Fuzzy Positive Learning for Semi-supervised Semantic Segmentation

Pengchong Qiao[1,2,4*] Zhidan Wei[1,4*] Yu wang[1,4] Zhennan Wang[2] Guoli song[2]
Fan Xu[2] Xiangyang Ji[3] Chang Liu[3†] Jie Chen[1,2,4†]

[1] School of Electronic and Computer Engineering, Peking University, Shenzhen, China
[2] Peng Cheng Laboratory, Shenzhen, China
[3] Department of Automation and BNRist, Tsinghua University, Beijing, China
[4] AI for Science (AI4S)-Preferred Program, Peking University Shenzhen Graduate School, China

{pcqiao,adan,rain_wang}@stu.pku.edu.cn, {wangzhn,songgl,xuf01,chenj}@pcl.ac.cn,
{xyji,liuchang2022}@tsinghua.edu.cn

Abstract

Semi-supervised learning (SSL) essentially pursues class boundary exploration with less dependence on human annotations. Although typical attempts focus on ameliorating the inevitable error-prone pseudo-labeling, we think differently and resort to exhausting informative semantics from multiple probably correct candidate labels. In this paper, we introduce Fuzzy Positive Learning (FPL) for accurate SSL semantic segmentation in a plug-and-play fashion, targeting adaptively encouraging fuzzy positive predictions and suppressing highly-probable negatives. Being conceptually simple yet practically effective, FPL can remarkably alleviate interference from wrong pseudo labels and progressively achieve clear pixel-level semantic discrimination. Concretely, our FPL approach consists of two main components, including fuzzy positive assignment (FPA) to provide an adaptive number of labels for each pixel and fuzzy positive regularization (FPR) to restrict the predictions of fuzzy positive categories to be larger than the rest under different perturbations. Theoretical analysis and extensive experiments on Cityscapes and VOC 2012 with consistent performance gain justify the superiority of our approach. Codes are provided in https://github.com/qpc1611094/FPL.

1. Introduction

Semantic segmentation models enable accurate scene understanding [1, 29, 45] with the help of fine pixel-level annotations. Yet, collecting labeled segmentation datasets is time-consuming and labor-costing [6]. Considering unlabeled data are annotation-free and easily accessible, semi-supervised learning (SSL) is introduced into semantic segmentation [5, 34, 43, 49, 51, 53] to encourage the model to generalize better on unseen data with less dependence on artificial annotations.

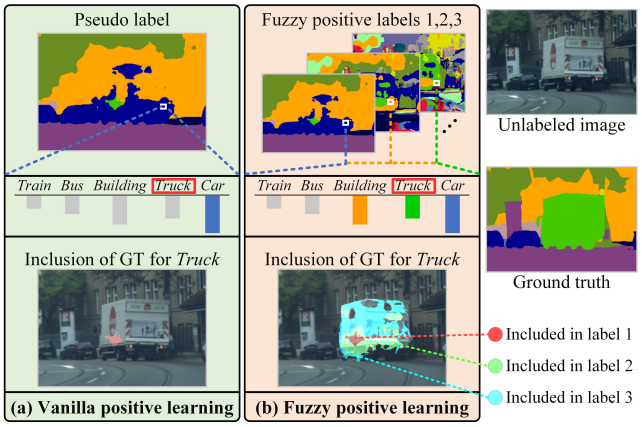

Figure 1. (a) Existing methods using pseudo label to utilize unlabeled data. (b) The proposed FPL that provides multiple fuzzy positive labels for each pixel to utilize unlabeled data. The example of 'Truck' shows that our method covers ground truth (GT) more comprehensively than vanilla positive learning.

The semi-supervised segmentation task faces a scenario where only a subset of training images are assigned segmentation labels while the others remain unlabeled. Current state-of-the-art (SOTA) methods utilize unlabeled data via consistency regularization, which aims to obtain invariant predictions for unlabeled pixels under various perturbations [5, 34, 49, 53]. Their general paradigm is to use the pseudo label generated under weak (or none) perturbations as the learning target of predictions under strong perturbations. Though achieving promising results, errors are inevitable in the pseudo label used in these methods, misguid-

*Equal contribution.
†Corresponding author.

ing the training of their models [24, 33]. An intuitive example is that some pixels may be confused in categories with similar semantics. As Fig. 1 (a), some pixels belonging to 'Truck' are wrongly classified into the 'Car' category (e.g., white boxed pixel). To mitigate this problem, typical methods focus on ameliorating the learning of pseudo labels by filtering low-confidence pseudo labels out [14,21,38,51,53] and generating pseudo labels more accurately [8,20,26,48]. However, the semantics of ground truth buried in other unselected labels are ignored in existing methods.

In this paper, we propose **Fuzzy Positive Learning** (FPL), a new SSL segmentation method that exhausts informative semantics from multiple probably correct candidate labels. We name these labels "fuzzy positive" labels since each of them has the probability to be the ground truth. As shown in Fig. 1 (b), our fuzzy positive labels cover the ground truth more comprehensively, facilitating our FPL to exploit the semantics of ground truth better. Extending learning from one pseudo label to learning from multiple fuzzy positive labels is not a simple implementation, which contains two pending issues. One is how to provide an adaptive number of labels for each pixel. And the other one is how to exploit the possible GT semantics from fuzzy positive labels. For these two issues, a fuzzy positive assignment (FPA) algorithm is first proposed to select which labels should be appended to the fuzzy positive label set of each pixel. Afterward, a fuzzy positive regularization (FPR) is developed to regularize the predictions of fuzzy positive categories to be larger than the predictions of the rest negative categories under different perturbations.

Our FPL achieves consistent performance gain on Cityscapes and Pascal VOC 2012 datasets using CPS [5] and AEL [14] as baselines. Moreover, we theoretically and empirically analyze that the superiority of FPL lies in revising the gradient of learning ground truth when pseudo-labels are wrongly-assigned. Our main contributions are:

- FPL provides a new perspective for SSL segmentation, that is, learning informative semantics from multiple fuzzy positive labels instead of only one pseudo label.
- A fuzzy positive assignment is proposed to provide an adaptive number of labels for each pixel. Besides, a fuzzy positive regularization is developed to learn the semantics of ground truth from fuzzy positive labels.
- FPL is easy to implement and could bring stable performance gains on existing SSL segmentation methods in a plug-and-play fashion.

2. Related Work

2.1. Semi-supervised Learning

Modern SSL classification approaches typically learn semantics from unlabeled data by introducing techniques of entropy minimization and consistency regularization. Entropy minimization enforces the predicted probability distribution to be sharp by training upon pseudo labels [2, 3, 23, 27, 38, 46]. On the other hand, consistency regularization aims to obtain prediction invariance under various perturbations, including input perturbation [31,38,46], feature perturbation [34], network perturbation [10,18,35,40], *etc*. Variants of their combination have achieved great success [35,38,44,47,50], whose core inspiration is computing consistency regularization via pseudo labeling.

2.2. Semi-supervised Semantic Segmentation

Semi-supervised semantic segmentation methods benefit from the development of general semi-supervised learning, which could be also roughly divided into two types of approaches: consistency regularization based methods [11, 17, 19, 34] and entropy-minimization based methods [4, 9, 16, 28, 30, 52]. More recently, SOTA semi-supervised segmentation methods combine both two technologies together to train their models. PseudoSeg [53], AEL [14], UCC [8] and Jianglong Yuan et al. [49] propose to use the pseudo label generated from weak augmented image to supervise the prediction of strong augmented image. CPS [5] designs a mutual learning mechanism that trains two student models with pseudo labels from each other. PC^{2}Seg [51] proposes a negative sampling technique to provide reliable negative samples for SSL segmentation. Different from existing methods, we propose for the first time to exploit the informative semantics of unlabeled data from multiple fuzzy positive labels, resulting in less interference from wrong pseudo labels and accurate segmentation.

Pseudo-label learning is the key technology in current SSL segmentation methods, but it has a limitation in that wrong pseudo labels mislead the training of SSL models. Typical approaches design filter-out mechanisms to use only high-confidence pseudo-labels for training [8, 14, 21, 51,53] and develop complex training mechanisms to predict accurate pseudo-labels [8, 20, 26, 48]. Apart from the above methods, U^{2}PL [43] introduces the idea of negative learning into SSL segmentation, which has similarities to our FPL. It thinks uncertain pixels usually get confused among only a few classes. Hence, it uses uncertain pixels as negative samples for those unlikely classes. We analyze that our FPL and negative learning have mathematically different optimization objectives. That is, negative learning implicitly maximizes only the prediction of the pseudo-label, while our FPL learns all fuzzy positive labels. (cf. Appendix).

3. Method

3.1. Preliminaries

Overview: For the SSL segmentation task, we have a small labeled dataset $D_l = \{(x_l, y_l)\}_{l=1}^{L}$ and a large unla-

beled dataset $D_u = \{x_u\}_{u=1}^{U}$, where L is the size of the labeled dataset, and U is the size of the unlabeled dataset ($L \ll U$). The x_l, y_l, x_u are the image and label of the l-th labeled data and the image of the u-th unlabeled data, respectively. The purpose of SSL segmentation is to learn the parameters θ of a segmentation model $\mathbb{F}(\bullet;\theta)$ by optimizing a loss function that contains both supervised and unsupervised loss:

$$\mathcal{L} = \frac{1}{L}\sum_{l=1}^{L}\mathcal{L}^{sup}(\mathbb{F}(x_l;\theta)) + \frac{\beta}{U}\sum_{u=1}^{U}\mathcal{L}^{uns}(\mathbb{F}(x_u;\theta)), \quad (1)$$

where $\mathcal{L}^{sup}$ and $\mathcal{L}^{uns}$ are supervised loss and unsupervised loss, and β is a regularization weight.

In current SOTA methods [5, 8, 14, 20, 26, 49, 53], the unsupervised loss in Eq. 1 is formulated as the cross-entropy loss between model predictions and pseudo labels, which are also predicted by their models. The paradigm is:

$$\begin{aligned} \overline{\boldsymbol{y}_u} &= \mathbb{1}(\arg\max(\mathbb{F}(x_u))), \quad \boldsymbol{z}_u = \hat{\mathbb{F}}(\hat{x}_u) \\ \mathcal{L}_u^v &= \frac{1}{S}\sum_{s=1}^{S}\mathcal{L}_{us}^v(\boldsymbol{z}_{us}, \overline{\boldsymbol{y}_{us}}) \\ &= \frac{1}{S}\sum_{s=1}^{S}\sum_{c=1}^{C} -\overline{y_{us}^c}\log(\frac{\exp(z_{us}^c)}{\sum_{n=1}^{C}\exp(z_{us}^n)}), \end{aligned} \quad (2)$$

where the $\overline{\boldsymbol{y}_u}$ is the one-hot encoding of the pseudo label generated from a segmentation model $\mathbb{F}$, and $\mathbb{1}$ is the one-hot-encoding function. The $\boldsymbol{z}_u$ is the prediction vector from disturbed model $\hat{\mathbb{F}}$ with disturbed input $\hat{x}_u$. The disturbed model is often realized by adding dropout layers [22, 34] into the model structure, or injecting random noises into the feature maps [26, 34]. And the disturbed input is usually realized by data augmentations [5, 14, 49, 53]. The S is the number of pixels in image x_u and C is the number of categories, and $\overline{y_{us}^c}$ and z_{us}^c are the elements of $\overline{\boldsymbol{y}_u}$ and $\boldsymbol{z}_u$ for the c-th class of the s-th pixel. This vanilla positive loss $\mathcal{L}_u^v$ has only one learning target, the pseudo label.

Motivation: By the definition of $\mathcal{L}_{us}^v$, its gradient with respect to the prediction $\boldsymbol{z}_{us}$ in backpropagation is:

$$\frac{\partial\mathcal{L}_{us}^v}{\partial z_{us}^c} = \begin{cases} p_{us}^c - 1, & if \ \ \overline{y_{us}^c} = 1, \\ p_{us}^c, & else, \end{cases} \quad (3)$$

where the $p_{us}^c = \frac{\exp(z_{us}^c)}{\sum_{n=1}^{C}\exp(z_{us}^n)}$ is the predicted probability for the c-th class computed by softmax. According to the gradient descent algorithm [37], only the prediction for the pseudo label category ($\overline{y_{us}^c} = 1$) is optimized to increase, and the predictions for other categories ($\overline{y_{us}^c} = 0$) are optimized to decrease. This means that once the pseudo-label is assigned incorrectly, the training of the SSL model will be misled since the prediction of ground truth is suppressed.

Algorithm 1 K value selection strategy

Input: sorted prediction $\mathbf{p} = (p^1, p^2, ..., p^C)$
Output: K value
Initialize: cumulative probability upper bound T, category numbers C, cumulative probability $\boldsymbol{V} = \boldsymbol{p}$
Compute cumulative probability:
for $n = 1$ **to** C **do**
  if $V^n > T$ **or** $n = C$ **then**
    return n
  end if
  $V^{n+1} = V^n + p^{n+1}$
end for
Determine K value:
$K = \max(n-1, 1)$
Return K

To reduce interference from wrong pseudo labels, we propose an FPL to exploit informative semantics from unlabeled data via multiple fuzzy positive labels, as shown in Fig. 2. Concretely, in Sec. 3.2, we propose a fuzzy positive assignment (FPA) algorithm, which assigns the top-K predicted categories of each pixel as its fuzzy positive labels, where K is computed according to our elaborate K value selection strategy. In Sec. 3.3, we develop a fuzzy positive regularization (FPR), which enables our model to exploit the possible ground truth in the fuzzy positive label set by regularizing the predictions of fuzzy positive categories to be larger than the rest negative categories.

3.2. Fuzzy Positive Assignment

The assignment of fuzzy positive labels determines from which our FPL exploits the semantics of ground truth. To provide an adaptive number of labels for each pixel, we first propose to choose the categories with top-K predicted probabilities as fuzzy positive labels since high-confidence predictions are prone to be correct [3]. We then design an easy but effective K value selection strategy to adaptively determine the K value for each pixel, as shown in Alg. 1. Specifically, we set a hyperparameter T that represents the upper bound of cumulative probability. For each pixel, we compute the cumulative probability of its top-n predicted categories and record the value of n where the cumulative probability exceeds T for the first time. Finally, the K value for this pixel is set as $\max(n-1, 1)$.

Selecting top-n predicted categories whose cumulative probability exceeds T guarantees that the ground truth has a high probability of being selected. A counter-intuitive design in our Alg. 1 is choosing $K = n-1$ instead of $K = n$. This is because setting $K < n$ alleviates the gradient vanishing problem in training our FPL (cf. Appendix). Another noteworthy point is that our algorithm provides $K = 1$ for pixels with high confidence, while $K > 1$ are usually sup-

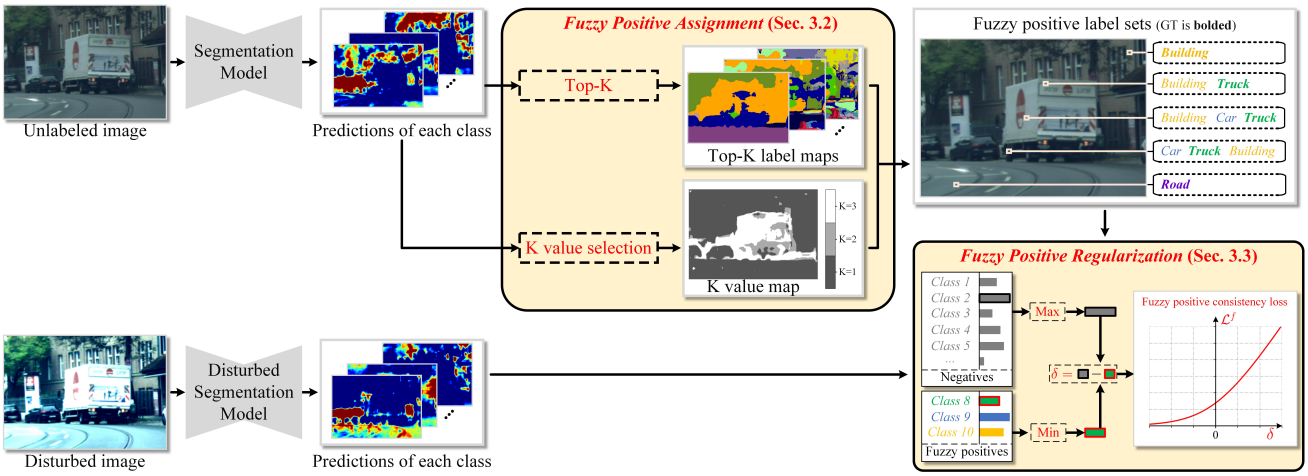

Figure 2. **Pipeline illustration of our FPL**, where FPA densely allocates multiple labels as a fuzzy positive label set for each pixel, while FPR enforces the discrimination of the fuzzy positive assigns with the rest negative labels to facilitate more reliable semantic generalization.

plied for uncertain pixels, as illustrated in Fig. 4 and Fig. 5. This property is in line with semantic intuition because a certain pixel should learn an explicit label, while an uncertain pixel needs to learn from multiple fuzzy labels. The ablation study about the K value selection is in Appendix.

3.3. Fuzzy Positive Regularization

In our FPA, we generate a fuzzy positive label set $\mathbb{Y}_{us} = \{y^1_{us}, y^2_{us}, ..., y^K_{us}\}$ that contains K labels for each unlabeled pixel instead of only one pseudo label as in previous works. Hence we need to propose a new loss function to learn the possible ground truth from $\mathbb{Y}_{us}$.

Our FPL regards all categories in the fuzzy positive label set $\mathbb{Y}_{us}$ are probable to be the ground truth, but the categories outside the $\mathbb{Y}_{us}$ are unlikely to be the ground truth. Therefore, we hope that the predictions of our model for the K fuzzy positive categories to be larger than the predictions for the rest $C - K$ negative categories. We refer to some works in metric learning [25, 39, 41, 42] and formulate our optimization objective for each pixel as:

$$\min_{i \in \mathbb{Y}_{us}} (z^i_{us}) \quad > \quad \max_{j \notin \mathbb{Y}_{us}} (z^j_{us}), \tag{4}$$

where z^i_{us} represents the prediction of our model for the i-th category. Eq. 4 means we regularize the minimum of the predictions for categories in $\mathbb{Y}_{us}$ to be larger than the maximum of the predictions for other categories. In other words, we enforce all the predictions for fuzzy positive categories to be larger than those for negative categories. From Eq. 4, a straightforward loss function can be formulated as:

$$\mathcal{L}^f_{us} = ReLU(\max_{j \notin \mathbb{Y}_{us}} (z^j_{us}) - \min_{i \in \mathbb{Y}_{us}} (z^i_{us})). \tag{5}$$

However, this $\mathcal{L}^f_{us}$ is globally non-differentiable with respect to $\boldsymbol{z}_{us} = \{z^1_{us}, z^2_{us}, ..., z^C_{us}\}$ since the max and min functions in Eq. 5 are globally non-differentiable [27, 36]. And the $ReLU$ function also has a singularity at $x = 0$. Thanks to existing functional approximations [7, 12, 27, 32], we approximate the Eq. 5 to make $\mathcal{L}^f_{us}$ differentiable:

$$\begin{aligned} \max(z^1, z^2, ..., z^n) &\approx \log(\sum_{i=1}^{n} \exp(z^i)) \\ \min(z^1, z^2, ..., z^n) &\approx -\log(\sum_{i=1}^{n} \exp(-z^i)) \\ ReLU(z) = \max(z, 0) &\approx \log(1 + \exp(z)). \end{aligned} \tag{6}$$

Based on these functional approximations, our fuzzy positive consistency loss $\mathcal{L}^f$ for one pixel x_{us} (i.e., the s-th pixels of the u-th unlabeled image) could be converted to:

$$\mathcal{L}^f_{us} = \log(1 + \sum_{i \in \mathbb{Y}_{us}} e^{-z^i_{us}} \times \sum_{j \notin \mathbb{Y}_{us}} e^{z^j_{us}}). \tag{7}$$

Next, we analyze the behavior of $\mathcal{L}^f_{us}$ in backpropagation. The gradient of $\mathcal{L}^f_{us}$ with respect to the prediction $\boldsymbol{z}_{us}$ of our model is computed as:

$$\begin{aligned} \frac{\partial \mathcal{L}^f_{us}}{\partial z^i_{us}} &= \frac{-\sum_{j \notin \mathbb{Y}_{us}} e^{z^j_{us}} \times e^{-z^i_{us}}}{1 + \sum_{j \notin \mathbb{Y}_{us}} e^{z^j_{us}} \times \sum_{i \in \mathbb{Y}_{us}} e^{-z^i_{us}}}, i \in \mathbb{Y}_{us} \\ \frac{\partial \mathcal{L}^f_{us}}{\partial z^j_{us}} &= \frac{\sum_{i \in \mathbb{Y}_{us}} e^{-z^i_{us}} \times e^{z^j_{us}}}{1 + \sum_{j \notin \mathbb{Y}_{us}} e^{z^j_{us}} \times \sum_{i \in \mathbb{Y}_{us}} e^{-z^i_{us}}}, j \notin \mathbb{Y}_{us}, \end{aligned} \tag{8}$$

where the $\frac{\partial \mathcal{L}^f_{us}}{\partial z^i_{us}}$ and $\frac{\partial \mathcal{L}^f_{us}}{\partial z^j_{us}}$ denote the derivatives with respect to predictions for fuzzy positive categories and other negative categories, respectively. From Eq. 7 and Eq. 8, we see that our $\mathcal{L}^f_{us}$ has following characteristics:

1) The prediction for the ground truth increases when it appears in $\mathbb{Y}_{us}$. This is because predictions for fuzzy posi-

tive categories have gradients less than 0, and thus are optimized to increase by gradient descent.

2) The existing $\mathcal{L}_{us}^{v}$ is a special case of our $\mathcal{L}_{us}^{f}$ when we set $K = 1$, as shown in Eq. 9.

$$\mathcal{L}_{us}^{v} = \log(1 + e^{-z_{us}^{i}} \times \sum_{j \neq i} e^{z_{us}^{j}}), \tag{9}$$

where i is the index of the top-1 predicted pseudo label.

Adaptive weight for each pixel: From Eq. 4, it can be seen that our model learns informative semantics based on the assumption that the ground truth exists in the fuzzy positive label set $\mathbb{Y}_{us}$. Thus, we propose to integrate the confidence of this assumption into the training of FPL. When our assumption is not tenable, the ground truth will be outside $\mathbb{Y}_{us}$, and its largest predicted probability is $\max_{j \notin \mathbb{Y}_{us}}(p_{us}^{j})$. Therefore, the $\max_{j \notin \mathbb{Y}_{us}}(p_{us}^{j})$ is negatively correlated with the assumption confidence since high $\max_{j \notin \mathbb{Y}_{us}}(p_{us}^{j})$ means ground truth has a low probability inside $\mathbb{Y}_{us}$, and vice versa.

Formulately, the range of $\max_{j \notin \mathbb{Y}_{us}}(p_{us}^{j})$ is derived as:

$$\frac{1-T}{C-K_{us}} < \max_{j \notin \mathbb{Y}_{us}}(p_{us}^{j}) < \frac{\sum_{i \in \mathbb{Y}_{us}} p_{us}^{i}}{K_{us}}. \tag{10}$$

In practice, T is close to 1 (e.g., 0.9), thus $\frac{1-T}{C-K_{us}}$ is close to 0. For simplicity, we obtain the approximate range of $\max_{j \notin \mathbb{Y}_{us}}(p_{us}^{j})$ as $(0, \frac{\sum_{i \in \mathbb{Y}_{us}} p_{us}^{i}}{K_{us}})$. We then define our adaptive weight as a monotonically decreasing concave function:

$$w_{us} = \frac{\log\left[1 + A \times \left(\frac{\sum_{i \in \mathbb{Y}_{us}} p_{us}^{i}}{K_{us}} - \max_{j \notin \mathbb{Y}_{us}}(p_{us}^{j})\right)\right]}{\log\left[1 + A \times \left(\frac{\sum_{i \in \mathbb{Y}_{us}} p_{us}^{i}}{K_{us}}\right)\right]}, \tag{11}$$

where A is a scalar used to control the radian of this function, which is fixed as 50. It is worth noting that our adaptive weight is different from the weights computed by top-1 confidence used to filter out or re-weight low-confidence pixels [11, 17, 34]. Those weights are small for pixels with low top-1 probability, resulting in those pixels not being sufficiently used in training [43]. But our weight is only small when the prediction of a pixel is confused in the top-(K+1) categories, thus our model still uses the information that its prediction should not belong to other C-K-1 categories.

3.4. Analysis

Ideally, we hope to learn the semantics of ground truth in unlabeled data, but in practice, we can only learn the semantics of positive categories and suppress the rest. Here, we propose a *positive gradient score* R to measure how properly the ground truth is learned :

$$R_{us} = \frac{\partial \mathcal{L}_{us}}{\partial z_{us}^{gt}} \Big/ \sum_{i \in Y_{us}} \frac{\partial \mathcal{L}_{us}}{\partial z_{us}^{i}}, \tag{12}$$

where the Y_{us} represents the fuzzy positive label set $\mathbb{Y}_{us}$ when $\mathcal{L}_{us}$ is $\mathcal{L}_{us}^{f}$, and Y_{us} represents the pseudo label when $\mathcal{L}_{us}$ is $\mathcal{L}_{us}^{v}$. The positive gradient score R_{us} is the ratio of the gradient for the ground truth to the sum of the gradients for all positive categories. It ranges from $[-1, 1]$ and a positive R_{us} means the GT prediction is encouraged to increase, while a negative R_{us} means the GT prediction is incorrectly suppressed to decrease. Based on actual training, we consider R_{us} in three cases:

Case 1. The pseudo label is correct, that is, the ground truth is the top-1 predicted category. In this case, the positive gradient score R_{us} computed by $\mathcal{L}_{us}^{v}$ and $\mathcal{L}_{us}^{f}$ are:

$$R_{us}^{v} = \frac{p_{us}^{gt} - 1}{p_{us}^{pse} - 1} = 1, \quad R_{us}^{f} = \frac{e^{-z_{us}^{gt}}}{\sum_{i \in \mathbb{Y}_{us}} e^{-z_{us}^{i}}} \in [0, 1], \tag{13}$$

where p_{us}^{gt} and p_{us}^{pse} are the predicted probabilities for ground truth and the pseudo-label category. When the size of $\mathbb{Y}_{us}$ (i.e., K value) is 1, the R_{us}^{f} will be equal to R_{us}^{v} as 1. We see that R_{us}^{f} and R_{us}^{v} are both greater than 0, meaning they both encourage the GT prediction to increase. In practice, the statistics of R_{us}^{f} is close to 1. This is because most pixels in this case have $K = 1$ (cf. Appendix).

Case 2. The top-1 prediction is wrong, but the ground truth is in the categories with top-K probabilities, where K is computed by our K value selection strategy in Alg. 1. For Case 2, the positive gradient score R_{us}^{v} and R_{us}^{f} are computed as:

$$R_{us}^{v} = \frac{p_{us}^{gt}}{p_{us}^{pse} - 1} \in [-1, 0], \quad R_{us}^{f} = \frac{e^{-z_{us}^{gt}}}{\sum_{i \in \mathbb{Y}_{us}} e^{-z_{us}^{i}}} \in [0, 1]. \tag{14}$$

We see that R_{us}^{f} is larger than 0 while R_{us}^{v} is less than 0. This is because the ground truth is missed by the pseudo label but captured by our fuzzy positive label set. It means that vanilla $\mathcal{L}_{us}^{v}$ erroneously suppresses GT prediction, but our $\mathcal{L}_{us}^{f}$ encourages GT prediction, reflecting FPL remarkably reduces the interference from wrong pseudo labels.

Case 3. The pseudo label is wrong, and the ground truth is also outside the fuzzy positive labels $\mathbb{Y}_{us}$. In this case, the positive gradient score R_{us}^{v} and R_{us}^{f} are:

$$R_{us}^{v} = \frac{p_{us}^{gt}}{p_{us}^{pse} - 1} \in [-1, 0], \quad R_{us}^{f} = \frac{-e^{z_{us}^{gt}}}{\sum_{j \notin \mathbb{Y}_{us}} e^{z_{us}^{j}}} \in [-1, 0]. \tag{15}$$

It is obvious that R_{us}^{f} and R_{us}^{v} are both less than 0, meaning neither $\mathcal{L}_{us}^{v}$ nor $\mathcal{L}_{us}^{f}$ is beneficial for learning the semantics of ground truth in this case. In Fig. 3 (a), we display some examples which intuitively reflect the advantages of R_{us}^{f} over R_{us}^{v}. That is, many parts of R_{us}^{v} less than 0 (colored in blue) becomes larger than 0 in R_{us}^{f} (colored in red). In Fig. 3 (b), the statistics of positive gradient score show R_{us}^{f} significantly outperforms the existing R_{us}^{v} in Case 2, and they perform similarly in Case 1 and Case 3.

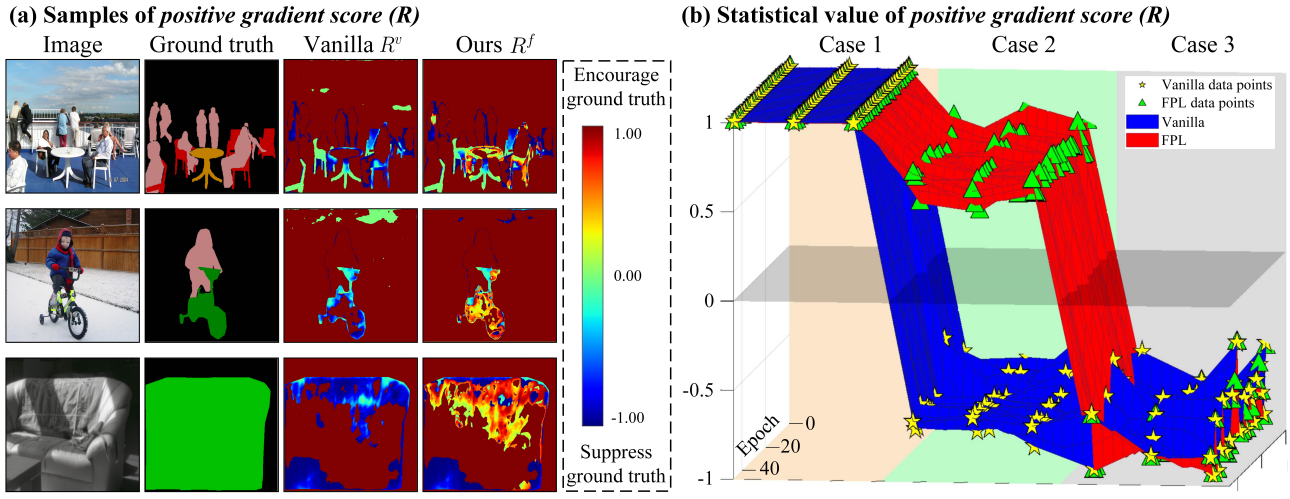

Figure 3. **Positive gradient score *R*.** (a) shows the positive gradient score maps of some unlabeled examples, where the red color means the prediction of ground truth is encouraged, while the blue color indicates suppression. (b) is the statistics value of the positive gradient score in three cases (Sec. 3.4). This figure is plotted on VOC2012 with 1/16 labeled data.

Method	ResNet 50				ResNet 101			
	1/32 (93)	1/16 (186)	1/8 (372)	1/4 (744)	1/32 (93)	1/16 (186)	1/8 (372)	1/4 (744)
MT [40]	-	66.14	72.03	74.47	-	68.08	73.71	76.53
CCT [34]	-	66.35	72.46	75.68	-	69.64	74.48	76.35
GCT [17]	-	65.81	71.33	75.30	-	66.90	72.96	76.45
U^2PL [43]	-	-	-	-	-	74.90	76.48	78.51
CPS w/o cutmix† [5]	54.40	68.68	73.06	75.75	59.70	71.22	74.98	77.45
FPL+CPS w/o cutmix	55.77(↑1.37)	69.71(↑1.03)	74.43(↑1.37)	76.76(↑1.01)	61.00(↑1.30)	72.05(↑0.83)	75.67(↑0.69)	77.57(↑0.12)
CPS w/ cutmix† [5]	71.33	74.05	76.92	77.77	72.51	74.72	77.62	78.93
FPL+CPS w/ cutmix	**72.39**(↑1.06)	**74.80**(↑0.75)	**77.32**(↑0.40)	**78.53**(↑0.76)	73.20(↑0.69)	75.74(↑1.02)	**78.47**(↑0.85)	**79.19**(↑0.26)
AEL† [14]	68.39	74.03	75.83	76.18	73.00	75.26	78.07	78.26
FPL+AEL	71.21(↑2.82)	74.54(↑0.51)	76.25(↑0.42)	76.88(↑0.70)	**75.01**(↑2.01)	**76.58**(↑1.32)	78.19(↑0.12)	78.46(↑0.20)

Table 1. **The mIoU on Cityscapes.** Results marked by † are reproduced in the same experimental environment as FPL.

4. Experiments

4.1. Implementation Details

Frameworks and dataset: We evaluate the effectiveness of our FPL on two widely used frameworks, CPS [5] and AEL [14], and two datasets *PASCAL VOC 2012* and *Cityscapes*. The Cityscapes is a large-scale dataset designed for urban street scene segmentation which consists of 19 semantic classes containing 2,975 images for training, 500 for validation, and 1,525 for testing. The PASCAL VOC 2012 is a generic object segmentation benchmark that consists of 20 object classes and 1 background class. It is divided into training, validation, and test sets including 1,464, 1,449, and 1,456 images, respectively. There is also an augmented set [13] adding 10,582 images into the standard training set. Following the setting of previous works [5,53], we implement two splits on VOC2012: standard split (with augmented set) and low data split (without augmented set).

Experimental setting: Following the default settings of CPS and AEL, we use Deeplab v3+ with pre-trained ResNet-50 and ResNet-101 as backbones. Specifically, on Cityscapes using CPS as the baseline, we use SGD optimizer with a weight decay of 1e-4. The initial learning rate is set to 0.02 and the momentum is fixed at 0.9. We use the default 'poly' learning rate decay policy to scale the learning rate by $(1 - iter/max\ iter)^{0.9}$, and this policy is used in all our experiments. The input images are cropped to 800×800 and the batchsize is 64. When using AEL as the baseline, the batchsize, learning rate, and image size are changed to 16, 0.01, and 769. On VOC2012 using CPS as the baseline, we use SGD optimizer with a weight decay of 1e-4. The initial learning rate is set to 0.01 and the momentum is fixed at 0.9. The input images are cropped to 512×512 and the batchsize is 32. When using AEL as the baseline, the batchsize is changed to 16. The cumulative probability upper bound T in all our experiments is set from $\{0.95, 0.9, 0.85\}$. More details are in Appendix.

4.2. Quantitative Results

Our FPL model is trained with the same hyperparameters as the baseline model, only replacing the vanilla positive learning using one pseudo label with our fuzzy posi-

Method	ResNet 50			ResNet 101		
	1/16 (662)	1/8 (1323)	1/4 (2646)	1/16 (662)	1/8 (1323)	1/4 (2646)
MT [40]	66.77	70.78	73.22	70.59	73.20	76.62
CCT [34]	65.22	70.87	73.43	67.94	73.00	76.17
CutMix-Seg [11]	68.90	70.70	72.46	72.56	72.69	74.25
GCT [17]	64.05	70.47	73.45	69.77	73.30	75.25
CAC [21]	70.10	72.40	74.00	72.40	74.60	76.30
CPS w/o cutmix† [5]	68.13	72.79	74.24	72.50	74.97	77.14
FPL+CPS w/o cutmix	68.67(↑0.54)	73.03(↑0.36)	74.80(↑0.56)	73.18(↑0.68)	75.74(↑0.77)	77.47(↑0.33)
CPS w/ cutmix† [5]	71.78	73.44	74.90	74.48	76.44	77.68
FPL+CPS w/ cutmix	**72.52**(↑0.74)	**73.74**(↑0.30)	75.35(↑0.45)	74.98(↑0.50)	**77.75**(↑1.31)	78.30(↑0.62)
AEL† [14]	69.93	73.17	75.50	74.20	76.58	77.98
FPL+AEL	71.01(↑1.08)	73.69(↑0.52)	**76.61**(↑1.11)	**74.98**(↑0.78)	76.73(↑0.15)	**78.35**(↑0.37)

Table 2. **The mIoU on VOC2012.** Results marked by † are reproduced in the same experimental environment as FPL.

Method	1/16 (92)	1/8 (183)	1/4 (366)	1/2 (732)
AdvSemSeg [15]	39.69	47.58	59.97	65.27
CCT [34]	33.10	47.60	58.80	62.10
VAT [31]	36.92	49.35	56.88	63.34
MT [40]	48.70	55.81	63.01	69.16
GCT [17]	46.04	54.98	64.71	70.67
CutMix-Seg [11]	52.16	63.47	69.46	73.73
PseusoSeg [53]	57.60	65.50	69.14	72.41
PC^2Seg [51]	57.00	66.28	69.78	73.05
U^2PL [43]	67.98	69.15	73.66	76.16
CPS w/ cm† [5]	67.53	70.41	75.27	78.69
FPL+CPS w/ cm	**69.30**(↑1.77)	**71.72**(↑1.31)	**75.73**(↑0.46)	**78.95**(↑0.26)

Table 3. **The mIoU on VOC2012 LowData**. Results marked by † are reproduced in the same experimental environment as FPL. The 'cm' is the cutmix.

tive learning using multiple fuzzy positive labels. The segmentation results on Cityscapes, VOC2012, and VOC2012 LowData are presented in Table 1, Table 2, and Table 3, where red numbers represent the improvement brought by FPL to the baseline. We see that FPL achieves stable improvements over baseline models across all data splits. Besides, FPL improves the CPS baseline under both with and without CutMix settings, indicating that the performance gain from FPL and data augmentation (e.g., CutMix) can be accumulated. Furthermore, FPL is effective on multiple baselines, i.e., CPS and AEL, which means FPL is universal for various existing SSL frameworks.

4.3. Empirical Study

4.3.1 The Hyperparameter T

The T is the only new hyperparameter brought by FPL, which controls the K values of pixels in training. Here we summarize two rules for setting a proper T value. First, a T value around 0.9 (e.g., 0.85, 0.9, 0.95) is usually a promising setting. Second, a T value set negatively correlated to the number of labeled data usually brings high performance.

The effect of T on the training behaviors. In training, T affects the number of fuzzy positive labels for each pixel (K value), which reflects the degree of fuzziness of our FPL. Specifically, a higher T leads to large K values meaning more labels will be selected as candidates, thus the ground truth will be captured with a higher probability. However, too large T value (e.g. 0.99) makes our model learn from too many labels, which is also not suitable for a single-label classification task. In Table 4, we present the mIoU of our FPL models trained with various T values, and we find a T value around 0.9 always provides promising results.

T value	0.5	0.75	0.85	0.9	0.95	0.99
mIoU	68.80	68.97	69.34	**69.71**	69.08	67.52

Table 4. **The performances of FPL models with various T.** These results are obtained on Cityscapes with 1/16 labeled data using CPS as the baseline.

T	1/32	1/16	1/8
0.85	55.22 (↑0.90)	69.34 (↑0.66)	74.37 (↑1.31)
0.9	55.40 (↑1.08)	69.71 (↑**1.03**)	74.43 (↑**1.37**)
0.95	55.77 (↑**1.45**)	69.08 (↑0.40)	74.03 (↑0.97)

Table 5. **The relationship between cumulative probability upper bound T and the amount of labeled data.** The results are obtained on Cityscapes using CPS as the baseline.

Weight functions	Baseline	Convex	Linear	Concave	w/o weight
mIoU	76.44	76.68	77.02	**77.75**	77.01

Table 6. FPL+CPS w/ cutMix on VOC2012 with 1/8 labels. The 'w/o' represents the FPL model trained without adaptive weight.

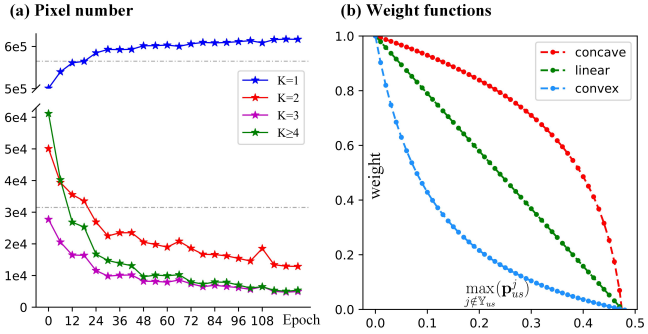

Figure 4. (a) The K values during training, where the input size is 800×800 meaning there are 640000 pixels in total. (b) The adaptive weights plotted in the setting of $T = 0.95$ and $K = 2$.

The relationship between T and the amount of la-

beled data. We find a high T usually obtains good performance when labeled data are limited, while a low T usually performs better when labeled data are sufficient. As shown in Table 5, in the 1/32 labeled data setting, $T = 0.95$ obtains the highest improvement about 1.45%, while $T = 0.85$ and $T = 0.9$ only obtain improvements about 1%. In the 1/16 labeled data setting, $T = 0.9$ obtains the best performance, improving baseline by 1.03%, and the rest two T values improve baseline by about 0.5%. In the 1/8 labeled data setting, $T = 0.9$ and $T = 0.85$ obtains close performances which improve baseline by 1.3%, while $T = 0.95$ performs not as well as the previous two T settings. It is obvious that setting the T value negatively according to the amount of labeled data significantly benefits the performance.

4.3.2 Adaptive Weight

In Sec. 3.3, we show that the adaptive weight function should be inversely proportional to $\max_{j \notin \mathbb{Y}_{us}}(p_{us}^j)$. Here we provide an experiment showing that the used concave decreasing function performs better than linear or convex decreasing functions. The function curves are illustrated in Fig. 4 (b), and the formulas of convex and linear functions are expressed as:

$$
\begin{aligned}
w_{convex} &= \frac{\frac{\sum_{i \in \mathbb{Y}_{us}} p_{us}^i}{K_{us}} - \max_{j \notin \mathbb{Y}_{us}}(p_{us}^j)}{\frac{\sum_{i \in \mathbb{Y}_{us}} p_{us}^i}{K_{us}} + 4 * \max_{j \notin \mathbb{Y}_{us}}(p_{us}^j)}, \\
w_{linear} &= \frac{\frac{\sum_{i \in \mathbb{Y}_{us}} p_{us}^i}{K_{us}} - \max_{j \notin \mathbb{Y}_{us}}(p_{us}^j)}{\frac{\sum_{i \in \mathbb{Y}_{us}} p_{us}^i}{K_{us}}}.
\end{aligned}
\tag{16}
$$

The segmentation performances are shown in Table 6, where we see that the used convex function performs better than other alternatives.

4.3.3 K Values in Training

The number of pixels with different K values is shown in Fig. 4 (a). We see that within training, the number of pixels with $K > 1$ decreases and the number of pixels with $K = 1$ increases. At the late stage of training, the K values for more than 93.75% (i.e., 6e5 / 6.4e5) pixels are 1. This indicates the K values automatically converge to 1, meaning FPL could progressively achieve clear pixel-level semantic discrimination. In Fig. 4 (b), we illustrate that our FPL provides $K = 1$ for certain pixels with low entropy while providing $K > 1$ for uncertain pixels with high entropy.

Moreover, we show the K value maps of some examples during training in Fig. 5. We see that the K values of most pixels in the background are 1 since background pixels are usually easy to classify. In the early stage of training, the pixels with $K > 1$ are mainly located on objects, since the classification of objects for our model is uncertain at early training. As the training progresses, the number of pixels with $K > 1$ gradually decreases and these pixels are mainly located at the boundary of objects. This is because our model has certain predictions for most pixels in the later stage of training. But for pixels located at the object boundary, their categories are fuzzy, for which our model makes uncertain predictions for them. Our FPL provides multiple labels (i.e., $K > 1$) for these uncertain pixels to learn, which is in line with their fuzzy property.

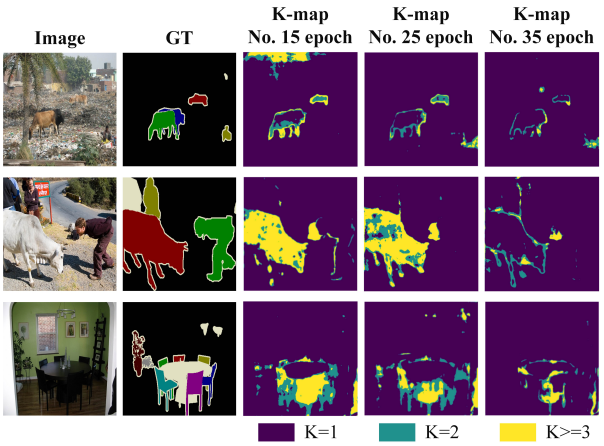

Figure 5. **K value visualization,** plotted on the VOC2012 with 1/16 labeled data using CPS+FPL w/ CutMix.

5. Conclusion

In this paper, we introduce a novel plug-and-play method named FPL for semi-supervised semantic segmentation. Our method is the first to explore learning the semantics of ground truth from multiple fuzzy positive labels. Specifically, We first propose a fuzzy positive assignment algorithm to provide an adaptive number of labels for each pixel. We then develop a fuzzy positive regularization to learn the possible ground truth from these fuzzy positive labels. Extensive experiments on two commonly used benchmarks with consistent performance gain demonstrate the effectiveness of our method. Moreover, we provide an analysis showing the superiority of FPL in that it revises the gradient of learning ground truth when pseudo labels are wrong. There are still directions worth continuing to explore in FPL, e.g., "extending discrete K values to continuous form for finer-grained fuzzy positive labels."

Acknowledgements. This work was supported in part by the National Key R&D Program of China (No. 2022ZD0118201) Natural Science Foundation of China (No. 61972217, 32071459, 62176249, 62006133, 62271465), and the Natural Science Foundation of Guangdong Province in China (No. 2019B1515120049).

References

[1] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *ICCV*, pages 9297–9307, 2019. 1

[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. *ICLR*, 2019. 2

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

[4] Liang-Chieh Chen, Raphael Gontijo Lopes, Bowen Cheng, Maxwell D Collins, Ekin D Cubuk, Barret Zoph, Hartwig Adam, and Jonathon Shlens. Naive-student: Leveraging semi-supervised learning in video sequences for urban scene segmentation. In *ECCV*, pages 695–714. Springer, 2020. 2

[5] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *CVPR*, pages 2613–2622, 2021. 1, 2, 3, 6, 7

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, pages 3213–3223, 2016. 1

[7] Charles Dugas, Yoshua Bengio, François Bélisle, Claude Nadeau, and René Garcia. Incorporating second-order functional knowledge for better option pricing. *NeurIPS*, pages 472–478, 2001. 4

[8] Jiashuo Fan, Bin Gao, Huan Jin, and Lihui Jiang. Ucc: Uncertainty guided cross-head co-training for semi-supervised semantic segmentation. In *CVPR*, pages 9947–9956, 2022. 2, 3

[9] 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*, 1(2):5, 2020. 2

[10] 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. *Pattern Recognition*, page 108777, 2022. 2

[11] Geoff French, Timo Aila, Samuli Laine, Michal Mackiewicz, and Graham Finlayson. Semi-supervised semantic segmentation needs strong, high-dimensional perturbations. In *BMVC*, 2020. 2, 5, 7

[12] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *AISTATS*, pages 315–323, 2011. 4

[13] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, pages 991–998. IEEE, 2011. 6

[14] Hanzhe Hu, Fangyun Wei, Han Hu, Qiwei Ye, Jinshi Cui, and Liwei Wang. Semi-supervised semantic segmentation via adaptive equalization learning. *NeurIPS*, 34, 2021. 2, 3, 6, 7

[15] 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. 7

[16] Mostafa S Ibrahim, Arash Vahdat, Mani Ranjbar, and William G Macready. Semi-supervised semantic image segmentation with self-correcting networks. In *CVPR*, pages 12715–12725, 2020. 2

[17] Zhanghan Ke, Di Qiu, Kaican Li, Qiong Yan, and Rynson WH Lau. Guided collaborative training for pixel-wise semi-supervised learning. In *ECCV*, pages 429–445. Springer, 2020. 2, 5, 6, 7

[18] 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 *ICCV*, pages 6728–6736, 2019. 2

[19] Jongmok Kim, Jooyoung Jang, and Hyunwoo Park. Structured consistency loss for semi-supervised semantic segmentation. *arXiv preprint arXiv:2001.04647*, 2020. 2

[20] Donghyeon Kwon and Suha Kwak. Semi-supervised semantic segmentation with error localization network. In *CVPR*, pages 9957–9967, 2022. 2, 3

[21] 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*, pages 1205–1214, 2021. 2, 7

[22] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *ICLR*, 2017. 3

[23] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. *ICMLW*, 3(2):896, 2013. 2

[24] Yu-Feng Li, Han-Wen Zha, and Zhi-Hua Zhou. Learning safe prediction for semi-supervised regression. In *AAAI*, volume 31, 2017. 2

[25] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, pages 212–220, 2017. 4

[26] Yuyuan Liu, Yu Tian, Yuanhong Chen, Fengbei Liu, Vasileios Belagiannis, and Gustavo Carneiro. Perturbed and strict mean teachers for semi-supervised semantic segmentation. In *CVPR*, pages 4258–4267, 2022. 2, 3

[27] Richard McElreath. *Statistical rethinking: A Bayesian course with examples in R and Stan*. Chapman and Hall/CRC, 2018. 2, 4

[28] Robert Mendel, Luis Antonio De Souza, David Rauber, João Paulo Papa, and Christoph Palm. Semi-supervised segmentation based on error-correcting supervision. In *ECCV*, pages 141–157. Springer, 2020. 2

[29] Shervin Minaee, Yuri Y Boykov, Fatih Porikli, Antonio J Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE TPAMI*, 2021. 1

[30] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high-and low-level consistency. *IEEE TPAMI*, 2019. 2

[31] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE TPAMI*, 41(8):1979–1993, 2018. 2, 7

[32] Frank Nielsen and Ke Sun. Guaranteed bounds on information-theoretic measures of univariate mixtures using piecewise log-sum-exp inequalities. *Differential Geometrical Theory of Statistics*, 18(442):287, 2017. 4

[33] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D Cubuk, and Ian J Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *NeurIPS*, pages 3239–3250, 2018. 2

[34] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *CVPR*, pages 12674–12684, 2020. 1, 2, 3, 5, 6, 7

[35] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V Le. Meta pseudo labels. In *CVPR*, pages 11557–11568, 2021. 2

[36] János D Pintér. Globally optimized spherical point arrangements: model variants and illustrative results. *Annals of Operations Research*, 104(1):213–230, 2001. 4

[37] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985. 3

[38] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*, 33, 2020. 2

[39] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *CVPR*, pages 6398–6407, 2020. 4

[40] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, pages 1195–1204, 2017. 2, 6, 7

[41] Feng Wang, Jian Cheng, Weiyang Liu, and Haijun Liu. Additive margin softmax for face verification. *IEEE SPL*, 25(7):926–930, 2018. 4

[42] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *CVPR*, pages 5265–5274, 2018. 4

[43] 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*, pages 4248–4257, 2022. 1, 2, 5, 6, 7

[44] Chen Wei, Kihyuk Sohn, Clayton Mellina, Alan Yuille, and Fan Yang. Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. In *CVPR*, pages 10857–10866, 2021. 2

[45] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, pages 418–434, 2018. 1

[46] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *NeurIPS*, 2019. 2

[47] Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Yu-Feng Li, Baigui Sun, Hao Li, and Rong Jin. Dash: Semi-supervised learning with dynamic thresholding. In *ICML*, pages 11525–11536, 2021. 2

[48] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. St++: Make self-training work better for semi-supervised semantic segmentation. In *CVPR*, pages 4268–4277, 2022. 2

[49] Jianlong Yuan, Yifan Liu, Chunhua Shen, Zhibin Wang, and Hao Li. A simple baseline for semi-supervised semantic segmentation with strong data augmentation. *ICCV*, 2021. 1, 2, 3

[50] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *NeurIPS*, 34, 2021. 2

[51] Yuanyi Zhong, Bodi Yuan, Hong Wu, Zhiqiang Yuan, Jian Peng, and Yu-Xiong Wang. Pixel contrastive-consistent semi-supervised semantic segmentation. *ICCV*, 2021. 1, 2, 7

[52] Yi Zhu, Zhongyue Zhang, Chongruo Wu, Zhi Zhang, Tong He, Hang Zhang, R Manmatha, Mu Li, and Alexander J Smola. Improving semantic segmentation via efficient self-training. *IEEE TPAMI*, 2021. 2

[53] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. Pseudoseg: Designing pseudo labels for semantic segmentation. In *ICLR*, 2020. 1, 2, 3, 6, 7