{y}_i^u = G(T, p_i^u), P(\{\tilde{y}_i^u\}_{i=1}^{N_u}) = \alpha\%. \tag{3}$$
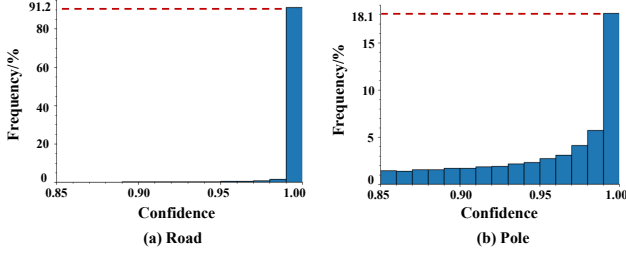
(a) Road **(b) Pole**

Figure 3. Confidence overlapping issue in Cityscapes semantic segmentation task [10] with PSPNet50 backbone [57]. a) Histograms of confidence values for class Road; b) Histograms of confidence values for class Pole.

Here, given the labeled data, $R(.)$ is a frequency counting function which outputs the labeled (pseudo-labeled) pixel percentage r_i^l (or r_i^u) of category i, i.e. $\{r_i^l\}_{i=1}^C = R(\{y_i^l\}_{i=1}^{N_l})$ or $\{r_i^u\}_{i=1}^C = R(\{\tilde{y}_i^u\}_{i=1}^{N_u})$, $D_{KL}(.,.)$ calculates the Kullback-Libeler (KL) divergence measuring the distance of two distributions. Besides, $G(.,.)$ is to generate a valid pseudo label to pixels if the confidence value is not smaller than the threshold of the corresponding category, otherwise, assign an ignore label to the pixel and $P(.)$ returns the percentage of pseudo-labeled pixels. Notably, pixels with ignore label will not contribute to the training.

D_{KL} is minimized when $R(\{y_i^l\}_{i=1}^{N_l})$ and $R(\{\tilde{y}_i^u\}_{i=1}^{N_u})$ are the same, and thus the desirable number of pseudo-labeled pixels for each category j is:

$$n_j^u = \alpha\% \times N_u \times H \times W \times r_j^l. \tag{4}$$

Further, the above optimization problem is readily solvable if the confidence values are distinct with no overlapped values: t_j corresponds to the n_j^u-th prediction value if we sort in descending order the prediction confidence of all pixels with predicted category j.

Confidence Overlapping. However, in semantic segmentation, our observation shows many pixels have similar and indistinguishable confidence values ("confidence overlapping" for short), which is largely due to the fact DCNNs are prone to producing over-confident prediction values [20]. We observe this issue is serious for head categories in semantic segmentation as shown in Fig. 3 (a), the confidence values for the road category are distributed in a very narrow range, and the confidence of 81.3% road pixels is 1.

This renders our previous solution to Eq. (3) not viable, which means the number of pixels after thresholding $\tilde{n}_j^u$ should be larger than n_j^u, and the serious confidence overlapping issue makes this not ignorable, i.e. $\tilde{n}_j^u$ is larger than n_j^u by a significant percentage. The class distribution, therefore, still deviates from the true distribution. This issue is especially severe for the head categories such as road in Cityscapes. The overlapping in confidence values also suggests that the optimal solution to Eq. (3) is not unique.

Though calibration methods, among which temperature scaling [23, 20, 25] has been shown to be the most effective
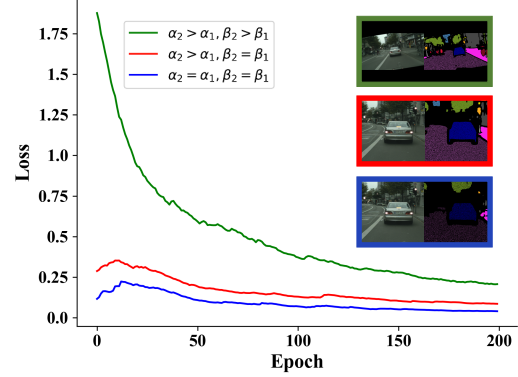
one for DNNs, have been studied to make DCNN's prediction calibrated and distinguishable, and there are also recent works focus on long-tailed recognition such as Focal Loss [34], our ablation studies demonstrate that they fail to offer distribution alignment and are sub-optimal to our problem, while introducing additional cost to validate the parameters.

In the following, we present our simple yet effective method to find one solution to the above problem by alignment and sampling with a few lines of code in Algorithm 1.

Distribution Alignment and Random Sampling (DARS). Firstly, we assume no confidence overlapping and perform distribution alignment with the optimal solution to Eq. (3), as shown in Algorithm 1 (Line $2-3$). For categories that do not suffer from serious confidence overlapping, we can derive the desirable number of pixels n_j^u for category j by ignoring all pixels for category j with confidence lower than threshold t_j. This stage resolves the distribution mismatch problem to some extent especially for tail categories which do not suffer from serious "confidence" overlapping issue, such as pole in Cityscapes, shown in Fig. 3 (b).

Algorithm 1 DARS

Input: Labeled set labels $\{y_i^l\}_{i=1}^{N_l}$, network predictions on the unlabeled set $\{p_i^u\}_{i=1}^{N_u}$ and labeling ratio α.
Output: Pseudo labels $\{\tilde{y}_i^u\}_{i=1}^{N_u}$
1: # Distribution Alignment
2: Calculate $\{n_i^u\}_{i=1}^C$, $\{t_j\}_{j=1}^C$ according to Eq. (3) and Eq. (4)
3: Obtain initial pseudo labels $\{\tilde{y}_i^u\}_{i=1}^{N_u}$ by ignoring low confidence labels in $\text{argmax}\{p_i^u\}_{i=1}^{N_u}$ compared with $\{t_j\}_{j=1}^C$
4: # Random Sampling
5: Count sampling ratio: $\{s_j\}_{j=1}^C \leftarrow n_j^u / \tilde{n}_j^u$
6: Update $\{\tilde{y}_i^u\}_{i=1}^{N_u}$ by randomly ignoring $1 - s_j$ percent pseudo-labeled pixels for each class j

---

Due to confidence overlapping, especially for head categories, the number of pixels derived after distribution alignment might be larger than the desirable number of pixels.

Figure 4. Training loss of pseudo labels in iterative training (k=2), where α_k and β_k denote the labeling ratio and the strength of data augmentation at round k. Right are examples of image crop and pseudo label pair for each case.

Thus, we study how to effectively sample pixels for these categories. Here, we use the random sampling strategy to get the desirable number of pixels as described in Algorithm 1 (Line $5-6$). The reason lies in the following folds: 1) random sampling helps redistribute the centralized high-confident pixels to different regions, effectively enlarging its spatial coverage; and 2) random sampling functions as a way of data augmentation to enhance model performance.

3.3. Progressive Data Augmentation and Labeling

Self-training can benefit from iterative learning. However, during the iterative learning process, if we only update pseudo labels by the latest model while keeping the labeling ratio and data augmentation magnitude the same, the training loss starts from a very low value shown in Fig. 4 (blue curve), which implies that the model has already fit the pseudo-labeled data well, and the data cannot further improve the model performance.

Motivated by these observations, we propose to progressively enlarge the labeling ratio α similar to [61, 17] and increase the strength of data augmentation. Progressively enlarging the labeling ratio helps the model harvest novel data samples without sacrificing the quality of pseudo-labeled data, benefited from an improved model. Though inducing novel data by enlarging the ratio could benefit model training, enlarging the labeling ratio alone still provides quite little new information for iterative process since the loss curve only raises a little at the start as shown in Fig. 4 (red curve), and the model still easily fits the pseudo-labeled data which are typical high-confident easy samples.

Hence, we propose an orthogonal strategy to introduce new data samples for iterative training through progressively increasing the magnitude of data augmentation. Stronger data augmentations could bring unseen cases for the model and turn easy samples into challenging ones while not affecting the quality of pseudo labels, and thus providing new information for model updates.

For the semantic segmentation task, random scaling is the most useful data augmentation strategy. Here, we also focus on strengthening the random scaling effect. At the initial stage, we use weak data augmentations to prevent the model from being influenced by challenging hard examples from augmentation as the model still struggles with easy examples at that time. Then, we increase the range of scales at different self-training stages. Given the range of random scales $[s_{\min}, s_{\max}]$, the upper bound will be increased by β_{max} and the lower bound will be decreased by β_{min} with the new random scale range in $[(1 - \beta_{min})s_{\min}, (1 + \beta_{max})s_{\max}]$. As shown in Fig. 4 (green curve), after increasing the magnitude, the loss for pseudo-label data starts from a relatively high point and gradually converges to zero, which suggests further model updates and improvements.

4. Experiments

4.1. Experimental Setup

Datasets. To evaluate our method, we conduct the main experiments and ablation studies on the Cityscapes dataset [10], which contains 5K fine annotated images and is divided into 2975, 500 and 1525 three image sets for training, validation and test respectively. 19 urban-scene semantic classes are defined in Cityscapes for semantic segmentation. Similar to previous standards [24, 39, 18, 41, 17, 38] in semi-supervised semantic segmentation, we randomly sample $1/8$ and $1/4$ training images to construct the labeled set, and the remaining training images consist of the unlabeled set. To further explore the effectiveness of the proposed method, we also conduct experiments on the PASCAL VOC 2012 dataset (VOC12) [14], which provides 20 semantic classes and 1 background class. The VOC12 dataset consists of 1464 training, 1449 validation, and 1456 test images. Following previous common practice [42, 24] for semi-supervised settings, we use the official 1464 training images as labeled data and the 9k augmented set [21] as unlabeled data.

Comparison Methods. We denote the model trained with only the labeled set as **Baseline**, and with both the labeled set and ground truth labels of the unlabeled set as **Oracle**.

Backbone. It is noteworthy that most previous works employ the Deeplabv2 [24, 39, 18, 17, 38] framework. Nevertheless, we argue that exploring semi-supervised approaches based on a strong baseline could further illustrate their effectiveness and is more practical for real-world scenarios. Hence, we use PSPNet [57] with ResNet-50 [22] as our backbone segmentation network for the main experiments. Our reported Oracle is similar to the results reported by the paper [57]. Moreover, we also provide the results of our method with Deeplabv2 backbone for a fair comparison.

Note that we employ PSPNet instead of the top ones on leaderboards, because the SOTA methods often contain heavy engineering and parameter tunning, accompanying large computation costs, and PSPNet is our best trade-off between reproducibility, performance and costs.

Implementation Details. We implement our method using the PyTorch [43] framework and set the batch size to 16. In self-training, a batch of 16 images is composed of 8 labeled images and 8 unlabeled images, and an epoch is defined as training once on all unlabeled images as previous standard [17]. The number of epochs in each round is 200 for Cityscapes and 50 for VOC12. During training, we employ the SGD [4] with an initial learning rate of 0.01, momentum 0.9 and weight decay 0.0001 respectively. Also, we use a polynomial learning annealing procedure [6] to schedule the learning rate. For data augmentation, we use random scaling, random horizontal flipping, random rota-

Method	Split	mIoU (%)			
		Baseline	Result	Oracle	Gain
Hung *et al*. [24]	1/8	55.5	58.8	67.7	3.3
	1/4	59.9	62.3		2.4
Mittal *et al*. [39]	1/8	56.2	59.3	65.8	3.1
	1/4	60.2	61.9		1.7
CutMix [18]	1/8	55.25±0.66	60.34±1.24	67.53±0.35	5.09
	1/4	60.57±1.13	63.87±0.71		3.30
DST-CBC [17]	1/8	56.7	60.5	66.9	3.8
	1/4	61.1	64.4		3.3
Mendel *et al*. [38]	1/8	55.96±0.86	60.26±0.84	66.9	4.3
	1/4	60.54±0.85	63.77±0.65		3.23
DARS (crop 361)	1/8	60.75±0.35	69.64±0.01	73.80±0.34	**8.89**
	1/4	66.54±0.48	71.30±0.08		4.76
DARS (crop 713)	1/8	65.54±0.34	**72.78±0.17**	76.60±0.67	7.24
	1/4	69.22±0.01	**74.32±0.12**		**5.10**

Table 1. Comparison with the state-of-the-arts on Cityscapes val set. DARS uses PSPNet50 backbone.

tion, and random Gaussian blur. Due to the high computation costs of large crop size, for Cityscapes, we only take large crop size (*i.e.* 713×713) for comparison with the state-of-the-art methods, while a small crop size (*i.e.* 361×361) in our ablation studies for efficient training and evaluation. For VOC12, we take crops of 321×321 as previous standard [42]. To be noted, we empirically find 2 self-training rounds are enough for our implementation, and all results with iterative training (IT) undergo 2 self-training rounds. All of our results are derived by running the experiments on the same setting for **three** times.

4.2. Comparison with Previous Work

All results of DARS reported in this section contain iterative training, and are tested only by a single scale.

4.2.1 Cityscapes

PSPNet50. We construct experiments to compare the proposed DARS on PSPNet50 backbone with several state-of-the-art semi-supervised semantic segmentation approaches. We report the result of each approach with 1/8 and 1/4 labeled proportions on Cityscapes. As shown in Table 1, it is noteworthy that the performance gaps between baseline and oracle of previous works and ours are pretty close (*e.g.* for 1/8 split results, the gaps of all methods lie around 12% mIoU). Our method, though based on a stronger model, obtains more performance gains across all labeled splits. Remarkably, with only 1/8 labeled data, our method outperforms our baseline 8.89% in terms of mIoU and is only 4.16% apart from the fully supervised oracle model. In addition, with a larger crop size (*i.e.* 713), our performance is further boosted to achieve 74.32% mIoU with 1/4 labeled data on Cityscapes, only 2.28% lower than the fully supervised oracle.

Deeplabv2. Here, we also follow the setting of previous works [24, 39, 18, 17, 38] and report the experimental results with Deeplabv2 backbone for a fair comparison. We study the setting of 1/8 split of Cityscapes, which is the most challenging setting. As shown in Table 2, equipped with the

same backbone, DARS still outperforms previous state-of-the-art methods by around 4% mIoU, and is 8% higher than the baseline and only 2.7% to the oracle (66.9%).

Method	mIoU
Deeplabv2 [6]	56.2
Hung *et al*. [24]	57.1
Mittal *et al*. [39]	59.3
CutMix [18]	60.34
DST-CBC [17]	60.5
Mendel *et al*. [38]	60.26
DARS	**64.20**

Table 2. Comparison with the state-of-the-arts with DeepLabv2 backbone in 1/8 split setting for Cityscapes.

4.2.2 VOC12

We further verify the effectiveness of our method by comparing with state-of-the-arts on VOC12 in Table 3. Compared with the most recent CCT method also with PSPNet50 backbone, our DARS significantly outperforms it (with 4.49% mIoU) as well as other previous methods.

Method	Backbone	mIoU
GANSeg ([48])	VGG16	64.10
AdvSemSeg ([24])	DeepLabv2-101	68.40
CCT ([42])	PSPNet50	69.40
DARS	PSPNet50	**73.89**

Table 3. Comparison with the state-of-the-arts on VOC12 val set.

4.3. Ablation Studies

For all of our ablation studies, we conduct experiments in the setting of 1/8 split on Cityscapes dataset, with crop size 361×361 and PSPNet50 backbone.

4.3.1 Ablation Study for Pseudo-labeling Process

Comparison of different pseudo-labeling methods. Since our goal is to re-distribute the biased pseudo labels towards the distribution of labeled data in iterative self-training, here, we compare our DARS method with the following different pseudo-labeling methods:

- ST: the single confidence thresholding method like [52, 60], regraded as the self-training baseline method;
- CBST: the class balanced confidence thresholding method [17, 61] based on prediction results, known as the SOTA method in self-training;
- DA: Our proposed distribution aligning method without following random sampling to address confidence overlapping;
- TS: Temperature scaling, incorporating with DA, CBST or ST method to facilitate distribution alignment by calibrating model predictions as mentioned in Sec. 3.2.

In the upper part of Table 4, we compare our DARS with ST, ST+TS, CBST, CBST+TS, DA+TS in a single self-training round. DARS achieves 68.01% in terms of mIoU, outperforming the single thresholding ST and class-balanced CBST by 2.31% and 1.72%, respectively.

Method	road	sidewalk	building	wall	fence	pole	light	sign	veg	terrain	sky	person	rider	car	truck	bus	train	mbike	bike	Tail mIoU	mIoU	Gain
Baseline	96.7	75.4	88.2	35.2	35.0	45.8	50.3	63.0	89.6	53.9	92.3	72.1	46.7	90.0	38.4	47.9	33.0	33.7	67.0	47.2	60.75±0.35	0.0
ST	**97.4**	**78.3**	89.5	43.4	38.1	47.6	55.7	68.1	**90.9**	56.6	**93.3**	75.1	51.4	**91.7**	49.0	67.9	38.0	46.5	**69.9**	54.0	65.70±0.40	4.95
ST + TS	**97.3**	77.7	89.5	43.2	39.1	48.3	57.8	68.7	**90.8**	56.3	**93.4**	74.9	50.7	91.4	48.5	67.9	40.2	46.3	**69.8**	54.3	65.89±0.30	5.14
CBST	97.1	77.6	89.5	42.4	44.9	50.2	**58.9**	**69.8**	90.6	**57.1**	93.1	**75.2**	52.6	91.5	49.2	68.4	35.5	46.3	69.6	55.0	66.29±0.05	5.54
CBST + TS	96.8	77.1	89.4	43.7	44.2	50.1	58.6	68.9	90.4	55.5	92.7	75.0	**52.6**	91.3	48.8	66.9	42.7	**51.1**	69.7	55.5	66.61±0.20	5.86
DA + TS	97.2	**77.9**	**89.7**	**44.8**	**45.6**	50.7	59.2	69.1	90.6	56.1	93.0	**75.1**	52.9	91.6	**54.8**	**69.4**	**43.1**	48.3	69.6	56.7	**67.31±0.12**	6.56
DARS	97.1	77.7	**89.8**	**50.2**	**46.3**	**50.8**	58.6	**69.5**	90.7	**57.4**	92.8	75.0	52.6	**91.8**	**57.6**	**70.3**	**44.3**	**49.9**	69.7	**57.9**	**68.01±0.12**	**7.26**
ST + IT	97.5	78.8	89.6	43.4	38.5	47.2	55.1	69.4	90.9	56.1	93.3	75.1	51.4	91.9	49.5	67.5	47.3	52.3	70.4	55.2	66.59±0.14	5.84
CBST + IT	**97.8**	**79.2**	**90.4**	44.6	**48.1**	**51.7**	59.8	70.3	**91.1**	58.1	**93.5**	74.9	52.7	**92.6**	53.1	70.5	24.2	53.6	**70.4**	56.1	67.20±0.38	6.45
DARS + IT	97.2	78.5	90.1	**49.3**	47.7	50.9	**59.9**	70.1	90.8	**59.6**	92.9	**75.2**	**54.4**	92.5	**67.7**	**73.0**	**48.7**	**54.7**	69.9	**60.6**	**69.64±0.01**	**8.89**

Table 4. Ablation study for different pseudo-labeling methods. The upper part reports results in a single self-training round (k=1, labeling ratio α =20%), and the lower part reports results with iterative training (IT). The tail classes are highlighted in blue. We make the top-2 results bold for the upper part, and top-1 bold for the lower part. Tail mIoU shows the mean IoU of tail classes.

Equipped with temperature scaling (TS), ST+TS, CBST + TS obtain 0.19%, 0.32% improvements, while our DARS is still 0.7% superior to DA + TS strategy. This result validates TS is effective but sub-optimal in alleviating confidence overlapping and aligning pseudo label distribution.

Especially, our DARS achieves the top-2 best performance on 10 out of all 13 tail classes. Besides, we design to calculate mIoU on tail class (denoted as Tail mIoU), where our DARS further outperforms CBST 2.9%. These experimental results demonstrate the superiority of our method on tail classes by aligning pseudo labels distribution which prevents the model from collapsing into head classes, *e.g.* our DARS outperforms others on truck by over 8.4% mIoU.

In the lower part of Table 4, we further compare the proposed DARS with other pseudo-labeling methods after combining iterative training. With iterative training, DARS achieves more performance gain compared with ST and CBST, and outperforms CBST+IT by 2.44% mIoU, leading to a significant performance boost of 8.89% mIoU, which indicates that DARS helps resolve the bias in pseudo-labeling that may hurt iterative self-training.

Ablation study for distribution matching. While we have shown that combining Temperature scaling with previous pseudo-labeling methods lead to inferior performance compared with DARS, here, we directly compare the distribution mismatch (D_{KL}) between true labels from the labeled set and pseudo labels generated by different off-the-shelf techniques, including calibration methods like Temperature Scaling, Matrix Platting Scaling, Histogram Binning and approaches aim at long-tailed recognition like Focal Loss [34]. All calibration parameters are optimized using the cross-entropy loss over the validation set as in [20] (T=1.27 for TS in our experiments), and hyper-parameters for focal loss adopt the advice from the paper (γ=2).

As shown in Table 5, some techniques such as TS, Matrix Platting Scaling, and Focal Loss may help to alleviate the distribution mismatch in naive self-training (ST), but fail to eliminate it, whereas our DARS is much simpler and en-

ables perfect distribution alignment.

Method	D_{KL}
ST	0.0491
ST + TS	0.0357
ST + Matrix Platting Scaling	0.0340
ST + Histogram Binning	0.4188
ST + Focal Loss	0.0396
DARS	**0.0005**

Table 5. KL divergence between the distribution of true labels and pseudo labels generated by different methods.

4.3.2 Ablation Study for the Progressive Strategy

We explore the effectiveness of enlarging labeling ratio α and data augmentation magnitude (*i.e.* β_{min} and β_{max}) in iterative self-training, as discussed in Section 3.3.

Round	α (%)	β_{min}	β_{max}	mIoU
1	20	0	0	68.01±0.12
1	20	0.2	0.5	(-0.47) 67.54±0.26
2	20	0	0	(+0.26) 68.27±0.12
2	50	0	0	(+0.92) 68.93±0.16
2	50	0.2	0.5	(+1.63) 69.64±0.01

Table 6. Results for different labeling ratio α and data augmentation setups at self-training round 0,1 at split 1/8. β_{min}, β_{max} are parameters for data augmentation magnitude defined in 3.3.

As shown in Table 6, directly using the model obtained by round k=1 to generate new pseudo labels and train round k=2 can only obtain 0.26% mIoU improvements (*i.e.* $\beta_{min} = \beta_{max} = 0$). However, with larger labeled ratio $\alpha = 50\%$, our round k=2 trained model delivers 0.92% performance gains compared to round k=1 results. Furthermore, equipped our round k=2 self-training procedure with larger labeled ratio $\alpha = 50\%$ as well as stronger data augmentation magnitude $\beta_{min} = 0.2, \beta_{max} = 0.5$, our method obtains 1.63% gains. These results provide strong evidence for the necessity to introduce novel yet hard examples for iterative self-training and echo our demonstrations in Section 3.3. To validate whether we just need stronger data augmentation on each round, we show that directly applying stronger data augmentation on round k=1 could even cause performance degradation.

4.4. Additional Results and Analysis

Held-out test results. For all the reported results above, we only provide experimental results on the validation sets, following the standard in semi-supervised semantic segmentation for a fair comparison. However, to verify the generalization ability of our method and show that we do not heavily tune hyper-parameters, we also provide the results on held-out test sets. As shown in Table 7, we obtain similar gains on held-out test sets compared with validation sets.

Method	cs (crop 361)	cs (crop 713)	voc (crop 321)
baseline	59.73	64.70	66.87
DARS	(+8.10) 67.83	(+5.50) 70.20	(+4.97) 71.84

Table 7. Held-out test set: Cityscapes (cs) 1/8 split setting; VOC12 (voc) 1.4k labeled + 9k augmented unlabeled setting.

Improving fully-supervised models with extra unlabeled data. In the above experiments, we have shown the effectiveness of DARS in the low-data regime. Here, we further explore how much performance boost DARS can bring to fully-supervised models (trained with all 3K fine annotated training examples in Cityscapes, with crop size 713×713).

Concretely, we utilize the given 20K coarse annotated images in Cityscapes. However, we ignore the original coarse labels and instead generate pseudo labels for them by DARS. To simulate the real-world scenarios, we study the impact of the number of pseudo labels, denoted as N_p. We conduct experiments with $N_p = 3K, 6K, 9K$, where we keep the training iterations to be the same.

As shown in Table 8, more unlabeled data increases the performance till saturation. However, the performance gain in the high-data regime is relatively small compared with the low-data regime and seems to meet a bootleneck.

N_p	0	3k	6k	9k
mIoU	76.60	78.43	78.64	78.70

Table 8. Improving fully-supervised models with various number of extra unlabeled data.

Analysis for potential bottlenecks in high-data regimes. Here, we try to analyze the bottlenecks in the high-data regime for semi-supervised semantic segmentation. Without loss of generalization, we take Cityscapes as an example to explore. While previous works usually group classes into head and tail classes, here we present a four-group categorization for classes in semantic segmentation from the perspective of object size and appearing frequency: (1) Classes that frequently appear (frequency>50%) with large sizes (size>10% of the crop area), usually the head classes, *e.g.* road, sky; (2) Classes with high appearing frequency but small sizes, *e.g.* pole, traffic light/sign; (3) Classes that rarely appear but with large sizes, *e.g.* truck, bus; (4) Classes that both rarely occur and with small sizes, *e.g.* motorcycle.

To better understand why only limited performance boosts could be achieved in the high-data regime, we plot the performance gain of the four groups in both low-data (*i.e.* 1/8, 1/4 split) and high-data regimes in Fig 5 (a). Since
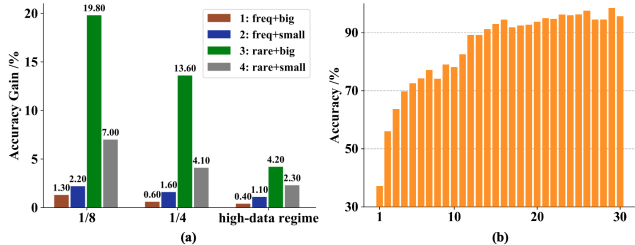

Figure 5. (a) Performance gains of four groups in both low-data (*i.e.* 1/8, 1/4 split) and high-data regimes. (b) Accuracy as a function of size on Cityscapes val set. For crop size 713×713=500k pixels, we use 3 different bin sizes (0.5k, 12k, 40k respectively) and each bin size for 10 bins.

group 1 classes are all head classes, it is natural that we do not harvest much boost on them. However, for the rest three groups that are all tail classes, only groups with rare occurrence attribute (*i.e.* 3, 4) achieve noticeable gain, while group 2 achieves similar limited gains as 1. Therefore, we conclude that self-training mainly contributes to classes with rare occurrence. As a result, in the high-data regime, the occurrence of all classes increases and classes in group 3 could gradually evolve to group 1 with the growth of data, which explains the decreased performance gain in group 3 from low to high-data regime. Hence, one potential bottleneck is the gradually saturated performance for originally rare classes as training data increases.

In order to boost performance, we would expect classes in group 2, 4 (both with small sizes) also turn into group 1, which inspires us that another bottleneck comes from the network's systematic error towards small size objects. As shown in Fig 5 (b), we test the baseline model's accuracy of different object size ranges. The accuracy increases as the object size grows. The network's systematic error for small objects leads to low pseudo label quality for them, and further limits self-training performance on small objects. Hence, we suggest that the second bottleneck for high-data regimes is the network's systematic error towards different object sizes. A promising direction is to develop size agnostic architecture for semantic segmentation, which we leave as future work.

5. Conclusion

We have presented a simple and yet effective DARS method to calibrate the bias in pseudo-labeling, together with a progressive data augmentation and labeling strategy for iterative self-training. Experiments on Cityscapes and VOC12 demonstrate that our simple method can outperform existing sophisticated designed approaches. We hope our formulation, encouraging results, and analysis for bottlenecks could inspire more research efforts in this direction.

Acknowledgement

References

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. 2

[2] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019. 2

[3] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5049–5059, 2019. 2

[4] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010. 5

[5] Liang-Chieh Chen, Raphael Gontijo Lopes, Bowen Cheng, Maxwell D Collins, Ekin D Cubuk, Barret Zoph, Hartwig Adam, and Jonathon Shlens. Semi-supervised learning in video sequences for urban scene segmentation. *arXiv preprint arXiv:2005.10266*, 2020. 1, 3

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 2, 5, 6

[7] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2

[8] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016. 1, 2

[9] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 2

[10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 1, 4, 5

[11] Jifeng Dai, Kaiming He, and Jian Sun. Convolutional feature masking for joint object and stuff segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3992–4000, 2015. 2

[12] Qi Dong, Shaogang Gong, and Xiatian Zhu. Imbalanced deep learning by minority class incremental rectification. *IEEE transactions on pattern analysis and machine intelligence*, 41(6):1367–1381, 2018. 1

[13] WeiWang Dong-DongChen and Zhi-HuaZhou WeiGao. Tri-net for semi-supervised deep learning. In *Proceedings of twenty-seventh international joint conference on artificial intelligence*, pages 2014–2020, 2018. 2

[14] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015. 5

[15] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 1

[16] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1915–1929, 2012. 2

[17] Zhengyang Feng, Qianyu Zhou, Guangliang Cheng, Xin Tan, Jianping Shi, and Lizhuang Ma. Semi-supervised semantic segmentation via dynamic self-training and class-balanced curriculum. *arXiv preprint arXiv:2004.08514*, 2020. 1, 3, 5, 6

[18] Geoff French, Timo Aila, Samuli Laine, Michal Mackiewicz, and Graham Finlayson. Semi-supervised semantic segmentation needs strong, high-dimensional perturbations. *arXiv preprint arXiv:1906.01916*, 2019. 2, 5, 6

[19] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, pages 529–536, 2005. 2

[20] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. *arXiv preprint arXiv:1706.04599*, 2017. 2, 4, 7

[21] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 International Conference on Computer Vision*, pages 991–998. IEEE, 2011. 5

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[23] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 4

[24] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. *arXiv preprint arXiv:1802.07934*, 2018. 2, 5, 6

[25] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957. 4

[26] Zhanghan Ke, Di Qiu, Kaican Li, Qiong Yan, and Rynson WH Lau. Guided collaborative training for pixel-wise semi-supervised learning. *arXiv preprint arXiv:2008.05258*, 2020. 3

[27] Zhanghan Ke, Daoye Wang, Qiong Yan, Jimmy Ren, and Rynson WH Lau. Dual student: Breaking the limits of the teacher in semi-supervised learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6728–6736, 2019. 2

[28] Jaehyung Kim, Youngbum Hur, Sejun Park, Eunho Yang, Sung Ju Hwang, and Jinwoo Shin. Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning. *arXiv preprint arXiv:2007.08844*, 2020. 2

[29] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016. 1, 2

[30] Hanchao Li, Pengfei Xiong, Jie An, and Lingxue Wang. Pyramid attention network for semantic segmentation. *arXiv preprint arXiv:1805.10180*, 2018. 2

[31] Qing Lian, Fengmao Lv, Lixin Duan, and Boqing Gong. Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6758–6767, 2019. 1, 3

[32] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1925–1934, 2017. 2

[33] Guosheng Lin, Chunhua Shen, Anton Van Den Hengel, and Ian Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3194–3203, 2016. 2

[34] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 4, 7

[35] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1

[36] Wei Liu, Andrew Rabinovich, and Alexander C Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015. 1, 2

[37] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1, 2

[38] Robert Mendel, Luis Antonio de Souza, David Rauber, João Paulo Papa, and Christoph Palm. Semi-supervised segmentation based on error-correcting supervision. In *European Conference on Computer Vision*, pages 141–157. Springer, 2020. 2, 5, 6

[39] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high-and low-level consistency. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 2, 5, 6

[40] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018. 2

[41] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. *arXiv preprint arXiv:2007.07936*, 2020. 5

[42] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12674–12684, 2020. 2, 5, 6

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019. 5

[44] Pedro Pinheiro and Ronan Collobert. Recurrent convolutional neural networks for scene labeling. In *International conference on machine learning*, pages 82–90, 2014. 2

[45] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[46] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013. 1, 2

[47] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020. 1, 2, 3

[48] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In *Proceedings of the IEEE international conference on computer vision*, pages 5688–5696, 2017. 6

[49] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204, 2017. 1, 2

[50] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell. Understanding convolution for semantic segmentation. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 1451–1460. IEEE, 2018. 2

[51] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*, 2019. 1, 2

[52] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020. 1, 2, 3, 6

[53] Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. St3d: Self-training for unsupervised domain adaptation on 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10368–10378, 2021. 2

[54] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 2

[55] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 2

[56] Yiheng Zhang, Zhaofan Qiu, Ting Yao, Chong-Wah Ngo, Dong Liu, and Tao Mei. Transferring and regularizing prediction for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9621–9630, 2020. 1, 3

[57] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 1, 2, 4, 5

[58] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019. 1

[59] Yi Zhu, Zhongyue Zhang, Chongruo Wu, Zhi Zhang, Tong He, Hang Zhang, R Manmatha, Mu Li, and Alexander Smola. Improving semantic segmentation via self-training. *arXiv preprint arXiv:2004.14960*, 2020. 1, 3

[60] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D Cubuk, and Quoc V Le. Rethinking pre-training and self-training. *arXiv preprint arXiv:2006.06882*, 2020. 1, 3, 6

[61] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pages 289–305, 2018. 1, 3, 5, 6