This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Towards the Uncharted: Density-Descending Feature Perturbation for Semi-supervised Semantic Segmentation

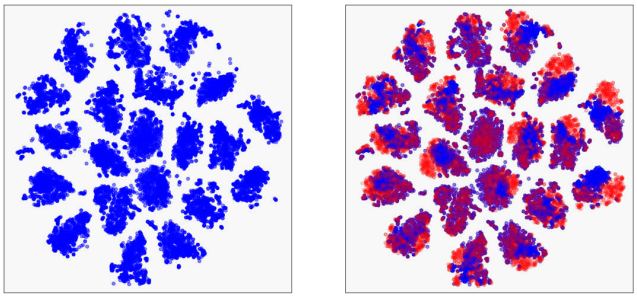
Xiaoyang Wang[1,2,3] Huihui Bai[4] Limin Yu[1] Yao Zhao[4] Jimin Xiao[1*]
[1]XJTLU [2]University of Liverpool [3]Metavisioncn [4]Beijing Jiaotong University

Abstract

Semi-supervised semantic segmentation allows model to mine effective supervision from unlabeled data to complement label-guided training. Recent research has primarily focused on consistency regularization techniques, exploring perturbation-invariant training at both the image and feature levels. In this work, we proposed a novel feature-level consistency learning framework named Density-Descending Feature Perturbation (DDFP). Inspired by the low-density separation assumption in semi-supervised learning, our key insight is that feature density can shed a light on the most promising direction for the segmentation classifier to explore, which is the regions with lower density. We propose to shift features with confident predictions towards lower-density regions by perturbation injection. The perturbed features are then supervised by the predictions on the original features, thereby compelling the classifier to explore less dense regions to effectively regularize the decision boundary. Central to our method is the estimation of feature density. To this end, we introduce a lightweight density estimator based on normalizing flow, allowing for efficient capture of the feature density distribution in an online manner. By extracting gradients from the density estimator, we can determine the direction towards less dense regions for each feature. The proposed DDFP outperforms other designs on feature-level perturbations and shows state of the art performances on both Pascal VOC and Cityscapes dataset under various partition protocols. The project is available at https://github.com/Gavinwxy/DDFP.

1. Introduction

Semantic segmentation is a fundamental task in visual understanding, involving pixel-level classification on input images [7, 24, 34, 55]. However, segmentation models often exhibit a strong dependence on large amounts of annotated data. Unfortunately, collecting such training data can be both time-consuming and laborious, thereby impeding the practical application of segmentation models. To address the challenge, semi-supervised semantic segmentation is drawing growing attention recently [42]. Such learning paradigm aims to enhance label-efficiency by utilizing a limit amount of labeled data alongside massive unlabeled data. The key lies in mining effective training signal from unlabeled data to allow better model generalization.



(a) Original features (b) Perturbed features

Figure 1. t-SNE visualization of per-pixel features from Pascal VOC 2012 dataset [13]. (a) Features extracted from encoder. (b) Features after the proposed DDFP strategy (shown in red). The perturbed features significantly deviate from high density centers and move towards low density regions within and out of clusters.

Recent research in semi-supervised semantic segmentation has witnessed a transition from primal adversarial learning approaches [21, 35, 42] to self-training methods [16, 19, 50, 52]. Recent studies focus on consistency regularization frameworks, which aim to enforce prediction agreement across diverse views of unlabelled images [15]. Notably, alternative data views can also be created at feature level, which is explored by a line of works [33, 37, 51]. Among these methods, uniformly sampled noise, random channel dropout, and perturbations adversarial to predictions are applied to image features. Subsequently, predictions on perturbed features are supervised by those derived from the original ones, enabling feature-level consistency learning. These methods have demonstrated their efficacy. Nonetheless, previous design of feature-level perturbations

*Corresponding author.

seems to be designed for general purpose but not tailored for the context of semi-supervised learning.

The low-density separation assumption in semi-supervised learning states that decision boundaries should ideally reside in low-density regions within the feature space [4]. While previous efforts implicitly move towards this objective, the question is whether a more direct and effective approach can be devised. In an effort to address this question, we propose a novel feature perturbation strategy called Density-Descending Feature Perturbation (DDFP) for semi-supervised semantic segmentation. We assume that density information in feature space can shed a light on the direction to improve decision boundary. With density information, features with reliable label guidance in self-training as shown in Fig. 1 (a) can be perturbed toward regions of lower density as in Fig. 1 (b), while still be supervised by its original semantic. Hence, the decision boundary will be forced to explore less dense regions in feature space to prevent classifier overfitting easy patterns.

The crux in our method is the acquisition of feature density distribution. Normalizing flow [12, 27], designed for generative modelling, is a perfect fit for this task. Hence, we propose a density estimator based on a normalizing flow to learn and predict feature density in real time upon the training of segmentation model. The estimator constructs bijective mappings transforming a predefined base distribution into the target feature probability density, with the mappings optimized by likelihood maximization. Inspired by previous work [22], we initialize the base distribution as a Gaussian mixture model, where pair-wise links are built between each Gaussian component and semantic category, enabling more fine-grained optimization and density description. Once the density information is obtained, the density-descending direction on features can be obtained from the density objective as the gradient over original features. Hence, density-descending features can be created with such perturbation injected, which are then leveraged in consistency regularization framework. The density estimator solely act as an observer upon the main segmentation network, online tracking the feature density but not directly contribute to the training of the main model. In inference, the estimator is discarded thereby avoiding any computational overhead. The knowledge learned on the feature distribution indirectly benefits the segmentation classifier, providing effective hints for its optimization.

To verify the effectiveness of the proposed method, we evaluate our method on mainstream benchmarks Pascal VOC 2012 [13] and Cityscapes [10] dataset under different data partition settings, where our method achieves state-of-the-art performance. The contributions of our method can be summarised as following:

- Inspired by the low-density separation assumption, we propose to utilize density information in feature space and design a novel density-descending feature-level perturbations for consistency regularization framework.
- We propose to leverage a normalizing-flow-based density estimator to online capture feature density through likelihood maximization training, from which density-descending directions can be obtained.
- The proposed feature-level consistency regularization achieves competitive performance on mainstream benchmark for semi-supervised semantic segmentation.

2. Related Works

Semi-supervised Learning. Semi-supervised learning (SSL) aims to mine effective supervision from unlabeled data. A fundamental technique is self-training [38, 39, 48, 60], which generates pseudo labels for unlabeled data based on the knowledge from labeled samples, followed by re-training on the combined data to improve model generalization. Recent research focuses on consistency regularization [6, 23, 29–31, 40, 43], where models are benefited from perturbation-invariant training on unlabeled samples. Among these approaches, MixMatch [3] introduces label-guessing by averaging predictions on multiple augmented versions of unlabeled data. FixMatch [41] employs a weak-to-strong consistency strategy where pseudo labels are generated from weakly augmented samples and used to supervise strongly augmented counterparts. Subsequent works have proposed sophisticated pseudo label filtering strategies. Among them, FlexMatch [53] takes into account the varying learning difficulties among categories and designs class-specific thresholds. FreeMatch [46] proposes an adaptive threshold that adjusts based on the model's training status, while SoftMatch [5] introduces a truncated Gaussian function as confidence threshold for unlabeled samples.

Semi-supervised Semantic Segmentation. Research in semi-supervised semantic segmentation has been influenced by advancements in SSL techniques. Notably, self-training and co-training methods [9, 14, 47, 50] have demonstrated success by extracting pseudo labels from either a single model or multiple models. For instance, ST++ [50] progressively and selectively generates pseudo labels to ensure high-quality re-training. CPS [9] utilizes two differently initialized models and exchanges pseudo labels between them to facilitate cross-supervision, while CCVC [47] follows a similar framework but incorporates discrepancy loss to enhance model diversity. Another research line explores the utilization of contrastive learning [1, 28, 32, 44, 45, 54, 58, 59]. Among these methods, ReCo [32] performs contrastive learning on hard negative samples to regularize the feature space, and U2PL [45] extracts negative samples from unreliable predictions to contrast against positive samples.

Consistency regularization has also shown progress in semi-supervised semantic segmentation [15, 20, 33, 37, 51, 56, 57]. French et al. [15] adapts CutMix and CutOut tech-

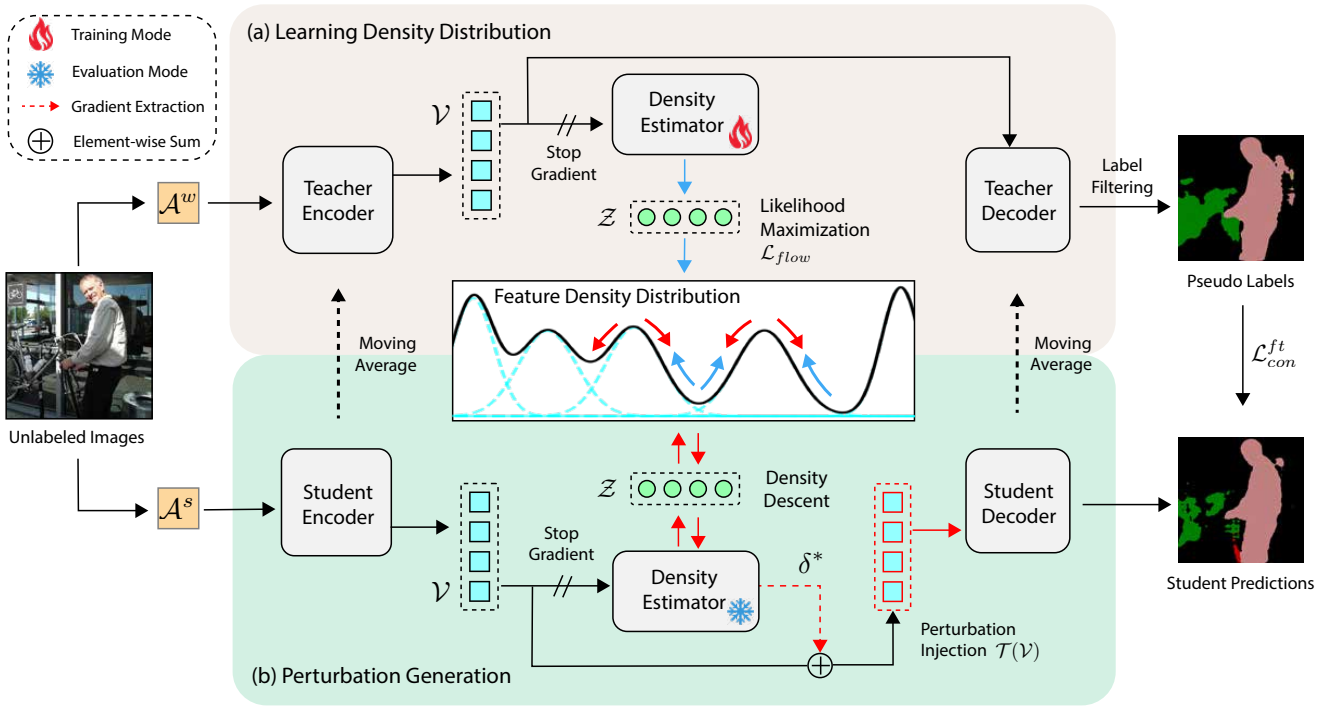


Figure 2. Overview of the proposed density-descending feature perturbation strategy. Based on the weak-to-strong consistency regularization, our method consists of two main components: (a) Learning Density Distribution and (b) Perturbation Generation. In phase (a), a lightweight normalizing-flow-based estimator is adopted to learn the density information on unlabeled features from teacher encoder. A mixture of Gaussian distribution is initialized and the estimator is optimized to maximize the feature likelihood on that distribution, which is denoted by blue arrows. Meanwhile in phase (b), the density estimator is set as evaluation mode and applied on the student features. Once feature distribution is approximated, the density-descending direction can be extracted by the gradient of the density objective on unlabeled features. Predictions on the density-descending features are supervised by pseudo labels from teacher model. The whole perturbation generation and injection process is indicated by red arrows.

niques from image classification to the segmentation domain, serving as a baseline for image-level strong augmentations. Subsequently, AEL [20] proposes adaptive CutMix, which targets under-performing categories during training. Consistency has also been explored at the feature level. UniMatch [51] introduces random channel dropout on features and enforces consistency between predictions on perturbed features and the original ones. PS-MT [33] further explores adversarial perturbations that induce the most disagreement among multiple teacher models, effectively regularizing the training of the student model.

Our research also focuses on feature-level consistency regularization. However, our novelty lies in the design of feature-level perturbations that leverage density information in the feature space to regulate the decision boundary.

3. Methodology

3.1. Problem Statement

In semi-supervised semantic segmentation, an labeled set $\mathcal{D}^l = \{(x_i^l, y_i^l)\}_{i=1}^{|\mathcal{D}^l|}$ is given with images $x_i^l \in \mathbb{R}^{H\times W\times C}$ of size $H \times W$ and channel number C. $y_i^l \in \mathbb{R}^{H\times W\times K}$ are annotations with K classes. Meanwhile, massive unlabeled data $\mathcal{D}^u = \{x_i^u\}_{i=1}^{|\mathcal{D}^u|}$ are also provided, where $|\mathcal{D}^u| \gg |\mathcal{D}^l|$. Then, semi-supervised segmentation models are designed to be optimized on both data sets, aiming to achieve stronger model generalization beyond labeled data.

3.2. Basic Framework

Before diving into the proposed feature perturbation strategy, we first go through the basic framework. We adopt the widely used teacher-student model combined with image-level consistency regularization in this work. The segmentation model $f = g \circ h$ consists of a feature encoder $h(\cdot) : \mathcal{X} \to \mathcal{V}$, which maps images $\mathcal{X}$ into feature space $\mathcal{V}$, and a mask decoder $g(\cdot) : \mathcal{V} \to \mathcal{P}$, which then decodes features into class probabilities $\mathcal{P}$. While the student model is optimized, the teacher model $f' = g' \circ h'$ is updated as the exponential moving average (EMA) of student.

For the supervised learning part, a mini-batch of labeled images $\mathcal{B}^l = \{(x_i^l, y_i^l)\}_{i=1}^{|\mathcal{B}^l|}$ is given. With i for i-th image and j indicating pixel index, the cross-entropy loss is

applied to supervise the model predictions:

$$\mathcal{L}_{sup} = \frac{1}{|\mathcal{B}^l|}\frac{1}{HW}\sum_{i=1}^{|\mathcal{B}^l|}\sum_{j=1}^{HW}\ell_{ce}(f(x_{ij}^l), y_{ij}^l). \quad (1)$$

The image-level consistency regularization is performed on unlabeled images $\mathcal{B}^u = \{x_i^u\}_{i=1}^{|\mathcal{B}^u|}$. Let $\mathcal{A}^w(\cdot)$ and $\mathcal{A}^s(\cdot)$ denote the weak and strong image augmentation strategies, respectively. For an unlabeled image x_i^u, we obtain the probability predictions on its augmented versions as:

$$p_i^w = f'(\mathcal{A}^w(x_i^u)) \quad (2)$$
$$p_i^s = f(\mathcal{A}^s(x_i^u)). \quad (3)$$

Pseudo labels y_{ij}^u are extracted by one-hot encoding on teacher predictions p_{ij}^w and then used to supervised the student predictions p_{ij}^s. The image-level consistency loss $\mathcal{L}_{con}^{im}$ is calculated as:

$$\mathcal{L}_{con}^{im} = \frac{1}{|\mathcal{B}^u|}\frac{1}{HW}\sum_{i=1}^{|\mathcal{B}^u|}\sum_{j=1}^{HW}\ell_{ce}(p_{ij}^s, y_{ij}^u)\cdot\mathbb{1}(\max(p_{ij}^w) > \tau), \quad (4)$$

where a fixed probability threshold τ is applied to screen out potential noisy labels with low prediction confidence.

3.3. Density-Descending Feature Exploration

Upon the basic learning framework, we introduce a novel feature-level consistency regularization to boost the model generalization ability. This section gives details of our density-descending feature perturbation strategy. We will first introduce the proposed density estimator which is based on normalizing flow. Then, the whole process is explained as shown in Fig. 2, which mainly consists two main sessions: (a) Learning Density Distribution where density estimator is optimized by likelihood maximization on image features to capture their density, and (b) Perturbation Generation where the learned density information is leveraged to generate density-descending perturbations on features.

3.3.1 Feature Density Estimation Module

To address the challenge of tracking the unknown density distribution $p_{\mathcal{V}}$ in a dynamic feature space $\mathcal{V}$, we propose a density estimator based on normalizing flows. Normalizing flows are specifically designed for generative modeling tasks, allowing for the learning of complex density functions by transforming a known distribution through a series of invertible mappings. In our approach, given a known latent distribution $p_{\mathcal{Z}}$, the density estimator is defined as the mapping $\varphi(v)$ that transforms features $\mathcal{V} \in \mathbb{R}^d$ into the latent space $\mathcal{Z} \in \mathbb{R}^d$, with the inverse mapping $\varphi^{-1}(z)$ to map them back to the original feature space.

The density of the unknown distribution $p_{\mathcal{V}}(v)$ can be modeled using the transformed variable $\varphi(v)$ through the change of variable formula:

$$p_{\mathcal{V}}(v) = p_{\mathcal{Z}}(\varphi(v)) \cdot \left|\det \frac{\partial \varphi}{\partial v}\right|, \quad (5)$$

where $\frac{\partial \varphi}{\partial v}$ represents the Jacobian matrix of the transformations in φ, which are carefully designed to ensure that the Jacobian determinant is tractable. The parameters in the normalizing flow are optimized through likelihood maximization on $p_{\mathcal{V}}$.

3.3.2 Learning Feature Density

During the training of segmentation model, the pixel-level features on both labeled and unlabeled images are extracted by encoders. Since teacher encoder shows higher stability in training, we focus on optimizing the density estimator φ_θ, parameterised by θ, on teacher features. In the latent space, we assign distinct Gaussian distributions to each class. Specifically, the distribution for class k is initialized with mean μ_k and covariance Σ_k, while the overall distribution across all categories is modeled as a mixture of Gaussians, weighted by π_k where $\sum_{k=1}^{K}\pi_k = 1$. This allows us to estimate the likelihood of a latent variable z with an unknown label using the following expression:

$$p_{\mathcal{Z}}(z) = \sum_{k=1}^{K}\pi_k\mathcal{N}(z|\mu_k, \Sigma_k). \quad (6)$$

Specifically, in each training iteration, a batch of labeled features $\mathcal{V}^l = \{v_m^l, y_m\}_{m=1}^{|\mathcal{V}^l|}$ and unlabeled features $\mathcal{V}^u = \{v_n^u\}_{n=1}^{|\mathcal{V}^u|}$ are collected from teacher encoder with gradient cut off. For a labeled feature v_m^l of class k, its likelihood is estimated by the target Gaussian component as

$$p_{\mathcal{V}}(v_m^l|y=k;\theta) = \mathcal{N}(\varphi_\theta(v_m^l)|\mu_k, \Sigma_k) \cdot \left|\det \frac{\partial \varphi_\theta}{\partial v_m^l}\right|. \quad (7)$$

For unlabeled feature, since no trustworthy labels are provided, their density can be captured in unsupervised manner to alleviate the potential bias in classifier predictions:

$$p_{\mathcal{V}}(v_n^u;\theta) = \sum_{k=1}^{K}\pi_k\mathcal{N}(\varphi_\theta(v_n^u)|\mu_k, \Sigma_k) \cdot \left|\det \frac{\partial \varphi_\theta}{\partial v_n^u}\right|. \quad (8)$$

The optimization objective $\mathcal{L}_{flow}$ for density estimator is the unified log-likelihood combining estimation from both labeled and unlabeled features:

$$\mathcal{L}_{flow} = -\frac{1}{|\mathcal{V}^l| + |\mathcal{V}^u|}\left(\sum_{m=1}^{|\mathcal{V}^l|}\log p_{\mathcal{V}}(v_m^l|y_m;\theta) + \sum_{n=1}^{|\mathcal{V}^u|}\log p_{\mathcal{V}}(v_n^u;\theta)\right). \quad (9)$$

Then, θ is optimized to maximize likelihood in feature space. By learning upon $p_{\mathcal{Z}}$, we can effectively harness the knowledge present in the labeled set to establish connections between classes and their corresponding Gaussian components and to guide the initialization of the density estimator φ_θ. This approach allow us to capture the inherent structure and relationships between different classes. Additionally, the optimization process on the unlabeled features through unsupervised likelihood maximization enables the density estimator to adapt and refine its representation to better align with the underlying data distribution.

3.3.3 Generating Density-descending Perturbation

During the optimization on the density estimator φ_θ, we employ it to generate density-descending perturbations on student features for consistency learning. The log-likelihood evaluation of a student feature v in the Gaussian mixtures, denoted as $\log p_{\mathcal{V}}(v;\theta)$, is determined based on the current φ_θ using Eq. 8. In such setting, each feature is evaluated in the unified distribution rather than a single Gaussian component, assuring the following density descent happens in the global scale.

The generation of density-descending perturbations draws inspiration from adversarial learning practices. However, unlike traditional adversarial learning, where perturbations primarily aims to attack classification results, our strategy focuses on the density objective, guided by the evaluation of the estimator φ_θ. Within a predefined exploration range ϵ, the objective is to find perturbations δ^* that result in the most substantial decrease in feature density:

$$\delta^* = \underset{\|\delta\|_2 \leq \epsilon}{\operatorname{argmax}} \left(-\log p_{\mathcal{V}}(v+\delta)\right). \tag{10}$$

The direction of perturbation can be determined by computing the gradient of the likelihood minimization objective $-\log p_{\mathcal{V}}(v)$ with respect to the target feature v, denoted as $\nabla_v(-\log p_{\mathcal{V}}(v))$. The calculation is as follows:

$$\delta^* = \epsilon \cdot \|\nabla_v(-\log p_{\mathcal{V}}(v))\|_2, \tag{11}$$

where $\|\cdot\|_2$ represents L2-normalization, and ϵ the magnitude of the perturbation, determining the exploration step. Consequently, the density-descending version of the original feature can be obtained by injecting the perturbation:

$$\mathcal{T}(v) = v + \delta^*, \tag{12}$$

where $\mathcal{T}(\cdot)$ denotes the perturbation injection operation. The perturbed features are expected to shift towards lower-density regions, based on the current estimation of φ_θ.

3.4. Unified Training Objective

With the proposed perturbation generation strategy, we employ feature-level consistency regularization. The features to be perturbed are shared with image-level consistency learning for computational efficiency, which are extracted from the student encoder h as follows:

$$v_i^s = h(\mathcal{A}^s(x_i^u)). \tag{13}$$

Subsequently, predictions on these features are generated using the student decoder g:

$$p_{ij}^{ft} = g(\mathcal{T}(v_{ij}^s)). \tag{14}$$

To align the predictions between the original and perturbed features, we introduce the consistency loss $\mathcal{L}_{con}^{ft}$, which is calculated as:

$$\mathcal{L}_{con}^{ft} = \frac{1}{|\mathcal{B}^u|}\frac{1}{HW}\sum_{i=1}^{|\mathcal{B}^u|}\sum_{j=1}^{HW} \ell_{ce}(p_{ij}^{ft}, y_{ij}^u)\cdot \mathbb{1}(\max(p_{ij}^w) > \tau). \tag{15}$$

Here, pseudo labels are derived from teacher predictions, and label filtration is employed to ensure that density-descending features are only guided by reliable labels.

The overall optimization is achieved by unifying three learning objectives: the supervised learning loss $\mathcal{L}_{sup}$, the image-level consistency loss $\mathcal{L}_{con}^{im}$, and the feature-level consistency loss with density-descending feature perturbations. The unified objective $\mathcal{L}_{uni}$ is defined as:

$$\mathcal{L}_{uni} = \mathcal{L}_{sup} + \mathcal{L}_{con}^{im} + \lambda_{ft}\mathcal{L}_{con}^{ft}. \tag{16}$$

Here, λ_{ft} represents the weight assigned to the feature-level consistency loss term.

4. Experiments

4.1. Implementation Details.

Datasets. The experiments are performed on two standard datasets, which are Pascal VOC 2012 [13] and Cityscapes [10]. Pascal VOC contains 20 foreground and 1 background classes. It is initially built with 1464 training images with high-quality annotations and 1449 validation images, which is denoted as *classic* set. Then, SBD dataset [17] with coarse annotations is introduced to extend the training set to 10582 images to form the *blended* set. We conduct experiments on both *classic* and *blended* sets. Cityscapes contains 19 semantic categories of urban scenes. It consists of 2975 and 500 annotated images for training and validation, respectively. For both datasets, we follow data partitions in CPS [9] to generate subsets of 1/16, 1/8, 1/4, and 1/2 from the training set as labeled data, while the remaining images are utilized as the unlabeled set.

Evaluation Protocols. We report the mean of intersection over union (mIoU) for all settings. For Pascal VOC, single-scale evaluation on center-cropped images are performed.

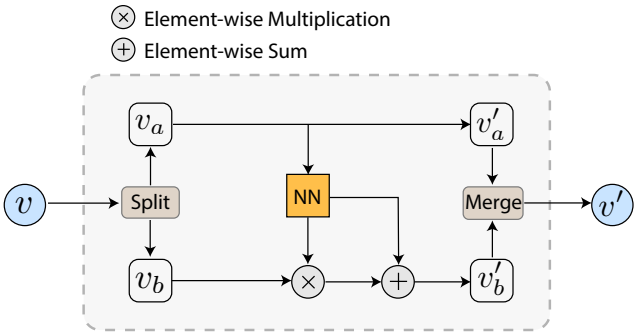


Figure 3. Block design for the proposed density estimator.

For Cityscapes, we adopt sliding window evaluation on the validation images with resolution of 1024×2048.

Implementation Details. We adopt DeepLabV3+ [8] with ResNet-101 [18] pretrained on ImageNet [11] as the segmentation model. We use SGD optimizer with momentum of 0.9 and polynomial learning rate scheduler. For Pascal VOC, the model is trained for 80 epochs with initial learning rate of 0.001. The images are copped to 513×513 for training. For Cityscapes, we train 200 epochs with initial learning rate of 0.01 under a crop size of 769×769. The batch size is set to 16 for both datasets. The prediction thresholds τ in Eq. 4 and 15 are set to 0.95 for Pascal VOC and 0.7 for Cityscapes. The momentum for updating teacher model is set to 0.999 for all experiments.

For the weak image-level augmentation $\mathcal{A}^w(\cdot)$, we adopt random resize within the scale range $[0.5, 2.0]$, random crop and random flip. The strong data augmentation $\mathcal{A}^s(\cdot)$ is implemented by the random combination of CutMix [15], Gaussian blur, color jitter and random grayscale.

Density Estimator. Our density estimator is designed as a modified RealNVP [12] module. Since the features are regulated by the segmentation objective, they show more clear patterns and also with lower dimensions compared to natural images, which largely ease the process of density learning. This allows a highly lightweight density learner with negligible amount of parameters. Fig. 3 shows a single block in the proposed density estimator, where the input feature v is split in half in channel and then merged into v' with identical dimensions. The whole module contains two cascaded blocks with channel permutation between them, transforming from v to z. The NN part is implemented by two cascaded Linear (256) layers with learnable parameters. The module is implemented with FrEIA [2] library.

For the GMM in latent space, the number of Gaussian components is set as 21 and 19 for Pascal VOC and Cityscapes, respectively. The mean for k-th component is drawn from the standard normal distribution $\mu_k \sim \mathcal{N}(0, I)$ and the covariance matrix is set as an identity matrix $\Sigma_k = I$. Only the flow module φ_θ is updated by likelihood maximization. In each iteration, 20k feature vectors are sampled equally from both labeled and unlabeled samples for training. For all the experiments, we adopt Adam optimizer and step learning rate scheduler with initial learning rate of 0.001. The training of density estimator starts from the second epoch during the training of segmentation model.

4.2. Comparison with State-of-the-Art Methods

In this section, we compare our method with the state-of-the-art on both Pascal VOC and Cityscapes under various partition protocols. The data splits in our experiments strictly follow previous works [9, 33, 50]. All the results are produced by DeepLabV3+ segmentation decoder with ResNet-101 as backbone.

Results on Pascal VOC 2012. In Tab. 1, we compare our DDFP with other methods on *classic* Pascal VOC set and our method shows competitive results. DDFP brings significant performance improvement over supervised baseline by +29.18%, +23.09% and +13.63% on 1/16, 1/8 and 1/4 splits. Compared with previous method PS-MT [33] that focuses on designing feature-level perturbations, our DDFP yields stronger performance especially in low-data regime. Specifically, we improve over PS-MT by +9.15% and +8.43% under 92 and 183 labeled images, respectively.

Tab. 2 reports the results on *blended* Pascal VOC dataset that contains noisy annotations, which is a more challenging setting. Our DDFP consistently produce competitive results. Compared with supervised baseline, our method achieves performance gain of +10.45%, +7.33%, +4.03% and +3.77% on 1/16, 1/8, 1/4 and 1/2 labeled image proportions, respectively. Compared with previous methods, our method yields best results in most cases.

Results on Cityscapes. In Tab. 3, we evaluate our method on Cityscapes. The proposed DDFP significantly improves supervised baseline by +11.36%, +5.66%, +5.45% and +2.99% on 1/16, 1/8, 1/4 and 1/2 splits. Our method outperforms previous best method UniMatch [51] by +0.68% and +1.32% on 1/4 and 1/2 data settings.

4.3. Ablation Studies

In this section, we conduct a series of experiments to investigate the effectiveness of our proposed feature perturbation strategy. All the experiments are based on 1/4 (366) and 1/2 (732) data partitions in *classic* Pascal VOC 2012 dataset.

Effectiveness of the density-descending perturbation. In Tab. 4, we ablate the framework to manifest the effectiveness of DDFP. We set the model with image-level consistency regularization in Experiment II as the main baseline. We first introduce noise sampled from normal distribution to see if randomly perturbed features can enhance model performance. The noise vectors are normalized and then applied on features. By comparing results in Experiment III

Table 1. Comparison with state-of-the-art methods on PASCAL VOC 2012 validation set with mIoU results (%) ↑. Labeled images are sampled from the high-quality *classic* set comprising 1464 samples. The fractions $1/n$ and the following integers (m) denote the proportions and numbers of labeled images, respectively.

Method	1/16 (92)	1/8 (183)	1/4 (366)	1/2 (732)	Full (1464)
Supervised	45.77	54.92	65.88	71.69	72.50
CutMix [15]	52.16	63.47	69.46	73.73	76.54
CPS [9]	64.07	67.42	71.71	75.88	-
U^{2}PL [45]	67.98	69.15	73.66	76.16	79.49
ST++ [50]	65.20	71.00	74.60	77.30	79.10
PS-MT [33]	65.80	69.58	76.57	78.42	80.01
PCR [49]	70.06	74.71	77.16	78.49	80.65
GTA-Seg [25]	70.02	73.16	75.57	78.37	80.47
UniMatch [51]	**75.20**	77.20	78.80	79.90	81.20
CCVC [47]	70.20	74.40	77.40	79.10	80.50
AugSeg [57]	71.09	75.45	78.80	80.33	81.36
Ours	74.95	**78.01**	**79.51**	**81.21**	**81.96**

Table 2. Comparison with state-of-the-art methods on PASCAL VOC 2012 validation set with mIoU results (%) ↑. Labeled images are sampled from the extended *blended* set which consists of 10582 samples. * means reproduced results on CPS [9] splits.

Method	1/16 (662)	1/8 (1323)	1/4 (2646)	1/2 (5291)
Supervised	67.87	71.55	75.80	77.13
MT [43]	70.51	71.53	73.02	76.58
CutMix [15]	71.66	75.51	77.33	78.21
CCT [37]	71.86	73.68	76.51	77.40
GCT [26]	70.90	73.29	76.66	77.98
CPS [9]	74.48	76.44	77.68	78.64
U^{2}PL* [45]	74.43	77.60	78.70	79.94
PS-MT [33]	75.50	78.20	78.72	79.76
UniMatch [51]	78.10	78.40	79.20	-
CCVC [47]	76.80	**79.40**	79.60	-
AugSeg [57]	77.01	77.31	78.82	-
Ours	**78.32**	78.88	**79.83**	**80.90**

$*$ Results are reproduced on CPS [9] splits.

Table 3. Comparison with state-of-the-art methods on Cityscapes validation set with mIoU results (%) ↑. Labeled images are sampled from Cityscapes *train* set which contains 2975 samples.

Method	1/16 (186)	1/8 (372)	1/4 (744)	1/2 (1488)
Supervised	65.74	72.53	74.43	77.83
MT [43]	69.03	72.06	74.20	78.15
CutMix [15]	67.06	71.83	76.36	78.25
CCT [37]	69.32	74.12	75.99	78.10
GCT [26]	66.75	72.66	76.11	78.34
CPS [9]	74.72	77.62	79.21	80.21
U^{2}PL [45]	70.30	74.37	76.47	79.05
PS-MT [33]	-	76.89	77.60	79.09
PCR [49]	73.41	76.31	78.40	79.11
GTA-Seg [25]	69.38	72.02	76.08	-
UniMatch [51]	76.60	77.90	79.20	79.50
AugSeg [57]	75.22	77.82	79.56	80.43
Ours	**77.10**	**78.19**	**79.88**	**80.82**

Table 4. Ablation study on main components of DDFP. $\mathbf{L}_{con}^{im}$: Self-training with image-level consistency regularization. $\mathcal{L}_{con}^{ft}$: Feature-level consistency regularization. Random: Random noise sampled from normal distribution as perturbations. DD: The proposed density-descending perturbations.

	DDFP Framework				mIoU (%)	
	Supervised	$\mathcal{L}_{con}^{im}$	$\mathcal{L}_{con}^{ft}$ (Random)	$\mathcal{L}_{con}^{ft}$ (DD)	366	732
I	✓				65.88	71.69
II	✓	✓			75.44	77.56
III	✓	✓	✓		76.26	78.51
IV	✓	✓		✓	**79.51**	**81.21**

and II, random perturbations improve baseline by +0.82% and +0.95% on 366 and 732 settings, respectively. Then, in Experiment IV, we inject density-descending perturbations under the same magnitude to replace random noise, which significantly boosts the model performance by +4.07% and +3.65% on two splits compared with baseline. This indicates that most performance gain in DDFP is brought by our density-descending design.

Impact of perturbation step size ϵ. We also investigate the impact of perturbation step size ϵ in Eq. 11, which indicates the magnitude of a normalized vector. In Tab. 5, we compare the performance under different step sizes. We found that the model performance improves when step size increases from 1, and the optimal step size slightly varies on different data splits. We choose step size of 4 as the default setting where overall best performance can be achieved. Overly large step size can hurt the performance since ag-

Table 5. Ablation study on the magnitude of perturbation vectors. The magnitude refers to ϵ in Eq. 11.

Magnitude	1	2	4	6	8
366	78.76	78.96	**79.51**	79.43	79.41
732	81.03	**81.23**	81.21	81.07	80.79

Table 6. Ablation study on the impact of loss weight λ_{ft} in Eq. 16.

λ_{ft}	0.2	0.5	1	1.5	2
366	78.89	**79.51**	79.28	79.40	77.78
732	80.87	**81.21**	81.13	80.56	80.22

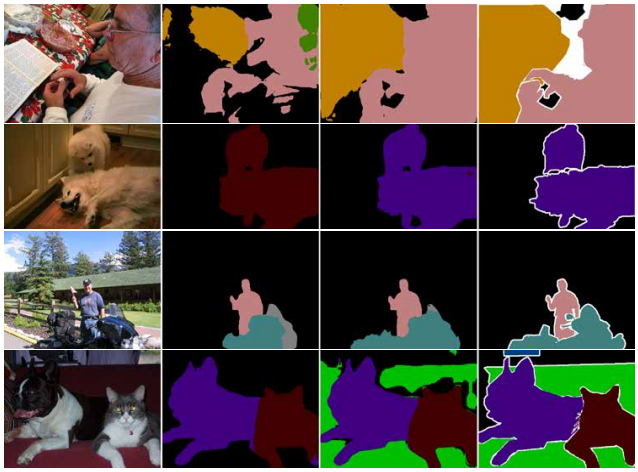
Table 7. Comparison with other feature-level perturbations. All perturbations are implemented on the same baseline framework.

Perturbation Type	366	732
Uniform Noise [37]	75.69	78.52
Channel Dropout [51]	77.76	80.38
VAT [33]	76.87	78.38
Density-descending	**79.51**	**81.21**

gressive exploring towards low density regions might end up with out-of-distribution samples.

Impact of loss weight λ_{ft}. We also examine the model performance under different λ_{ft} in Eq. 16. As shown in Tab. 6, the optimal weight for our feature-level consistency loss is around 0.5, which is the default setting in our experiments. Further increasing the contribution of the loss can lead to degenerated performance. Our guess is the image-level consistency serves as the foundation for the proposed perturbation strategy to function and the overly aggressive optimization of feature-level consistency might interfere the consistency learning at image-level.

Comparison with other feature perturbations. To further validate the effectiveness of DDFP, we compare it with different types of feature-level perturbations. As shown in Tab. 7, Uniform Noise is the perturbation sampled from a uniform distribution as in CCT [37]. Channel Dropout refers to randomly zeroing out half of the feature channels following UniMatch [51]. VAT indicates the perturbation in virtual adversarial training [36] which is introduced in semi-supervised semantic segmentation by PS-MT [33]. All the perturbations are implemented on the same baseline framework which is self-training with single-stream weak-to-strong image-level consistency regularization. In Tab. 7, the quantitative results show the superiority of our density-descending perturbation. Specifically, our DDFP significantly outperforms the second best Channel Dropout strategy by +1.75% and +0.83% on 366 and 732 splits.



(a) (b) (c) (d)

Figure 4. Qualitative results on Pascal VOC 2012 dataset. Models are trained under *classic* 732 partitions. (a) Input images. (b) Results from baseline model with image-level consistency regularization only. (c) Results by our DDFP. (d) Ground truth.

Qualitative Results. We show the qualitative results on Pascal VOC dataset in Fig. 4. By observation, baseline method only with image-level consistency regularization performs poorly to capture the complete outline of the object as shown in the first row in Fig. 4 (b). Also, in certain context, it has difficulty distinguishing visually similar classes like cat and dog as shown in the second row of Fig. 4 (b). Equipped with DDFP, the model predicted more accurately on shapes and classes, which is shown in Fig. 4 (c).

5. Conclusion

In this work, we propose a novel feature-level consistency regularization strategy name Density-Descending Feature Perturbation (DDFP) for semi-supervised semantic segmentation. The aim of DDFP is to create perturbed features in low density regions in feature space, to force decision boundary to explore less dense regions thus enhancing model generalization. Density estimation is the heart of our method, which is achieved by our proposed lightweight density estimator based on normalizing flow. Extensive experiments under various data settings have shown that our DDFP can effectively boost model performance and outperform other types of feature-level perturbation designs.

Acknowledgments. This work was supported by the National Key R&D Program of China (No.2022YFE0200300), the National Natural Science Foundation of China (No. 61972323, 62331003), Suzhou Basic Research Program (SYG202316) and XJTLU REF-22-01-010, XJTLU AI University Research Centre, Jiangsu Province Engineering Research Centre of Data Science and Cognitive Computation at XJTLU and SIP AI innovation platform (YZCXPT2022103).

References

[1] Iñigo Alonso, Alberto Sabater, David Ferstl, Luis Montesano, and Ana Cristina Murillo. Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank. In *ICCV*, 2021. 2

[2] Lynton Ardizzone, Till Bungert, Felix Draxler, Ullrich Köthe, Jakob Kruse, Robert Schmier, and Peter Sorrenson. Framework for easily invertible architectures (freia), 2022. 6

[3] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 2019. 2

[4] Olivier Chapelle and Alexander Zien. Semi-supervised classification by low density separation. In *International Conference on Artificial Intelligence and Statistics*, 2005. 2

[5] Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xingxu Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. In *ICLR*, 2023. 2

[6] Jinpeng Chen, Runmin Cong, Yuxuan Luo, Horace Ho Shing Ip, and Sam Kwong. Saving 100x storage: Prototype replay for reconstructing training sample distribution in class-incremental semantic segmentation. In *NeurIPS*, 2023. 2

[7] 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. *PAMI*, 40(4):834–848, 2018. 1

[8] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 6

[9] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *CVPR*, 2021. 2, 5, 6, 7

[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 *CVPR*, 2016. 2, 5

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 6

[12] Laurent Dinh, Jascha Narain Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *ICLR*, 2017. 2, 6

[13] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2009. 1, 2, 5

[14] Zhengyang Feng, Qianyu Zhou, Qiqi Gu, Xin Tan, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. Dmt: Dynamic mutual training for semi-supervised learning. *PR*, 130:108777, 2020. 2

[15] Geoffrey French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham D. Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. In *BMVC*, 2020. 1, 2, 6, 7

[16] Dayan Guan, Jiaxing Huang, Aoran Xiao, and Shijian Lu. Unbiased subclass regularization for semi-supervised semantic segmentation. In *CVPR*, 2022. 1

[17] Bharath Hariharan, Pablo Arbeláez, Lubomir D. Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, 2011. 5

[18] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[19] Ruifei He, Jihan Yang, and Xiaojuan Qi. Re-distributing biased pseudo labels for semi-supervised semantic segmentation: A baseline investigation. In *ICCV*, 2021. 1

[20] Hanzhe Hu, Fangyun Wei, Han Hu, Qiwei Ye, Jinshi Cui, and Liwei Wang. Semi-supervised semantic segmentation via adaptive equalization learning. In *NeurIPS*, 2021. 2, 3

[21] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. In *BMVC*, 2018. 1

[22] Pavel Izmailov, P. Kirichenko, Marc Finzi, and Andrew Gordon Wilson. Semi-supervised learning with normalizing flows. In *ICML*, 2019. 2

[23] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *NeurIPS*, 2019. 2

[24] Siyu Jiao, Yunchao Wei, Yaowei Wang, Yao Zhao, and Humphrey Shi. Learning mask-aware clip representations for zero-shot segmentation. 2023. 1

[25] Ying Jin, Jiaqi Wang, and Dahua Lin. Semi-supervised semantic segmentation via gentle teaching assistant. In *NeurIPS*, 2022. 7

[26] Zhanghan Ke, Di Qiu, Kaican Li, Qiong Yan, and Rynson W. H. Lau. Guided collaborative training for pixel-wise semi-supervised learning. In *ECCV*, 2020. 7

[27] Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *NeurIPS*, 2018. 2

[28] Xin Lai, Zhuotao Tian, Li Jiang, Shu Liu, Hengshuang Zhao, Liwei Wang, and Jiaya Jia. Semi-supervised semantic segmentation with directional context-aware consistency. In *CVPR*, 2021. 2

[29] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *ICLR*, 2017. 2

[30] Man Liu, Chunjie Zhang, Huihui Bai, Riquan Zhang, and Yao Zhao. Cross-part learning for fine-grained image classification. *TIP*, 31:748–758, 2022.

[31] Man Liu, Feng Li, Chunjie Zhang, Yunchao Wei, Huihui Bai, and Yao Zhao. Progressive semantic-visual mutual adaption for generalized zero-shot learning. In *CVPR*, 2023. 2

[32] Shikun Liu, Shuaifeng Zhi, Edward Johns, and Andrew J Davison. Bootstrapping semantic segmentation with regional contrast. In *ICLR*, 2022. 2

[33] Yuyuan Liu, Yu Tian, Yuanhong Chen, Fengbei Liu, Vasileios Belagiannis, and G. Carneiro. Perturbed and strict mean teachers for semi-supervised semantic segmentation. In *CVPR*, 2022. 1, 2, 3, 6, 7, 8

[34] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1

[35] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high- and low-level consistency. *PAMI*, 43(4):1369–1379, 2021. 1

[36] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *PAMI*, 41(8):1979–1993, 2019. 8

[37] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *CVPR*, 2020. 1, 2, 7, 8

[38] Hieu Pham, Qizhe Xie, Zihang Dai, and Quoc V. Le. Meta pseudo labels. In *CVPR*, 2021. 2

[39] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. In *WACV*, 2005. 2

[40] Mehdi S. M. Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *NeurIPS*, 2016. 2

[41] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin Dogus Cubuk, Alexey Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020. 2

[42] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In *ICCV*, 2017. 1

[43] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017. 2, 7

[44] Xiaoyang Wang, Bingfeng Zhang, Limin Yu, and Jimin Xiao. Hunting sparsity: Density-guided contrastive learning for semi-supervised semantic segmentation. In *CVPR*, 2023. 2

[45] Yuchao Wang, Haochen Wang, Yujun Shen, Jingjing Fei, Wei Li, Guoqiang Jin, Liwei Wu, Rui Zhao, and Xinyi Le. Semi-supervised semantic segmentation using unreliable pseudo-labels. In *CVPR*, 2022. 2, 7

[46] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Zhen Wu, and Jindong Wang. Freematch: Self-adaptive thresholding for semi-supervised learning. In *ICLR*, 2023. 2

[47] Zicheng Wang, Zhen Zhao, Luping Zhou, Dong Xu, Xiaoxia Xing, and Xiangyu Kong. Conflict-based cross-view consistency for semi-supervised semantic segmentation. In *CVPR*, 2023. 2, 7

[48] Qizhe Xie, Eduard H. Hovy, Minh-Thang Luong, and Quoc V. Le. Self-training with noisy student improves imagenet classification. In *CVPR*, 2020. 2

[49] Hai-Ming Xu, Lingqiao Liu, Qiuchen Bian, and Zhengeng Yang. Semi-supervised semantic segmentation with prototype-based consistency regularization. In *NeurIPS*, 2022. 7

[50] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. St++: Make self-training work better for semi-supervised semantic segmentation. In *CVPR*, 2022. 1, 2, 6, 7

[51] Lihe Yang, Lei Qi, Litong Feng, Wayne Zhang, and Yinghuan Shi. Revisiting weak-to-strong consistency in semi-supervised semantic segmentation. In *CVPR*, 2023. 1, 2, 3, 6, 7, 8

[52] Jianlong Yuan, Yifan Liu, Chunhua Shen, Zhibin Wang, and Hao Li. A simple baseline for semi-supervised semantic segmentation with strong data augmentation. In *ICCV*, 2021. 1

[53] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In *NeurIPS*, 2021. 2

[54] Zekang Zhang, Guangyu Gao, Jianbo Jiao, Chi Harold Liu, and Yunchao Wei. Coinseg: Contrast inter- and intra- class representations for incremental segmentation. In *ICCV*, 2023. 2

[55] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017. 1

[56] Xinqiao Zhao, Feilong Tang, Xiaoyang Wang, and Jimin Xiao. Sfc: Shared feature calibration in weakly supervised semantic segmentation. In *AAAI*, 2024. 2

[57] Zhen Zhao, Lihe Yang, Sifan Long, Jimin Pi, Luping Zhou, and Jingdong Wang. Augmentation matters: A simple-yet-effective approach to semi-supervised semantic segmentation. In *CVPR*, 2023. 2, 7

[58] Yuanyi Zhong, Bodi Yuan, Hong Wu, Zhiqiang Yuan, Jian Peng, and Yu-Xiong Wang. Pixel contrastive-consistent semi-supervised semantic segmentation. In *ICCV*, 2021. 2

[59] Yanning Zhou, Hang Xu, Wei Zhang, Bin-Bin Gao, and Pheng-Ann Heng. C3-semiseg: Contrastive semi-supervised segmentation via cross-set learning and dynamic class-balancing. In *ICCV*, 2021. 2

[60] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc V. Le. Rethinking pre-training and self-training. In *NeurIPS*, 2020. 2