This WACV 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.

Interactive Network Perturbation between Teacher and Students for Semi-Supervised Semantic Segmentation

Hyuna Cho, Injun Choi, Suha Kwak, Won Hwa Kim
Pohang University of Science and Technology (POSTECH)
{hyunacho, surung9898, suha.kwak, wonhwa}@postech.ac.kr

Abstract

The current golden standard of semi-supervised semantic segmentation is to generate and exploit pseudo-supervision on unlabeled images. This approach is however susceptible to the quality of pseudo-supervision—training often becomes unstable particularly at early stages and biased to incorrect supervision. To address these issues, we propose a new semi-supervised learning framework, dubbed Guided Pseudo Supervision (GPS). GPS comprises three networks, i.e., a teacher and two separate students. The teacher is first trained with a small set of labeled data and provides stable initial pseudo-supervision on the unlabeled data to the students. The students interactively train each other under the supervision of the teacher, and once they are sufficiently trained, they offer feedback supervision to the teacher so that the teacher improves in subsequent iterations. This strategy enables more stable and faster convergence than previous works, and consequently, GPS achieved state-of-the-art performance on Pascal VOC 2012 and Cityscapes datasets in various experiment settings.

1. Introduction

Semantic segmentation [32, 33] aims to identify semantically meaningful regions (e.g., objects) in a given image. This is often done by performing pixel-wise classification, where the objective is to determine to which class each pixel belongs to over the entire image. Training such a model requires a dataset of densely annotated images, as precise pixel-wise supervisions over multiple objects (including background) of each image are needed [1, 4, 9]. Exhaustive human labor is sacrificed to embed human perception of the objects into the annotation, and the problem becomes even worse when a large-scale dataset is curated to train deep and complex models.

The issue of exhaustive labeling is well-addressed in various label-efficient strategies for semantic segmentation, where the segmentation approaches use less expensive annotations to learn class-specific patterns at each pixel. While weakly supervised segmentation uses image-level or coarsely annotated labels as a weak supervision to guide segmentation [11, 28] and domain adaptation methods use supervision from synthetic data [23, 39], semi-supervised segmentation still uses pixel-wise labels on the real images but only from a *small fraction* of the entire dataset. It uses a small set of labeled images (i.e., expensive data) together with a large set of unlabeled images (i.e., cheap data) to train a segmentation model, such that the knowledge from the labeled set are propagated to the unlabeled set to generate pseudo-labels and secure sufficient sample-size [22, 26].

Recent studies on semi-supervised semantic segmentation heavily rely on perturbation-based (i.e., consistency regularization) approaches [7, 17, 25, 43]. The cornerstone of consistency regularization is the smoothness assumption: the model predictions should be consistent even if realistic perturbations are applied on data points [14, 43] or network parameters [7, 17, 31]. To ensure the effect of consistency regularization, existing schemes require much more data than its given unlabeled images. Works done in [2, 3, 7, 13, 17, 30] take out labels of labeled images to extend the size of unlabeled dataset, and a prerequisite data augmentation is often used for generating a diverse perturbation [3, 25, 30, 43], resulting in an extra cost to secure a sufficient number of samples.

Specifically, the Cross Pseudo Supervision (CPS) [7] and its extended version n-CPS [13] representatively leverage the network perturbation mechanism. CPS consists of two differently initialized networks of the same architecture, and the pair of networks supervise each other by exchanging their output to evaluate their prediction. The n-CPS extended the CPS to multiple networks demonstrating that providing more pseudo-supervisions can potentially improve segmentation performance. Recently, Perturbed and Strict Mean Teachers (PS-MT) [25] also proposed a perturbation-based method with three networks by replacing the loss of the Mean Teacher [31] with a more strict confidence-weighted loss. Although these approaches have brought substantial improvements, notice that adopting multiple networks causes an exhaustive number of su-

pervision exchanges which may be computationally intractable. Also, to train a diverse set of models, they often use a variety of weak and strong image augmentations and feature perturbations at the cost of additional computation.

Therefore, it is critical to *efficiently* utilize multiple networks with a *strategical training* approach. In this paper, we propose a novel perturbation-based training strategy for semi-supervised semantic segmentation. Using a network perturbation, our method co-trains one teacher and two student networks in a data-efficient manner by clearly *separating the roles of labeled and unlabeled data*. Unlike other works [7, 13, 25, 27] where the labeled images are identically used to train all models within their frameworks with a supervised loss, we propose to use the labeled data (small but expensive) 1) to first train the teacher network such that it gains feasible knowledge to teach student networks and 2) as a validation benchmark to assess the current performance of the networks to selectively regulate the interaction between the teacher and student networks. With unlabeled data (large and inexpensive), consistency regularization is imposed to propagate knowledge from better networks to under-trained networks iteratively, either from the teacher to students or from the students to teacher, according to the validation from the labeled data.

Notice that, the student networks are very immature at the initial training stages. However, we expect that they will eventually become smarter than the initial teacher network as they supervise each other. Assessing the loss of the student and teacher networks via the labeled data computed during training, we gradually increase the influence of feedback from the student networks to the teacher network as the students become better. Once the students are trained as much as the teacher, they return a full feedback pseudo-supervision on the unlabeled data to perturb the teacher network on its unseen samples. The loss comparison between the differently trained networks enables adaptive adjustment of the effect of pseudo-supervision from the student networks to the teacher network. Also, the teacher network can avoid being overfitted on the precedent knowledge on the labeled set and be further improved with reliable feedback from students to consistently provide better guidance.

In this way, our training scheme strengthens both the teacher and student networks by interactively exchanging informative pseudo-labels. As our model does not require specialized augmentation, it is computationally efficient compared to [31] which must use various augmentation schemes to train multiple networks. In addition, existing methods [7, 13] train multiple networks in an identical way during the entire training, whereas our phased training scheme adaptively trains multiple networks by establishing different strategies depending on characteristics of the data.

Our proposed idea makes the following **contributions**:

- We propose a novel network perturbation method for semi-supervised semantic segmentation by clearly separating the "roles" of Teacher/Student networks and the roles of labeled/unlabeled data,
- Our method accelerates model convergence and stabilizes training in the early stages using much fewer data compared to existing state-of-the-art methods,
- Our method flexibly controls the influence of student networks on a teacher network, so that the teacher network also gets sufficiently perturbed and improved to provide better pseudo-supervision.

As a result, our method achieves *state-of-the-art* performance on *Pascal VOC 2012* and *Cityscapes*, which are two representative datasets for semantic segmentation.

2. Related Works

Semi-supervised Semantic Segmentation. Recent semi-supervised learning methods for semantic segmentation are categorized into two classes: 1) self-training and 2) perturbation-based methods. Self-training [41, 42] consists of iterative two-step procedures: 'pseudo-labeling' and 'training'. First, a model is trained with the small labeled set. In the pseudo-labeling step, the supervised model predicts pseudo-labels of unlabeled images. Then the unlabeled data with pseudo-labels and the labeled data are typically combined and used to train the classifier as a whole.

Apart from self-training, perturbation-based methods rely on the assumption that the model prediction should be consistent even if inputs, features, or model parameters are perturbed. Owing to this property to maintain consistency, perturbation-based learning is known to impose a consistency regularization. A body of research in this direction includes Mean Teacher (MT) [31], Cross-Consistency Training (CCT) [27], CutMix-Seg [14], and Guided Collaborative Training (GCT) [17]. Also, PseudoSeg [43] inspired by FixMatch [30] performs the pseudo-segmentation using differently augmented images, and Cross Pseudo Supervision (CPS) [7] and n-CPS [13] perturb network parameters. Recently, Perturbed and Strict MT (PS-MT) [25] extended MT with one student and two teacher networks by unifying input, feature, and network perturbations to generalize the consistency regularization leveraging heavy augmentation.

Teacher-Student Framework. Teacher-student models [6, 25, 31] have been adopted for label-efficient learning of semantic segmentation. They are trained via a knowledge distillation strategy [10, 24, 35] that improves student models under the guidance of teacher models. Shen *et al*. [29] employed an ensemble of large-scale teacher networks to provide accurate pseudo-supervision to a student. On the other hand, networks of the same architecture but with different initialization have been used for teacher and student in [36, 40], or even a single network has played both teacher and student roles in [30]. Another recent approach is to build the teacher as an exponential moving average (EMA)

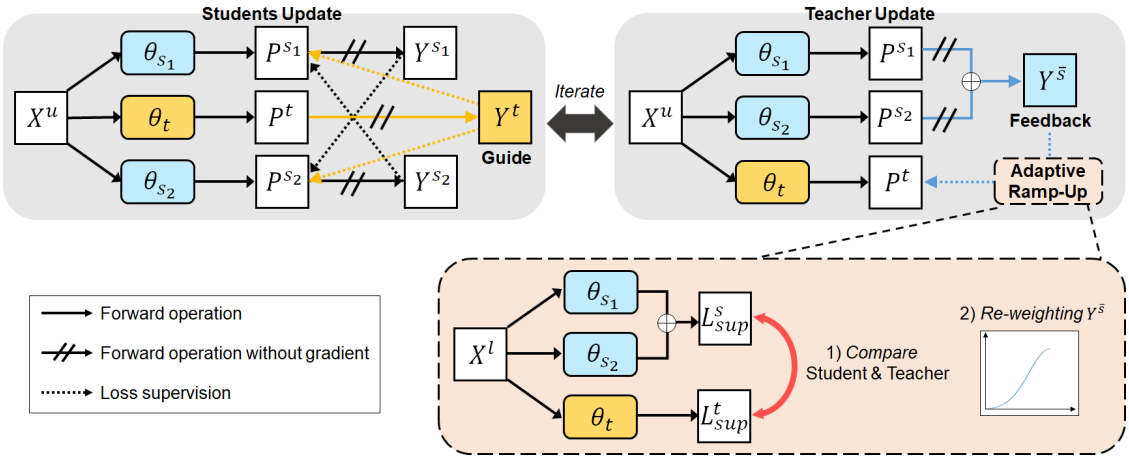


Figure 1. An overview of training architecture. Two student networks (θ_{s_1} and θ_{s_2}) and one teacher network (θ_t) are iteratively updated using the counterpart's pseudo label from the unlabeled image $X^u \in D^u$. For each network, P is a class probability map as a network output, and Y is a one-hot encoded pseudo-supervision. To perturb the teacher with reliable feedback, the magnitude of feedback from students is controlled with Adaptive Ramp-Up. To do so, the teacher and student networks are quantitatively compared via supervised losses on the labeled image $X^l \in D^l$. Once the average loss of students is smaller than the teacher's (i.e., $L^s_{sup}/2 < L^t_{sup}$), the feedback weight is maximized and the teacher can be further improved with informative perturbations on the unseen dataset D^u.

of the student [19,21,25,31], which has been known to provide stable pseudo-supervision.

3. Method

Let $D^l = \{X^l_i\}^n_{i=1}$ be a labeled image set of n samples with an individual ground truth label Y^l_i which represents pixel-wise object classes, and let $D^u = \{X^u_j\}^m_{j=1}$ be a set of m unlabeled images. Our method aims to solve semi-supervised semantic segmentation by jointly training three separate networks using the two independent image sets. The segmentation model consists of a Teacher Network (TN) and two Student Networks (SNs) which share the same structure with different parameter initialization.

The training strategy is comprised of three phases: **a)** training a TN using D^l with human annotation, **b)** co-training SNs, and **c)** updating the TN, and b) and c) are iterated until all models converge as shown in Fig. 1. The TN is first trained with ground truth supervision Y^l from D^l (expensive data), in order to learn reasonable knowledge to teach SNs with pseudo-supervision. The SNs receive a pseudo-supervision Y^t from the teacher on D^u (cheap data) and offer their pseudo-supervisions Y^{s_1} and Y^{s_2} to each other. Using the powerful guide Y^t from the TN, the SNs grow rapidly to follow up the TN. While existing works with Teacher-Student architecture [25, 31, 34] perform unilateral supervision from a teacher to a student, our SNs provide their averaged feedback $Y^{\bar{s}}$ to the TN on D^u so that the TN improves to provide reliable supervision to SNs.

Notice that the feedback from SNs $Y^{\bar{s}}$ is not reliable at the beginning of the training as they are learning the segmentation from scratch. Therefore, we designed an Adaptive Ramp-Up scheme that leverages the D^l as a "validation set" to fairly evaluate and compare the performance of SNs and TN via their supervised losses. With the Adaptive Ramp-Up, the influence of feedback from SNs is flexibly controlled. We expect that the students eventually become better than the teacher, and the teacher also improves to further guide the students by effectively perturbing each other with improved pseudo-supervision.

3.1. Establishing Teacher Network

Given a labeled image $X^l_i \in D^l$, the TN $f_{\theta_t}(\cdot)$ produces a segmentation confidence map $P^{l,t}_i$ which contains class probability as

$$P^{l,t}_i = f(X^l_i; \theta_t). \tag{1}$$

Suppose an input image has a resolution (W, H). The supervision loss L^t_{sup} of the TN is computed with a standard pixel-wise cross-entropy loss l_{ce} using a confidence vector $\mathbf{p}^t_{ip}$ at p-th pixel of $P^{l,t}_i$ for the whole images in D^l as

$$L^t_{sup}(X^l, Y^l) = \frac{1}{n \times W \times H} \sum_{X^l_i \in D^l} \sum_{p=1}^{W \times H} l_{ce}(\mathbf{p}^t_{ip}, \mathbf{y}^l_{ip}) \tag{2}$$

where $\mathbf{y}^l_{ip}$ is a one-hot encoded ground truth vector from Y^l_i. With this loss, TN is trained on the accurately labeled image set and attains the ability to generate pseudo-supervision to guide SNs in an indirect manner.

3.2. Guiding Student Networks

Using both D^l and D^u, our method jointly trains a pair of SNs $f_{\theta_{s_1}}(\cdot)$ and $f_{\theta_{s_2}}(\cdot)$. Both SNs have the same structure as the TN's, but their parameters θ_{s_1} and θ_{s_2} are differently initialized. The segmentation confidence maps of the two SNs ($k \in \{1, 2\}$) on D^l and D^u are produced as

$$P^{l,s_k}_i = f(X^l_i; \theta_{s_k}) \quad \text{and} \quad P^{u,s_k}_j = f(X^u_j; \theta_{s_k}). \tag{3}$$

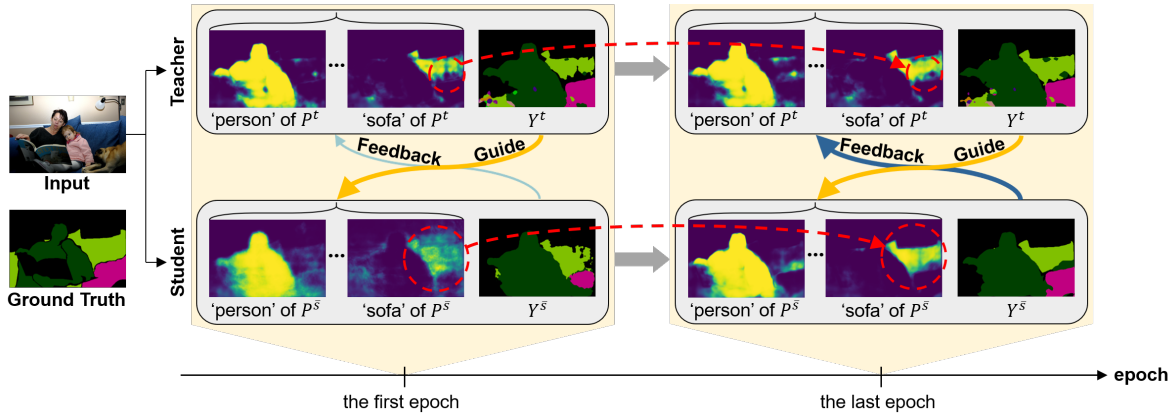


Figure 2. Example of interactions between TN and SNs via the confidence maps (for slices of 'person' and 'sofa' classes) and pseudo-label maps at the first and last epoch. The networks iteratively exchange their pseudo-labels through guiding (with GPS) and feedback (with FPS) phases. The GPS allows SNs to effectively learn the supervised knowledge of the TN from the beginning, and FPS with Adaptive Ramp-Up allows the TN to improve its guiding quality. At the last epoch, the pseudo-label qualities of SNs are substantially improved compared to the blurred prediction at the first epoch (in red circle).

where i and j denote indices in D^l and D^u respectively. In this step, the objective function consists of three losses for the SNs: 1) a supervision loss from the students L^s_{sup} with the ground truth Y^l, 2) L_{gps} from the TN with Y^t and 3) L_{cps} from SNs with Y^{s_k}. Note that the ground truth Y^l is from D^l, and the pseudo-labels Y^t and Y^{s_k} are from D^u.

First, as in Eq. 2, a classical pixel-wise cross-entropy is calculated for the supervision loss L^s_{sup} over the confidence maps (i.e., P^{l,s_1}_i and P^{l,s_2}_i) of two SNs as

$$L^s_{sup}(X^l, Y^l) = \frac{1}{n \times W \times H} \sum_{X^l_i \in D^l} \sum_{p=1}^{W \times H} \big(l_{ce}(\mathbf{p}^{s_1}_{ip}, \mathbf{y}^l_{ip}) + l_{ce}(\mathbf{p}^{s_2}_{ip}, \mathbf{y}^l_{ip})\big) \quad (4)$$

where $\mathbf{p}^{s_k}_{ip}$ is a class probability vector at p-th pixel of P^{l,s_k}_i and $\mathbf{y}^l_{ip}$ is a corresponding one-hot encoded ground truth.

Second, the loss to train SNs with the TN's pseudo-labels Y^t from unlabeled data D^u via cross-entropy is defined as,

$$L_{gps}(X^u, Y^t) = \frac{1}{m \times W \times H} \sum_{X^u_j \in D^u} \sum_{p=1}^{W \times H} \big(l_{ce}(\mathbf{p}^{s_1}_{jp}, \mathbf{y}^t_{jp}) + l_{ce}(\mathbf{p}^{s_2}_{jp}, \mathbf{y}^t_{jp})\big) \quad (5)$$

where $\mathbf{p}^{s_k}_{jp}$ is a class probability vector at p-th pixel of P^{u,s_k}_j and $\mathbf{y}^t_{jp}$ is a corresponding one-hot encoded pseudo-label vector from Guided Pseudo Supervision (GPS) Y^t_j. Minimizing the L_{gps} enforces the consistency between the TN and SNs, and let the predictions of SNs be similar to the TN's for the same input. Also, we expect that the guide from the TN will help SNs train faster (especially at the early stages of training) as the TN already has some knowledge on the segmentation although it may not be optimal.

Unlike the GPS, the CPS [7] operates only between SNs without intervention from the TN. It is a bilateral operation to exchange pseudo-labels $\mathbf{y}^{s_1}_{jp}$ and $\mathbf{y}^{s_2}_{jp}$ (i.e., trained knowledge) between the SNs as

$$L_{cps}(X^u, Y^s) = \frac{1}{m \times W \times H} \sum_{X^u_j \in D^u} \sum_{p=1}^{W \times H} \big(l_{ce}(\mathbf{p}^{s_1}_{jp}, \mathbf{y}^{s_2}_{jp}) + l_{ce}(\mathbf{p}^{s_2}_{jp}, \mathbf{y}^{s_1}_{jp})\big). \quad (6)$$

Finally, the overall loss function for SNs is given as:

$$L_{guide} = L^s_{sup} + \alpha L_{gps} + \beta L_{cps} \quad (7)$$

where α and β are hyperparameters to balance the losses. In this way, the L_{gps} and L_{cps} simultaneously evaluate the predictions of SNs for X^u_j (i.e., P^{u,s_1}_j and P^{u,s_2}_j). Each SN is trained to improve their predictions by approximating the GPS from the TN, and the consistency from each other is enforced providing different pseudo-labels. This perturbation from different pseudo-labels on the same input acts as a regularizer to prevent overfitting that may be caused by the TN which is already trained with partial labeled dataset.

3.3. Perturbing Teacher with Adaptive Ramp-Up

After the SNs learn sufficient knowledge from the trained TN with L_{gps}, they pass what they have learned to the TN by providing their averaged pseudo-supervision. At the same iterative step of the guiding phase, the updated SNs take the identical unlabeled image X^u_j once again and produce a one-hot encoded pseudo-label map $Y^{\bar{s}}_j$, i.e., Feedback Pseudo Supervision (FPS), which is computed from averaged segmentation confidence map $(P^{u,s_1}_j + P^{u,s_2}_j)/2$.

The TN learns from D^u by leveraging a per-pixel pseudo-label vector $\mathbf{y}^{\bar{s}}_{jp}$ from FPS $Y^{\bar{s}}_j$. The loss to train the TN with FPS is written as

$$L_{fps}(X^u, Y^{\bar{s}}) = \frac{1}{m \times W \times H} \sum_{X^u_j \in D^u} \sum_{p=1}^{W \times H} l_{ce}(\mathbf{p}^t_{jp}, \mathbf{y}^{\bar{s}}_{jp}). \quad (8)$$

Adaptive Gaussian Ramp-up. Since the supervision from SNs is hardly reliable at the beginning of the training, we apply *adaptive* Gaussian ramp-up $w(\cdot)$ to Eq. (8). Given a number for maximum training epochs E, the $w(\cdot)$ is defined with a current training epoch e and a maximum ramp-up epoch r ($1 \leq e, r \leq E$) as

$$w(e) = \begin{cases} \exp(-(1-\frac{e}{r})^2), & \text{if } e \leq r \\ 1, & \text{otherwise} \end{cases} \quad (9)$$

where r is initialized as E. The point to maximize the ramp-up is automatically determined by evaluating the SNs' performance on D^l: if $L^s_{sup}/2 \leq L^t_{sup}$, the r is set to e (i.e., $w(e) = 1$). From $L^s_{sup}/2 = L^t_{sup}$ when the performance of SNs is as good as the TN for D^l, they are expected to provide sensible feedback to the TN. With a hyperparameter γ, the feedback loss function can be written as

$$L_{fb} = L^t_{sup} + w(e) \cdot \gamma L_{fps}. \quad (10)$$

Without this Adaptive Ramp-Up, the TN will receive poor feedback from the SNs in the early training stages, and its performance becomes even worse. Eventually, it will provide a poor guide back to the SNs and slow the model convergence. Also, without this feedback step for the TN, the TN will not improve and affect the GPS step, limiting the SNs' performance as the TN and SNs are tied with consistency regularization. However, with this feedback step, SNs alleviate the overconfidence of TN acting as regularizers. The interaction between TN and SNs explained above is visualized in Fig. 2 with the variation of their pseudo-label maps across the beginning and the end of training.

4. Experiments

In this section, we quantitatively and qualitatively evaluate our method with various recent methods for semi-supervised segmentation on two independent datasets. Ablation study is also introduced to empirically examine the roles of individual components within our model.

4.1. Experimental Setup

Datasets. We conducted experiments on two standardized benchmarks for semantic segmentation: *Pascal VOC 2012* [12] (Pascal) and *Cityscapes* [8]. Pascal contains 20 object classes and 1 background class. The original dataset is comprised of 1464, 1449, and 1456 images for training, validation, and test sets, respectively. We used an augmented training dataset (10582 images) proposed in [16] for full training. With 19 classes, Cityscapes consists of 2975, 500, and 1525 images for training, validation, and test sets, respectively. The images are fine-annotated with a resolution of 2048×1024. To construct a labeled partition, we follow partition protocols suggested in GCT [17] as setting $1/16$, $1/8$, $1/4$, and $1/2$ of the whole dataset for labeled set and the other for unlabeled set.

Implementation. Pytorch framework and SGD optimizer with momentum 0.9 were used to implement our work. We set the learning rate 0.01 and 0.02 for Pascal and Cityscapes, respectively, and they were multiplied by $(1 - \frac{iter}{max\_iter})^{0.9}$ using a poly-learning rate scheduler. The batch size is set to 8 for each labeled and unlabeled data and the scale of image is randomly selected from $\{0.5, 0.75, 1, 1.5, 1.75, 2\}$. CutMix [37] is used in our framework as in other recent methods for semi-supervised segmentation [7, 13, 25, 34]. After the supervised TN is obtained, we fixed the learning rate of the TN as $(max\_epoch \times max\_iter)^{-0.9}$ multiplied by the initial learning rate. The trade-off weights were set as $\alpha = 0.5$ (1.0), $\beta = 1.5$ (5.0), and $\gamma = 0.5$ (1.0) for Pascal (Cityscapes). The number of epochs were set as 32 (128) / 34 (137) / 40 (160) / 60 (240) for $1/16$, $1/8$, $1/4$, and $1/2$ partition protocols on Pascal (Cityscapes), respectively. As a segmentation network, DeepLabv3+ [5] with ResNet-50 or ResNet-101 pretrained on ImageNet is used and its segmentation head is randomly initialized. As only three networks could be rested even on NVIDIA RTX A6000 GPUs with 48GB memory, using three networks was our best.

Evaluation. For all experiments, we performed a single-scale inference with mean Intersection-over-Union (mIoU) metric on validation sets of both benchmarks. The results of our approach are reported using one student network, and we did not use any ensemble techniques for all evaluations.

4.2. Quantitative Analysis

In this section, our method is compared with state-of-the-art baselines in segmentation performance (mIoU) and efficiency (size of required data and convergence speed).

4.2.1 Comparisons with SOTA Baselines

Comparison on Computational Costs. In Table 1, we compare the number of unlabeled data required by various network-perturbation-based methods with our method. Along with the unlabeled samples from D^u, the CPS and n-CPS additionally use the *labeled* set D^l without the ground truth as if it were an unlabeled set. On the other hand, our method does not adopt such a scheme by default and thus

Table 1. **Comparison on the number of unlabeled data** to train all networks of network perturbation-based methods. Under the same supervised partitions, different amounts of data and epochs are required for each method to converge the networks.

Method	# Network	Pascal VOC 2012				Cityscapes			
		1/16	1/8	1/4	1/2	1/16	1/8	1/4	1/2
# of unlabeled data for 1 epoch									
CPS [7] (CVPR '21)	2	10.5k	10.5k	10.5k	10.5k	2.9k	2.9k	2.9k	2.9k
3-CPS [13] (arXiv)	3	10.5k	10.5k	10.5k	10.5k	2.9k	2.9k	2.9k	2.9k
ELN [19] (CVPR '22)	2	19.8k	18.5k	15.8k	10.5k	5.5k	5.2k	4.4k	2.9k
PS-MT [25] (CVPR '22)	3	19.8k	18.5k	15.8k	10.5k	5.5k	5.2k	4.4k	2.9k
ST++ [36] (CVPR '22)	4	14.8k	13.8k	11.9k	7.9k	4.1k	3.9k	3.3k	2.2k
GPS (Ours)	3	**9.9k**	**9.2k**	**7.9k**	**5.2k**	**2.7k**	**2.6k**	**2.2k**	**1.4k**
# of unlabeled data for the whole epochs									
CPS [7] (CVPR '21)	2	339k	360k	423k	635k	381k	408k	476k	714k
3-CPS [13] (arXiv)	3	339k	360k	423k	635k	381k	408k	476k	714k
PS-MT [25] (CVPR '22)	3	1587k	1481k	3174k	3174k	-	1666k	2008k	1636k
ST++ [36] (CVPR '22)	4	1184k	1104k	952k	632k	984k	936k	792k	528k
GPS (Ours)	3	**317k**	**315k**	**317k**	**317k**	**357k**	**357k**	**357k**	**357k**

Table 2. **Performance comparison on Cityscapes** with the *state-of-the-art* methods under different supervised partitions.

Method	ResNet-50				ResNet-101			
	1/16	1/8	1/4	1/2	1/16	1/8	1/4	1/2
SupOnly	64.30	66.00	70.70	72.00	65.74	72.53	74.43	77.13
MT [31] (NeurIPS '17)	66.14	72.03	74.47	77.43	68.08	73.71	76.53	78.59
CCT [27] (CVPR '20)	66.35	72.46	75.68	76.78	69.64	74.48	76.35	78.29
GCT [17] (ECCV '20)	65.81	71.33	75.30	77.09	66.90	72.96	76.45	78.58
DCC [20] (CVPR '21)	-	69.70	72.70	-	-	-	-	-
CPS [7] (CVPR '21)	74.47	76.61	77.83	78.77	74.72	77.62	79.21	80.21
ST [36] (CVPR '22)	-	71.60	73.40	-	-	-	-	-
ST++ [36] (CVPR '22)	-	72.70	73.80	-	-	-	-	-
U2PL [34] (CVPR '22)	-	-	-	-	70.30	74.37	76.47	79.05
ELN [19] (CVPR '22)	-	70.33	73.52	75.33	-	-	-	-
USRN [15] (CVPR '22)	71.20	75.00	-	-	-	-	-	-
PS-MT [25] (CVPR '22)	-	77.12	78.38	79.22	-	-	-	-
PGCL [18] (WACV '23)	-	71.20	73.90	76.80	-	-	-	-
GPS (Ours)	**74.86**	**77.32**	**78.71**	**79.33**	**75.64**	**77.79**	**79.27**	**80.40**

Table 3. **Performance comparison on Pascal VOC 2012** with the *state-of-the-art* methods under different supervised partitions.

Method	ResNet-50				ResNet-101			
	1/16	1/8	1/4	1/2	1/16	1/8	1/4	1/2
SupOnly	63.90	68.20	70.40	73.12	65.74	72.53	74.43	77.83
MT [31] (NeurIPS '17)	66.77	70.78	73.22	75.41	70.59	73.20	76.62	77.61
CCT [27] (CVPR '20)	65.22	70.87	73.43	74.75	67.94	73.00	76.17	77.56
CutMix-Seg [14] (BMVC '20)	68.90	70.70	72.46	74.49	72.56	72.69	74.25	75.89
GCT [17] (ECCV '20)	64.05	70.47	73.45	75.20	69.77	73.30	75.25	77.14
DCC [20] (CVPR '21)	70.10	72.40	74.00	76.50	72.40	74.60	76.30	78.20
CPS [7] (CVPR '21)	71.98	73.67	74.90	76.15	74.48	76.44	77.68	78.64
3-CPS [13] (arXiv)	71.11	73.56	74.68	75.86	74.98	76.98	77.95	79.67
USCS [38] (ACCV '22)	72.30	74.88	76.15	76.45	74.52	76.20	77.09	78.63
ST [36] (CVPR '22)	71.60	73.30	75.00	-	72.90	75.70	76.40	-
ST++ [36] (CVPR '22)	72.60	74.40	75.40	-	74.50	76.30	76.60	-
U2PL [34] (CVPR '22)	-	-	-	-	74.43	77.60	78.70	79.94
ELN [19] (CVPR '22)	-	73.20	74.63	-	-	75.10	76.58	-
PS-MT [25] (CVPR '22)	72.83	75.70	76.43	77.88	75.50	**78.20**	78.72	79.76
PGCL [18] (WACV '23)	-	75.20	76.00	-	-	76.80	77.90	-
GPS (Ours)	**72.91**	75.72	76.33	77.07	**75.66**	77.56	79.18	79.88
GPS† (Ours)	-	**76.03**	**76.56**	**77.97**	-	77.67	**79.61**	**80.61**

†: Additional unlabeled data were used by removing labels of the D^l.

uses only half of the unlabeled samples used in CPS and n-CPS under 1/2 labeled partition protocol.

As in our method, PS-MT and ELN do not take out the labels of D^l, however, they must double the unlabeled samples with diverse data augmentations to feed differently augmented images to the student and teacher networks. Therefore, these methods require unlabeled data *at least* twice more than ours, and the gap becomes even larger considering the number of epochs as the data augmentation is performed for every iterative step. Specifically, under the same setting, while our networks converged fast within 32 to 60, and 137 to 240 epochs for Pascal and Cityscapes, respectively, PS-MT required 80 to 300, and 320 to 550 epochs to fully train three networks. Similarly, ST++ does not remove the label of D^l, however, ST++ requires 1.5 times more unlabeled data than ours to perform its 2-step self-training with four networks. ST++ requires 80 and 240 epochs for Pascal and Cityscapes, respectively, for all partition protocols for model convergence.

Note that, even including the training steps to establish a TN, the number of epochs in GPS is still smaller than that of these baselines. With comparatively much less computational costs, our method uses only 10% of the data used by PS-MT in 1/4 and 1/2 partition protocols on Pascal. Also, our method uses only 26% and 36% of the data used by ST++ in the 1/16 partition protocol on Pascal and Cityscapes, respectively. Substantially reducing overhead costs with fewer data and faster convergence speed, GPS

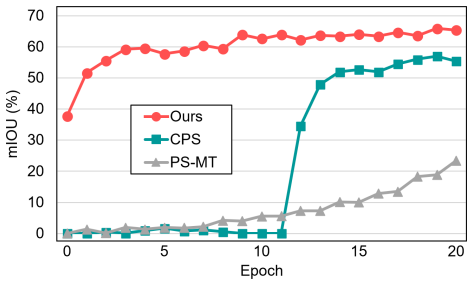


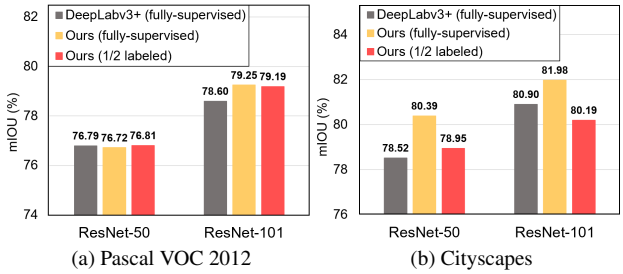
Figure 3. **Comparison of mIoU at the initial epochs** on validation set of Cityscapes. Our method and CPS are trained without CutMix augmentation and PS-MT is trained with multiple input augmentation along with the CutMix. ResNet-50 is used under 1/16 supervised partition protocol.

outperformed these *state-of-the-art* methods in many partition protocols for both datasets as described below.

Comparison on Segmentation Performance. In Table 2 and 3, our approach outperforms all baselines in both benchmarks. Notably, *using much fewer data and epochs for network convergence*, our method surpassed SOTA baselines such as PS-MT, 3-CPS, ST, and ST++ which adopt 3-4 networks. Our method surpasses the supervised baseline by 3∼12%p and 2∼10%p for diverse settings on Cityscapes and Pascal, respectively. On Pascal, we additionally tested our method with more unlabeled data by removing the label of the labeled set. This setting lets us utilize more data and epochs as in CPS and 3-CPS as shown in Table 1, which are *still smaller* than other SOTA methods [19,25,36] in most settings. We observed that the use of more data brings a 0.1∼0.9%p improvement in our method, which intensifies the gap over baselines. Therefore, it is worth noting that our method has the potential to be further boosted on Cityscapes if we adopt these larger-scale or heavily augmented datasets as in other baselines.

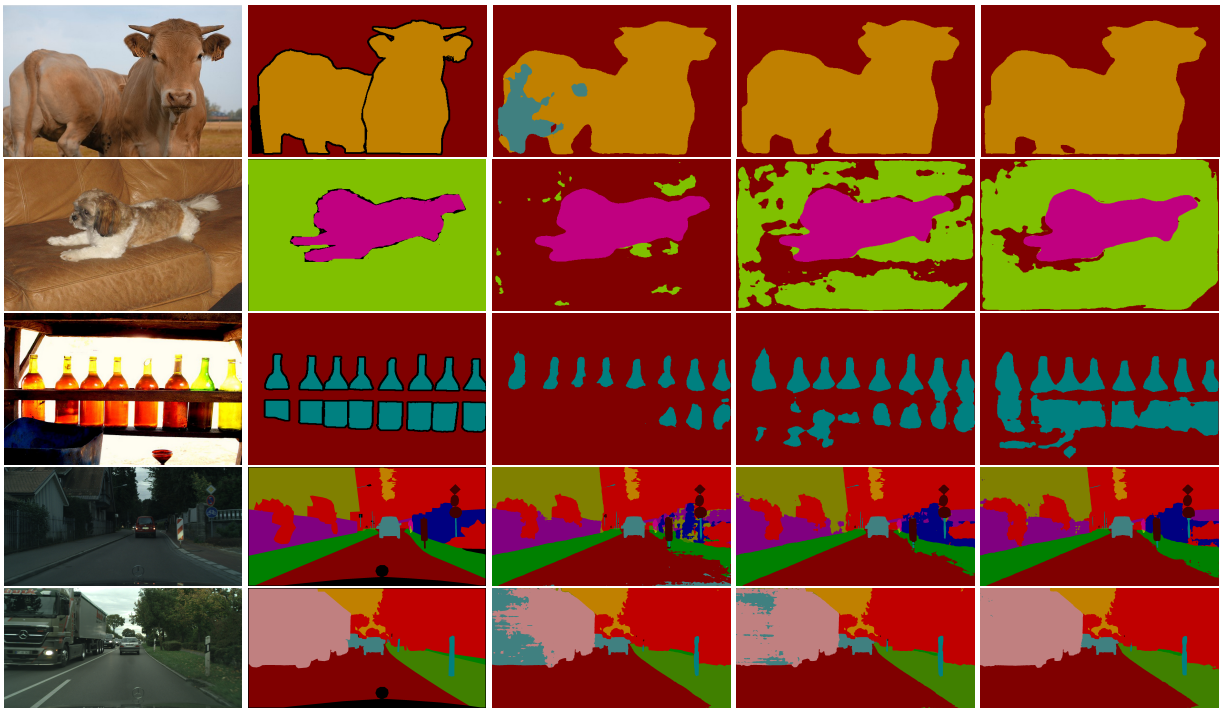
4.2.2 Comparison of Training Flows

Fig. 3 compares the performance of our method with current SOTA network-perturbation-based methods (i.e., PS-MT and CPS) at the initial training steps. The result of



(a) Pascal VOC 2012 (b) Cityscapes

Figure 4. **Comparison of fully-supervised methods** with our approach under 1/2 labeled partition protocol w/o CutMix. With half of the supervision, our method (red) outperforms the fully-supervised DeepLabv3+ network (grey) in all settings except the result with ResNet-101 on Cityscapes.



(a) Input (b) Ground Truth (c) Supervised only (d) Ours w/o CutMix (e) Ours w/ CutMix

Figure 5. **Resultant Samples** from Pascal (top 1-3 rows) and Cityscapes (bottom 4-5 rows). (a) Input, (b) Ground Truth, (c) Supervised only, (d) Ours w/o CutMix, (e) Ours w/ CutMix. DeepLabv3+ with ResNet-50 under 1/8 partition protocol is used to produce the results.

our method at the first epoch (37.82%) outperforms both of the CPS's (0.06%) and PS-MT's (0.02%) by 37.76%p and 37.80%p respectively and this trend continues until epoch 12, where the network in CPS starts to train towards its convergence. Although the PS-MT leverages more data with various augmentations (including CutMix), GPS (without CutMix) consistently shows much better mIoU even with faster convergence across all 20 epochs. This fast and stable convergence demonstrates the effect of our initial guiding phase, which eventually makes smart SNs in the early training process via consistency regularization. As the SNs rapidly grow with GPS, the TN can quickly receive useful feedback on the previously unknown knowledge (i.e., unlabeled samples) from the SNs. Ultimately, this positive guiding-feedback loop strengthens both TN and SNs even in the later training process. Without the guide from TN, the CPS learns the whole data from scratch, and one can easily see that it requires more epochs to converge.

4.2.3 Comparison with Fully-Supervised Models

In this section, we compare GPS with a fully supervised model to demonstrate that GPS performs as good as or even better than the fully supervised model using significantly less labeled data. In Fig 4, DeepLabv3+ [5] (grey), GPS under 1/2 partition (red) and GPS with full supervision (yellow) — unlabeled data are curated by removing labels from the original data — are compared. Notably, although the number of labeled images has been reduced by half, our method under $1/2$ labeled partition protocol (red) outperformed the fully-supervised networks in almost all settings.

4.3. Qualitative Results

In Fig. 5, we visualize partial results of prediction from our method on Pascal and Cityscapes. The results of a supervised-only network show poor performance compared to our methods. For example, in the second row, the supervised-only method barely predicts background pixels as a sofa, while our method recognizes most of them as the true label. In the third row, although there exists an occlusion in front of the bottles, our method detects the bottom half of the bottles while it was tricky in the supervised-only method. Moreover, in the fourth row, our method correctly predicts the wall (blue) at the right side, and notably, the difference can be found in the bus (pink) of the fifth row.

Table 4. **Ablation study over various loss configurations** using ResNet-50 under 1/8 supervised partition protocol without CutMix. Notably, the comparison of the last row (w/ L_{gps}) and the 4-th row (w/o L_{gps}) shows the improvement from GPS.

L^t_{sup}	L^s_{sup}	L_{gps}	L_{cps}	L_{fps}	Pascal VOC	Cityscapes
✓	✓	✓	-	-	72.25	72.36
✓	✓	-	-	✓	70.89	71.25
✓	✓	✓	-	✓	72.82	72.39
✓	✓	-	✓	✓	72.99	73.51
✓	✓	✓	✓	-	72.72	72.92
✓	✓	✓	✓	✓	**73.95**	**74.62**

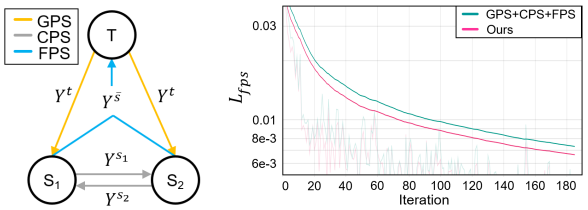


Figure 6. **Comparison of the united loss update structure and our method** on loss flows of L_{fps} at the first epoch. (a) A united loss update structure. (b) Loss flows of L_{fps} on Cityscapes under 1/2 labeled partition protocol with ResNet-50. The loss curves are smoothed for visualization.

4.4. Ablation Study

4.4.1 Analysis on Perturbing the Teacher Network

In our training scheme, we added the perturbing step to improve the generalization of the TN. Intuitively, this step is essential since the initial TN can solely observe a small fraction of the whole dataset for supervision which may result in overfitting. Furthermore, if the FPS is excluded, the training becomes a one-way pseudo-supervision in terms of unilaterally delivering pseudo-labels from the TN to SNs only. This single-network pseudo-supervision is known to be inferior to the CPS, mainly because the training highly depends on the training quality of the TN [7]. In other words, if the TN is weakly trained, the training of the SNs also fails as they consistently approximate the TN's pseudo-labels.

To verify the necessity of this step, we perform the ablation study over different loss combinations in Table 4. Compared to the case of only adopting L_{gps} (72.25%) among the three losses for pseudo-labeling (i.e., L_{gps}, L_{cps} and L_{fps}), one can observe that adding L_{fps} directly improves the performance of SN by $0.57\%p$ on Pascal VOC 2012. Also, the last two rows of Table 4 indicate that the application of L_{fps} with the rest of the losses improves the result without L_{fps} (72.72% and 72.92%) by $1.23\%p$ and $1.70\%p$ on Pascal and Cityscapes, respectively.

4.4.2 Merging vs. Separating GPS, CPS, and FPS

We illustrate why the networks are iteratively trained (i.e., the loss updates are performed *twice* in a single iterative step) with the Eq. (7) (with L_{gps} and L_{cps}) and Eq. (10) (with L_{fps}). As shown in Fig. 6a, the entire pseudo-supervision losses suggested in our method can be combined as $\mathcal{L} = L_{guide} + L_{fb} = L^t_{sup} + L^s_{sup} + \alpha L_{gps} + \beta L_{cps} + \gamma L_{fps}$ with the ground truth supervision losses and trained at the same time.

The significance of our iterative loss update scheme is compared to the $\mathcal{L}$ with the following assumption. For each iteration i, the TN's parameter θ^i_t in the unified loss is updated as: $\theta^i_t = \text{argmin}_\theta \mathcal{L}(X; \theta^{i-1}_{s_1}, \theta^{i-1}_{s_2}, \theta^{i-1}_t)$, where the parameters of the three networks at the previous step are used for loss calculation. On the contrary, our method uses the revised SNs' parameters at the current step as: $\theta^i_t = \text{argmin}_\theta \mathcal{L}(X; \theta^i_{s_1}, \theta^i_{s_2}, \theta^{i-1}_t)$. The SNs of our method send feedback to the TN soon after they are trained with GPS, while the SNs of the unified loss need to wait until the next iteration to provide FPS. We assumed that $\theta^i_{s_1}$ and $\theta^i_{s_2}$ contain richer knowledge on the data compared to $\theta^{i-1}_{s_1}$ and $\theta^{i-1}_{s_2}$, and thus they serve as better parameters to minimize L_{fps} to enforce the consistency between TN and SNs.

We experimentally proved the assumption above by comparing the loss flow of our approach with that of the single loss update structure on the Cityscapes benchmark. According to the resultant loss flows shown in Fig. 6b, we can see that our method with two-step loss update (pink) has a stronger convergence rate compared to the merged loss update structure (blue), demonstrating our approach can intensify the consistency for the three networks.

4.4.3 Does TN Help SN in Later Epochs?

Here, we analyze the necessity of a TN at latter epochs where its mIoU is lower than that of SNs. Using the same condition (i.e., $L^s_{sup}/2 \leq L^t_{sup}$) as in Eq. 9, we applied adaptive *ramp-down* to L_{gps} to reduce the amount of pseudo-supervision from a TN at latter epochs. We observed mIoU of 70.73% and 73.57% in the $1/16$ and $1/8$ supervised partition protocols, respectively, using ResNet-50 on the Pascal dataset. At the same settings, our method *without ramp-down* showed 72.91% and 75.72% mIoU. These results demonstrate that the knowledge of the TN is not totally subsumed to the SNs and the TN still provides useful guidance to some samples although its overall performance is worse than that of the SNs. Also, unlike EMA teachers [25, 31] whose parameters are extracted from SNs, our TN is independently initialized from SNs such that it is expected to provide much more diverse and robust supervision during the entire training.

5. Conclusion

We present GPS, a novel data-efficient approach based on the teacher-student framework for semi-supervised semantic segmentation. Unlike prior studies, GPS assigns different roles to the teacher and students during training, strategically utilizing labeled and unlabeled data. The method enables fast and stable learning for the students at the early training stages and regulates feedback from students with adaptive ramp-up scheme. Consequently, the teacher can consistently provide informative knowledge back to the students and models trained by GPS achieved the state of the art on both PASCAL VOC 2012 and Cityscapes benchmarks.

Acknowledgement. This research was supported by IITP-2022-0-00290 (50%), NRF-2022R1A2C2092336 (40%), and IITP-2019-0-01906 (AI Graduate Program at POSTECH, 10%).

References

[1] Sean Bell, Paul Upchurch, Noah Snavely, et al. Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3479–3487, 2015. 1

[2] David Berthelot, Nicholas Carlini, Ekin D Cubuk, et al. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *International Conference on Learning Representations (ICLR)*, 2020. 1

[3] David Berthelot, Nicholas Carlini, Ian Goodfellow, et al. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in neural information processing systems (NeurIPS)*, 2019. 1

[4] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE TPAMI)*, 40(4):834–848, 2017. 1

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 5, 7

[6] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:22243–22255, 2020. 2

[7] Xiaokang Chen, Yuhui Yuan, Gang Zeng, et al. Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 2, 4, 5, 6, 8

[8] Marius Cordts, Mohamed Omran, Sebastian Ramos, et al. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 3213–3223, 2016. 5

[9] Jifeng Dai, Kaiming He, and Jian Sun. Convolutional feature masking for joint object and stuff segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3992–4000, 2015. 1

[10] Peijie Dong, Lujun Li, and Zimian Wei. Diswot: Student architecture search for distillation without training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11898–11908, 2023. 2

[11] Thibaut Durand, Taylor Mordan, Nicolas Thome, et al. Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[12] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, et al. The pascal visual object classes (voc) challenge. *International journal of computer vision*, (2):303–338, 2010. 5

[13] Dominik Filipiak, Piotr Tempczyk, and Marek Cygan. n-cps: Generalising cross pseudo supervision to n networks for semi-supervised semantic segmentation. arXiv preprint arXiv:2112.07528, 2021. 1, 2, 5, 6

[14] Geoff French, Samuli Laine, Timo Aila, et al. Semi-supervised semantic segmentation needs strong, varied perturbations. In *The British Machine Vision Conference (BMVC)*, 2020. 1, 2, 6

[15] Dayan Guan, Jiaxing Huang, Aoran Xiao, and Shijian Lu. Unbiased subclass regularization for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9968–9978, 2022. 6

[16] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, et al. Semantic contours from inverse detectors. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 991–998, 2011. 5

[17] Zhanghan Ke, Di Qiu, Kaican Li, et al. Guided collaborative training for pixel-wise semi-supervised learning. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 5, 6

[18] Heejo Kong, Gun-Hee Lee, Suneung Kim, and Seong-Whan Lee. Pruning-guided curriculum learning for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5914–5923, 2023. 6

[19] Donghyeon Kwon and Suha Kwak. Semi-supervised semantic segmentation with error localization network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9957–9967, 2022. 3, 5, 6

[20] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1205–1214, 2021. 6

[21] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016. 3

[22] Jungbeom Lee, Eunji Kim, Sungmin Lee, et al. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[23] Sangrok Lee, Eunsoo Park, Hongsuk Yi, et al. Strdan: Synthetic-to-real domain adaptation network for vehicle re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[24] Lujun Li and Zhe Jin. Shadow knowledge distillation: Bridging offline and online knowledge transfer. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:635–649, 2022. 2

[25] Yuyuan Liu, Yu Tian, Yuanhong Chen, et al. Perturbed and strict mean teachers for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Com-*

puter Vision and Pattern Recognition (CVPR), pages 4258–4267, 2022. 1, 2, 3, 5, 6, 8

[26] Robert Mendel, Luis Antonio Souza, David Rauber, et al. Semi-supervised segmentation based on error-correcting supervision. In *European Conference on Computer Vision (ECCV)*, 2020. 1

[27] Yassine Ouali, Celine Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 6

[28] George Papandreou, Liang-Chieh Chen, Kevin Murphy, et al. Weakly- and semi-supervised learning of a dcnn for semantic image segmentation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015. 1

[29] Zhiqiang Shen, Zhankui He, and Xiangyang Xue. Meal: Multi-model ensemble via adversarial learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4886–4893, 2019. 2

[30] Kihyuk Sohn, David Berthelot, Chun-Liang Li, et al. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in neural information processing systems (NeurIPS)*, 2020. 1, 2

[31] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems (NeurIPS)*, 2017. 1, 2, 3, 6, 8

[32] Martin Thoma. A survey of semantic segmentation. *arXiv preprint arXiv:1602.06541*, 2016. 1

[33] Panqu Wang, Pengfei Chen, Ye Yuan, et al. Understanding convolution for semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1451–1460, 2018. 1

[34] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4248–4257, 2022. 3, 5, 6

[35] Liu Xiaolong, Li Lujun, Li Chao, and Anbang Yao. Norm: Knowledge distillation via n-to-one representation matching. In *International Conference on Learning Representations (ICLR)*, 2023. 2

[36] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. St++: Make self-training work better for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4268–4277, 2022. 2, 5, 6

[37] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, et al. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. 5

[38] Yunyang Zhang, Zhiqiang Gong, Xiaoyu Zhao, Xiaohu Zheng, and Wen Yao. Semi-supervised semantic segmentation with uncertainty-guided self cross supervision. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 4631–4647, 2022. 6

[39] Sicheng Zhao, Bo Li, Xiangyu Yue, et al. Multi-source domain adaptation for semantic segmentation. In *Advances in neural information processing systems (NeurIPS)*, 2019. 1

[40] 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 Transactions on Pattern Analysis and Machine Intelligence (IEEE TPAMI)*, 2021. 2

[41] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, et al. Rethinking pre-training and self-training. In *Advances in neural information processing systems (NeurIPS)*, 2020. 2

[42] Yang Zou, Zhiding Yu, Xiaofeng Liu, et al. Confidence regularized self-training. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, page 5982 – 5991, 2019. 2

[43] Yuliang Zou, Zizhao Zhang, Han Zhang, et al. Pseudoseg: Designing pseudo labels for semantic segmentation. In *International Conference on Learning Representations (ICLR)*, 2021. 1, 2