Semi-supervised Semantic Segmentation with Directional Context-aware Consistency

Xin Lai[1*] Zhuotao Tian[1*] Li Jiang[1] Shu Liu[2]
Hengshuang Zhao[3] Liwei Wang[1] Jiaya Jia[1,2]
[1]The Chinese University of Hong Kong [2]SmartMore [3]University of Oxford
{xinlai,zttian,lijiang,lwwang,leojia}@cse.cuhk.edu.hk sliu@smartmore.com hengshuang.zhao@eng.ox.ac.uk

Abstract

Semantic segmentation has made tremendous progress in recent years. However, satisfying performance highly depends on a large number of pixel-level annotations. Therefore, in this paper, we focus on the semi-supervised segmentation problem where only a small set of labeled data is provided with a much larger collection of totally unlabeled images. Nevertheless, due to the limited annotations, models may overly rely on the contexts available in the training data, which causes poor generalization to the scenes unseen before. A preferred high-level representation should capture the contextual information while not losing self-awareness. Therefore, we propose to maintain the context-aware consistency between features of the same identity but with different contexts, making the representations robust to the varying environments. Moreover, we present the Directional Contrastive Loss (DC Loss) to accomplish the consistency in a pixel-to-pixel manner, only requiring the feature with lower quality to be aligned towards its counterpart. In addition, to avoid the false-negative samples and filter the uncertain positive samples, we put forward two sampling strategies. Extensive experiments show that our simple yet effective method surpasses current state-of-the-art methods by a large margin and also generalizes well with extra image-level annotations.

1. Introduction

Semantic segmentation, as a fundamental tool, has profited many downstream applications, and deep learning further boosts this area with remarkable progress. However, training a strong segmentation network highly relies on sufficient finely annotated data to yield robust representations for input images, and dense pixel-wise labeling is rather time-consuming, *e.g.*, the annotation process costs more than 1.5h on average for a single image in Cityscapes [12].

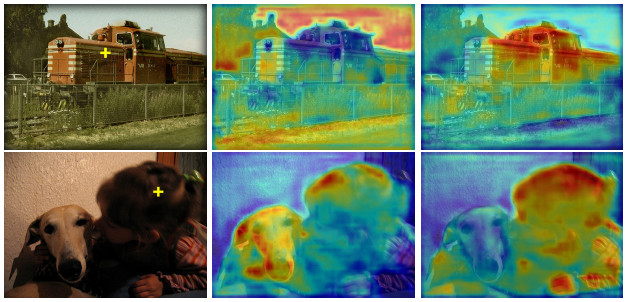

Figure 1. Grad-CAM [46] visualizations of the regional contribution to the feature of interest (*i.e.*, the yellow cross shown in the input). The red region corresponds to high contribution. SupOnly: the model trained with only 1/8 labeled data. More illustrations are shown in the supplementary.

To alleviate this problem, weaker forms of segmentation annotation, *e.g.*, bounding boxes [13, 48], image-level labels [56, 1, 44] and scribbles [34, 51, 52], have been exploited to supplement the limited pixel-wise labeled data. Still, collecting these weak labels requires additional human efforts. Instead, in this paper, we focus on the semi-supervised scenario where the segmentation models are trained with a small set of labeled data and a much larger collection of unlabeled data.

Segmentation networks can not predict a label for each pixel merely based on its RGB values. Therefore, the contextual information is essential for semantic segmentation. Iconic models (*e.g.*, DeepLab [7] and PSPNet [60]) have also shown satisfying performance by adequately aggregating the contextual cues to individual pixels before making final predictions. However, in the semi-supervised setting, models are prone to overfit the quite limited training data, which results in poor generalization on the scenes unseen during training. In this case, models are easy to excessively rely on the contexts to make predictions. Empirically, as shown in Fig. 1, we find that after training with only the labeled data, features of *train* and *person* overly focus on the contexts of *sky* and *dog* but overlook themselves. Therefore, to prevent the model abusing the contexts and also

*Equal Contribution

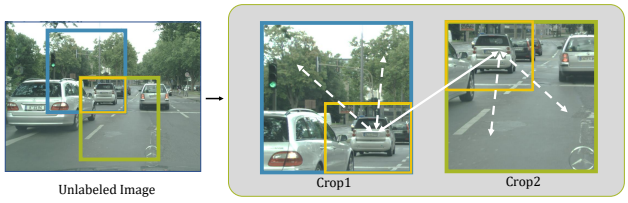

Figure 2. Crop1 and Crop2 are randomly cropped from the same image with an overlapping region. The consistency (represented by the solid white line) is maintained between representations for the overlapping region in the two crops under different contexts (represented by the dashed white line), in a pixel-to-pixel manner.

help enhance self-awareness, our solution in this work is to make the representations more robust to the changing environments, which we call the **context-aware consistency**.

Specifically, as shown in Fig. 2, we crop two random patches from an unlabeled image and they are confined to have an overlapping region, which can be deemed that the overlapping region is placed into two different environments, *i.e.*, *contextual augmentation*. Even though the ground-truth labels are unknown, the consistency of high-level features under different environments can still be maintained because there exists a pixel-wise one-to-one relationship between the overlapping regions of the two crops. To accomplish the consistency, we propose the Directional Contrastive Loss that encourages the feature to align towards the one with generally higher quality, rather than bilaterally in the vanilla contrastive loss. Also, we put forward two effective sampling strategies that filter out the common false negative samples and the uncertain positive samples respectively. Owing to the context-aware consistency and the carefully designed sampling strategies, the proposed method brings significant performance gain to the baseline.

The proposed method is simple yet effective. Only a few additional parameters are introduced during training and the original model is kept intact for inference, so it can be easily applied to different models without structural constraints. Extensive experiments on PASCAL VOC [15] and Cityscapes [12] show the effectiveness of our method.

In sum, our contributions are three-fold:

- To alleviate the overfitting problem, we propose to maintain context-aware consistency between pixels under different environments to make models robust to the contextual variance.
- To accomplish the contextual alignment, we design the Directional Contrastive Loss, which applies the contrastive learning in a pixel-wise manner. Also, two effective sampling strategies are proposed to further improve performance.
- Extensive experiments demonstrate that our proposed model surpasses current state-of-the-art methods by a large margin. Moreover, our method can be extended to the setting with extra image-level annotations.

2. Related Work

Semantic Segmentation Semantic segmentation is a fundamental yet rather challenging task. High-level semantic features are used to make predictions for each pixel. FCN [47] is the first semantic segmentation network to replace the last fully-connected layer in a classification network by convolution layers. As the final outputs of FCN are smaller than the input images, methods based on encoder-decoder structures [40, 2, 45] are demonstrated to be effective by refining the outputs step by step. Although the semantic information has been encoded in the high-level output features, it cannot well capture the long-range relationships. Therefore, dilated convolution [6, 58], global pooling [36], pyramid pooling [60, 59, 57] and attention mechanism [24, 21, 61, 63] are used to better aggregate the contexts. Despite the success of these models, they all need sufficient pixel-wise annotations to accomplish representation learning, which costs lots of human effort.

Semi-Supervised Learning Semi-supervised learning aims to exploit unlabeled data to further improve the representation learning given limited labeled data [16, 30, 35, 28]. Adversarial based methods [14, 32, 50] leverage discriminators to align the distributions of labeled and unlabeled data in the embedding space. Our method in this paper follows another line based on consistency. VAT [39] applies adversarial perturbations to the output and Π-Model [29] applies different data augmentations and dropout to form the perturbed samples and aligns between them. Dual Student [26] generates perturbed outputs for the same input via two networks with different initializations. Data interpolation is another feasible way to get perturbed samples in MixMatch [4] and ReMixMatch [3]. Besides, consistency training can be accomplished with confident target samples. Temporal Model [29] ensembles the predictions over epochs as the targets and makes the outputs consistent with them. Mean Teacher [53] yields the target samples via exponential moving average. Also, ideas of self-supervised learning have been exploited to tackle the semi-supervised learning recently [54, 5], and we also incorporate the contrastive loss that has been well studied in the self-supervised learning [17, 19, 11, 27, 9, 10] as the constraint to accomplish consistency training.

Semi-Supervised Semantic Segmentation Pixel-wise labelling is more costly than image-level annotations. Weak labels including bounding boxes [13, 48], image-level labels [56, 1, 44, 55] and scribbles [34, 51, 52] are used to alleviate this issue, but they still require human efforts. To exploit the unlabeled data, adversarial learning and consistency training are leveraged for semi-supervised segmentation. Concretely, both AdvSemiSeg [23] and S4GAN [38] utilize a discriminator to provide additional supervision to unlabeled samples. Similar to Mean Teacher, S4GAN [38]

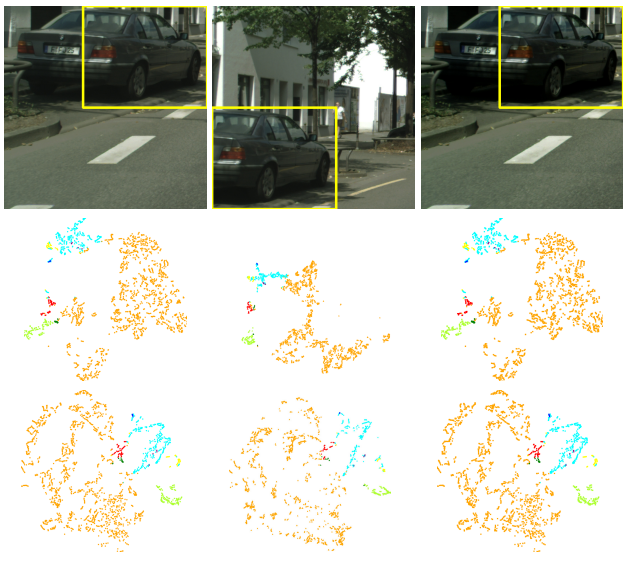

Figure 3. Visual comparison between *contextual augmentation* (I and II) and *low-level augmentation* (I and III) using t-SNE visualization for features of the overlapping region (shown in yellow box). **Top**: input crops from the same image, where II and III apply the *contextual* and *low-level augmentation* respectively. **Middle**: t-SNE results of the model trained with labeled data only. Note that the three visualizations are in the same t-SNE space, and the dots with the same color represent the features of the same class. **Bottom**: t-SNE results of our method.

also uses the teacher-student framework and the final multi-class classifier to filter out uncertain categories by scaling the predictions. [49] adds new samples that are synthesized based on the unlabeled data. The idea of self-correction has been exploited in ECS [37] and GCT [25] by creating the Correction Network and Flaw Detector respectively to amend the defects in predictions. Nevertheless, CCT [41] aligns the outputs of the main decoder and several auxiliary decoders with different perturbations to enforce a consistency that improves feature representations. Unlike these methods, our proposed context-aware consistency brings significant performance gain by explicitly alleviating the contextual bias caused by limited training samples.

3. Method

In the following sections, we firstly present our motivation in Sec. 3.1, and then elaborate the proposed context-aware consistency in Sec. 3.2. Also, to accomplish the consistency, we propose the Directional Contrastive Loss in Sec. 3.3. Moreover, two sampling strategies further improve the baseline as shown in Sec. 3.4. In Sec. 3.5, our method generalizes well with extra image-level annotations.

3.1. Motivation

Consistency-based methods [29, 53, 39] have achieved decent performance in semi-supervised learning by maintaining the consistency between perturbed images or features to learn robust representations. To accomplish the consistency training in semantic segmentation, one can simply apply low-level data augmentations, such as Gaussian blur and color jitter, to the input images and then constrain the perturbed ones to be consistent. However, low-level augmentations only alter the pixel itself without changing the contextual cues. As shown in the **Middle** row of Fig. 3, we observe that the embedding distribution changes much more significantly under the *contextual augmentation* (*i.e.*, I and II) than *low-level augmentations* (*i.e.*, I and III). In other words, even when the model has achieved consistency between low-level augmentations, it still could be unable to produce consistent embedding distribution under different contexts, which implies that the consistency with contextual augmentation could be an additional constraint that supplements low-level augmentations.

Further, one of the reasons why features vary too much under different contexts is that the model overfits the limited training data, causing the features to excessively rely on the contextual cues without sufficient self-awareness. To this end, maintaining the consistency between features under different contexts can yield more robust features and also help to alleviate the overfitting problem to some extent.

Motivated by the above, we propose our simple yet effective method. Later experiments show that our method surpasses current state-of-the-art methods as well as the method based on only low-level augmentations.

3.2. Context-Aware Consistency

The overview of our framework is shown in Fig. 4. Specifically, there are two batches of inputs, *i.e.*, x_l and x_u, representing labeled and unlabeled data respectively. As common semantic segmentation models, the labeled images x_l pass through the encoder network $\mathcal{E}$ to get the feature maps $f_l = \mathcal{E}(x_l)$. Then, the classifier $\mathcal{C}$ makes predictions $p_l = \mathcal{C}(f_l)$, which are supervised by ground truth labels y_l with the standard cross entropy loss $\mathcal{L}_{ce}$.

As for the unlabeled image x_u, two patches x_{u1} and x_{u2} are randomly cropped with an overlapping region x_o (Fig. 4 (a)). Then, x_{u1} and x_{u2} are processed by different low-level augmentations, and further pass through the encoder $\mathcal{E}$ to get the feature maps f_{u1} and f_{u2} respectively (Fig. 4 (b)). Next, similar to [9], they are projected as $\phi_{u1} = \Phi(f_{u1})$ and $\phi_{u2} = \Phi(f_{u2})$ by the non-linear projector Φ. The features of the overlapping region x_o in ϕ_{u1} and ϕ_{u2} are denoted as ϕ_{o1} and ϕ_{o2} respectively (Fig. 4 (c)).

The context-aware consistency is then maintained between ϕ_{o1} and ϕ_{o2} by the Directional Contrastive Loss (DCL) that encourages the representations of the overlapping region x_o to be consistent under different contexts, *i.e.*, the non-overlapping regions in x_{u1} and x_{u2}.

Even though the context-aware consistency is used, we don't intend to make the features totally ignore the contexts.

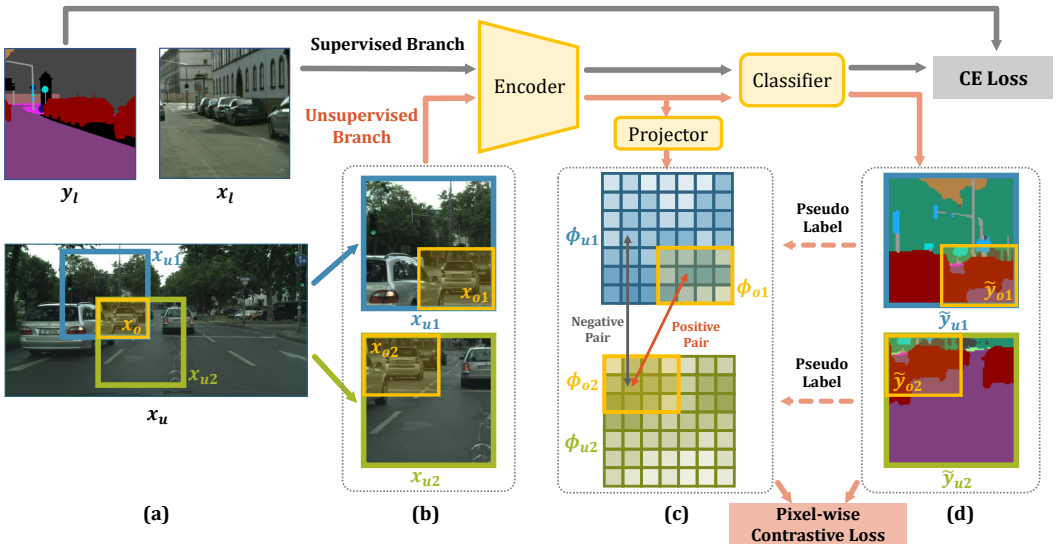

Figure 4. Overview of our framework. In the unsupervised branch, two patches are randomly cropped from the same image with a partially overlapping region. We aim to maintain a pixel-to-pixel consistency between the feature maps corresponding to the overlapping region.

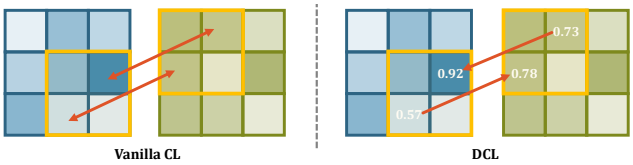

Figure 5. Comparison between vanilla Contrastive Loss (CL) and Directional Contrastive Loss (DCL). Each grid represents a feature. The scalar in the grid means the confidence of that feature.

Our purpose is just to alleviate the excessive contextual reliance and make the contexts get used more properly. On the one hand, the supervised loss $\mathcal{L}_{ce}$ is used to prevent the model from degrading into totally ignoring the contexts. On the other hand, the non-linear projector Φ is employed for the alignment. The projector Φ projects the features into a lower dimension, and only the projector output features are directly required to be invariant to the contexts rather than the encoder output features. Therefore, the projector actually plays the role of information bottleneck that prevents the original features losing useful contextual information for segmentation. In Table 3, we also conduct an experiment to highlight the contribution of the projector.

3.3. Directional Contrastive Loss (DC Loss)

The context-aware consistency requires each feature in ϕ_{o1} to be consistent with the corresponding feature of the same pixel in ϕ_{o2}, making the high-level features less vulnerable to the varying environments. To accomplish this alignment, the most straightforward solution is to apply ℓ_2 loss between ϕ_{o1} and ϕ_{o2}. However, ℓ_2 loss is too weak to make the features discriminative without pushing them away from the negative samples, which is shown in later experiments (Table 3). Instead, we take the inspiration from the contrastive loss and propose Directional Contrastive Loss (DC Loss), which accomplishes contrastive learning in the pixel level.

Unlike the ℓ_2 loss, DC Loss not only forces the positive samples, *i.e.*, the features with the same class, to lie closer, but also separates the negative samples that belong to other classes. Specifically, as shown in Fig. 4 (c), we regard two features at the same location of ϕ_{o1} and ϕ_{o2} as a positive pair, because they both correspond to the same pixels in x_u but under different contexts, *i.e.*, the non-overlapping regions in x_{u1} and x_{u2}. In addition, any two features in ϕ_{u1} and ϕ_{u2} at different locations of the original image can be deemed as a negative pair (in Fig. 4 (c)).

Furthermore, the proposed DC Loss additionally incorporates a directional alignment for the positive pairs. Specifically, we compute the maximum probability among all classes, *i.e.*, $\max(\mathcal{C}(f_i))$, as the confidence of each feature ϕ_i, where $\mathcal{C}$ is the classifier. As the prediction with higher confidence generally is more accurate as shown in the supplementary file, the less confident feature is required to be aligned towards the more confident counterpart (in Fig. 5), which effectively prevents the more confident feature from corrupting towards the less confident one.

Formally, for the b-th unlabeled image, the DC Loss $\mathcal{L}_{dc}^{b}$ can be written as follows.

$$l_{dc}^{b}(\phi_{o1},\phi_{o2}) = -\frac{1}{N}\sum_{h,w}\mathcal{M}_{d}^{h,w}\cdot\log\frac{r(\phi_{o1}^{h,w},\phi_{o2}^{h,w})}{r(\phi_{o1}^{h,w},\phi_{o2}^{h,w})+\sum_{\phi_n\in\mathcal{F}_u}r(\phi_{o1}^{h,w},\phi_n)} \quad (1)$$

$$\mathcal{M}_{d}^{h,w} = \mathbf{1}\{\max\mathcal{C}(f_{o1}^{h,w}) < \max\mathcal{C}(f_{o2}^{h,w})\} \quad (2)$$

$$\mathcal{L}_{dc}^{b} = l_{dc}^{b}(\phi_{o1},\phi_{o2}) + l_{dc}^{b}(\phi_{o2},\phi_{o1}) \quad (3)$$

where r denotes the exponential function of the cosine similarity s between two features with a temperature τ, *i.e.*, $r(\phi_1,\phi_2) = \exp\left(s(\phi_1,\phi_2)/\tau\right)$, h and w denote the 2-D

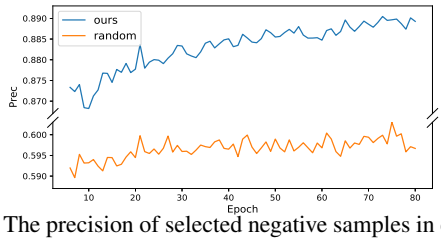

Figure 6. The precision of selected negative samples in each training epoch. The orange curve represents the result of random sampling, while the blue one represents the result of our negative sampling strategy. Note that the precision is computed by dividing the number of true negative samples by the number of all selected negative samples. Best viewed in zoom.

spatial locations, N denotes the number of spatial locations of the overlapping region, $\phi_n \in \mathbb{R}^c$ represents the negative counterpart of the feature $\phi_{o1}^{h,w}$, and $\mathcal{F}_u$ represents the set of negative samples. We note that the gradients of $l_{dc}^{b}(\phi_{o1}, \phi_{o2})$ are only back propagated to $\phi_{o1}^{h,w}$. As we observe that more negative samples lead to better performance, we select the negative samples ϕ_n not only from the current image, but also from all unlabeled images within the current training batch. Moreover, to further increase negative samples, we maintain a memory bank to store the features in the past few batches to get sufficient negative samples. We emphasize that the increasing negative samples only incurs minor additional computation and memory cost with our implementation, which is demonstrated in later experiments in Sec. 4.3.

Comparison with Vanilla Contrastive Loss The proposed Directional Contrastive Loss (DCL) differs from vanilla Contrastive Loss (CL) mainly in two ways. Firstly, CL is applied to the image-level feature, while DCL conducts contrastive learning in a pixel-wise manner. Secondly, CL as well as the Supervised Contrastive Loss [27] does not consider the confidence of features, and simply aligns them with each other bilaterally, which may even corrupt the better feature by forcing it to align towards the worse one. However, DCL only requires the less confident feature to be aligned towards the more confident counterpart.

3.4. Sampling Strategies

Negative Sampling Although image-level contrastive learning has made significant progress, it is hard to transfer the image-level success to pixel-level tasks like semantic segmentation. Because in segmentation, many different pixels in an image may belong to the same class, especially in the case of the background class or large objects such as sky and sidewalk. Therefore, when randomly selecting negative samples for an anchor feature, it is very common to select false negative samples, *i.e.*, those features that actually belong to the same class as the anchor feature. As shown in the orange curve of Fig. 6, random sampling causes that only less than 60% of the negative samples are actually true. In this case, the false negative pairs are forced to be separated from each other, which adversely affects or even corrupts the representation learning.

To avoid the common false negative pairs, we make use of pseudo labels as heuristics to eliminate those negative samples with high probability to be false. Specifically, when selecting negative samples, we also compute the probability for each class by forwarding feature maps of unlabeled data, *i.e.*, f_{u1} and f_{u2}, into the classifier $\mathcal{C}$, then get the class indexes with the highest probability, *i.e.*, $\tilde{y}_{u1}$ and $\tilde{y}_{u2}$ (Fig. 4 (d)). Formally, we have

$$\tilde{y}_{ui} = \arg\max\ \mathcal{C}(f_{ui}) \qquad i \in \{1, 2\} \tag{4}$$

For an anchor feature $\phi_o^{h,w}$ with pseudo label $\tilde{y}_o^{h,w}$, the selected negative samples should have different pseudo labels ($\tilde{y}_n \neq \tilde{y}_o^{h,w}$). Hence, the original equation (2) is accordingly updated as

$$l_{dc}^{b,ns}(\phi_{o1}, \phi_{o2}) = \\ -\frac{1}{N}\sum_{h,w} \mathcal{M}_d^{h,w} \cdot \log \frac{r(\phi_{o1}^{h,w}, \phi_{o2}^{h,w})}{r(\phi_{o1}^{h,w}, \phi_{o2}^{h,w}) + \sum_{\phi_n \in \mathcal{F}_u} \mathcal{M}_{n,1}^{h,w} \cdot r(\phi_{o1}^{h,w}, \phi_n)} \tag{5}$$

where $\mathcal{M}_{n,1}^{h,w} = \mathbf{1}\{\tilde{y}_{o1}^{h,w} \neq \tilde{y}_n\}$ is a binary mask indicating whether the pseudo labels $\tilde{y}_{o1}^{h,w}$ and $\tilde{y}_n$ for the two features $\phi_{o1}^{h,w}$ and ϕ_n are different.

As shown in Fig. 6, by exploiting pseudo labels to eliminate the false negative samples, the precision increased from around 60% to 89%. As most of the false negative pairs are filtered out, the training process becomes more stable and robust. Experiments in Table 3 show the huge improvement.

Positive Filtering Although we enable the less confident feature to align towards the more confident counterpart in the DC Loss, it may still cause the less confident feature to corrupt if the more confident counterpart is not confident enough. Therefore, to avoid this case, we also filter out those positive samples with low confidence. In particular, if the confidence of a positive sample is lower than a threshold γ, then this positive pair will not contribute to the final loss. Formally, the Eq. (5) is further revised as

$$l_{dc}^{b,ns,pf}(\phi_{o1}, \phi_{o2}) = \\ -\frac{1}{N}\sum_{h,w} \mathcal{M}_{d,pf}^{h,w} \cdot \log \frac{r(\phi_{o1}^{h,w}, \phi_{o2}^{h,w})}{r(\phi_{o1}^{h,w}, \phi_{o2}^{h,w}) + \sum_{\phi_n \in \mathcal{F}_u} \mathcal{M}_{n,1}^{h,w} \cdot r(\phi_{o1}^{h,w}, \phi_n)} \tag{6}$$

$$\mathcal{M}_{d,pf}^{h,w} = \mathcal{M}_d^{h,w} \cdot \mathbf{1}\{\max \mathcal{C}(f_{o2}^{h,w}) > \gamma\} \tag{7}$$

where $\mathcal{M}_{d,pf}^{h,w}$ is the binary mask that not only considers the directional mask $\mathcal{M}_d^{h,w}$, but also filters those uncertain positive samples. So, we have our final loss

$$\mathcal{L}_{dc}^{ns,pf} = \frac{1}{B}\sum_{b=1}^{B}(l_{dc}^{b,ns,pf}(\phi_{o1}, \phi_{o2}) + l_{dc}^{b,ns,pf}(\phi_{o2}, \phi_{o1})) \tag{8}$$

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{dc}^{ns,pf} \tag{9}$$

where B represents the training batch size, λ controls the contribution of the unsupervised loss.

3.5. Extension with Extra Image-level Annotations

Practically, common weak annotations such as image-level labels, bounding boxes and scribbles can be exploited to further boost the performance. Our method can be easily adapted to the setting where a small set of pixel-level labeled data and a much larger collection of image-level labeled data are provided.

Following the previous work [41], we first pre-train a classification network with the weakly labeled data, and then the pseudo labels y_p can be acquired by CAM [62]. During training, in addition to the main classifier $\mathcal{C}$ which is used to make predictions for the pixel-level labeled data, we add an extra classifier $\mathcal{C}_w$ for the weakly annotated images whose pseudo labels y_p are relatively coarse and inaccurate. In this way, it prevents the coarse labels from corrupting the main classifier $\mathcal{C}$. We keep all other components the same as semi-supervised setting, so the loss function becomes

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{dc}^{ns,pf} + \lambda_w \mathcal{L}_w \tag{10}$$

$$\mathcal{L}_w = \frac{1}{2} \cdot (CE(\mathcal{C}_w(f_{u1}), y_p) + CE(\mathcal{C}_w(f_{u2}), y_p)) \tag{11}$$

where $\mathcal{L}_w$ is the weakly-supervised loss and we follow CCT [41] to set up the weighting factor λ_w. During inference, we simply discard the auxiliary classifier $\mathcal{C}_w$.

4. Experiments

In the following, we show the implementation details in Sec. 4.1, followed by the comparison with state-of-the-art methods in Sec. 4.2. The ablation study is presented in Sec. 4.3 and we further apply our method with extra image-level labels in Sec. 4.4 to show the adaptation ability.

4.1. Implementation Details

Network Architecture Some previous works [23, 25, 38] use DeepLabv2 [8] as their base models, while some other methods are verified with different base models, *e.g.*, [37] and CCT [41] use DeepLabv3+ [7] and PSPNet [60] respectively. Since DeepLabv3+ shares a very similar structure with DeepLabv2 and the former achieves better performance, we use DeepLabv3+ as the main segmentation network. We also implement our method on PSPNet [60] to show the generalization ability.

The encoder in Fig. 4 refers to all other components except the final classifier, *i.e.*, the encoders of PSPNet and DeepLabv3+ also include the PPM and ASPP modules respectively. The projector Φ is implemented by an MLP that consists of two FC layers (128 output channels) and one intermediate ReLU layer.

Different segmentation models vary in their output resolutions. For DeepLabv3+, the spatial size of the output feature map before the classifier is 1/4 of the input image. To be more memory-efficient, the feature map is firstly downsampled via an 2×2 average pooling layer and then sent to the projector Φ. For PSPNet, the spatial size of output feature map is 1/8 of input images, so the feature map is directly sent to Φ without additional processing.

Datasets The experiments are conducted on PASCAL VOC [15, 18] and Cityscapes [12]. The details are illustrated in the supplementary file.

Experimental Setting During training, labeled images are first randomly resized by a ratio between 0.5 and 2 followed by the random cropping (320×320 for PASCAL VOC and 720×720 for Cityscapes). Then we apply the horizontal flip with a probability of 0.5 to the cropped patches. Differently, for each unlabeled image, after being randomly resized, two patches x_{u1} and x_{u2} are randomly cropped from the same resized image and the Intersection-over-Union(IoU) value of these two patches is supposed to be within the range $[0.1, 1.0]$. After that, we apply random mirror and standard pixel-wise augmentations (*i.e.*, Gaussian blur, color jitters and gray scaling) to x_{u1} and x_{u2}.

Following the common practice, we use 'poly' learning rate decay policy where the base learning rate is scaled by $(1 - iter/max\_iter)^{power}$ and $power$ is set to 0.9 in our experiments. SGD optimizer is implemented with weight decay 0.0001. The base learning rate values are set to 0.001 and 0.01 for backbone parameters and the others respectively for PASCAL VOC, while 0.01 and 0.1 for Cityscapes. The temperature τ is set to 0.1. Two NVIDIA GeForce RTX 2080Ti GPUs are used for training our method unless specified for PASCAL VOC while four are used for Cityscapes, and a training batch includes 8 labeled and 8 unlabeled images. The unsupervised loss weight λ and the threshold for positive filtering γ are set to 0.1 and 0.75. To stabilize training, we only use the supervised cross entropy loss to train the main segmentation model in the first 5 epochs. All models are trained entirely for 80 epochs on both datasets.

The mean Intersection-over-Union (mIoU) is adopted as our evaluation metric and each image is tested in its original size. We compare our methods in the settings with different labeled data proportions, *i.e.*, full, 1/4, 1/8 and 1/16, and all results are averaged over 3 runs. Note that in the full data setting, images fed to the unsupervised branch are simply collected from the labeled set.

4.2. Results

To demonstrate the superiority of our method, we make comparisons with recent state-of-the-art models. However, it is hard to compare these methods that are implemented with various settings, *e.g.*, different segmentation models, randomly sampled data lists and inconsistent baseline performance. Therefore, we reproduce the representative models [38, 25, 41] within an unified framework according to their official code, where all methods are applied upon the same base segmentation model and trained with the same

Method	SegNet	Backbone	1/16	1/8	1/4	Full
SupOnly	PSPNet	ResNet50	57.4	65.0	68.3	75.1
CCT [41]	PSPNet	ResNet50	62.2	68.8	71.2	75.3
Ours	PSPNet	ResNet50	**67.1**	**71.3**	**72.5**	**76.4**
SupOnly	DeepLabv3+	ResNet50	63.9	68.3	71.2	76.3
ECS [37]	DeepLabv3+	ResNet50	-	70.2	72.6	76.3
Ours	DeepLabv3+	ResNet50	**70.1**	**72.4**	**74.0**	**76.5**
SupOnly	DeepLabv3+	ResNet101	66.4	71.0	73.5	77.7
S4GAN [38]	DeepLabv3+	ResNet101	69.1	72.4	74.5	77.3
GCT [25]	DeepLabv3+	ResNet101	67.2	72.5	75.1	77.5
Ours	DeepLabv3+	ResNet101	**72.4**	**74.6**	**76.3**	**78.2**

Table 1. Comparison with the baseline (SupOnly, *i.e.*, with only supervised loss) and current state-of-the-art methods evaluated on PASCAL VOC with 1/16, 1/8, 1/4 and full labeled data. We use DeepLabv3+ and PSPNet as the main segmentation network, ResNet101 and ResNet50 [20] as the backbone.

Methods	1/8	1/4	Full
SupOnly	66.0	70.7	**77.7**
Ours	**69.7**	**72.7**	77.5

Table 2. Comparison with SupOnly on Cityscapes with 1/8, 1/4 and full labeled data. All results are based on Deeplabv3+ [7] with ResNet-50 backbone.

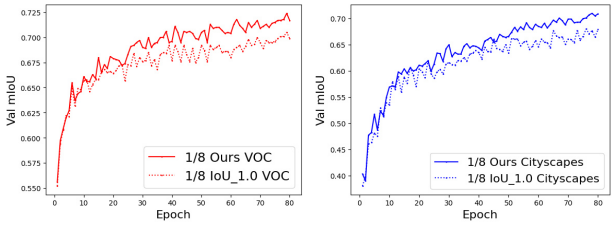

Figure 7. Performance evaluated on the validation sets of PASCAL VOC and Cityscapes respectively during training. IoU_1.0 means that two patches of unlabeled data totally overlap and only low-level augmentations are applied.

data lists. As the implementation of ECS [37] is not publicly available, we directly use the results reported in the original paper. It is worth noting that our reproduced results are better than those reported in the original papers, and we will also release our code publicly.

The comparison on PASCAL VOC is shown in Table 1, where our model surpasses other methods by a large margin. S4GAN [38] uses an additional discriminator to obtain extra supervision, and both GCT [25] and ECS [37] refine the flaw or error by exploiting unlabeled images. However, they do not explicitly maintain the contextual consistency with the unlabeled images. Although CCT [41] enables features with different contexts to be consistent by aligning the perturbed high-level features to the main features, the perturbation directly applied to features is unnatural, and also the alignment does not push away the features in different classes. Moreover, in Table 2, the experimental results on Cityscapes further demonstrate the generalization ability of our method.

ID	Proj	Context	CL	Dir	NS	PF	mIoU
SupOnly							64.7
ST							66.3
I	✓	✓					64.2
II	✓	✓	✓				56.4
III	✓	✓	✓	✓			64.8
IV	✓	✓	✓	✓	✓		71.6
V	✓	✓	✓		✓	✓	71.2
VI	✓		✓	✓	✓	✓	70.5
VII		✓	✓	✓	✓	✓	61.5
VIII	✓	✓	✓	✓	✓	✓	**72.4**

Table 3. Ablation Study. Exp.I uses ℓ_2 loss to align positive feature pairs. **ST**: Self-Training. **Proj**: Non-linear Projector Φ. **Context**: Context-aware Consistency. **CL**: Vanilla Contrastive Loss. **Dir**: Directional Mask $\mathcal{M}_d^{h,w}$ defined in Eq. (2). **NS**: Negative Sampling. **PF**: Positive Filtering.

4.3. Ablation Study

We conduct an extensive ablation study in Table 3 to show the contribution of each component. The ablation study is based on 1/8 labeled data on PASCAL VOC. We use PSPNet with ResNet50 as the segmentation network. We set up two baselines, *i.e.*, the model trained with only supervised loss (SupOnly) and the model with the self-training technique (ST). We follow [64] to implement ST. We find that the results of three data lists do not vary much, so we only run on a single data list for ablation study.

Context-aware Consistency To manifest the effectiveness of the proposed context-aware consistency, we make a comparison between the model with context-aware consistency and that only with low-level transformations. Intuitively, when the two patches cropped from an unlabeled image totally overlap with each other, *i.e.*, the IoU is confined to [1, 1], there will be no contextual augmentation between them. In Table 3, the experiments VI and VIII show that the model with our proposed context-aware consistency (Exp.VIII) is superior to that with only low-level transformations (Exp.VI) by 1.9 points. To further highlight the improvement throughout the training process, we present the validation curves on PASCAL VOC and Cityscapes in Fig. 7, where a huge gap can be observed.

Directional Contrastive Loss The proposed DC Loss is a stronger constraint than ℓ_2 loss, because ℓ_2 loss does not consider pushing negative samples away. To illustrate this, in Table 3, we compare the models with ℓ_2 loss (Exp.I) and DC Loss (Exp.VIII). It shows that simply using ℓ_2 loss even worsens the performance from 65.0 to 64.2. Though the result of Exp.III without negative sampling also falls behind that of the baseline (SupOnly), Exp.IV shows that it is caused by the overwhelming false negative samples. After addressing the negative sampling problem, the proposed DC Loss (Exp.IV) surpasses the simple ℓ_2 alignment (Exp.I) as well as the baseline (SupOnly) by a large margin.

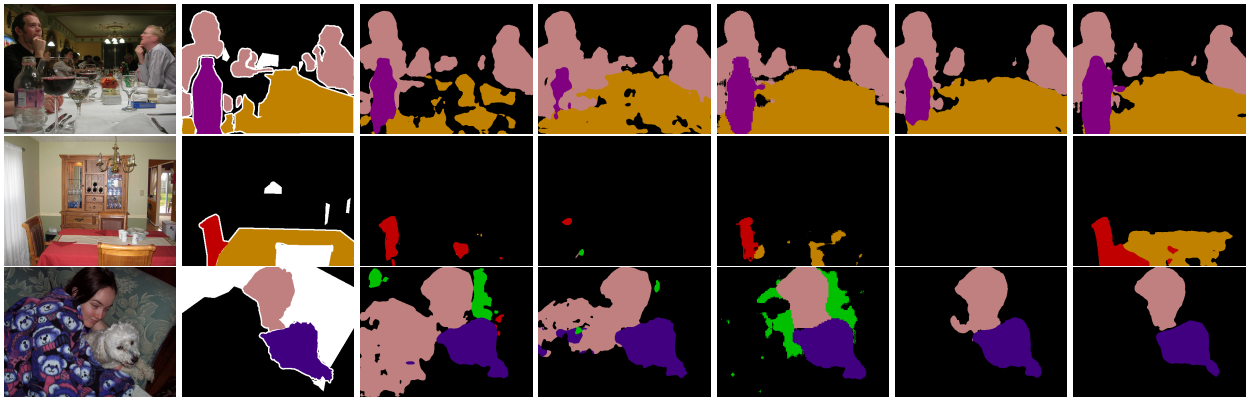

Figure 8. Visual comparison between SupOnly (*i.e.*, trained with only supervised loss) and current state-of-the-art methods with ours.

NumNeg	500	1k	2k	6.4k	12.8k	19.2k	25.6k
mIoU	70.9	71.0	71.7	71.3	71.9	72.4	71.9

Table 4. Performance (mIoU) evaluated on PASCAL VOC under different number of negative samples per GPU.

Also, by comparing the experiments V and VIII, we observe that with the directional mask, the DC Loss improves the vanilla Contrastive Loss by 1.2 points. This demonstrates the effectiveness of the proposed directional alignment compared to the bilateral one.

Negative Sampling The experiments III and IV in Table 3 show that the proposed negative sampling strategy with pseudo labels significantly improves the DC Loss.

In Table 4, we also notice that, within a certain scope, the more negative samples we use in training, the better performance we will get until it reaches an upper bound. More importantly, it is worth noting that the increased number of negative samples does not affect the training efficiency much by using the gradient checkpoint function provided in PyTorch [43]. Specifically, by increasing negative samples from 500 to 19.2k, the training memory consumption merely increases by about 800M on each GPU, and the average training time of each iteration only increases from 0.96s to 1.18s. The implementation details are elaborated in the supplementary file.

Positive Filtering We show the effect of positive filtering in Table 3. The comparison between experiments IV and VIII demonstrates that the proposed positive filtering strategy makes further improvements by 0.8 points.

The Role of Projector Φ In Table 3, we compare the performance with and without the projector in experiment VII and VIII, which shows the contribution of the projector.

4.4. Extension with Extra Image-level Annotations

Following [41], the original 1464 training images of PASCAL VOC [15] are given the pixel-wise annotations and the rest of 9118 augmented images in SBD [18] are provided with image-level annotations. In Table 5, our method reaches 76.1% mIoU and surpasses CCT [41] by a large margin. Amazingly, it is 1.0 points higher than the model trained with the full pixel-level annotations, which demonstrates the generalization ability of the proposed method.

Methods	Backbone	Semi	Weakly
WSSN [42]	VGG-16	-	64.6
GAIN [33]	VGG-16	-	60.5
MDC [56]	VGG-16	-	65.7
DSRG [22]	VGG-16	-	64.3
Souly *et al.* [49]	VGG-16	64.1	65.8
FickleNet [31]	ResNet-101	-	65.8
CCT [41]	ResNet-50	69.4	73.2
Ours	VGG-16	68.7	69.3
CCT‡	ResNet-50	72.8	74.6
Ours	ResNet-50	**74.5**	**76.1**

Table 5. Results with extra image-level annotations. CCT‡: Reproduced with the same setting as ours. Semi: Semi-supervised setting. Weakly: the setting with extra image-level labels.

4.5. Visual Comparison

Fig. 8 presents the visual comparison with the SupOnly and current state-of-the-art methods. We observe that the results of our method are generally superior to others.

5. Conclusion

In this work, we focus on the semi-supervised semantic segmentation problem. In order to alleviate the problem of excessively using contexts and enhance self-awareness, we have presented the context-aware consistency, where we explicitly require features of the same identity but with different contexts to be consistent. In addition, we propose Directional Contrastive Loss to conduct the alignment. Also, two effective sampling strategies are put forward to make further improvements. Extensive experiments show our method achieves new state-of-the-art results, and also generalizes well with extra image-level annotations.

References

[1] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018. 1, 2

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *TPAMI*, 2017. 2

[3] David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *ICLR*, 2020. 2

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

[5] Lucas Beyer, Xiaohua Zhai, Avital Oliver, and Alexander Kolesnikov. S4L: self-supervised semi-supervised learning. In *ICCV*, 2019. 2

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2018. 2

[7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 1, 6, 7

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2017. 6

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2, 3

[10] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. In *NeurIPS*, 2020. 2

[11] Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv*, 2020. 2

[12] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 1, 2, 6

[13] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, 2015. 1, 2

[14] Zihang Dai, Zhilin Yang, Fan Yang, William W. Cohen, and Ruslan Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *NeurIPS*, 2017. 2

[15] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 2, 6, 8

[16] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *NeurIPS*, 2004. 2

[17] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006. 2

[18] Bharath Hariharan, Pablo Arbelaez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, 2011. 6, 8

[19] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 2

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 7

[21] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *ICCV*, 2019. 2

[22] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *CVPR*, 2018. 8

[23] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. In *BMVC*, 2018. 2, 6

[24] Haijie Tian Yong Li Yongjun Bao Zhiwei Fang and Hanqing Lu Jun Fu, Jing Liu. Dual attention network for scene segmentation. In *CVPR*, 2019. 2

[25] Zhanghan Ke, Di Qiu, Kaican Li, Qiong Yan, and Rynson W.H. Lau. Guided collaborative training for pixel-wise semi-supervised learning. In *ECCV*, 2020. 3, 6, 7

[26] Zhanghan Ke, Daoye Wang, Qiong Yan, Jimmy S. J. Ren, and Rynson W. H. Lau. Dual student: Breaking the limits of the teacher in semi-supervised learning. In *ICCV*, 2019. 2

[27] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv*, 2020. 2, 5

[28] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017. 2

[29] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *ICLR*, 2017. 2, 3

[30] Dong-Hyun Lee. Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. In *ICMLW*, 2013. 2

[31] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *CVPR*, 2019. 8

[32] Chongxuan Li, Taufik Xu, Jun Zhu, and Bo Zhang. Triple generative adversarial nets. In *NeurIPS*, 2017. 2

[33] Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Tell me where to look: Guided attention inference network. In *CVPR*, 2018. 8

[34] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, 2016. 1, 2

[35] Bin Liu, Zhirong Wu, Han Hu, and Stephen Lin. Deep metric transfer for label propagation with limited annotated data. In *ICCVW*, 2019. 2

[36] Wei Liu, Andrew Rabinovich, and Alexander C. Berg. Parsenet: Looking wider to see better. *arXiv*, 2015. 2

[37] Robert Mendel, Luis Antonio, De Souza Jr, David Rauber, and Christoph Palm. Semi-supervised segmentation based on error-correcting supervision. In *ECCV*, 2020. 3, 6, 7

[38] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high- and low-level consistency. *TPAMI*, 2019. 2, 6, 7

[39] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *TPAMI*, 2019. 2, 3

[40] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015. 2

[41] Yassine Ouali, Celine Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *CVPR*, 2020. 3, 6, 7, 8

[42] George Papandreou, Liang-Chieh Chen, Kevin P. Murphy, and Alan L. Yuille. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *ICCV*, 2015. 8

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 8

[44] Pedro H. O. Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015. 1, 2

[45] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 2

[46] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *IJCV*, 2020. 1

[47] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *TPAMI*, 2017. 2

[48] Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In *CVPR*, 2019. 1, 2

[49] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In *ICCV*, 2017. 3, 8

[50] Jost Tobias Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. In *ICLR*, 2016. 2

[51] Meng Tang, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers. Normalized cut loss for weakly-supervised CNN segmentation. In *CVPR*, 2018. 1, 2

[52] Meng Tang, Federico Perazzi, Abdelaziz Djelouah, Ismail Ben Ayed, Christopher Schroers, and Yuri Boykov. On regularized losses for weakly-supervised CNN segmentation. In *ECCV*, 2018. 1, 2

[53] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017. 2, 3

[54] Phi Vu Tran. Semi-supervised learning with self-supervised networks. In *NeurIPS Workshop*, 2019. 2

[55] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *CVPR*, 2017. 2

[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 *CVPR*, 2018. 1, 2, 8

[57] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *CVPR*, 2018. 2

[58] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 2

[59] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images. In *ECCV*, 2018. 2

[60] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017. 1, 2, 6

[61] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. Psanet: Point-wise spatial attention network for scene parsing. In *ECCV*, 2018. 2

[62] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 6

[63] Zhen Zhu, Mengde Xu, Song Bai, Tengteng Huang, and Xiang Bai. Asymmetric non-local neural networks for semantic segmentation. In *ICCV*, 2019. 2

[64] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le. Rethinking pre-training and self-training. *arXiv*, 2020. 7