# Improving Semi-Supervised Semantic Segmentation with Dual-Level Siamese Structure Network

Zhibo Tian
tianzhb21@lzu.edu.cn
School of Information Science and Engineering,
Lanzhou University
Lanzhou, China

Xiaolin Zhang
solli.zhang@gmail.com
Independent Researcher
Shenzhen, China

Peng Zhang
pengzhang_skd@sdust.edu.cn
College of Computer Science and Engineering,
Shandong University of Science and Technology
Qingdao, China

Kun Zhan*
kzhan@lzu.edu.cn
School of Information Science and Engineering,
Lanzhou University
Lanzhou, China

## ABSTRACT

Semi-supervised semantic segmentation (SSS) is an important task that utilizes both labeled and unlabeled data to reduce expenses on labeling training examples. However, the effectiveness of SSS algorithms is limited by the difficulty of fully exploiting the potential of unlabeled data. To address this, we propose a dual-level Siamese structure network (DSSN) for pixel-wise contrastive learning. By aligning positive pairs with a pixel-wise contrastive loss using strong augmented views in both low-level image space and high-level feature space, the proposed DSSN is designed to maximize the utilization of available unlabeled data. Additionally, we introduce a novel class-aware pseudo-label selection strategy for weak-to-strong supervision, which addresses the limitations of most existing methods that do not perform selection or apply a predefined threshold for all classes. Specifically, our strategy selects the top high-confidence prediction of the weak view for each class to generate pseudo labels that supervise the strong augmented views. This strategy is capable of taking into account the class imbalance and improving the performance of long-tailed classes. Our proposed method achieves state-of-the-art results on two datasets, PASCAL VOC 2012 and Cityscapes, outperforming other SSS algorithms by a significant margin. The source code is available at https://github.com/kunzhan/DSSN.

## CCS CONCEPTS

• **Computing methodologies** → **Image segmentation**.

## KEYWORDS

Semi-supervised segmentation, pixel-wise contrastive learning, class-aware pseudo-label generation

**ACM Reference Format:**
Zhibo Tian, Xiaolin Zhang, Peng Zhang, and Kun Zhan. 2023. Improving Semi-Supervised Semantic Segmentation with Dual-Level Siamese Structure Network. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3581783.3611816

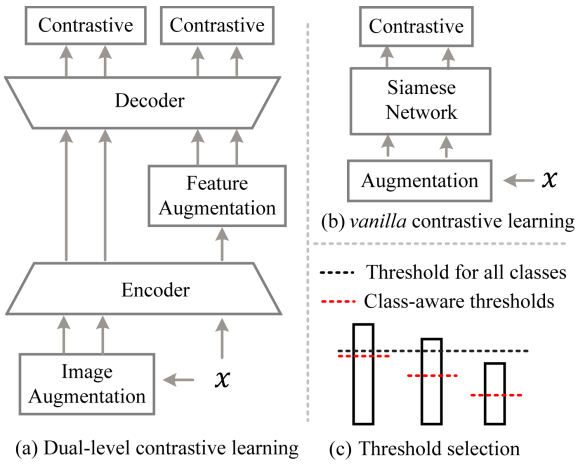


**Figure 1: Illustration of the motivation. (a) demonstrates the proposed dual-level contrastive structure for exploiting the maximum potential of unlabelled samples. (b) depicts the structure of the *vanilla* contrastive learning. (c) compares the threshold selection strategies of the proposed class-aware pseudo-label generation method and the classical approaches of utilizing a threshold for all classes.**

## 1 INTRODUCTION

Deep learning methods for supervised segmentation have shown remarkable performance. However, they heavily rely on a large amount of annotated images, which is labor cost and time-consuming.

*Corresponding author.


*MM '23, October 29-November 3, 2023, Ottawa, ON, Canada*

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3611816

Alternatively, semi-supervised semantic segmentation (SSS) offers a viable solution to address this fundamental weakness by exploiting the readily available unlabeled data to improve model performance.

Existing semi-supervised learning methods typically use unlabeled samples in two ways: pseudo supervision [1, 27] and consistency regularization [20, 28, 31]. Pseudo supervision is to generate pseudo labels for the unlabeled images and gradually incorporates them into the training process to supervise model learning. For example, preliminary works [17, 24] in SSS tend to utilize the generative adversarial networks [6] as auxiliary supervision for unlabeled images. Consistency regularization promotes agreement among model predictions on unlabeled samples that are subjected to various perturbations, thus improving model generalization by ensuring that different views of the same unlabeled image are consistent. Modern SSS algorithms combine pseudo supervision and consistency regularization into a two-view network architecture, where one view generates pseudo labels to supervise the other view for prediction consistency. For instance, the intuition of CPS [3] is that using one view generates pseudo labels of unlabeled images to expand the training set of the other view. PseudoSeg [37] generates pseudo labels in a weak augmented view to supervise the other strong augmented view. PS-MT [22] employs higher-confidence pseudo labels than CPS by averaging the predictions of two views. To search for high-quality pseudo labels, CCT [26] employs a fixed threshold for all classes and pixels with confidence scores above the threshold to participate in network updates. In CCT [26], it mainly uses consistency learning between one weak view and two strong augmented views of a high-level feature.

However, many existing SSS algorithms do not fully exploit the potential of unlabelled data. To address this issue, we propose a Dual-level Siamese structure network (DSSN) to fully exploit feature diversities. In addition to the two strategies commonly used in most algorithms, we introduce a new variant of contrastive learning. Fig. 1(b) illustrates a typical structure of the *vanilla* contrastive learning, which excels at providing extraordinary generalization abilities for unlabeled samples [4, 13]. Specifically, the proposed DSSN simultaneously employs pixel-wise contrastive learning and two-level strong augmented views. Accordingly, contrastive objectives in terms of image-level and feature-level augmentations are introduced to guide the network training. Such structure guarantees fully exploiting the potential of unlabeled data. As shown in Fig. 1(a), at the image level, two different views of unlabeled samples are obtained with different strong augmentations, and a pixel-wise contrastive objective is added to train DSSN using the corresponding predictions. At the feature level, high-level latent features from the encoder produce two strong augmented views and also conduct a contrastive loss. This DSSN design enables us to fully exploit the available unlabeled data.

Given that most real-world datasets exhibit imbalanced or long-tailed label distributions [23], we propose a class-aware pseudo label generation (CPLG) strategy that selects class-specific high-confidence pseudo labels from weak views to supervise the strong views. Our CPLG strategy differs from previous approaches [11, 26], which apply a fixed threshold to all categories. By treating each class differently, our method aims to improve the performance of long-tailed categories. Without any selection, low-quality pseudo labels generated from the weak augmented view are used to supervise the strong augmented view, which could negatively affect the model training. Using a constant threshold for all classes may result in long-tailed classes being poorly trained, as their confidence may be lower than the threshold and thus not involved in training. Using a fixed threshold may also result in useful pseudo-labels being ignored in some classes that fall below the predefined threshold. For each class has pseudo labels, we select top high-confidence pixels in each class since most segments in an image are imbalances and also it is imbalances in the whole dataset. A schematic illustrating this strategy is presented in Fig. 1(c). This approach increases the contribution of long-tailed classes and addresses the learning difficulties of different classes.

In summary, DSSN makes the following contributions:

(1) DSSN offers a novel approach to leverage unlabeled data in training SSS models by utilizing dual-level pixel-wise contrastive learning. This approach is a valuable addition to the existing techniques of exploiting unlabeled data, such as pseudo-supervision and consistency regularization.

(2) DSSN's design enables the maximal utilization of available unlabeled data. The dual-level structure is not only utilized in contrastive learning but also in weak-to-strong pseudo-supervision.

(3) We introduce a novel class-aware pseudo-label selection strategy for weak-to-strong supervision, known as CPLG. This strategy effectively improves the performance of long-tailed classes.

## 2 RELATED WORK

SSS has two mainstream methods, pseudo supervision and consistency regularization. Preliminary works [17, 24] use the generative adversarial networks [6] to generate pseudo supervision. Specifically, consistency regularization methods encourage consistency prediction of unlabeled samples with various perturbation. The CutMix-Seg [11] approach incorporates the CutMix [35] augmentation into semantic segmentation in order to supply consistency restrictions on unlabeled data and also revealed Cutout [8] and CutMix [35] are critical to the success of consistency regularization. Alternatively, CCT [26] proposes a feature-level perturbation and a cross-consistency training method that enforce consistency between the main decoder predictions and auxiliary decoders. By using two segmentation models with the same structure but different initialization, GCT [19] conducts network perturbation and promotes consistency between the predictions from the two models. In the meantime, CPS [3] constructs two parallel networks to provide cross-pseudo labels for one another. DMT [10] re-weights the loss on different regions based on the disagreement of two different initialized models. Self-training by pseudo labeling is a classic technique that dates back about a decade, taking the most likely class as a pseudo label and training models on unlabeled data is a common method for achieving minimum entropy. Concurrently ST++ [34] also demonstrates that employing suitable data perturbations on unlabeled samples is really quite beneficial for self-training. Unimatch [33] explores the effectiveness of weak-to-strong supervision, leveraging dual strong augmentations.

Contrastive learning is one of the alternative methods that stands out. RoCo [21] and U$^2$PL [30] use InfoNCE loss [25] on the predicted logits, but they not use Siamese structure network as shown in

Fig. 1(b). DSSN obtains better performance than them, which can be seen in the experiment section.

# 3 METHOD

## 3.1 Preliminaries

Following SSS works [3, 21, 34], we use both a small fraction of labeled data $\mathcal{D}_l = \{(X_i, \boldsymbol{T}_i)\}_{i=1}^{M}$ and a large fraction of unlabeled data $\mathcal{D}_u = \{X_i\}_{i=1+M}^{N+M}$. $X_i$ denotes an image, and $\boldsymbol{T_i}$ represents its ground-truth label if $X_i$ is a labeled image. $N$ and $M$ indicate the number of labeled and unlabeled images, respectively, where $N \gg M$ in most cases. To facilitate the calculation of loss functions, we represent each pixel in an image as a vector $\boldsymbol{x}$ since a pixel has values in different channels. Thus, in subsequent sections, we represent each pixel as a vector $\boldsymbol{x}$ with $\boldsymbol{t}$ as its one-hot ground-truth label. Given an image $X = [\boldsymbol{x}_i]$ with the size of $W \times H$ where $W$ and $H$ are the width and height, we denote the pixel by $\boldsymbol{x}_i, i \in \{1, ..., W \times H\}$. The latent high-level feature $\boldsymbol{z}$ corresponding to $\boldsymbol{x}$ is obtained by an encoder $f(\boldsymbol{x}|\theta)$ where $\theta$ is the learnable parameters of the encoder. We yield the predicted logits $\boldsymbol{h}$ by feeding the latent representations $\boldsymbol{z}$ into a decoder $g(\boldsymbol{z}|\varphi)$ where $\varphi$ is the learnable parameters of the decoder. Finally, a softmax layer is added to obtain the ultimate probability for each class, i.e., $\boldsymbol{y} = \text{softmax}(\boldsymbol{h})$.

Given a labeled image, we use a supervised cross-entropy loss,

$$\mathcal{L}_{\text{sup}} = -\sum_{i}\sum_{j\in C} t_{ij} \log y_{ij} \tag{1}$$

where $C = \{1, \ldots, C\}$ and $C$ is the total number of classes. For a unlabeled image, a simple way to generate their pseudo labels $\hat{\boldsymbol{t}}_i$ is to apply a one-hot operation to the predictions, i.e., $\boldsymbol{y}_i$. For the $i$-th pixel of an unlabeled image, we represent the predicted probability of the $i$-th pixel belonging to the $j$-th class as $y_{ij}$ . Specifically, we use the following operation to generate pseudo labels:

$$c = \arg\max_{j\in C}(y_{ij}), \tag{2}$$

$$\hat{t}_{ij} = \begin{cases} 1, & \text{if } j = c \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $c$ denotes the maximal probability within the class $j \in C$, the $\hat{\boldsymbol{t}}_i = [\hat{t}_{ij}]$ is the one-hot pseudo label.

## 3.2 Dual-Level Contrastive Learning

To fully exploit the potential of available unlabeled data, we propose to use DSSN for extracting pixel-wise contrastive positive pairs in different abstraction levels. The low-level image is subjected to two-view strong augmentations,

$$\boldsymbol{x}_i^{ls1} = \text{AugL}_s(\boldsymbol{x}_i), \tag{4}$$

$$\boldsymbol{x}_i^{ls2} = \text{AugL}_s(\boldsymbol{x}_i) \tag{5}$$

where $\boldsymbol{x}_i^{ls1}$ denotes the strong augmented low-level pixel in the first view. The output, $\text{AugL}_s(\cdot)$, is random. $\text{AugL}_s(\cdot)$ generates varying outputs using the same input to augment the data diversity. This increases the diversity, resulting in an improvement in the robustness and generalization ability of the training model.

We use two-view augmented images to obtain its decoded logits,

$$\boldsymbol{h}_i^{ls1} = g(f(\boldsymbol{x}_i^{ls1}|\theta)|\varphi)\,. \tag{6}$$

$$\boldsymbol{h}_i^{ls2} = g(f(\boldsymbol{x}_i^{ls2}|\theta)|\varphi)\,. \tag{7}$$

Analogous to [16], we apply the contrastive objective, i.e., $\mathcal{L}_{\text{cl}}$ to pairwise pixels for learning better representations:

$$\begin{aligned}\mathcal{L}_{\text{cl}} = &-\frac{1}{|\mathcal{P}|}\sum_{(i,i)\in\mathcal{P}} \log d(\boldsymbol{h}_i^{ls1}, \boldsymbol{h}_i^{ls2}) \\ &-\frac{1}{|\mathcal{N}|}\sum_{(i,j)\in\mathcal{N}} \log\big(1 - d(\boldsymbol{h}_i^{ls1}, \boldsymbol{h}_j^{ls2})\big)\end{aligned} \tag{8}$$

where $d(\cdot,\cdot)$ is a similarity score of a pair of logits. $\boldsymbol{h}_i^{ls1}$ and $\boldsymbol{h}_i^{ls2}$ are belong to positive pairs $(i,i) \in \mathcal{P}$ while $\boldsymbol{h}_i^{ls1}$ and $\boldsymbol{h}_j^{ls2}$ are negative pairs $(i,j) \in \mathcal{N}, \forall\, i \neq j$. We use $\mathcal{P}$ and $\mathcal{N}$ to denote the sets of positive and negative pairs, respectively.

Inspired by BYOL [12], we only use the positive pairs. The similarity $d(\cdot,\cdot)$ of positive logits is defined by a Gaussian function,

$$d(\boldsymbol{h}_i^{ls1}, \boldsymbol{h}_i^{ls2}) = \exp\Big(-\big\|\boldsymbol{h}_i^{ls1} - \boldsymbol{h}_i^{ls2}\big\|_2^2\Big)\,. \tag{9}$$

The similarity defined by Eq. (9) implies the similarity is 1 if the pairwise logits are the same while it tends to 0 if their distance is far from each other. From a different perspective, the error $\|\boldsymbol{h}_i^{ls1} - \boldsymbol{h}_i^{ls2}\|_2^2$ of two-view logits is governed by the Gaussian distribution due to the central limit theorem [29], so we also obtain Eq. (9).

Substituting Eq. (9) into Eq. (8) obtains the following loss.

$$\mathcal{L}_{\text{cl}}^{ls} = \frac{1}{W\times H}\sum_{i}\big\|\boldsymbol{h}_i^{ls1} - \boldsymbol{h}_i^{ls2}\big\|_2^2 \tag{10}$$

where we only use pixel-wise positive pairs.

For the high-level feature contrastive learning, we obtain the high-level latent feature with the encoder,

$$\boldsymbol{z}_i^{hw} = f(\text{AugL}_w(\boldsymbol{x}_i)|\theta) \tag{11}$$

where $\text{AugL}_w(\cdot)$ is a weak augmentation for the low-level pixel. The high-level feature is subjected to two-view strong augmentations,

$$\boldsymbol{z}_i^{hs1} = \text{AugH}_s(\boldsymbol{z}_i^{hw}), \tag{12}$$

$$\boldsymbol{z}_i^{hs2} = \text{AugH}_s(\boldsymbol{z}_i^{hw}) \tag{13}$$

We use the two-view augmented features to obtain its decoded logits, $\boldsymbol{h}_i^{hs1} = g(\boldsymbol{z}_i^{hs1}|\varphi)$ and $\boldsymbol{h}_i^{hs2} = g(\boldsymbol{z}_i^{hs12}|\varphi)$ . Then, we use them to construct the contrastive loss,

$$\mathcal{L}_{\text{cl}}^{hs} = \frac{1}{W\times H}\sum_{i}\big\|\boldsymbol{h}_i^{hs1} - \boldsymbol{h}_i^{hs2}\big\|_2^2\,. \tag{14}$$

## 3.3 Weak-to-Strong Pseudo Supervision

To leverage the four predictions generated by a strongly augmented image, we feed the corresponding weakly augmented image into DSSN. Next, we use the prediction of the weak view to generate its pseudo label and supervise the four strong views. Given our dual-level structure, weak-to-strong pseudo supervision is also performed in both levels. Specifically, we use the pseudo labels of the weak view, denoted as $\hat{\boldsymbol{t}}$, to supervise the predictions of the strong views, denoted as $\boldsymbol{y}$,.

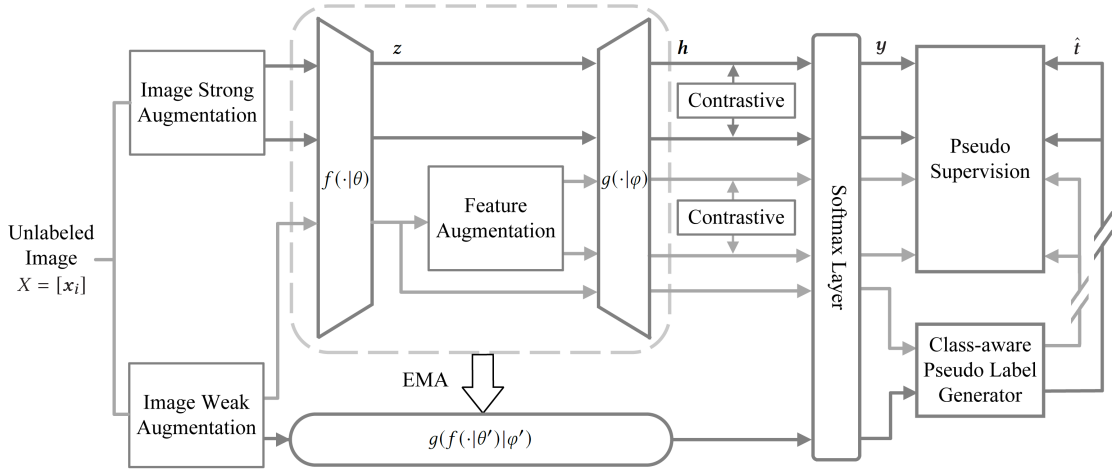


**Figure 2: The overview of DSSN . Dual-level contrastive learning and weak-to-strong pseudo supervision.**

The weak pseudo supervisions are obtained by

$$\boldsymbol{y}^{lw} = \text{softmax}(g(f(\boldsymbol{x}|\theta')|\varphi')) \tag{15}$$

$$\boldsymbol{y}^{hw} = \text{softmax}(g(\boldsymbol{z}^{hw}|\varphi)) \tag{16}$$

where $(\theta', \varphi')$ of the teacher are updated from the student $(\theta, \varphi)$ by the exponential moving average (EMA)

$$(\theta', \varphi') = \alpha(\theta', \varphi') + (1-\alpha)(\theta, \varphi) \tag{17}$$

where $\alpha$ is a momentum parameter, with $\alpha \in [0, 1]$.

The pseudo labels $\hat{\boldsymbol{t}}^{lw}$ and $\hat{\boldsymbol{t}}^{hw}$ of $\boldsymbol{y}^{lw}$ and $\boldsymbol{y}^{hw}$ are calculated by using Eqs. (2) and (3), respectively.

The output probability of the strong augmented views, $\boldsymbol{y}_i^{ls1}$, $\boldsymbol{y}_i^{ls2}$, $\boldsymbol{y}_i^{hs1}$, and $\boldsymbol{y}_i^{hs2}$, are attained by the softmax layer.

The weak-to-strong pseudo-supervision loss functions are

$$\mathcal{L}_{\text{w2s}}^{(l)} = -\sum_i \sum_j m_{ij}^{lw} \left( \hat{t}_{ij}^{lw} \log y_{ij}^{ls1} + \hat{t}_{ij}^{lw} \log y_{ij}^{ls2} \right) \tag{18}$$

$$\mathcal{L}_{\text{w2s}}^{(h)} = -\sum_i \sum_j m_{ij}^{hw} \left( \hat{t}_{ij}^{hw} \log y_{ij}^{hs1} + \hat{t}_{ij}^{hw} \log y_{ij}^{hs2} \right) \tag{19}$$

where $m_{ij}$ is a class-wise binary mask to select the pixel with high-confidence score and we show how to obtain it in the next section.

## 3.4 Class-aware pseudo-label generation

As shown in Fig. 1(c), we show the class-aware pseudo-label generation (CPLG). For the $i$-th pixel, it has different probabilities belonging to different classes. $y_{ij}$ denotes the probability of the $i$-th pixel belonging to the $j$-th class. We observe all pixels in the same class, i.e., in the same channel of network output.

First, we find the pixel class-wisely that has the largest probability in the $j$-th class,

$$y_j^{\max} = \max_i (y_{ij}), \forall\, j \in C\,. \tag{20}$$

Second, we establish a class-wise threshold $\tau_j$ by multiplying the maximum probability by $r\%$. Pixels exceeding this class-wise threshold are selected. Additionally, we restrict the maximum probability by $\tau_{\text{low}}$ and exclude pixels with a low maximum probability since they indicate lower prediction confidence. Thus, the class-wise threshold $\tau_j$ is determined by

$$\tau_j = \begin{cases} y_j^{\max} \cdot r\%, & \text{if } y_j^{\max} > \tau_{\text{low}} \\ y_j^{\max}, & \text{otherwise} \end{cases} \tag{21}$$

where the ratio $r$ and the low bound $\tau_{\text{low}}$ are parameters.

Third, we select pixels in each class by $\tau_j$, i.e., pixels exceeding $\tau_j$ are selected:

$$m_{ij} = \begin{cases} 1, & \text{if } y_{ij} > \tau_j \\ 0, & \text{otherwise}\,. \end{cases} \tag{22}$$

The generation of the pseudo label is straightforward by using Eqs. (2) and (3). The refined class-aware pseudo labels are attained by multiplying them, i.e., $m_{ij}\hat{t}_{ij}$, as used in Eqs. (18) and (19). Our CPLG strategy considers the learning status and difficulties of different classes by adjusting thresholds for each class. As a result, we select useful pixels with low thresholds for training, which enhances the accuracy of challenging classes.

## 3.5 Overall Algorithm

Fig. 2 illustrates how we combine two distinct learning strategies for the unlabeled images: contrastive learning and weak-to-strong pseudo supervision.

In this section, we present the DSSN algorithm, which is illustrated in Algorithm 1. It takes a small fraction of labeled data and a large fraction of unlabeled data as input to train the model. The supervised loss between the model prediction on labeled data and the ground truth is computed using Eq. (1). Subsequently, the low-level and high-level contrastive learning losses are calculated using Eqs. (10) and (14), respectively. We then compute the weak-to-strong pseudo-supervision loss using Eqs. (18) and (19). The overall loss term is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{\text{sup}} + \gamma_1 \left( \mathcal{L}_{\text{cl}}^{ls} + \mathcal{L}_{\text{cl}}^{hs} \right) + \gamma_2 \left( \mathcal{L}_{\text{w2s}}^{(l)} + \mathcal{L}_{\text{w2s}}^{(h)} \right), \tag{23}$$

where $\gamma_1$ and $\gamma_2$ are the trade-off weight. Finally, we update the student model and the teacher model by using the error back-propagation algorithm and EMA, respectively.

**Algorithm 1** The DSSN algorithm.

**Input**: $\mathcal{D} = \{\mathcal{D}_u, \mathcal{D}_l\}$, and batch size $b$.
**Output**: $(\theta', \varphi')$.
**Initialization**: $epoch = 0$, $epoch_{\max}$, and $(\theta, \varphi)$.
**while** $epoch \leq epoch_{\max}$ **do**
  **for** mini-batch samples in $\mathcal{D}$ **do**
    Feed the samples into DSSN for forward propagation;
    Update $\mathcal{L}_{\text{sup}}$ by Eq. (1);
    Update $\mathcal{L}_{\text{cl}}^{ls}$ and $\mathcal{L}_{\text{cl}}^{hs}$ by Eqs. (10) and (14);
    Update $\mathcal{L}_{\text{w2s}}^{(l)}$ and $\mathcal{L}_{\text{w2s}}^{(h)}$ by Eqs. (18) and (19);
    Update $\mathcal{L}$ by Eq. (23);
    Update $(\theta, \varphi)$ by back propagation of $\sum_b \mathcal{L}$;
    Update $(\theta', \varphi')$ by Eq. (17);
    $epoch = epoch + 1$;
  **end for**
**end while**

## 4 EXPERIMENTS

In this section, we first present the details of the experiments. Second, we compare the proposed DSSN method to the recent state-of-the-art (SOTA) approaches to the SSS task. Third, we conduct extensive ablation experiments to demonstrate the effectiveness and robustness of the proposed method.

### 4.1 Experimental setup

**Datasets.** We evaluate the proposed method on two classical semantic segmentation datasets, i.e., PASCAL VOC 2012 [9] and Cityscapes [5]. In particular, PASCAL VOC 2012 [9] has 20 classes of objects and 1 class of background. The standard training, validation and test sets consist of 1,464, 1449 and 1,456 images, respectively. Following the previous work [3, 19, 34], we also use augmented set SBD [14] (9,118 images) and original training set (1,464 images) as our full training set (10,582 images). The labels from the SBD [14] dataset are noise-prone and of low quality. Cityscapes [5] has 19 semantic classes and is mostly intended for understanding urban scenes. It consists of 500 validation images, 1,525 test images, and 2,975 training images. All of the images have well-annotated masks. For a fair comparison with the benchmarks, we follow the partition procedure of CPS [3]. Specifically, the training set is divided into two partitions by randomly sampling 1/2, 1/4, 1/8, and 1/16 of the total set as the labeled samples and the remaining images as the unlabeled for the blended set.

**Implementation details.** Following the previous benchmarks CPS [3], we adopt DeepLab v3+ [2] based on ResNet [15] as the segmentation network for a fair comparison. The backbone i.e., ResNet, is initialized with the weights pre-trained on ImageNet [7]. The segmentation heads are randomly initialized. During training, each mini-batch contains eight labeled and eight unlabeled images. The stochastic gradient descent (SGD) optimizer is used, and the initial learning rates are set to 0.002 and 0.005 for the PASCAL VOC 2012 and Cityscapes, respectively. In accordance with other works [3, 26], we employ the following polynomial to decrease the learning rate while training: $(1 - epoch/epoch_{\max})^{0.9}$. The model is trained for 100 epochs on PASCAL VOC 2012 and 240 epochs for Cityscapes. For *weak augmentations*, we adopt the same operation as ST++[34], where the training images are random flipping and resizing (between 0.5 and 2.0 times), followed by a random crop operation to the resolutions of $513 \times 513$ and $801 \times 801$ for the two datasets, respectively. We employ several *strong augmentation*, including random color-jitter, grayscale, Gaussian blur, etc. For *strong feature augmentation*, we apply a random dropout of 50% on features from the encoder. The unsupervised trade-off weights $\gamma_1$ and $\gamma_2$ are set as 0.01 and 0.25. In CPLG, $r$ is set to 96% and $\tau_{\text{low}}$ is 0.92, respectively.

Additionally, we also apply CutMix [35] data augmentation to the student model images. The EMA smoothing factor $\alpha$ is set as 0.996. We follow U$^2$PL [30], the supervised loss is cross-entropy on PASCAL, and for Cityscapes the cross-entropy loss is replaced by the online hard example mining loss.

**Evaluation.** We use the mean of Intersection-over-Union(mIoU) for the validation set to evaluate the segmentation performance for both datasets. Following the previous works [3, 34], we employ the sliding evaluation to examine the efficacy of our model on the validation images from Cityscapes with a resolution of 1024×2048.

**Table 1: Comparison with SOTAs with ResNet-101. Labeled images are from the original high-quality original training set of PASCAL VOC 2012.**

| Method | 1/16(92) | 1/8(183) | 1/4(366) | 1/2(732) | Full(1464) |
|---|---|---|---|---|---|
| Baseline | 44.10 | 52.30 | 61.80 | 66.70 | 72.90 |
| CutMix-Seg [11] | 52.16 | 63.47 | 69.46 | 73.73 | 76.54 |
| PseudoSeg [37] | 57.60 | 65.50 | 69.14 | 72.41 | 73.23 |
| PC$^2$Seg [36] | 57.00 | 66.28 | 69.78 | 73.05 | 74.15 |
| CPS [3] | 64.07 | 67.42 | 71.71 | 75.88 | - |
| ReCo [21] | 64.78 | 72.02 | 73.14 | 74.69 | - |
| PS-MT [22] | 65.80 | 69.58 | 76.57 | 78.42 | 80.01 |
| ST++ [34] | 65.20 | 71.00 | 74.60 | 77.30 | 79.10 |
| U$^2$PL [30] | 67.98 | 69.15 | 73.66 | 76.16 | 79.49 |
| PCR [32] | 70.06 | 74.71 | 77.16 | 78.49 | 80.65 |
| GTA-Seg [18] | 70.02 | 73.16 | 75.57 | 78.37 | 80.47 |
| Unimatch [33] | 75.20 | 77.20 | 78.80 | 79.90 | 81.20 |
| **DSSN** | 75.24 | 76.75 | 78.68 | 80.61 | 81.18 |

### 4.2 Comparison to SOTA Methods

To demonstrate the superiority of our proposed DSSN method, we conduct a comparison with the current state-of-the-art methods across various settings. All results are reported on the validation set for both PASCAL VOC and Cityscapes datasets. Additionally, we present the corresponding baseline at the top of the table, representing the results of purely supervised learning trained on the same labeled data. To ensure a fair comparison, all methods employed the DeepLab v3+ architecture.

**PASCAL VOC 2012.** We report results of our experiments on the PASCAL VOC 2012 validation dataset in Tables 1 and 2, where we evaluate the mean Intersection over Union (mIoU) for different proportions of labeled samples. Additionally, we present the corresponding baseline at the top of the table, representing the results of purely supervised learning trained on the same labeled data.

**Table 2: Comparison with the state-of-the-art methods on blended PASCAL VOC 2012 under different partition protocols.**

| Method | ResNet-50 | | | | ResNet-101 | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/16 (662) | 1/8 (1323) | 1/4 (2646) | 1/2 (5291) | 1/16 (662) | 1/8 (1323) | 1/4 (2646) | 1/2 (5291) |
| Baseline | 61.20 | 67.30 | 70.80 | 74.75 | 65.6 | 70.40 | 72.80 | 76.65 |
| MT [28] | 66.77 | 70.78 | 73.22 | 75.41 | 70.59 | 73.20 | 76.62 | 77.61 |
| CutMix-Seg [11] | 68.90 | 70.70 | 72.46 | 74.49 | 72.56 | 72.69 | 74.25 | 75.89 |
| CCT [26] | 65.22 | 70.87 | 73.43 | 74.75 | 67.94 | 73.00 | 76.17 | 77.56 |
| GCT [19] | 64.05 | 70.47 | 73.45 | 75.20 | 69.77 | 73.30 | 75.25 | 77.14 |
| CPS [3] | 71.98 | 73.67 | 74.90 | 76.15 | 74.48 | 76.44 | 77.68 | 78.64 |
| ST++ [34] | 72.60 | 74.40 | 75.40 | - | 74.50 | 76.30 | 76.60 | - |
| $U^2$PL [30] | 72.00 | 75.10 | 76.20 | - | 74.43 | 77.60 | 78.70 | - |
| PS-MT [22] | 72.83 | 75.70 | 76.43 | 77.88 | 75.50 | 78.20 | 78.72 | 79.76 |
| Unimatch [33] | 75.80 | 76.90 | 76.80 | - | 78.10 | 78.40 | 79.20 | - |
| **DSSN** | 76.74 | 77.81 | 78.27 | 78.32 | 78.50 | 79.58 | 79.45 | 79.96 |

**Table 3: Comparison with state-of-the-art on Cityscapes, $*$ means the reproduced results in $U^2$PL [30].**

| Method | ResNet-50 | | | | ResNet-101 | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/16 (186) | 1/8 (372) | 1/4 (744) | 1/2 (1488) | 1/16 (186) | 1/8 (372) | 1/4 (744) | 1/2 (1488) |
| Baseline | 63.30 | 70.20 | 73.10 | 76.60 | 66.30 | 72.80 | 75.00 | 78.00 |
| MT [28] | 66.14 | 72.03 | 74.47 | 77.43 | 68.08 | 73.71 | 76.53 | 78.59 |
| CutMix-Seg [11] | - | - | - | - | 72.13 | 75.83 | 77.24 | 78.95 |
| CCT [26] | 66.35 | 72.46 | 75.68 | 76.78 | 69.64 | 74.48 | 76.35 | 78.29 |
| GCT [19] | 65.81 | 71.33 | 75.30 | 77.09 | 66.90 | 72.96 | 76.45 | 78.58 |
| CPS * [3] | - | - | - | - | 69.78 | 74.31 | 74.58 | 76.81 |
| ST++ [34] | - | 72.70 | 73.8 | - | - | - | - | - |
| $U^2$PL [30] | 69.03 | 73.02 | 76.31 | 78.64 | 70.30 | 74.37 | 76.47 | 79.05 |
| PS-MT [22] | - | 75.76 | 76.92 | 77.64 | - | 76.89 | 77.60 | 79.09 |
| Unimatch [33] | 75.00 | 76.80 | 77.50 | 78.60 | 76.60 | 77.90 | 79.20 | 79.50 |
| **DSSN** | 75.41 | 77.31 | 78.05 | 78.73 | 76.52 | 78.18 | 78.62 | 79.58 |

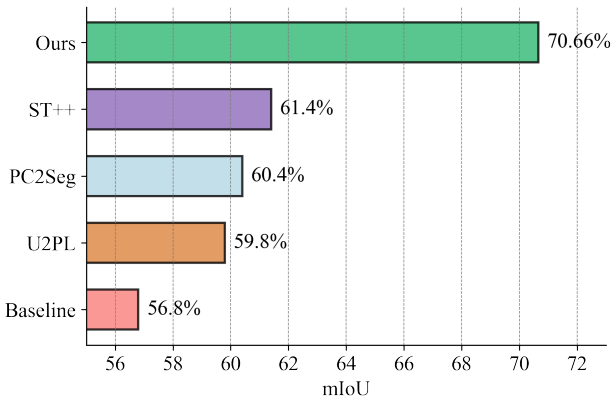


**Figure 3: The proposed DSSN method effectively utilizes unlabeled images, as demonstrated by its performance on the Cityscapes dataset with a 1/30 data split and ResNet-50. Compared to SOTAs, DSSN outperforms them significantly.**

Table 1 presents results on the classic PASCAL VOC 2012 dataset. It shows our method significantly outperforms current state-of-the-art methods. When employing ResNet-101 as the backbone, DSSN attains a 5.18% performance gain on the 1/16(92) split which surpass the performance obtain by the (1/3)183 data split in the prior study. Even with more labeled data, the performance differences become less evident; however, the proposed method still demonstrates performance improvements of 2.21% with 1/2 fine annotations over the previous SOTAs.

Table 2 illustrates the results on blender PASCAL VOC 2012 Dataset. Our method shows significant improvement on the 1/16, 1/8, 1/4, and 1/2 splits with ResNet-50, compared to the baseline, with improvements of 15.51%, 10.1%, 6.73%, and 3.57%, respectively. Similarly, with ResNet-101, our method achieves improvements of 12.9%, 9.18%, 6.65%, and 3.01% under the same partitions. Especially, our method shows significant improvements when the ratio of labeled data becomes smaller, such as under 1/8 or 1/16 partition protocols. In particular, when the labeled data is extremely limited,e.g., on the 1/16 partitions, our method achieves remarkable increases of 15.51% and 12.9% compared to the baseline with

**Table 4: Ablation of contrastive learning and CPLG.**

| $\mathcal{L}_{\text{cl}}^{ls} + \mathcal{L}_{\text{cl}}^{hs}$ | CPLG | mIoU |
|---|---|---|
| ✗ | ✗ | 76.12 |
| ✗ | ✓ | 78.33 |
| ✓ | ✗ | 78.70 |
| ✓ | ✓ | 79.58 |

**Table 5: Ablation of low- and high-level contrastive learning.**

| $\mathcal{L}_{\text{cl}}^{ls}$ | $\mathcal{L}_{\text{cl}}^{hs}$ | mIoU |
|---|---|---|
| ✗ | ✗ | 78.33 |
| ✗ | ✓ | 78.90 |
| ✓ | ✗ | 79.19 |
| ✓ | ✓ | 79.58 |

ResNet-50 and ResNet-101 as the backbone networks, respectively. Furthermore, our method demonstrates a considerable improvement over the previous state-of-the-art PS-MT [22], achieving a margin of 3.88% with ResNet-50 as the backbone, and 1.7% under the 1/8 partition protocol.

**Cityscapes.** In Table 3, we can see that our method consistently outperforms the supervised baseline by a significant margin, achieving improvements of 12.11%, 7.11%, 4.95%, and 2.13% with ResNet-50 under 1/16, 1/8, 1/4, and 1/2 partition protocols, respectively. Similarly, with ResNet-101, our method shows improvements of 10.22%, 5.38%, 3.62%, and 1.58% under 1/16, 1/8, 1/4, and 1/2 partition protocols, respectively. Furthermore, our method outperforms all other state-of-the-art methods across various settings. Specifically, under 1/8, 1/4, and 1/2 partitions, DSSN achieves a 1.55%, 1.13%, and 1.09% improvement over the previous state-of-the-art PS-MT [22] using ResNet-50, and a 1.29%, 1.02%, and 0.49% improvement using ResNet-101, respectively.

We evaluate DSSN using ResNet-50 on a 1/30 data split, which contained only 100 labeled images. As illustrated in Fig. 3, DSSN outperforms the current state-of-the-art significantly. This result indicates that our method effectively utilizes the unlabeled data through contrastive learning and the class-aware pseudo-label selection strategy (CPLG). Besides, although ReCo [21] and U$^2$PL[30] try to construct positive and negative pairs to use contrastive learning, the result shows our DSSN outperform them significantly.

Upon comparing performance on classic PASCAL VOC 2012 and blended training set, we observe that the quality of labeled samples is important. For example, DSSN achieves an exceptional performance of 80.61% by utilizing only 732 high-quality labels. However, even with significantly more labels (5291) from the blended dataset, a comparable score of 80.61% cannot be achieved.

## 4.3 Ablation Studies

In this subsection, we discuss the contribution of each component to our framework using ResNet-101 and a 1/8 labeled ratio on PASCAL VOC 2012 dataset.

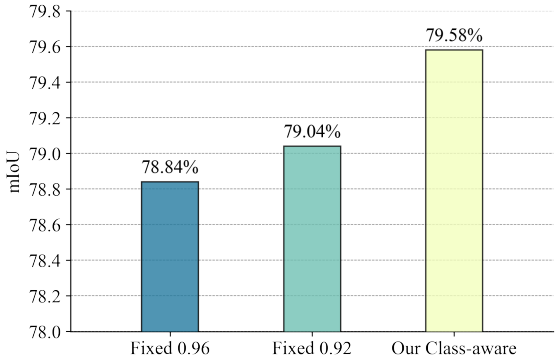


**Figure 4: Comparsion CPLG to the fixed threshold.**

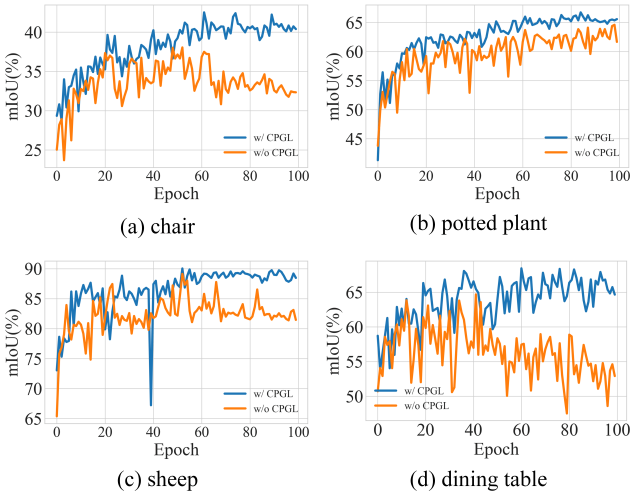


**Figure 5: The mIoU of four long-tailed classes.**

**Effectiveness of the DSSN components.** We conduct a step-by-step ablation study of each component to comprehensively assess their effectiveness. Table 4 presents the results of our study. Without our proposed dual-Level contrastive learning and CPLG, applying a plain consistency method yields an accuracy of 76.12%. However, employing dual-level contrastive learning leads to an accuracy of 78.33%, while the proposed CPLG results in 78.70%. Combining both dual-level contrastive learning and CPLG produces the highest accuracy of 79.58%, demonstrating the effectiveness of each component in the proposed DSSN method.

**Effectiveness of contrastive Learning.** In our study, we incorporate both low-level and high-level contrastive learning in our dual-level contrastive learning approach. Table 5 presents the results of our study. Without the use of both low-level contrastive and high-level contrastive, the accuracy was 78.33%. Using low-level contrastive alone results in a 0.57% improvement, while using high-level contrastive alone improves the accuracy by 0.86%. Notably, using both low-level and high-level contrastive further improves the accuracy by 1.25%, which shows the efficacy of our method.

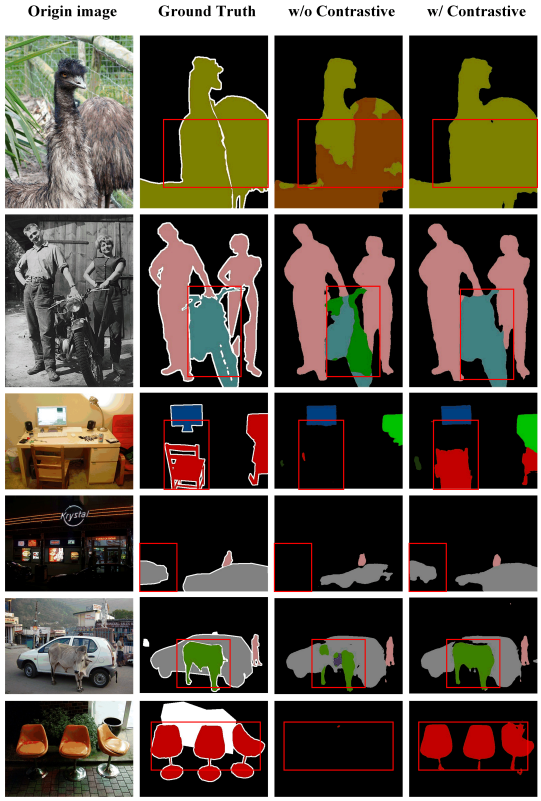


**Figure 6: Visualization on PASCAL VOC 2012. Columns from left to right denote the input images, the ground-truth, DSSN without/with contrastive learning, respectively.**

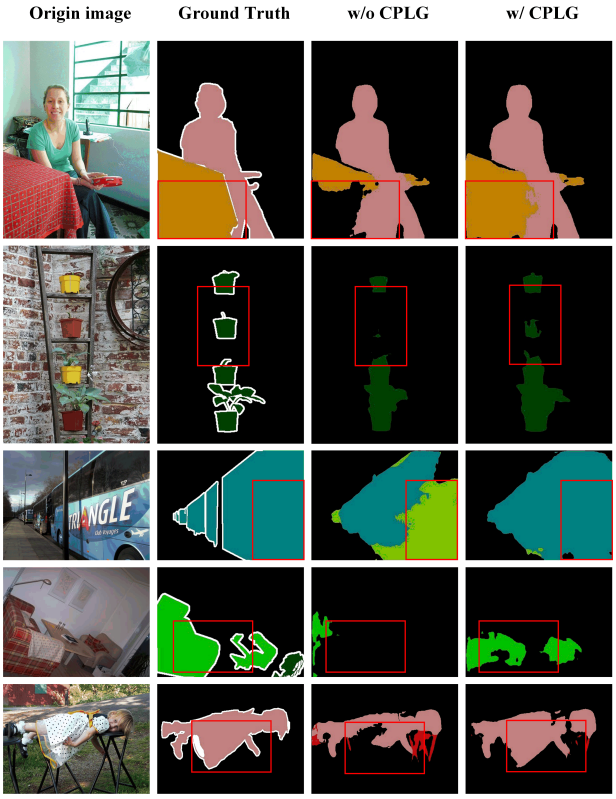


**Figure 7: Visualization on PASCAL VOC 2012, from left to right, we show the raw images, the ground-truth, DSSN without/with CPLG, respectively.**

**Effectiveness of CPLG.** As discussed in §3.4, the CPLG strategy considers difficulties of different classes and long-tailed classes, instead of using a fixed threshold during the pseudo-label generation. To test our method against a fixed threshold, we conduct experiments using a fixed threshold. Fig. 4 shows that our strategy outperforms using a fixed threshold of 0.96 and 0.92 since we set $r$ to 0.96 and $\tau_{\text{low}}$ to 0.92 in CPLG. This finding further highlights the effectiveness of our proposed DSSN method. We chose these specific thresholds because, following our experiments, we establish 0.92 as the lowest threshold and used 0.96 as the factor for the maximum probability value. Additionally, Fig. 5 presents mIoU values of classes with long tails and those that are hard to learn during training, which demonstrates the effectiveness of CPLG strategy.

**Qualitative Results.** In Figs. 6 and 7, we present the qualitative results of our study on the PASCAL VOC 2012 validation set. DSSN is based on the DeepLab v3+ with ResNet-101 network and a 1/8 ratio. The integration of contrastive learning into our method improve the performance of our model for contour and ambiguous regions, while also enhancing the accuracy of some scenarios, as illustrated in Fig. 6. Furthermore, our proposed CPLG achieved substantial precision in certain classes that are typically challenging to learn, as illustrated in Fig. 7.

## 5 CONCLUSION

In this paper, we introduce DSSN, a novel method that utilizes pixel-wise contrastive learning to address the SSS problem. DSSN is equipped with a dual-level structure that can effectively leverage unlabeled data. In DSSN, both contrastive learning and weak-to-strong consistency learning are utilized to maximize the utilization of available unlabeled data. Furthermore, we propose a class-aware pseudo label selection strategy that generates high-quality pseudo labels and significantly improves performance on long-tailed classes without incurring additional computation. DSSN achieves state-of-the-art performance on two benchmarks, and the effectiveness of our proposed novelties is confirmed by the ablation study.

## ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under the Grant No. 62176108, Natural Science Foundation of Qinghai Province of China under No. 2022-ZJ-929, Fundamental Research Funds for the Central Universities under Nos. lzujbky-2021-ct09 and lzujbky-2022-ct06, Natural Science Foundation of Shandong Province of China, No. ZR2021QF017, and Supercomputing Center of Lanzhou University.

# REFERENCES


[1] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. 2019. MixMatch: A holistic approach to semi-supervised learning. *NeurIPS* 32 (2019).

[2] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*. Springer, 801–818.

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

[4] S. Chopra, R. Hadsell, and Y. LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*. IEEE, 539–546.

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

[6] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. 2018. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine* 35, 1 (2018), 53–65.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*. IEEE, 248–255.

[8] Terrance DeVries and Graham W Taylor. 2017. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552* (2017).

[9] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. 2015. The pascal visual object classes challenge: A retrospective. *IJCV* 111 (2015), 98–136.

[10] Zhengyang Feng, Qianyu Zhou, Qiqi Gu, Xin Tan, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. 2022. DMT: Dynamic mutual training for semi-supervised learning. *Pattern Recognition* 130 (2022), 108777.

[11] Geoff French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham Finlayson. 2019. Semi-supervised semantic segmentation needs strong, varied perturbations. In *BMVC*.

[12] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. In *NeurIPS*, Vol. 33. 21271–21284.

[13] Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *CVPR*. IEEE, 1735–1742.

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

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. IEEE, 770–778.

[16] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2019. Learning deep representations by mutual information estimation and maximization. In *ICLR*.

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

[18] Ying Jin, Jiaqi Wang, and Dahua Lin. 2022. Semi-supervised semantic segmentation via gentle teaching assistant. In *NeurIPS*, Vol. 35. 2803–2816.

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

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

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

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

[23] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. 2021. Long-tail learning via logit adjustment. In *ICLR*.

[24] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. 2019. Semi-supervised semantic segmentation with high-and low-level consistency. *TPAMI* 43, 4 (2019), 1369–1379.

[25] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[26] Yassine Ouali, Céline Hudelot, and Myriam Tami. 2020. Semi-supervised semantic segmentation with cross-consistency training. In *CVPR*. IEEE, 12674–12684.

[27] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. 2020. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, Vol. 33. 596–608.

[28] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, Vol. 30.

[29] A. M. Walker. 1969. On the Asymptotic Behaviour of Posterior Distributions. *Journal of the Royal Statistical Society: Series B (Methodological)* 31, 1 (1969), 80–88.

[30] Yuchao Wang, Haochen Wang, Yujun Shen, Jingjing Fei, Wei Li, Guoqiang Jin, Liwei Wu, Rui Zhao, and Xinyi Le. 2022. Semi-supervised semantic segmentation using unreliable pseudo-labels. In *CVPR*. IEEE, 4248–4257.

[31] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. *NeurIPS* 33 (2020), 6256–6268.

[32] Haiming Xu, Lingqiao Liu, Qiuchen Bian, and Zhen Yang. 2022. Semi-supervised semantic segmentation with prototype-based consistency regularization. In *NeurIPS*, Vol. 35. 26007–26020.

[33] Lihe Yang, Lei Qi, Litong Feng, Wayne Zhang, and Yinghuan Shi. 2023. Revisiting weak-to-strong consistency in semi-supervised semantic segmentation. In *CVPR*.

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

[35] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*. IEEE, 6023–6032.

[36] Yuanyi Zhong, Bodi Yuan, Hong Wu, Zhiqiang Yuan, Jian Peng, and Yu-Xiong Wang. 2021. Pixel contrastive-consistent semi-supervised semantic segmentation. In *ICCV*. IEEE, 7273–7282.

[37] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. 2021. PseudoSeg: Designing pseudo labels for semantic segmentation. In *ICLR*.