arXiv:2306.04300v1 [cs.CV] 7 Jun 2023

CorrMatch: Label Propagation via Correlation Matching for Semi-Supervised Semantic Segmentation

Boyuan Sun[1] **Yu-Qi Yang**[1] **Le Zhang**[2] **Ming-Ming Cheng**[1] **Qibin Hou**[1]

[1]School of Computer Science, Nankai University
[2]School of Information and Communication Engineering, UESTC

Abstract

In this paper, we present a simple but performant semi-supervised semantic segmentation approach, termed CorrMatch. Our goal is to mine more high-quality regions from the unlabeled images to leverage the unlabeled data more efficiently for consistency regularization. The key contributions of our CorrMatch are two novel and complementary strategies. First, we introduce an adaptive threshold updating strategy with a relaxed initialization to expand the high-quality regions. Furthermore, we propose to propagate high-confidence predictions through measuring the pairwise similarities between pixels. Despite its simplicity, we show that CorrMatch achieves great performance on popular semi-supervised semantic segmentation benchmarks. Taking the DeepLabV3+ framework with ResNet-101 backbone as our segmentation model, we receive a 76%+ mIoU score on the Pascal VOC 2012 segmentation benchmark with only 92 annotated images provided. We also achieve a consistent improvement over previous semi-supervised semantic segmentation models. Code is available at https://github.com/BBBBchan/CorrMatch.

1 Introduction

Semantic segmentation is one of the core tasks in image processing and computer vision, which aims to provide pixel-level classification predictions for images. With the development of deep learning techniques, especially convolutional neural networks (CNNs) [19, 13, 59, 67], many significant semantic segmentation methods [40, 69, 5, 16, 43] have achieved remarkable results. However, methods based on deep learning often require large-scale pixel-wise annotated datasets with a massive amount of labeled images. Compared to the image classification and object detection tasks [8, 37], the accurate annotations for segmentation datasets are very expensive and time-consuming.

Recently, many researchers have sought to address the above challenge by reducing the demand for large-scale accurately annotated data in the semantic segmentation task by presenting weakly-supervised [56, 24, 55, 25], semi-supervised [20, 21, 11, 42], or even unsupervised segmentation methods [22, 12, 52, 17]. Among these schemes, semi-supervised semantic segmentation only requires a small amount of labeled data accompanied with a large amount of unlabeled data for training, which approaches real-world scenarios more and hence attracts the favor of more and more researchers from both academia and industry.

In the literature of semi-supervised semantic segmentation, most works use multiple networks [21, 39, 60], self-training [28, 63, 64], or contrastive learning [54, 71, 31] to enforce the consistency between results from different networks or unearth connections between training samples with different views. These methods often require multiple networks or training stages, making the training process complicated. As a contrast, recent works show that a simple single-stage pipeline is sufficient for learning a strong segmentation model under the semi-supervised setting. A typical example should be UniMatch [62], which leverages perturbation techniques for weak-to-strong consistency learning.

However, UniMatch [62] struggles to utilize unlabeled data efficiently due to selecting a strict and fixed threshold (0.95) to screen high-confidence pseudo labels for consistency training, making many correct predicted pixels ignored. Beyond that, in this paper, we investigate how to mine more accurate high-confidence regions for consistency regularization. Our technical contributions are two-fold: threshold relaxing and label propagation via correlation matching. We first consider how to select reliable regions with high-confidence predictions. Instead of leveraging a fixed threshold as done in UniMatch [62], we propose to relax the threshold selection by using a dynamic updating policy. To be specific, we maintain a global threshold for all classes to expand the high-quality regions. During the optimization process, we gradually update the threshold using the averaged scores of the highest probability for each class. This strategy is insensitive to the initial value of the threshold.

In addition, motivated by previous works showing that the correlations between pixels can reflect the pairwise similarities [38, 44, 36, 35], we propose to construct correlation maps from extracted features and then propagate them into predictions via correlation matching. This strategy is complementary to threshold relaxing and enables us to gradually mine more reliable high-confidence regions with accurate predictions and consequently improve the utilization of unlabeled data.

We find that the above two strategies cooperate well with each other. Our method, named CorrMatch, is able to receive great results using only a single network with a single-stage training process. Notably, CorrMatch achieves 78.3% mIoU on the Pascal VOC 2012 dataset with 1/8 (183) annotated images and 80.3% mIoU on the Cityscapes dataset with 1/2 (1488) annotated images, outperforming UniMatch by 1.1% and 0.8% in terms of mIoU.

2 Related Work

2.1 Semi-Supervised Learning

Semi-supervised learning [73, 46] is proposed to settle a paradigm that how to construct models using both labeled and unlabeled data and has been studied long before the deep learning era [27, 3, 2]. And certainly, semi-supervised learning has been studied more extensively as deep learning has brought great advances in computer vision [34, 58, 14, 74].

Since Bachman *et al.* [1] proposed a consistency regularization based method, many approaches have migrated it into the semi-supervised learning field. Π-Model [33, 45] assumes that model predictions should be consistent among the same inputs with stochastic perturbation. MixMatch [4] proposes to combine consistency regularization with entropy minimization. Mean Teacher [50] and Dual Student [30] are also based on consistent learning, aiming for the same outputs for different networks.

Recently, FixMatch [48] provides a simple yet effective weak-to-strong consistency regularization framework and serves as many other relevant methods' baseline [49, 15, 51, 62]. However, many follow-up works [53, 66, 61] have pointed out that simply using a manually fixed threshold (e.g., 0.95) under all different experimental settings may lead to inferior performance and slow convergence speed. Among them, FreeMatch [53] provides a dynamic threshold scheme connected with the model's learning process. However, it is designed for the image classification task but not suitable for semi-supervised semantic segmentation as there are often multiple categories existing in each image.

2.2 Semi-Supervised Semantic Segmentation

As semi-supervised learning has achieved surprising results in the image classification task [34, 50, 4, 48, 33], there are some works considering the same setting for semantic segmentation [20, 6, 42].

One type of methods [11, 72, 21, 54, 39, 60, 68] adopt the Mean Teacher architecture. U^{2}PL [54] attempts to make better use of unreliable predictions via contrastive learning. PS-MT [39] builds a stricter teacher with the VAT [41] technology. There are also some methods, like CPS [6] and CTT [57], using the dual student framework. All of these methods demand multi-networks during the training process. Meanwhile, there is another type of methods, self-training based methods [28, 63, 64, 9], which often require multiple training stages. ST++ [63] proposes to use a three-stage paradigm and emphasizes the importance of strong augmentations. SimpleBase [64] demands a pre-trained teacher model and uses separated batch normalization [23] for images with strong augmentation. More recently, FST [9] proposes two variants to explore the future states deeply and widely.

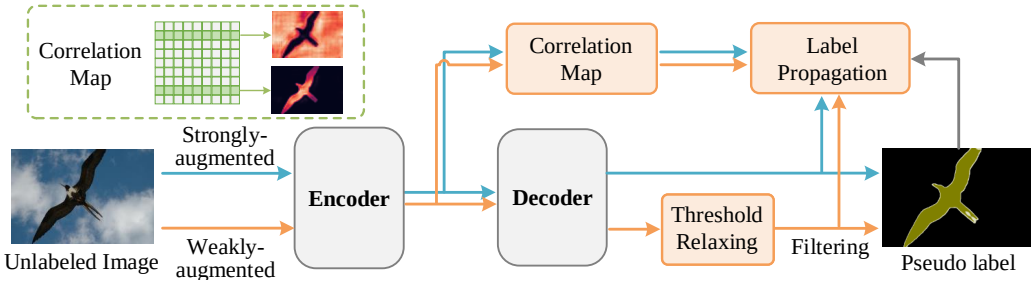


Figure 1: Illustration of our CorrMatch pipeline for unlabeled images. We build it upon the DeepLabv3+ framework [5]. Besides consistency regularization, CorrMatch adopts a threshold relaxing strategy to filter pseudo labels with a dynamic threshold, and a label propagation strategy with correlation matching.

Since FixMatch [48] combines self-training and consistency regularization into a single stage, it has become the baseline for many semi-supervised semantic segmentation approaches. PC^{2}Seg [71] uses feature-space contrastive learning besides consistency training. And recently, UniMatch [62] leverages multiple strong augmentation branches and feature perturbations via consistency regularization. Our CorrMatch also follows this single-stage framework. However, previous works mostly concentrate on how to leverage high-confidence regions and neglect mining high-confidence regions itself. Different from all above, CorrMatch focuses on mining more reliable high-confidence regions.

3 CorrMatch

The goal of semi-supervised semantic segmentation is to train a semantic segmentation network $\mathcal{F}$ with a small labeled image set $\mathcal{D}^l = \{(x_i^l, y_i^l)\}_{i=1}^{N_l}$ and a large unlabeled image set $\mathcal{D}^u = \{x_i^u\}_{i=1}^{N_u}$, where x_i^l/x_i^u and y_i^l are the training examples and labels, respectively, and N_l and N_u are the number of labeled and unlabeled images, respectively. In this section, we present CorrMatch, which leverages threshold relaxing and label propagation with pairwise correlations to mine more accurate high-confidence regions of unlabeled data for semi-supervised semantic segmentation.

3.1 Framework

CorrMatch is built upon a simple single-stage framework with weak-to-strong consistency regularization as shown in Fig. 1. For labeled images, like most methods, a supervised loss is applied. Given a mini-batch of labeled images $\mathcal{B}^l \subset \mathcal{D}^l$, the supervised loss $\mathcal{L}_s^h$ can be written as:

$$\mathcal{L}_s^h = \frac{1}{|\mathcal{B}^l|} \sum_{i=1}^{|\mathcal{B}^l|} \ell_{ce}(\mathcal{F}(\mathcal{A}^w(x_i^l)), y_i^l), \tag{1}$$

where $\mathcal{A}^w(\cdot)$ denotes weak data augmentation, $\mathcal{F}(\cdot)$ denotes the predictions of the semantic segmentation network $\mathcal{F}$, and ℓ_{ce} is the pixel-wise cross-entropy loss function.

For the unlabeled images in a mini-batch $\mathcal{B}^u \subset \mathcal{D}^u$, we provide a consistent training process to encourage the outputs to be consistent for both weakly and strongly augmented inputs:

$$\mathcal{L}_u^h = \frac{1}{|\mathcal{B}^u|} \sum_{i=1}^{|\mathcal{B}^u|} \ell_{ce}(\mathcal{F}(\mathcal{A}^s(x_i^u)), \hat{y}_i^u \circ \mathcal{M}_i), \tag{2}$$

where $\mathcal{A}^s(\cdot)$ denotes strong data augmentation, $\hat{y}_i^u = \mathcal{F}(\mathcal{A}^w(x_i^u))$ is the pseudo label from the predictions of the weakly augmented images, $\circ$ is the element-wise multiplication. And $\mathcal{M}_i$ is a binary map indicating the positions with high confidence predictions in $\hat{y}_i^u$, which can be written as:

$$\mathcal{M}_i = \mathbb{1}(\max(\hat{\mathcal{F}}(\mathcal{A}^w(x_i^u))) > \tau), \tag{3}$$

where $\hat{\mathcal{F}}(\cdot)$ is the logits produced by the semantic segmentation network $\mathcal{F}$ and τ is a threshold used to screen high confidence predicted pixels. And in most of the previous work, τ is a fixed and strict value, e.g., 0.95. We view the above single-stage method as our baseline.

However, the baseline only treats $\mathcal{F}(\mathcal{A}^w(x^u))$ as hard pseudo labels and thus ignores the information stored in logits $\hat{\mathcal{F}}(\mathcal{A}^w(x^u))$. Taking this into account, we further consider consistency between the logits of different augmented views. As aforementioned, we have attained the binary filter map $\mathcal{M}$ with high confidence. Thus, we enforce the consistency between the soft logits of weakly augmented images $\hat{\mathcal{F}}(\mathcal{A}^w(x^u))$ and strongly augmented images $\hat{\mathcal{F}}(\mathcal{A}^s(x^u))$ in high confidence regions. In Eqn. (4), we give the formula of $\mathcal{L}_u^s$ for calculating weak-strong consistency using soft pseudo labels.

$$\mathcal{L}_u^s = \frac{1}{|\mathcal{B}^u|} \sum_{i=1}^{|\mathcal{B}^u|} \mathrm{KL}(\hat{\mathcal{F}}(\mathcal{A}^w(x_i^u)), \hat{\mathcal{F}}(\mathcal{A}^s(x_i^u)) \circ \mathcal{M}_i), \tag{4}$$

where $\mathrm{KL}(\cdot)$ is the standard Kullback-Leibler Divergence loss function. Besides, to learn more robust feature representations, we also use the same feature perturbations as UniMatch in our framework. In what follows, we will describe how to mine more reliable high-confidence regions via a threshold relaxing strategy and label propagation with correlation matching.

3.2 Threshold Relaxing

As mentioned in FreeMatch [53], using a fixed threshold τ that is too strict or too loose is detrimental to model convergence. UniMatch [62] has shown that for different data splits, the most suitable fixed thresholds are different. Therefore, we provide a relaxed threshold update strategy that encourages the threshold to start from a looser initialization and gradually adapt it during the training process.

Considering that the threshold should not be too strict at the beginning, we give the threshold a relatively small value τ_0 as initialization. Then, the strategy of updating τ_t (threshold at the t-th iteration) depends on the logits predictions $\hat{\mathcal{F}}(\mathcal{A}^w(x_i^u))$, which show the confidence of the model on unlabeled data and can reflect the current overall learning state.

Specifically, we use the exponential moving average (EMA) to iteratively update the threshold. This process for each iteration is defined as:

$$\tau' = \frac{1}{|L|} \sum_{l \in L} \max[\mathbb{1}(\hat{y}_i^u = l) \circ \overset{c}{\max}(\hat{\mathcal{F}}(\mathcal{A}^w(x_i^u)))], \tag{5}$$

where L is the set of all classes that present in predictions $\hat{y}_i^u$, $\overset{c}{\max}(\cdot)$ denotes taking the maximum value along the channel dimension, and $\circ$ is the element-wise multiplication. This operation means we take the maximum confidence of all predicted classes in weakly augmented unlabeled images and consider their average as the increment for each iteration. Thus, the relaxed threshold is defined as:

$$\tau_t = \begin{cases} \tau_0, & \text{if } t = 0 \\ \lambda \tau_{t-1} + (1-\lambda)\tau', & \text{otherwise} \end{cases} \tag{6}$$

where λ is the momentum decay of EMA. We provide the corresponding pseudo code in the Appendix.

The threshold relaxing strategy maintains a global threshold that loosens first and then tightens, bringing about a fast first and then slow increase for the proportion of high-confidence regions. We will show in Sec. 4.4 that this strategy is insensitive to initialization.

3.3 Label Propagation via Correlation Matching

Besides the threshold relaxing strategy, we also consider treating correlation maps as a medium to measure the cost of matching between features. The correlation maps can be propagated to the final output so that each pixel incorporates the influence of its similarity measure with all pixels.

Our label propagation can be described as three steps: (1) feature extraction, (2) calculating correlation map, and (3) label propagation via correlation matching. Denote the network $\mathcal{F}$ as a combination of encoder $\mathcal{G}$ and decoder $\mathcal{H}$. Given a mini-batch of augmented unlabeled images $\mathcal{A}(\mathcal{B}^u)$, we can extract its features $\mathbf{e}$ through a simple extractor $\mathcal{E}$ from encoder $\mathcal{G}$, which can be written as $\mathbf{e} = \mathcal{E}(\mathcal{G}(\mathcal{A}(\mathcal{B}^u)))$. The extractor $\mathcal{E}$ comprises a 3×3 convolution, followed by batch normalization [23] and an activation layer. Since we have augmentations $\mathcal{A}^w$ and $\mathcal{A}^s$, we can get $\mathbf{e}^w = \mathcal{E}(\mathcal{G}(\mathcal{A}^w(\mathcal{B}^u)))$ and $\mathbf{e}^s = \mathcal{E}(\mathcal{G}(\mathcal{A}^s(\mathcal{B}^u)))$, which denote the extracted features of the unlabeled images with weak and strong data augmentations, respectively. These extracted features exhibit invariant to non-semantic variations and thus enable correlation matching to quantify the degree of pairwise similarity.

Given the features $\mathbf{e} \in \mathbb{R}^{D \times HW}$ of an unlabeled image, where D is the channel dimension, HW is the number of pixels, we then compute the correlation map $\mathcal{C}$ by performing a matrix multiplication operation between all pairs of feature vectors:

$$\mathcal{C} = W_1(\mathbf{e})^\top \cdot W_2(\mathbf{e}), \tag{7}$$

where W_1 and W_2 are two linear transformations and $^\top$ denotes matrix transpose. The correlation map $\mathcal{C} \in \mathbb{R}^{HW \times HW}$ is a 2D matrix. For weakly augmented images $\mathcal{A}^w(\mathcal{B}^u)$ and strongly augmented images $\mathcal{A}^s(\mathcal{B}^u)$, we denote their correlation maps as $\mathcal{C}^w$ and $\mathcal{C}^s$, respectively. These correlation maps enable precise delineation of the corresponding regions belonging to the same object as shown at the top-left corner of Fig. 1 and inspire us to construct representations using correlation matching.

To utilize the correlation map $\mathcal{C}$, we activate $\mathcal{C}$ with a $\mathrm{Softmax}$ function to yield pairwise similarities. Then, we propagate these similarities into model logits outputs $\hat{\mathcal{F}}(\mathcal{A}(\mathcal{B}^u))$ to attain another representation of the prediction $\mathbf{z}$ via label propagation:

$$\mathbf{z} = f(\hat{\mathcal{F}}(\mathcal{A}(\mathcal{B}^u))) \cdot \mathrm{Softmax}(\mathcal{C}/\sqrt{D}), \tag{8}$$

where $f(\cdot)$ is a bilinear interpolation for shape matching. Since the reshaped logits $f(\hat{\mathcal{F}}(\mathcal{A}(\mathcal{B}^u))) \in \mathbb{R}^{K \times HW}$, the output $\mathbf{z}$ also has the same shape $K \times HW$ after the matrix multiplication, where K is the class number. The resulting $\mathbf{z}$ emphasizes the regions of the same object through correlation matching, and hence can help our method more precisely distinguish different objects.

As a complement to the threshold relaxing strategy, $\mathbf{z}$ incorporates pairwise similarities and contains accurate predictions. Therefore, a correlation loss $\mathcal{L}_u^c$ should be calculated between the $\mathbf{z}$ for both the weakly and strongly augmented views and the high-confidence pseudo labels after label propagation, which can be written as follows:

$$\mathcal{L}_u^c = \frac{1}{|\mathcal{B}^u|} \sum_{i=1}^{|\mathcal{B}^u|} \frac{1}{2} (\ell_{ce}(\mathbf{z}_i^w, \hat{y}_i^u \circ \mathcal{M}_i) + \ell_{ce}(\mathbf{z}_i^s, \hat{y}_i^u \circ \mathcal{M}_i)) \tag{9}$$

where $\mathbf{z}_i^w$ and $\mathbf{z}_i^s$ denote the propagated representations of correlation maps $\mathcal{C}_i^w$ and $\mathcal{C}_i^s$ on logits $\hat{\mathcal{F}}(\mathcal{A}^w(x_i^u)), \hat{\mathcal{F}}(\mathcal{A}^s(x_i^u))$ of i-th unlabeled image calculated by Eqn. (8).

In addition, for the labeled images $\mathcal{B}^l$, we compute the cross-entropy loss between $\mathbf{z}^l$ and the ground truth y_i^l as supervised correlation loss $\mathcal{L}_s^c$, where $\mathbf{z}^l$ can be attained using Eqn. (8). It is also worth mentioning that correlation matching and label propagation only participate in the training process and hence do not bring additional computational burdens during the inference process.

3.4 Loss Function

The overall objective function $\mathcal{L}$ is a combination of supervised loss $\mathcal{L}_s$ and unsupervised loss $\mathcal{L}_u$: $\mathcal{L} = \frac{1}{2}(\mathcal{L}_s + \mathcal{L}_u)$. Like most methods, we use the cross-entropy loss function in $\mathcal{L}_s^h$ and $\mathcal{L}_s^c$ as the supervision of labeled data $\mathcal{D}^l$. Therefore, the supervised loss $\mathcal{L}_s$ is defined as: $\mathcal{L}_s = \frac{1}{2}(\mathcal{L}_s^h + \mathcal{L}_s^c)$. As for unsupervised loss $\mathcal{L}_u$ on unlabeled data $\mathcal{D}^u$, it can be expressed as follows:

$$\mathcal{L}_u = \lambda_1 \mathcal{L}_u^h + \lambda_2 \mathcal{L}_u^s + \lambda_3 \mathcal{L}_u^c \tag{10}$$

where $\mathcal{L}_u^h$, $\mathcal{L}_u^s$ and $\mathcal{L}_u^c$ denote the hard-pseudo-label loss, soft-pseudo-label loss, and correlation loss. $[\lambda_1, \lambda_2, \lambda_3]$ are set to $[0.5, 0.25, 0.25]$ by default.

4 Experiments

4.1 Experiment Setup

Datasets. We report results on the Pascal VOC 2012 and Cityscapes datasets. Pascal VOC 2012 is a standard semantic segmentation benchmark with 21 semantic classes, consisting of 1,464 high-quality annotated images for training and 1,449 images for evaluation originally [10]. Following recent works, we also conduct experiments on the aug Pascal VOC 2012 dataset, which contains more coarsely annotated images from the Segmentation Boundary Dataset (SBD) [18], resulting in 10,582 training images in total. Cityscapes is an urban scene understanding dataset, including 2,975 training

and 500 validation images with fine annotations [7]. It contains 19 semantic classes of urban scenes, and all images have the resolution of 1024×2048.

Implementation details. Following most previous semi-supervised semantic segmentation methods, we use DeepLabV3+ [5] with ResNet-101 [19] pre-trained on ImageNet [8] as the backbone. For the training on the Pascal VOC 2012 dataset, we use stochastic gradient descent (SGD) optimizer with an initial learning rate of 0.001, weight decay of 1e−4, crop size of 321×321 or 513×513, batch size of 16, and training epochs of 80. For the Cityscapes dataset, following UniMatch [62], we use stochastic gradient descent (SGD) optimizer with an initial learning rate of 0.005, weight decay of 1e−4, crop size of 801 × 801, batch size of 16, and training epochs of 240 with 4 × A40 GPUs.

As for evaluation metrics, we report the mean Intersection-over-Union (mIoU) with original images following previous papers [6, 11, 39] for the Pascal VOC 2012 dataset. For Cityscapes, same as previous methods [6, 54, 62], we apply slide window evaluation with a fixed crop in a sliding window manner and then calculate mIoU on these cropped images. All the results are measured on the standard validation set based on single-scale inference.

Table 1: Comparisons of our CorrMatch with the state-of-the-art approaches on the Pascal VOC 2012 val set in terms of mIoU (%). All methods are trained on the classic setting, i.e., the labeled images are selected from the original VOC train set, which consists of 1,464 samples in total.

Method	Training Size	1/16 (92)	1/8 (183)	1/4 (366)	1/2 (732)	Full (1464)
Supervised Only	321 × 321	45.4	54.7	65.4	71.5	72.4
ST++ [63]	321 × 321	65.2	71.0	74.6	77.3	79.1
UniMatch [62]	321 × 321	75.2	77.2	78.8	79.9	81.2
Mean Teacher [50]	513 × 513	51.7	58.9	63.9	69.5	71.0
CutMix-Seg [11]	513 × 513	52.2	63.5	69.5	73.7	76.5
PseudoSeg [75]	513 × 513	57.6	65.5	69.1	72.4	73.2
CPS [6]	513 × 513	64.1	67.4	71.7	75.9	-
PC ^{2} Seg [71]	513 × 513	57.0	66.3	69.8	73.1	74.2
U^2 PL [54]	513 × 513	68.0	69.2	73.7	76.2	79.5
PS-MT [39]	513 × 513	65.8	69.6	76.6	78.4	80.0
GTA [26]	513 × 513	70.0	73.2	75.6	78.4	80.5
PCR [60]	513 × 513	70.1	74.7	77.2	78.5	80.7
RC^{2}L [68]	513 × 513	65.3	68.9	72.2	77.1	79.3
CCVC [60]	513 × 513	70.2	74.4	77.4	79.1	80.5
CorrMatch	321×321 (513×513)	**76.2**	**78.3**	78.9 (**79.4**)	80.4 (**80.6**)	81.3 (**81.4**)

4.2 Comparison with State-of-the-art Methods

Results on classic Pascal VOC 2012. We show the performance of our method with other state-of-the-art methods on the classic Pascal VOC 2012 Dataset in Tab. 1. Our experiments are conducted on various splits of the original train set following the data partition in CPS [6]. Compared to the supervised baseline, our method gets +30.8%, +23.6%, +14.0%, and +9.1% on 1/16, 1/8, 1/4, and 1/2 split respectively. On the full split, our method gets 81.4% mIoU, 9.0% higher than the baseline. Also, CorrMatch achieves consistent performance gains compared to existing state-of-art approaches. Particularly, CorrMatch outperforms UniMatch by 1.0%, 1.1%, 0.6%, 0.7% and 0.2% on each split.

Results on aug Pascal VOC 2012. In Tab. 2, we show our performance with existing methods on the aug Pascal VOC 2012 Dataset. It is clear that our results are consistently much better than the existing best ones. Our experiments are conducted on 1/16, 1/8, and 1/4 splits, respectively. Under 321×321 training size, compared to the supervised baseline, CorrMatch gets +11.2%, +7.4%, and +5.5% improvements. Our approach outperforms UniMatch by 0.3%, 0.8%, and 1.1% on each split. As for 513×513 training size, we also consistently outperform the current SOTAs. For instance, we get 79.3% mIoU on the 1/8 split with a gain of around 1% compared to UniMatch.

We also report the results using the same splits as in U^{2}PL [54], which contain more well-annotated labels and have higher expectations on results. Compared to AugSeg [70], our method gain 1.1%, 0.4%, and 0.3% improvements on 1/16, 1/8, and 1/4 splits respectively. Also, same to other methods, we observe that, as the split size increases from 1/8 to 1/4, the performance decreases under this setting. This phenomenon is due to the fact that in the 1/8 split, almost all of the accurately labeled images are included, and most of the images added to the larger split are coarsely labeled, which leads to no improvement in performance.

Table 2: Comparisons of state-of-the-art methods on the Pascal VOC 2012 val set with mIoU (%) ↑ metric. All methods are trained on the aug setting, i.e., the labeled images are selected from the aug VOC train set, which consists of 10, 582 samples in total. † means using the same split as U^{2}PL [54].

Method	Train size	1/16 (662)	1/8 (1323)	1/4 (2646)
Supervised	321 × 321	65.6	70.4	72.8
ST++ [63]	321 × 321	74.5	76.3	76.6
CAC [32]	321 × 321	72.4	74.6	76.3
UniMatch [62]	321 × 321	76.5	77.0	77.2
CorrMatch	321 × 321	**76.8**	**77.8**	**78.3**
U2PL† [54]	513 × 513	77.2	79.0	79.3
GTA† [26]	513 × 513	77.8	80.4	80.5
PCR† [60]	513 × 513	78.6	80.7	80.7
CCVC† [60]	513 × 513	76.8	79.4	79.6
AugSeg† [70]	513 × 513	79.3	81.5	80.5
CorrMatch†	513 × 513	**80.4**	**81.9**	**80.8**

Method	Train size	1/16 (662)	1/8 (1323)	1/4 (2646)
CutMix-Seg [11]	513 × 513	71.7	75.5	77.3
CCT [42]	513 × 513	71.9	73.7	76.5
GCT [29]	513 × 513	70.9	73.3	76.7
CPS [6]	513 × 513	74.5	76.4	77.7
AEL [21]	513 × 513	77.2	77.6	78.1
FST [9]	513 × 513	73.9	76.1	78.1
ELN [31]	513 × 513	-	75.1	76.6
U2PL [54]	513 × 513	74.4	77.6	78.7
PS-MT [39]	513 × 513	75.5	78.2	78.7
AugSeg [70]	513 × 513	77.0	77.3	78.8
UniMatch [62]	513 × 513	78.1	78.4	79.2
CorrMatch	513 × 513	**78.4**	**79.3**	**79.6**

mei**Results on Cityscapes**. In Tab. 3, we compare the performance of CorrMatch with SOTAs on the more challenging Cityscapes dataset. We follow sliding window evaluation and online hard example mining (OHEM) loss [47] techniques, which have been widely applied in previous SOTA works [6, 54, 39, 62, 60, 21]. It can be clearly seen that our method can consistently outperform current SOTAs under all splits. Compared to UniMatch [62], CorrMatch achieves +0.7%, +0.2%, +0.2%, and +0.8% on 1/16, 1/8, 1/4, 1/2 splits.

Table 3: Comparing results of state-of-the-art algorithms on Cityscapes val set. All the experiments are conducted with ResNet-101 as the backbone.

Method	1/16 (186)	1/8 (372)	1/4 (744)	1/2 (1488)
Supervised	65.7	72.5	74.4	77.8
CCT [42]	69.3	74.1	76.0	78.1
CPS [6]	69.8	74.3	74.6	76.8
AEL [21]	74.5	75.5	77.5	79.0
U^{2}PL [54]	70.3	74.4	76.5	79.1
PS-MT [39]	-	76.9	77.6	79.1
UniMatch[62]	76.6	77.9	79.2	79.5
PCR [60]	73.4	76.3	78.4	79.1
CorrMatch	**77.3**	**78.1**	**79.4**	**80.3**

4.3 Ablations Studies

We conduct a series of ablations studies on the original Pascal VOC 2012 using ResNet-101 as the encoder with training size 321 × 321.

4.3.1 Effectiveness of Components

We first conduct ablation studies on different components of our CorrMatch to demonstrate their effectiveness in Tab. 4. Our baseline is with hard unsupervised loss and relaxed threshold, that is, under the supervision of $\mathcal{L}_s^h$ and $\mathcal{L}_u^h$. We get 79.3% on the 732 split and 80.0% on the 1464 split.

Based on the basic framework, adding soft pseudo-label loss $\mathcal{L}_u^s$ and correlation loss $\mathcal{L}_u^c$ respectively further brought 0.4% and 0.5%, 0.4% and 0.6% improvements. These results demonstrate the effectiveness of each of our components individually. Besides, we also tried to replace $\mathcal{L}_u^h$ with $\mathcal{L}_u^s$, resulting in a performance decrease, which illustrates the importance of $\mathcal{L}_u^h$. Finally, the complete CorrMatch achieves 80.4% and 81.3% mIoU, which is $+1.1\%$ and $+1.3\%$ compared to the baseline.

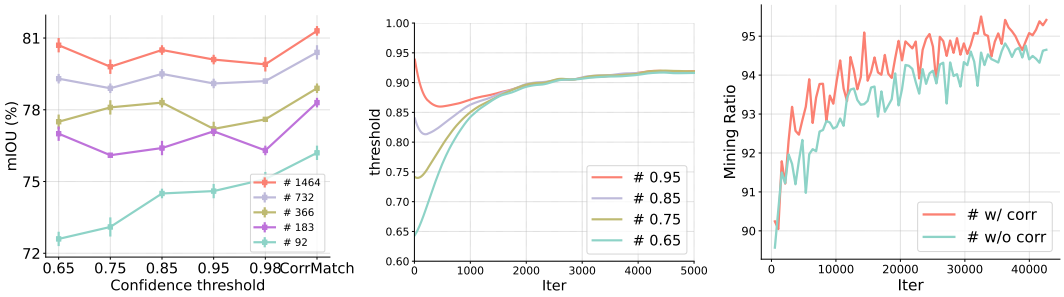
We also conduct experiments with the fixed threshold (0.95). It can be observed that compared to the fixed baseline (79.0% and 79.9%), changing it into a relaxed manner only brings $+0.3\%$ and $+0.1\%$. Meanwhile, after adding $\mathcal{L}_u^s$ and $\mathcal{L}_u^c$, the corresponding improvements came to +1.0% and +0.9%. That means only using the threshold relaxing strategy does not bring significant improvements, but with the complementary label propagation strategy, we can better mine more accurate high-confidence pixels, thus bringing performance improvements.

Table 4: Ablation study on the effectiveness of different components, including threshold τ, hard pseudo-label loss $\mathcal{L}_u^h$, correlation loss $\mathcal{L}_u^c$, soft pseudo-label loss $\mathcal{L}_u^s$.

τ	$\mathcal{L}_u^h$	$\mathcal{L}_u^s$	$\mathcal{L}_u^c$	732	1464
Fixed	✓			79.0	79.9
Fixed	✓	✓		79.2	79.9
Fixed	✓		✓	79.1	80.1
Fixed	✓	✓	✓	79.4	80.4
Relax	✓			79.3	80.0
Relax		✓		77.9	79.6
Relax	✓	✓		79.7	80.5
Relax	✓		✓	79.7	80.6
Relax	✓	✓	✓	**80.4**	**81.3**

Table 5: Ablation study on the position of feature extraction. We take features after these specific modules of DeepLabV3+ and use them to build correlation maps respectively.

Splits	Position	mIoU (%)
732	Backbone	**80.4**
	ASPP	79.5
	Fusion	79.1
	Classifier	79.5
1464	Backbone	**81.3**
	ASPP	80.6
	Fusion	80.1
	Classifier	80.8



(a) Different fixed threshold (b) Different threshold initialization (c) Mining ratio

Figure 2: Some statistics on the threshold strategy and label propagation. For (b) and (c), the experiments are conducted on the 1464 split with an input size of 321×321.

4.3.2 Where to Extract Features

In the default setting, we choose to extract features from the backbone, which makes correlation matching more convenient to be transplanted to other segmentation networks. Actually, given a specific network structure, the position of feature extraction can be flexible. Here, we consider the impact of different feature extraction positions on performance based on the Deeplabv3+ architecture.

In Tab. 5, we demonstrate the performance of extracting features after different positions for the Deeplabv3+ decoder under 732 and 1464 splits. The results show that using the backbone features consistently outperforms other alternatives.

4.4 Impact of the Threshold Strategy

Different fixed values. To further show the necessity of using a non-fixed threshold, we conduct experiments on different splits (92, 366, 732, 1464) with different fixed thresholds (0.65, 0.75, 0.85, 0.95, and 0.98) in Fig. 2a. We can clearly see that for different splits, the highest performance often appears at different fixed thresholds. For instance, using 0.98 is more suitable for the 92 split while 0.65 is better for the 1464 split. In the last column of Fig. 2a, we show the performance of CorrMatch. We can see that our CorrMatch can consistently achieve the best results on all splits, demonstrating the effectiveness of the proposed method.

Different initial values for CorrMatch. Since our EMA-based threshold relaxing strategy needs an initial value for τ, we discuss the impact of using different initialization values for τ in Fig. 2b. The conclusion is that our strategy is insensitive to different initialization values. Even with different threshold initialization values, all experiments tend to approach a similar value very quickly (around 1500 iterations) in the early stage of training (around 40000 iterations in total).

(a) image (b) objects (c) background (d) w/o corr (e) w/ corr (f) prediction (g) GT

Figure 3: Qualitative results on the Pascal VOC 2012 dataset. (a) input image; (b) correlation map on object; (c) correlation map on background; (d) pseudo label without correlation matching; (e) pseudo label with CorrMatch; (f) prediction of CorrMatch; (g) ground truth. White areas in (d) and (e) are ignored regions due to low confidence.

4.5 Correlation Matching Helps Mining Reliable Regions

Mining ratio. Ideally, all correctly predicted points should be regarded as pseudo-labels for the unlabeled data. To demonstrate the ability of correlation matching to help label propagation, we count the mining ratio, which is the proportion of selected high-confidence pixels among all correctly predicted pixels, in Fig. 2c. It can be clearly seen that with correlation matching, the proportion of pixels considered as pseudo-labels is significantly higher than that without it. This further illustrates the complementarity of the two proposed strategies. Threshold relaxing expands the confidence regions, and correlation matching can propagate pairwise similarity to the entire predictions, thereby assisting in discovering more accurate predictions with high confidence.

Qualitative analysis. In Fig. 3, we give some visualization results to further demonstrate the effectiveness of our correlation matching. Comparing Fig. 3e and Fig. 3d, it is obvious that with the support of correlation matching, the number of pixels and completeness of the high-confidence regions are significantly better than those without it. Besides the pseudo labels, we also show the correlation maps corresponding to an object pixel and a background pixel in Fig. 3b and Fig. 3c. It can be seen that the correlation maps with respect to both the object and the background can clearly capture their respective semantic regions. In the subsequent label propagation process, these attentive regions will be propagated to the corresponding pixels on the outputs and hence can help discover more high-confidence regions.

5 Conclusions

We present CorrMatch that utilizes an adaptive threshold strategy and label propagation with correlation matching to discover more accurate high-confidence regions for semi-supervised semantic segmentation. The key contribution of our CorrMatch is designing the complementary strategies to expand the high-quality regions with a relaxed dynamic threshold first and then propagate high-confidence predictions through correlation maps. We have experimentally demonstrated that our CorrMatch performs better than existing approaches on multiple benchmarks and various data splits.

6 Limitations and Discussions

A potential limitation of our CorrMatch is the additional computational burden of correlation matching during training, even though this process does not occur during model inference. As for broader impacts, our experiments are conducted on public datasets and to the best of our knowledge, have no ethical issues. We hope the efficiency of CorrMatch can inspire more researchers to explore how to more efficiently mine more high-confidence regions for the semi-supervised semantic segmentation.

A Algorithm for Threshold Relaxing

In Sec. 3.2 of our main paper, we propose the threshold relaxing strategy. Our core idea is maintaining a dynamic global threshold related to the model's learning process with a looser initialization. Specifically, during the optimization process, we gradually update the threshold using the average of the maximum confidence of all predicted classes in weakly augmented predictions. To make things more clear, we here present the pseudocode of the threshold relaxing strategy in a PyTorch-like style.

Algorithm 1 Pseudocode of threshold relaxing strategy in a PyTorch-like style.

```
# pred: logits prediction of weak augmented images
# thresh_global: current global threshold
# momentum: coefficient of EMA
def update(pred, thresh_global, momentum):
    # initialize update value
    update_value = 0.0

    # get predicted mask and confidence from pred
    mask_pred = torch.argmax(pred, dim=1)
    pred_conf = pred.softmax(dim=1).max(dim=1)

    # find all classes in the predicted mask
    unique_cls = torch.unique(mask_pred)
    cls_num = len(unique_cls)

    for cls in unique_cls:
        # find the highest confidence score for each predicted class
        cls_map = (mask_pred == cls)
        pred_conf_cls_all = pred_conf[cls_map]
        cls_max_conf = pred_conf_cls_all.max()
        update_value += cls_max_conf

    # get the mean of all confidence scores
    update_value = update_value / cls_num

    # update thresh_global in EMA style
    thresh_global = momentum * thresh_global + (1 - momentum) * update_value
```

B More Implementation Details

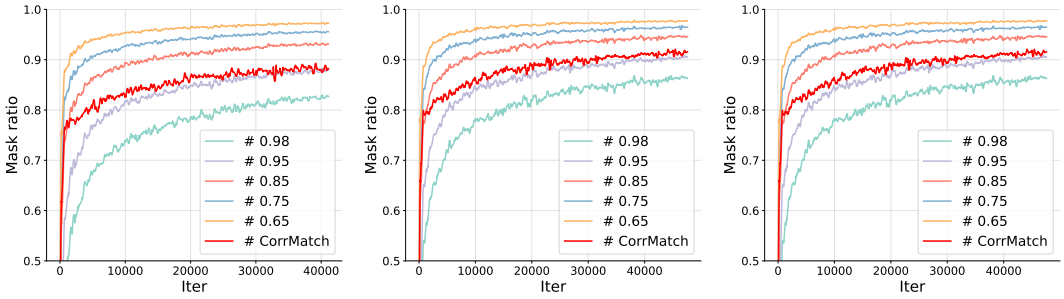
Data augmentations. We followed the common settings from previous works [63, 62, 75]. For weak data augmentation $\mathcal{A}^w$, we use the random scale with a range [0.5, 2.0], the random horizontal flip with a probability of 0.5, and the random crop with a certain size (321, 513, or 801). As for strong data augmentation $\mathcal{A}^s$, we use the colorjitter technology to change the brightness, contrast, saturation, and hue of the image with the same parameter setting as previous works [63, 62, 75]. Random grayscale and gaussian blur are also applied as strong data augmentations. We also use the CutMix [65] technology as done in UniMatch [62]. Besides, as for feature perturbation, same as UniMatch [62], we randomly dropout 50% of the channels from the encoder feature.

Others. We use the stochastic gradient descent (SGD) optimizer with momentum $= 0.9$ and the poly scheduling with $(1 - \frac{\text{iter}}{\text{total iter}})^{0.9}$ to decay the learning rate during the training process. Furthermore, we set the momentum of EMA to 0.999 for the proposed threshold relaxing strategy.

C More Analysis for Threshold Relaxing

C.1 Mask Ratio

In Fig. 4, we demonstrate the mask ratio (proportion of high-confidence pixels filtered by the threshold) during the training process. We compare the mask ratio statistics using fixed thresholds with using our CorrMatch. It is obvious that the lower the fixed threshold is, the higher the mask ratio will be. Moreover, a too-low mask ratio in the early training will lead to fewer predictions that constitute pseudo-labels, which will affect the convergence speed. On the contrary, a too-high mask ratio in the later training will contain more wrong predictions, which will affect the accuracy of pseudo-labels. Both situations are detrimental to model convergence. However, our CorrMatch tackles this problem by achieving a relatively higher mask ratio early and a relatively lower mask ratio later. This phenomenon maintains a consistent trend in Fig. 4a, Fig. 4b, and Fig. 4c, thus further verifying the stability of our method.



(a) 1464 split (b) 366 split (c) 92 split

Figure 4: Mask ratio during the training process of different splits with different fixed thresholds.

Table 6: Comparison of CorrMatch with and without per-class thresholding strategy on PASCAL VOC 2012 val set with mIoU (%) ↑ metric. * means with per-class threshold relaxing strategy.

Method	1 / 16(92)	1 / 8(183)	1 / 4(366)	1 / 2(732)	Full (1464)
CorrMatch	**76.2**	**78.3**	**78.9**	**80.4**	**81.3**
CorrMatch*	75.1	76.7	78.3	79.3	80.3

C.2 Why not Per-class Threshold Relaxing

Considering that we propose a dynamic threshold updating strategy in this paper, it might be argued that using a dynamic threshold update strategy for each class may lead to performance improvements since it has shown success in semi-supervised classification tasks [53, 66]. However, the classification and semantic segmentation tasks have different characteristics. That is, there may be multiple categories existing in each image and the confidence of the same class at different pixel positions are various. Therefore, a similar strategy may be not suitable for semi-supervised semantic segmentation tasks. To further illustrate this point, we conduct the following per-class thresholding strategy.

We first initialize a tensor with the same size as the number of categories, and its value is the same as the global initialization value. We use a similar EMA style to iteratively update strategy as global threshold relaxing. For each predicted class l in model predictions $\hat{y}_i^u$, the process for each iteration is defined as:

$$\tau_l' = \max[\mathbb{1}(\hat{y}_i^u = l) \circ \overset{c}{\max}(\hat{\mathcal{F}}(\mathcal{A}^w(x_i^u)))], \tag{11}$$

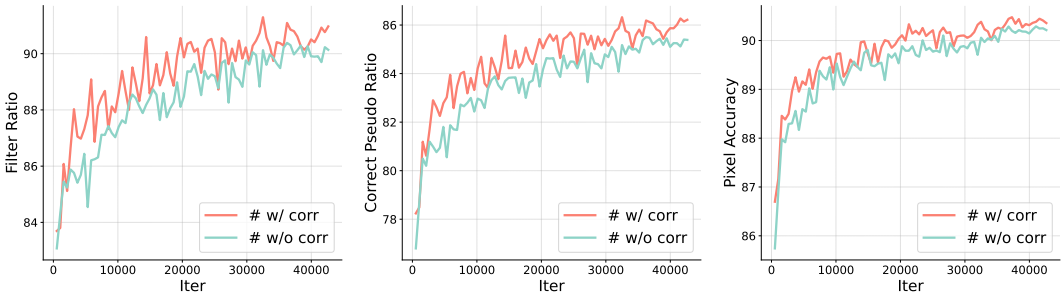
where $\hat{\mathcal{F}}(\mathcal{A}^w(x_i^u))$ is the logits prediction of unlabeled images with weak data augmentations. This operation means we take the maximum confidence of each predicted class in weakly augmented unlabeled images and consider them as the increment for each class at each iteration. Then, similar to FreeMatch [53], we use maximum normalization operation to integrate the global and local thresholds.

Here we conduct experiments on the original Pascal VOC 2012 dataset with 321×321 training size in Tab. 6. It can be clearly seen that converting it to a per-class scheme brings around 1% performance drop compared to the global threshold relaxing strategy.

D More Analysis for Label Propagation

D.1 Statistics

In Fig. 5, we demonstrate more statistics on the val set of the Pascal VOC 2012 dataset to further show the effectiveness of label propagation via correlation matching. We count the filter ratio, correct pseudo ratio, and pixel accuracy with and without adopting correlation matching in Fig. 5a, Fig. 5b, and Fig. 5c, respectively. The filter ratio is the proportion of high-confidence pixels that are regarded as pseudo-labels for the whole image, which can reflect the overall confidence of the model. The correct pseudo ratio is the proportion of accurately predicted pseudo labels to the whole image, which can reflect effective pseudo-labels numbers. And the pixel accuracy is all accurately predicted pixels to the whole image. All the experiments are conducted on the 1464 split with training size 321×321.



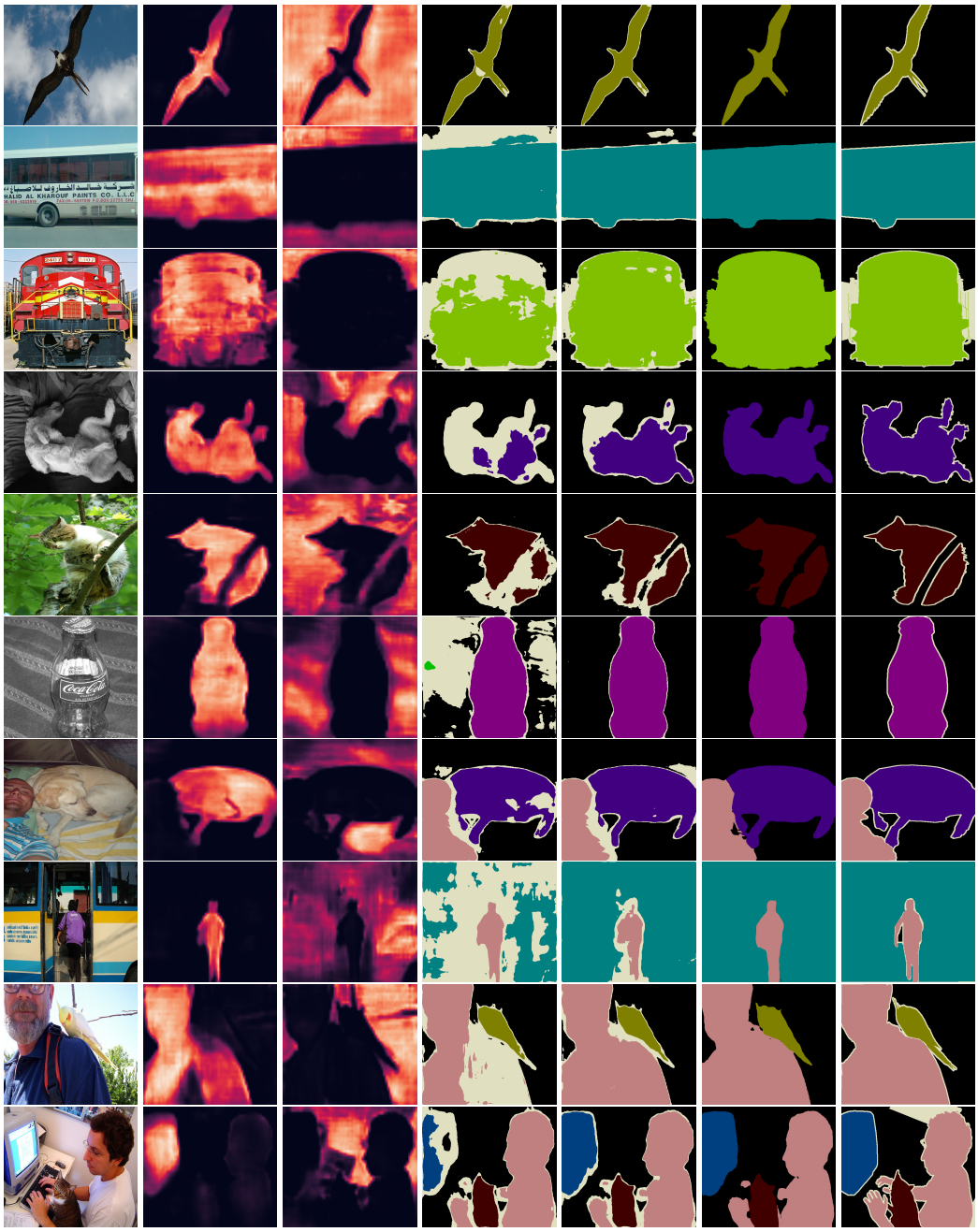
(a) Filter Ratio (b) Correct Pseudo Ratio (c) Pixel Accuracy

Figure 5: More statistics about label propagation via correlation matching.

It can be clearly seen that the trend of the two curves in these three figures is consistent. That is, using correlation matching can yield much better results. This means that not only does the model tend to make predictions with overall higher confidence, but the number of high-confidence pixels that are correctly predicted increases. Also, higher pixel accuracy with correlation matching indicates better performance of the model itself. These statistics further demonstrate that our proposed CorrMatch with the label propagation strategy can mine more accurate high-confidence regions and thus boost the model to learn more from the unlabeled data.

D.2 More Visualizations

In our main paper, we claim that label propagation via correlation matching helps mining reliable regions and we have verified this through both extensive quantitative and qualitative experiments. Here, we present more qualitative results in Fig. 6 to further support our conclusion.



(a) image (b) objects (c) background (d) w/o corr (e) w/ corr (f) prediction (g) GT

Figure 6: More qualitative results from the val set of Pascal VOC 2012 dataset. (a) input image; (b) correlation map on object; (c) correlation map on background; (d) pseudo label without correlation matching; (e) pseudo label with CorrMatch; (f) prediction of CorrMatch; (g) ground truth. White areas in (d) and (e) are ignored regions due to low confidence.

References

[1] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. *Advances in neural information processing systems*, 27, 2014. 2

[2] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning*

research, 7(11), 2006. 2

[3] Kristin Bennett and Ayhan Demiriz. Semi-supervised support vector machines. *Advances in Neural Information processing systems*, 11, 1998. 2

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32, 2019. 2

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 1, 3, 6

[6] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2613–2622, 2021. 2, 6, 7

[7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 6

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1, 6

[9] Ye Du, Yujun Shen, Haochen Wang, Jingjing Fei, Wei Li, Liwei Wu, Rui Zhao, Zehua Fu, and Qingjie Liu. Learning from future: A novel self-training framework for semantic segmentation. *arXiv preprint arXiv:2209.06993*, 2022. 2, 7

[10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–308, 2009. 5

[11] Geoffrey French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. In *British Machine Vision Conference*, 2020. 1, 2, 6, 7

[12] Shanghua Gao, Zhong-Yu Li, Ming-Hsuan Yang, Ming-Ming Cheng, Junwei Han, and Philip Torr. Large-scale unsupervised semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–20, 2022. 1

[13] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(2):652–662, 2021. 1

[14] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004. 2

[15] Sascha Grollmisch and Estefanía Cano. Improving semi-supervised learning for audio classification with fixmatch. *Electronics*, 10(15):1807, 2021. 2

[16] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zhengning Liu, Ming-Ming Cheng, and Shi-Min Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. *arXiv preprint arXiv:2209.08575*, 2022. 1

[17] Robert Harb and Patrick Knöbelreiter. Infoseg: Unsupervised semantic image segmentation with mutual information maximization. In *Pattern Recognition: 43rd DAGM German Conference, DAGM GCPR 2021, Bonn, Germany, September 28–October 1, 2021, Proceedings*, pages 18–32. Springer, 2022. 1

[18] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 international conference on computer vision*, pages 991–998. IEEE, 2011. 5

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 6

[20] Seunghoon Hong, Hyeonwoo Noh, and Bohyung Han. Decoupled deep neural network for semi-supervised semantic segmentation. *Advances in neural information processing systems*, 28, 2015. 1, 2

[21] Hanzhe Hu, Fangyun Wei, Han Hu, Qiwei Ye, Jinshi Cui, and Liwei Wang. Semi-supervised semantic segmentation via adaptive equalization learning. *Advances in Neural Information Processing Systems*, 34:22106–22118, 2021. 1, 2, 7

[22] Jyh-Jing Hwang, Stella X Yu, Jianbo Shi, Maxwell D Collins, Tien-Ju Yang, Xiao Zhang, and Liang-Chieh Chen. Segsort: Segmentation by discriminative sorting of segments. In

Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 7334–7344, 2019. 1

[23] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015. 2, 4

[24] Peng-Tao Jiang, Ling-Hao Han, Qibin Hou, Ming-Ming Cheng, and Yunchao Wei. Online attention accumulation for weakly supervised semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7062–7077, 2022. 1

[25] Peng-Tao Jiang, Yuqi Yang, Qibin Hou, and Yunchao Wei. L2g: A simple local-to-global knowledge transfer framework for weakly supervised semantic segmentation. In *IEEE CVPR*, 2022. 1

[26] Ying Jin, Jiaqi Wang, and Dahua Lin. Semi-supervised semantic segmentation via gentle teaching assistant. In *Advances in Neural Information Processing Systems*, 2022. 6, 7

[27] Thorsten Joachims et al. Transductive inference for text classification using support vector machines. In *Icml*, volume 99, pages 200–209, 1999. 2

[28] Rihuan Ke, Angelica I Aviles-Rivero, Saurabh Pandey, Saikumar Reddy, and Carola-Bibiane Schönlieb. A three-stage self-training framework for semi-supervised semantic segmentation. *IEEE Transactions on Image Processing*, 31:1805–1815, 2022. 1, 2

[29] Zhanghan Ke, Di Qiu, Kaican Li, Qiong Yan, and Rynson WH Lau. Guided collaborative training for pixel-wise semi-supervised learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 429–445. Springer, 2020. 7

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

[31] 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*, pages 9957–9967, 2022. 1, 7

[32] 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*, pages 1205–1214, 2021. 7

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

[34] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013. 2

[35] Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[36] Zhong-Yu Li, Shanghua Gao, and Ming-Ming Cheng. Exploring feature self-relation for self-supervised transformer. *arXiv preprint arXiv:2206.05184*, 2022. 2

[37] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 1

[38] Yuqi Lin, Minghao Chen, Wenxiao Wang, Boxi Wu, Ke Li, Binbin Lin, Haifeng Liu, and Xiaofei He. Clip is also an efficient segmenter: A text-driven approach for weakly supervised semantic segmentation. *arXiv preprint arXiv:2212.09506*, 2022. 2

[39] Yuyuan Liu, Yu Tian, Yuanhong Chen, Fengbei Liu, Vasileios Belagiannis, and Gustavo Carneiro. Perturbed and strict mean teachers for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4258–4267, 2022. 1, 2, 6, 7

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

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

[42] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *Proceedings of the IEEE/CVF Conference on Computer*

Vision and Pattern Recognition, pages 12674–12684, 2020. 1, 2, 7

[43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 1

[44] Lixiang Ru, Yibing Zhan, Baosheng Yu, and Bo Du. Learning affinity from attention: end-to-end weakly-supervised semantic segmentation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16846–16855, 2022. 2

[45] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in neural information processing systems*, 29, 2016. 2

[46] Matthias Seeger. Learning with labeled and unlabeled data, 2000. 2

[47] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 761–769, 2016. 7

[48] 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. *Advances in neural information processing systems*, 33:596–608, 2020. 2, 3

[49] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020. 2

[50] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017. 2, 6

[51] Pratima Upretee and Bishesh Khanal. Fixmatchseg: Fixing fixmatch for semi-supervised semantic segmentation. *arXiv preprint arXiv:2208.00400*, 2022. 2

[52] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10052–10062, 2021. 1

[53] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Zhen Wu, and Jindong Wang. Freematch: Self-adaptive thresholding for semi-supervised learning. *arXiv preprint arXiv:2205.07246*, 2022. 2, 4, 11

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

[55] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12275–12284, 2020. 1

[56] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7268–7277, 2018. 1

[57] Hui Xiao, Dong Li, Hao Xu, Shuibo Fu, Diqun Yan, Kangkang Song, and Chengbin Peng. Semi-supervised semantic segmentation with cross teacher training. *Neurocomputing*, 508:36–46, 2022. 2

[58] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698, 2020. 2

[59] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 1

[60] Hai-Ming Xu, Lingqiao Liu, Qiuchen Bian, and Zhen Yang. Semi-supervised semantic segmentation with prototype-based consistency regularization. *arXiv preprint arXiv:2210.04388*, 2022. 1, 2, 6, 7

[61] 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 *International Conference on Machine Learning*, pages 11525–11536. PMLR, 2021. 2

[62] Lihe Yang, Lei Qi, Litong Feng, Wayne Zhang, and Yinghuan Shi. Revisiting weak-to-strong consistency in semi-supervised semantic segmentation. In *IEEE conference on computer vision and pattern recognition*, 2023. 1, 2, 3, 4, 6, 7, 10

[63] 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*, pages 4268–4277, 2022. 1, 2, 6, 7, 10

[64] Jianlong Yuan, Yifan Liu, Chunhua Shen, Zhibin Wang, and Hao Li. A simple baseline for semi-supervised semantic segmentation with strong data augmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8229–8238, 2021. 1, 2

[65] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019. 10

[66] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34:18408–18419, 2021. 2, 11

[67] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2736–2746, 2022. 1

[68] Jianrong Zhang, Tianyi Wu, Chuanghao Ding, Hongwei Zhao, and Guodong Guo. Region-level contrastive and consistency learning for semi-supervised semantic segmentation. *arXiv preprint arXiv:2204.13314*, 2022. 2, 6

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

[70] Zhen Zhao, Lihe Yang, Sifan Long, Jimin Pi, Luping Zhou, and Jingdong Wang. Augmentation matters: A simple-yet-effective approach to semi-supervised semantic segmentation. In *CVPR*, 2023. 6, 7

[71] Yuanyi Zhong, Bodi Yuan, Hong Wu, Zhiqiang Yuan, Jian Peng, and Yu-Xiong Wang. Pixel contrastive-consistent semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7273–7282, 2021. 1, 3, 6

[72] Yanning Zhou, Hang Xu, Wei Zhang, Bin Gao, and Pheng-Ann Heng. C3-semiseg: Contrastive semi-supervised segmentation via cross-set learning and dynamic class-balancing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7036–7045, 2021. 2

[73] Xiaojin Jerry Zhu. Semi-supervised learning literature survey, 2005. 2

[74] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. Rethinking pre-training and self-training. *Advances in neural information processing systems*, 33:3833–3845, 2020. 2

[75] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. Pseudoseg: Designing pseudo labels for semantic segmentation. *arXiv preprint arXiv:2010.09713*, 2020. 6, 10