Learning Non-target Knowledge for Few-shot Semantic Segmentation

Yuanwei Liu[1] Nian Liu[2*] Qinglong Cao[1] Xiwen Yao[1] Junwei Han[1] Ling Shao[3]
[1]Northwestern Polytechnical University [2]Inception Institute of Artificial Intelligence
[3]Terminus Group, China
{liuyuanwei9809, liunian228, caoql19980603, yaoxiwen517, junweihan2010}@gmail.com
ling.shao@ieee.org

Abstract

Existing studies in few-shot semantic segmentation only focus on mining the target object information, however, often are hard to tell ambiguous regions, especially in non-target regions, which include background (BG) and Distracting Objects (DOs). To alleviate this problem, we propose a novel framework, namely Non-Target Region Eliminating (NTRE) network, to explicitly mine and eliminate BG and DO regions in the query. First, a BG Mining Module (BGMM) is proposed to extract the BG region via learning a general BG prototype. To this end, we design a BG loss to supervise the learning of BGMM only using the known target object segmentation ground truth. Then, a BG Eliminating Module and a DO Eliminating Module are proposed to successively filter out the BG and DO information from the query feature, based on which we can obtain a BG and DO-free target object segmentation result. Furthermore, we propose a prototypical contrastive learning algorithm to improve the model ability of distinguishing the target object from DOs. Extensive experiments on both PASCAL-5^i and COCO-20^i datasets show that our approach is effective despite its simplicity. Code is available at https://github.com/LIUYUANWEI98/NERTNet

1. Introduction

Due to the rapid development of fully convolutional network (FCN) [21] architectures, deep learning has made milestone progress in semantic segmentation. Most methods adopt the fully-supervised learning scheme and require a mass of annotated data for training. Although they can achieve good performance, their data-hungry nature demands time and labor-consuming image annotations. To alleviate this problem, few-shot semantic segmentation was proposed to segment unseen object classes in query images with only a few annotated samples, namely supports.

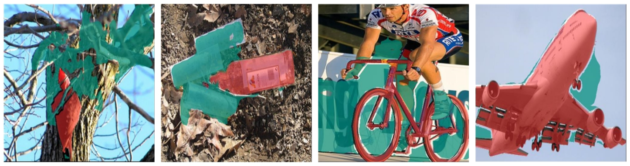
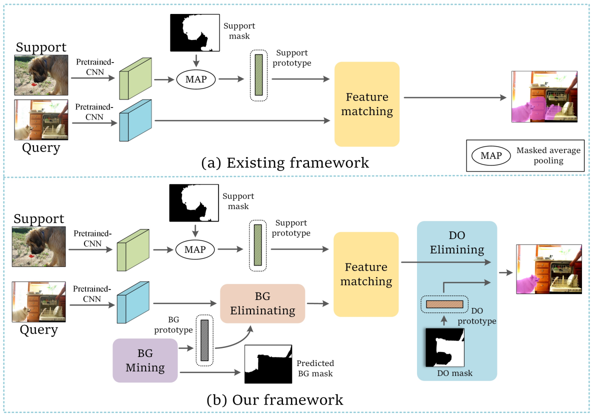


Figure 1. Previous methods often show false positive predictions in non-target regions. Pixels in red indicate the target objects, while pixels in green mean false positive predictions.



(a) Existing framework

(b) Our framework

Figure 2. Comparison between existing framework and ours for few-shot segmentation. The main difference is that the former only mines target category information, while we propose to eliminate co-existing pixels belonging to non-target regions, including the background (BG) and distracting objects (DO).

Currently, there are many existing researches exploring various deep learning methods for few-shot semantic segmentation [14, 20, 29, 31, 33, 42]. They usually extract features from both query and support images first, and then extract the class-specific representation using the support masks. Finally, a matching network is leveraged to segment the target object in the query image using the class representation.

Most typically, prototypical learning methods [6, 31, 33, 42, 44] use masked average pooling (MAP) on the target object regions of the support images to form a single or a few prototypes. Then, prototypes are used to segment the target object in the query image via conducting dense feature

[*] Corresponding author.

matching.

Although some achievements have been made, these methods all focus on digging out more effective target information from supports as much as possible, and then transferring them to the query image to achieve segmentation (see Figure 2 (a)). However, as illustrated in Figure 1, they often suffer from the false positive prediction in backgrounds (BG) and co-existing objects belonging to other classes, namely, distracting objects (DOs). The main reason is that solely focusing on target objects in the few-shot setting makes their models hard on learning discriminative features and differentiating ambiguous regions.

To alleviate this problem, we rethink the few shot semantic segmentation task from a new perspective, that is, mining and excluding non-target regions, *i.e.*, BG and DO regions, rather than directly segmenting the target object. From this point, in this paper, we propose a novel framework, namely non-target region eliminating (NTRE) network for few-shot semantic segmentation. As shown in Figure 2 (b), we first develop a BG mining module (BGMM) to obtain a BG prototype and segment the BG region. Then, a BG eliminating module (BGEM) is proposed to filter out the BG information from the query feature. Next, the target prototype from the support is utilized in a matching network to activate the target object in the query feature. Subsequently, we adopt a DO eliminating module (DOEM) to mine the DO region first and then filter out the DO information from the query feature. As such, finally, we can obtain an accurate target segmentation result without the distraction from the BG and DO regions.

In the BGMM, obtaining the BG prototype is not straightforward as we obtain the support prototype. Considering that BG regions universally exist in almost every image, such as sky, grass, walls, and etc, we propose to learn a general BG prototype from both query and support images in the training set. Based on this prototype, we can segment the BG regions for all images easily. Since having no ground truth BG segmentation masks to supervise the model learning, we specifically design a BG mining loss based on the known target segmentation masks.

Furthermore, considering that it's hard to learn a good prototype feature embedding space to differentiate DOs from the target object under the few-shot setting, we propose the prototypical contrastive learning (PCL) method to improve the object-discrimination ability of the network by refining the prototype feature embeddings. Specifically, for a query target prototype, we treat the corresponding support target prototype as the positive sample, while the DO prototypes both in query and support are considered as negative samples. We then propose a PCL loss to enforce the prototype embeddings to be similar within the target prototypes and dissimilar between target and DO prototypes. As such, the PCL could effectively help the network distinguish target objects from DOs.

In summary, our contributions are as follows:

- To the best of our knowledge, this is the first time to mine and eliminate non-target regions, including BG and DOs, for few-shot semantic segmentation, which can effectively decrease false positive predictions.
- We propose the BGMM, BGEM, and DOEM for effectively implementing the mining and eliminating of the BG and DO regions. A novel BG mining loss is also proposed for training the BGMM without using BG ground truth.
- We propose a PCL method to improve the model ability for better distinguishing target objects from DOs.
- Extensive experiments on PASCAL-5^i and COCO-20^i show that our proposed framework yields a new state-of-the-art performance, especially on the 1-shot setting.

2. Related Works

Semantic Segmentation. Compared with convolutional neural networks (CNNs) [13], the emergence of the FCN [21] has brought great progress to the semantic segmentation task. Specifically, the fully connected layers in CNNs are replaced by fully convolutional layers to enable pixel-level prediction. Based on the FCN, various network architectures are proposed to tackle the semantic segmentation problem in recent works. For example, [5,8,15,30,40,43,46] propose various attention mechanisms embedded in the FCN architecture. Some other works utilize different feature fusion methods, such as the pyramid pooling module [45], dilated convolution kernels [2], multi-scale feature aggregation [10], dense atrous spatial pyramid pooling (ASPP) [39]. However, these traditional semantic segmentation networks are powerless when dealing with unseen categories. Meanwhile, training such networks are computationally costly and also requires labor-consuming pixel-level annotations on large-scale data.

Few-shot Semantic Segmentation. Few-shot semantic segmentation aims to segment unseen object classes in query images with only a few annotated samples. There are two mainstream frameworks to segment the target objects in query images. One is the pixel-level matching framework, which is firstly proposed by [26]. This framework usually generates the target object prototype from the support features first, and then segments the query using dense feature matching. Another is the pixel-level measurement framework, proposed by [44], which measured the embedding similarity between the query and the supports. Following the first framework, CANet [42] utilized an Iterative Optimization Module (IOM) to refine the prediction progressively after concatenating the support prototype and the query features. PFENet [31] proposed a prior mask by calculating the cosine similarity between the support and query images on high-level features without learnable parameters. ASGNet [14] proposed a superpixel-guided clustering method to obtain multi-part prototypes from the support and used an allocation strategy to reconstruct the support feature map instead of using prototype expanding. Following the second framework, PANet [33] embedded different object classes into different prototypes with a pre-trained encoder. Then, the query image was labeled based on the distance between the representations of the query image and the prototypes. Yang *et al*. [38] proposed a novel joint-training framework via introducing additional base category prototypes to mine latent novel classes during training. Most of these previous methods focus on directly segmenting the target object. Differently, in this paper, we are the first to propose leveraging complementary non-target knowledge and eliminating distracting regions for few-shot segmentation.

Contrastive Learning Most previous computer vision researches focus on designing artificially preferred network architectures to tackle various computer vision tasks. The emergence of contrastive learning [11] brings our focus back to mining better deep feature representations via contrasting positive and negative samples. SimCLR [3] proposed a simple self-supervised contrastive learning paradigm by using different data augmentation methods to form positive and negative samples for each image instance. MoCo [4, 11] proposed to store negative samples using a dynamically updated queue, in which only the stored feature vectors from recent batches are used for training. As such, MoCo solved the inconsistency problem of the sampled features due to the optimization to the encoder. Very recently, Wang *et al*. [34] introduced contrastive learning in supervised semantic segmentation and proposed a pixel-wise contrastive algorithm. They treated the pixel embeddings of the same class and different classes as positive and negative samples, respectively. Different from them, in our work, we propose the PCL scheme to improve the objects-discrimination ability of the extracted prototypes, which could effectively help the network distinguish target objects from DOs.

3. Proposed Method

3.1. Problem Definition

Few-shot semantic segmentation aims to train a model on base classes, and segment unseen objects in query images with a few annotated support samples without re-training. Typically, all the datasets are divided into two subsets. One is the training set $\mathcal{D}_{base}$ with the base classes $\mathcal{C}_{base}$. The other is the testing set $\mathcal{D}_{novel}$ with the novel classes $\mathcal{C}_{novel}$. These two sets of classes are disjoint, *i.e.*, $\mathcal{C}_{base} \cap \mathcal{C}_{novel} = \varnothing$. Specifically, the training set $\mathcal{D}_{base}$ is partitioned into several episodes after randomly sampling $K+1$ image-mask pairs that contain objects from a specific class in $\mathcal{C}_{base}$. The testing set is composed of similar episodes, except that the data are sampled from the $\mathcal{C}_{novel}$. For one episode, K image-mask pairs are treated as the support set $\mathcal{S}= \{(\boldsymbol{I}_i^s, \boldsymbol{M}_i^s)\}_{i=1}^K$ to segment the objects of the target class in the remaining one sample, which is termed the query set $\mathcal{Q}$. Here, $\boldsymbol{I} \in \mathbb{R}^{H \times W \times 3}$ indicates the RGB image and $\boldsymbol{M} \in \mathbb{R}^{H \times W}$ indicates the corresponding mask. $\mathcal{Q}= \{(\boldsymbol{I}^q, \boldsymbol{M}^q)\}$ is provided with the ground truth only during training. Following this episode, the network is trained on $\mathcal{D}_{base}$ and evaluated on $\mathcal{D}_{novel}$.

3.2. Overview

As aforementioned, few-shot semantic segmentation models usually fail in ambiguous non-target regions. Motivated by this observation, we propose to mine and eliminate non-target regions, which include the background (BG) regions and the distracting object (DO) regions.

We first follow previous methods and use a pre-trained backbone to extract the query feature map $\boldsymbol{X}^s \in \mathbb{R}^{H \times W \times C}$ and the support feature map $\boldsymbol{X}^q \in \mathbb{R}^{H \times W \times C}$ from corresponding images, respectively. Then, the BG mining module (BGMM) is proposed to mine the BG region via learning a general BG prototype, which is randomly initialized and subsequently learned on the training set. We also propose a novel BG loss without using accurate BG segmentation ground truth. Next, a BG eliminating module (BGEM) is utilized to filter out the BG information from the query feature. Subsequently, we follow prototypical learning to activate the target region in the query feature using the support target prototype and feature matching (FM). An initial target object segmentation mask can be further obtained.

After that, a DO eliminating module (DOEM) is proposed to filter out DO information from the query feature. The DO region can be first mined by combining the BG segmentation map and the initial target prediction. Then, the DO prototype can be obtained from the query feature and used for DO elimination. The prototypical contrastive learning (PCL) is also adopted for better discriminating the target object from DOs. Finally, a segmentation network is used to achieve the BG and DO-free prediction.

3.3. Background Mining and Eliminating

3.3.1 Background Mining Module

Background regions, in which no obvious objects appear, commonly exist in most images. Based on this commonality, we propose to use a BG prototype to encode general BG knowledge. It can be represented as $\boldsymbol{P}_{BG} \in \mathbb{R}^{1 \times 1 \times D}$, where D is the channel dimension. Inspired by some saliency detection methods [17, 18], it is feasible to learn $\boldsymbol{P}_{BG}$ on a large number of images and then use it to detect BG regions on any natural image.

Hence, in this paper, we first randomly initialize $\boldsymbol{P}_{BG}$

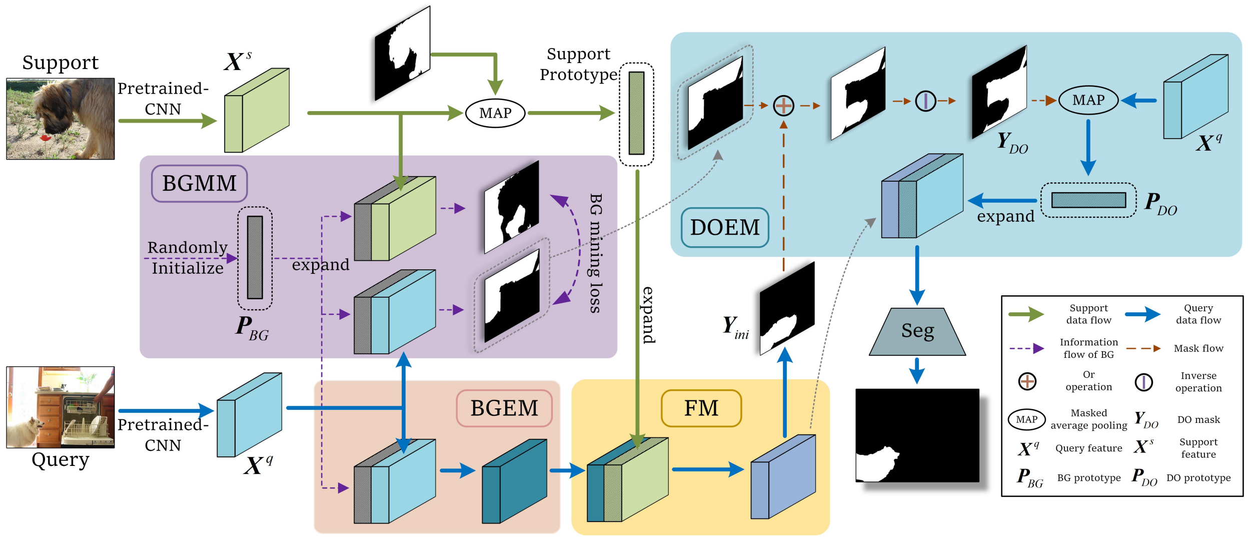


Figure 3. Overall architecture of the proposed method for few-shot semantic segmentation. Our network is composed of four parts. After extracting features from both the support and query images via a pre-trained backbone, our Background Mining Modul (BGMM) is performed to obtain a BG prototype and segment the BG regions. Meanwhile, Background Eliminating Module (BGEM) is performed to eliminate the BG regions. The third part is to obtain the activated query feature and further an initial target prediction via Feature Matching (FM). The last part is to eliminate the distracting objects by our proposed Distracting Objects Eliminating Module (DOEM).

and then learn it from both support and query images in the base classes during training. Specifically, given the query feature map $\boldsymbol{X}^Q$ and the support feature map $\boldsymbol{X}^S$ extracted from the backbone, we expand $\boldsymbol{P}_{BG}$ to the same size as them, obtaining $\hat{\boldsymbol{P}}_{BG} \in \mathbb{R}^{H\times W\times C}$. Then, we concatenate it with $\boldsymbol{X}^q$ and $\boldsymbol{X}^s$, respectively, and use a simple segmentation network $\mathcal{F}_{3\times3}(\cdot)$ to get the BG prediction for the query and the support:

$$\boldsymbol{y}^q_{BG} = \mathcal{F}_{3\times3}(\boldsymbol{X}^q \oplus \hat{\boldsymbol{P}}_{BG}), \tag{1}$$

$$\boldsymbol{y}^s_{BG} = \mathcal{F}_{3\times3}(\boldsymbol{X}^s \oplus \hat{\boldsymbol{P}}_{BG}), \tag{2}$$

where $\oplus$ denotes the concatenation operation along the channel dimension and $\mathcal{F}_{3\times3}$ is composed of two 3×3 convolutional layers and shares the same weights in both (1) and (2). $\boldsymbol{y}^{q/s}_{BG} \in \mathbb{R}^{H\times W\times 1}$ is the BG segmentation probability map of the query or the support.

Background Mining Loss. In the few-shot semantic segmentation task, we only have the ground truth of target object masks, *i.e.*, $\boldsymbol{M}^q$ and $\boldsymbol{M}^s$, and have no BG region ground truth. In order to force $\boldsymbol{P}_{BG}$ effectively predict the BG region as we expected, we design a BG mining loss to optimize this learning process as below:

$$\begin{aligned} L_{BG} = &-\frac{1}{N}\sum_i log(1-\boldsymbol{y}^{q/s}_{BG}(i))\boldsymbol{M}^{q/s}(i) \\ &-\alpha\frac{1}{Z}\sum_j log(\boldsymbol{y}^{q/s}_{BG}(j)), \end{aligned} \tag{3}$$

where i and j are the indexes of the spatial locations. $\boldsymbol{M}^{q/s}$ is the ground truth of target objects belonging to query or support. N denotes the total number of the target object pixels and Z is equal to $H\times W$. α is a hyperparameter to weight the second term.

The core idea of this loss is that the BG prediction should belong to the reverse region of the target object, *i.e.*, predicting zero in $\boldsymbol{y}^{q/s}_{BG}$ for the pixels that belong to the target object. However, solely using this constraint may lead to a trivial solution that predicts all zeros for $\boldsymbol{y}^{q/s}_{BG}$. To alleviate this problem, we add the second term as a regularization to force the module must predict valid BG regions for every image.

3.3.2 Background Eliminating Module

We further use the expanded BG prototype $\hat{\boldsymbol{P}}_{BG}$ to filter out the BG information from the query feature map via prototypical learning. Specifically, we first concatenate $\hat{\boldsymbol{P}}_{BG}$ with $\boldsymbol{X}^q$ and then use a convolutional layer $\mathcal{F}_{1\times1}(\cdot)$ to exclude the BG information in the query. The whole process can be denoted as:

$$\boldsymbol{X}^q_{BG} = \mathcal{F}_{1\times1}(\boldsymbol{X}^q \oplus \hat{\boldsymbol{P}}_{BG}), \tag{4}$$

where $\boldsymbol{X}^q_{BG} \in \mathbb{R}^{H\times W\times C}$ denotes the BG-filtered query feature and $\mathcal{F}_{1\times1}(\cdot)$ denotes a 1×1 convolutional layer.

3.4. Support Feature Matching

Following previous methods, we further use dense feature matching to activate the target object region on the $\boldsymbol{X}^q_{BG}$. Concretely, masked average pooling (MAP) is first used on the support feature map $\boldsymbol{X}^s$ to get the support prototype $\boldsymbol{P}^s \in \mathbb{R}^{1\times1\times C}$. Then, it is expanded to $\hat{\boldsymbol{P}}^s \in \mathbb{R}^{H\times W\times C}$ and concatenated with the BG-filtered query feature $\boldsymbol{X}^q_{BG}$. We also follow [31] and introduce a prior confidence map $\boldsymbol{C}_p \in \mathbb{R}^{\,H\times W\times 1}$ via computing the maximum similarity score at pixel-level. After that, we obtain the activated query feature $\boldsymbol{X}^q_{act} \in \mathbb{R}^{H\times W\times C}$ and further achieve the initial target object prediction $\boldsymbol{y}^q_{ini} \in \mathbb{R}^{H\times W\times 1}$:

$$\boldsymbol{X}^q_{act} = \mathcal{F}_{1\times1}(\boldsymbol{X}^q_{BG} \oplus \hat{\boldsymbol{P}}^s \oplus \boldsymbol{C}_p), \tag{5}$$

$$\boldsymbol{y}^q_{ini} = \mathcal{F}_{3\times3}(\boldsymbol{X}^q_{act}), \tag{6}$$

where $\mathcal{F}_{1\times1}(\cdot)$ is the same as in (4) and $\mathcal{F}_{3\times3}(\cdot)$ is the same as in (1).

3.5. Distracting Objects Eliminating

3.5.1 Distracting Object Eliminating Module

Although we have eliminated BG information in $\boldsymbol{X}^q_{act}$, it may still suffer from the distraction of DOs. To this end, we design the DOEM to further filter out DO information from $\boldsymbol{X}^q_{act}$ for more accurate target object prediction. To be specific, we mine the potential DO region in the query based on the known BG region in $\boldsymbol{y}^q_{BG}$ and the target object region in $\boldsymbol{y}^q_{ini}$. Intuitively, the DO region is complementary to the union of the BG region and target region. Hence, we have:

$$\boldsymbol{Y}^q_{DO} = 1 - (\boldsymbol{Y}^q_{BG} \cup \boldsymbol{Y}^q_{ini}), \tag{7}$$

where $\boldsymbol{Y}_{DO} \in \mathbb{R}^{H\times W\times 1}$ denotes the DO mask. $\boldsymbol{Y}^q_{BG}$ and $\boldsymbol{Y}^q_{ini}$ are the binary maps corresponding to $\boldsymbol{y}^q_{BG}$ and $\boldsymbol{y}^q_{ini}$, respectively.

Next, we utilize $\boldsymbol{Y}^q_{DO}$ to obtain the DO prototype $\boldsymbol{P}^q_{DO} \in \mathbb{R}^{1\times1\times C}$ via performing MAP on the query feature map:

$$\boldsymbol{P}^q_{DO} = \frac{\sum \boldsymbol{X}^q \otimes \boldsymbol{Y}^q_{DO}}{\sum \boldsymbol{Y}^q_{DO}}, \tag{8}$$

where $\otimes$ denotes the element-wise multiplication and the summation sums over all spatial locations.

After that, we expand the $\boldsymbol{P}^q_{DO}$ into $\hat{\boldsymbol{P}}^q_{DO} \in \mathbb{R}^{H\times W\times C}$ and combine it with the activated query feature map $\boldsymbol{X}^q_{ini}$ to eliminate the DO information. Finally, the combined feature is passed into a segmentation network, for which we use the Feature Enrichment Module (FEM) in [31], to obtain the BG and DO-free prediction:

$$\boldsymbol{y}^q = \mathrm{Seg}(\boldsymbol{X}^q_{ini} \oplus \hat{\boldsymbol{P}}^q_{DO}), \tag{9}$$

where $\boldsymbol{y}^q \in \mathbb{R}^{H\times W\times 1}$ is the final target object segmentation result of our whole model.

3.5.2 Prototypical Contrastive Learning

The DOEM only cares about the DO mask $\boldsymbol{Y}^q_{DO}$ in the DO eliminating process. However, a good DO eliminating model requires not only accurate DO masks, but also good prototype feature embeddings that can differentiate the target objects from DOs easily. Inspired by recent research on contrastive learning, we propose the prototypical contrastive learning (PCL) method to refine the feature embeddings of different prototypes. With the help of PCL, we want to make the prototype features between the target objects and the DOs more discriminative, and the prototypes between the target objects of the query and the support more similar.

To this end, we need to obtain the target prototypes and DO prototypes for both the query and the support first. For the target prototypes, we have obtained it for the support, *i.e.*, $\boldsymbol{P}^s$, from Section 3.4. For the query image, we first binarize the final target prediction $\boldsymbol{y}^q$ as the target mask and then adopt MAP on the query feature to generate the target prototype of the query $\boldsymbol{P}^q \in \mathbb{R}^{1\times1\times C}$. As for the DO prototypes, we have computed it for the query, *i.e.*, $\boldsymbol{P}^q_{DO}$, in (8). For the support image, we adopt the same workflow to compute the DO prototype of the support $\boldsymbol{P}^s_{DO}$, *i.e.*, using the target mask $\boldsymbol{M}^s$ and the BG mask $\boldsymbol{Y}^s_{BG}$ to generate the DO mask, and then conducting MAP on the support feature.

Prototypical Contrastive Learning Loss. According to the paradigm of contrastive learning, we propose a PCL loss to optimize the above prototype feature embeddings. For the query prototype $\boldsymbol{P}^q$, we treat the corresponding support prototype $\boldsymbol{P}^s$ as the positive sample, while the DO prototypes in both query and support as negative samples. Considering that a large number of negative samples is indispensable for contrastive learning, we build a DO prototype bank $\mathcal{B}$ to store the embeddings of 2000 DO prototypes in the latest batches during training. Note that they can be sampled across different episodes of the same class. At last, inspired by InfoNCE [23], we propose our PCL loss:

$$L_{PCL} = -log\frac{e^{cos(\boldsymbol{P}^q,\boldsymbol{P}^s)}}{\sum_{\mathcal{B}}\{e^{cos(\boldsymbol{P}^q,\boldsymbol{P}^q_{DO})} + e^{cos(\boldsymbol{P}^q,\boldsymbol{P}^s_{DO})}\}}, \tag{10}$$

where $cos(,)$ denotes the cosine similarity.

3.6. Total Training Loss

We use two binary cross-entropy losses to supervise the training of the initial target prediction $\boldsymbol{y}^q_{ini}$ and the final prediction $\boldsymbol{y}^q$, composing the target segmentation loss L_T. Finally, our total training loss includes L_T, the BG loss L_{BG} in (3), and the PCL loss L_{PCL} in (10):

$$L = \beta L_T + \lambda L_{BG} + \gamma L_{PCL}, \tag{11}$$

$$L_T = \mathrm{BCE}(\boldsymbol{y}^q_{ini}, \boldsymbol{M}^q) + \mathrm{BCE}(\boldsymbol{y}^q, \boldsymbol{M}^q), \tag{12}$$

where BCE denotes the binary cross-entropy loss and β, λ, γ are adjustable loss weights.

4. Experiments

4.1. Datasets and Evaluation Metrics

Datasets. We evaluate our model on two benchmark datasets, *i.e.*, the PASCAL-5^i dataset [26] and the COCO-20^i dataset [22]. PASCAL-5^i is constructed based on the PASCAL VOC 2012 dataset [7] and external annotations

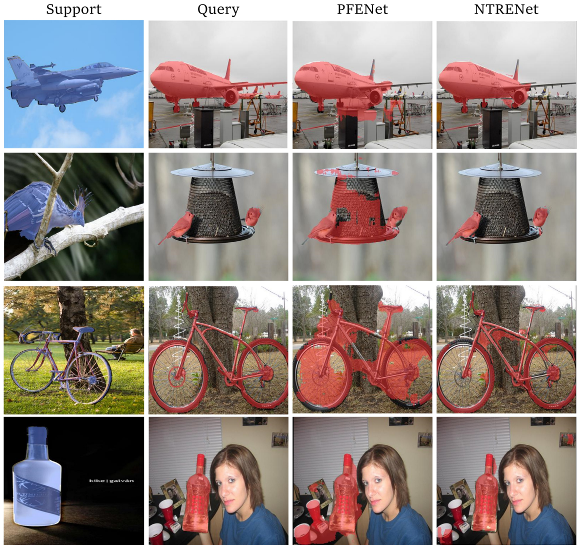


Figure 4. Qualitative results of our proposed NTRENet and PFENet. From left to right: support images, query images, prediction of PFENet, prediction of NTRENet.

from SDS [9]. The total 20 categories are partitioned into 4 folds as in [33] for cross validation and each fold contains 5 categories. COCO-20^i is a larger datasets based on the MSCOCO [16] dataset. Similar to PASCAL-5^i, the total 80 categories are also partitioned into 4 folds for cross validation, where each fold includes 20 categories. For both the datasets, we test on 1 fold and train on the remaining 3 folds.

Evaluation Metrics. Following previous methods [19,20, 26, 27], we adopt the class mean intersection over union (mIoU) as a primary evaluation metric for ablation studies and comparisons. In addition, we report the results of foreground-background IoU (FB-IoU), which only cares about the performance on target and non-target regions instead of differentiating categories, for a more comprehensive comparison. , the precision, whose formulation follows $\frac{TP}{TP+FP}$, also be leveraged to report our modules performance on decreasing false positives.

4.2. Implementation Details

Following previous works, we respectively use ResNet-50, ResNet-101 [12], and VGG-16 [28] as the backbones to construct our network for fair comparisons. These backbones are all pre-trained on the ImageNet classification task and their weights are fixed during training.

Our network is implemented using PyTorch [24] and all the experiments are conducted on one NVIDIA RTX 3090 GPU. We use random scaling, horizontal flipping, and random rotation within [-10,10] degrees as data augmentation to increase the training data. Finally, we randomly crop images and masks with the size of 473 × 473 to train our model. During training, we use SGD as our optimizer, where the initial learning rate, batch size, weight decay, and momentum are set as 0.03, 32, 0.0001, and 0.9, respectively. We train our model for 200 epochs on PASCAL-5^i and 50 epochs on COCO-20^i, respectively. The learning rate is decayed using the polynomial annealing policy with the power set to 0.9. During the evaluation, we follow [38] to randomly sample 1000 support-query pairs on PASCAL-5^i and 4000 pairs on COCO-20^i, respectively.

4.3. Comparison with State-of-the-art Methods

PASCAL-5^i. Table 1 shows the performance comparison on PASCAL-5^i between our method and several representative models. We can see that, on all the three backbones(*i.e.*, VGG-16, Resnet-50, and Resnet-101), our method outperforms all previous models by a large margin in terms of both mIoU and FB-IoU. Specifically, under the 1-shot setting, the averaged mIoU scores of our method are 59.0, 64.2, and 63.7 on the VGG-16, Resnet-50, and Resnet-101 backbones, respectively, surpassing state-of-the-art results by 1.2%, 3.4%, and 1.8%, respectively. Meanwhile, in terms of FB-IoU, our method outperforms the previous best results by 1.1%, 5.0%, and 3.3% on the three backbones, respectively. Under the 5-shot setting, our method only obtains the best results on the VGG-16 backbone in terms of mIoU, but outperforms previous state-of-the-art FB-IoU results by 2.6%, 5.7%, and 1% on all three backbones, respectively.

COCO-20^i. Although COCO-20^i is a more challenging dataset with a large number of images with realistic scenes, we still obtain superior performance, which is shown in Table 2. Here we follow previous works and only use the Resnet-50 and Resnet-101 backbones. Table 2 shows that our method respectively yields the averaged mIoU scores of 39.3 and 39.1 on the two backbones under the 1-shot setting, outperforming previous best results by a large margin of 4.8% and 5.7%, respectively. In the 5-shot setting, the FB-IoU results also verify the superiority of our method, despite the challenging scenarios.

Limitations. We find that the averaged mIoU results of our model do not achieve obvious advantages in the 5-shot setting, compared with previous state-of-the-arts. We argue that this is reasonable since our method mainly focuses on eliminating non-target regions instead of segmenting the target object. As such, although the number of support samples increases from one to five, it does not introduce additional non-target information to our method. However, we hope our work could provide a novel perspective on the opposite side of traditional methods for future works.

Qualitative Comparison. We show the qualitative comparison of the predicted segmentation masks generated by our method and a typical traditional model that focuses on segmenting the target object, *i.e.*, PFENet [31], in Figure 4. We can see that PFENet could not segment the target objects accurately due to the distraction of non-target regions.

Table 1. Class mIoU and FB-IoU results of four folds on PASCAL-5^i. The results of 'Mean' are the averaged class mIoU scores of all four folds. The detailed FB-IoU results of each fold are omitted in this table for simplicity. **Bold** indicates the best results.

Backbone	Methods	1-Shot						5-Shot					
		Fold-0	Fold-1	Fold-2	Fold-3	**Mean**	FB-IoU	Fold-0	Fold-1	Fold-2	Fold-3	**Mean**	FB-IoU
VGG-16	OSLSM [26]	33.6	55.3	40.9	33.5	40.8	61.3	35.9	58.1	42.7	39.1	44.0	61.5
	co-FCN [25]	36.7	50.6	44.9	32.4	41.1	60.1	37.5	50.0	44.1	33.9	41.4	60.2
	RPMM [37]	47.1	65.8	50.6	48.5	53.0	-	50.0	66.5	51.9	47.6	54.0	-
	PFENet [31]	56.9	**68.2**	54.4	52.4	58.0	72.3	59.0	**69.1**	54.8	52.9	59.0	72.3
	MMNet [35]	57.1	67.2	56.6	52.3	58.3	-	56.6	66.7	**63.6**	56.5	58.3	-
	NTRENet	**57.7**	67.6	**57.1**	**53.7**	**59.0**	**73.1**	**60.3**	68.0	55.2	**57.1**	**60.2**	**74.2**
ResNet-50	CANet [42]	52.5	65.9	51.3	51.9	55.4	66.2	55.5	67.8	51.9	53.2	57.1	69.6
	RPMM [37]	55.2	66.9	52.6	50.7	56.3	-	56.3	67.3	54.5	51.0	57.3	-
	PFENet [31]	61.7	69.5	55.4	56.3	60.8	73.3	63.1	70.7	55.8	57.9	61.9	73.9
	SCL [41]	63.0	70.0	56.5	57.7	61.8	71.9	64.5	70.9	57.3	58.7	62.9	72.8
	ASGNet [14]	58.8	67.9	56.8	53.7	59.3	69.2	63.7	70.6	64.2	57.4	63.9	74.2
	ReRPI [1]	59.8	68.3	62.1	48.5	59.7	-	64.6	71.4	71.1	59.3	**66.6**	-
	SAGNN [36]	64.7	69.6	57.0	57.2	62.1	73.2	64.9	70.0	57.0	59.3	62.8	73.3
	MLC [38]	59.2	71.2	**65.6**	52.5	62.1	-	63.5	71.6	**71.2**	58.1	66.1	-
	NTRENet	**65.4**	**72.3**	59.4	**59.8**	**64.2**	**77.0**	**66.2**	**72.8**	61.7	**62.2**	65.7	**78.4**
ResNet-101	DAN [32]	54.7	68.6	57.8	51.6	58.2	71.9	57.9	69.0	60.1	54.9	60.5	72.3
	PPNet [20]	52.7	62.8	57.4	47.7	55.2	70.9	60.3	70.0	69.4	60.7	65.1	77.5
	PFENet [31]	60.5	69.4	54.4	55.9	60.1	72.9	62.8	70.4	54.9	57.6	61.4	73.5
	ASGNet [14]	59.8	67.4	55.6	54.4	59.3	71.7	64.6	71.3	64.2	57.3	64.4	75.2
	ReRPI [1]	59.6	68.6	**62.2**	47.2	59.4	-	66.2	71.4	67.0	57.7	65.6	-
	MLC [38]	60.8	71.3	61.5	56.9	62.6	-	65.8	**74.9**	**71.4**	63.1	**68.8**	-
	NTRENet	**65.5**	**71.8**	59.1	**58.3**	**63.7**	**75.3**	**67.9**	73.2	60.1	**66.8**	67.0	**78.2**

Table 2. Class mIoU and FB-IoU results of four folds on COCO-20^i. The results of 'Mean' are the averaged class mIoU scores of all the four folds. The detailed FB-IoU results of each fold are omitted in this table for simplicity. **Bold** indicates the best results.

Backbone	Methods	1-Shot						5-Shot					
		Fold-0	Fold-1	Fold-2	Fold-3	**Mean**	FB-IoU	Fold-0	Fold-1	Fold-2	Fold-3	**Mean**	FB-IoU
ResNet-50	PPNet [20]	28.1	30.8	29.5	27.7	29.0	-	39.0	40.8	37.1	37.3	38.5	-
	RPMM [37]	29.5	36.8	28.9	27.0	30.6	-	33.8	42.0	33.0	33.3	35.5	-
	ASGNet [14]	-	-	-	-	34.6	60.4	-	-	-	-	**42.5**	67.0
	MMNet [35]	34.9	41.0	37.2	37.0	37.5	-	37.0	40.3	39.3	36.0	38.2	-
	MLC [38]	**46.8**	35.3	26.2	27.1	33.9	-	**54.1**	41.2	34.1	33.1	40.6	-
	NTRENet	36.8	**42.6**	**39.9**	**37.9**	**39.3**	**68.5**	38.2	**44.1**	**40.4**	**38.4**	40.3	**69.2**
ResNet-101	DAN [32]	-	-	-	-	24.4	62.3	-	-	-	-	29.6	63.9
	SCL [41]	36.4	38.6	37.5	35.4	37.0	-	38.9	40.5	41.5	38.7	39.9	-
	PFENet [31]	34.3	33.0	32.3	30.1	32.4	58.6	38.5	38.6	38.2	34.3	37.4	61.9
	MLC [38]	**50.2**	37.8	27.1	30.4	36.4	-	**57.0**	46.2	37.3	37.2	**44.4**	-
	SAGNN [36]	36.1	**41.0**	38.2	33.5	37.2	60.9	40.9	**48.3**	42.6	38.9	42.7	63.4
	NTRENet	38.3	40.4	**39.5**	**38.1**	**39.1**	**67.5**	42.3	44.4	**44.2**	**41.7**	43.2	**69.6**

However, our proposed NTRENet can obtain much more accurate results with much fewer false positive predictions in BG and DO regions, thus clearly demonstrating the effectiveness of our proposed method.

4.4. Ablation Study

Effectiveness of Different Modules. We conduct extensive ablation studies on PASCAL-5^i in the 1-shot setting to validate the effectiveness of our proposed key modules, *i.e.*, BGEM, DOEM, and PCL. We remove these three modules from our NTRENet as the baseline model, which only uses the support prototype to directly segment the target object as in [31]. As Table 3 shows, eliminating the BG regions using BGEM achieves as large as 4% performance improvement compared to the baseline model. Meanwhile, using DOEM to mine and eliminate DO regions obtains another 2% performance gain. Finally, using the PCL scheme to boost the model capability of discriminating different objects leads to further 1% performance improvement. These results clearly demonstrate the effectiveness of our proposed BGEM, DOEM, and PCL. In addition, we use precision to verify the performance of our method on decreasing false positives. The results show that our full method achieves 3% precision improvement compared to the baseline model and all of the proposed BGEM, DOEM, and PCL can progressively improve the precision, *i.e.*, decrease false positives.

Table 3. Ablation study of the key modules in our NERTNet. mIoU results are reported on the PASCAL-5^i dataset under the 1-shot setting.

BGEM	DOEM	PCL	Fold-0	Fold-1	Fold-2	Fold-3	**Mean**	Precision
			60.8	68.2	55.4	55.3	60.0	61.9
✓			63.2	71.1	57.7	57.4	62.4	62.8
✓	✓		64.7	71.9	58.8	59.0	63.6	63.3
✓	✓	✓	**65.4**	**72.3**	**59.4**	**59.8**	**64.2**	**63.6**

Choice of negative samples in PCL. The whole non-target region includes both BG and DO regions. Hence, we choose other alternative negative samples, *i.e.*, prototypes generated from the whole non-target region and the BG region, respectively, to apply in PCL. We conduct compara-

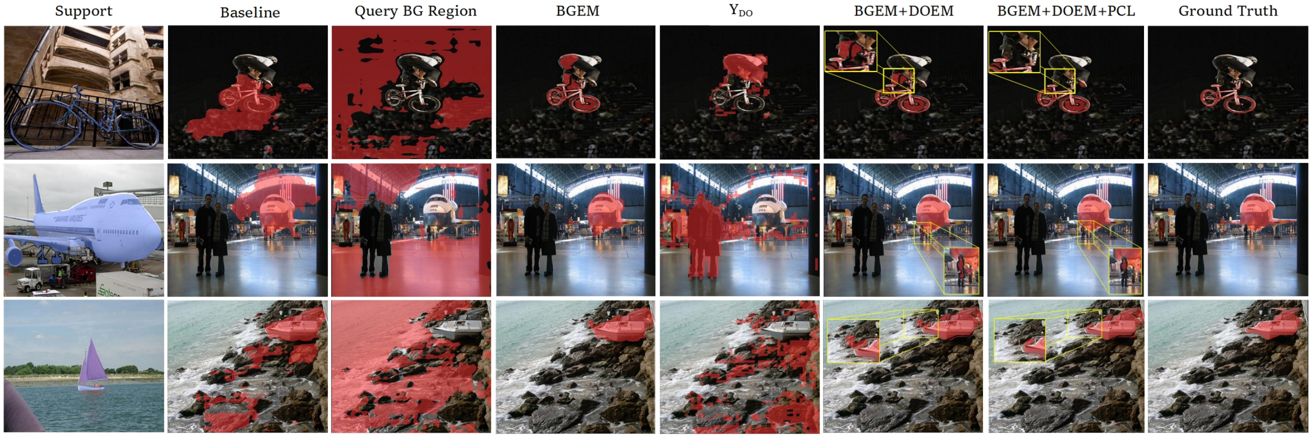


Figure 5. Visualization of different ablative results. From left to right: Support images, the results of baseline, BG prediction, the results of only using BGEM, DO prediction, the results of using BGEM+DOEM, the results of using BGEM+DOEM+PCL (*i.e.*, full model), Ground truth.

Table 4. Comparison of the choice of negative samples in PCL. mIoU results are reported on the PASCAL-5^i dataset under the 1-shot setting.

Negative samples	split0	split1	split2	split3	mean
Non-target Prototypes	63.7	69.0	57.8	56.8	61.8
BG Prototypes	64.4	68.7	58.4	58.6	62.6
DO Prototypes	**65.4**	**72.3**	**59.4**	**59.8**	**64.2**

tive experiments on PASCAL-5^i under the 1-shot setting. In Table 4, the results show that DO prototypes work more effectively than others since DO regions are more confusing and thus play a role of hard negative samples.

Qualitative Comparison. We further show some qualitative results in Figure 5 to prove the effectiveness of our proposed BGEM, DOEM, and PCL in an intuitionistic way. Column 2 shows predictions from baseline. In column 3, we show the BG prediction masks obtained from BGMM. We find that our BGMM can effectively mine the universal BG regions in query. Column 5 reveals the predicted masks of DO regions in DOEM. Columns 4 and 6 show that using BGEM and DOEM can effectively help eliminate the BG and DO regions compared with the baseline results. Finally, column 7 indicates that using PCL can further discriminate the target objects from DOs in detail.

Influence of the Channel Dimension of the BG Prototype. The channel dimension of the BG prototype is crucial since it determines how much general BG information it can encode. We conduct ablation experiments to explore its optimal value and the results are shown in Figure 6. It shows that using 512 channels achieves the best performance for fold 2 and 3, while using 640 channels outperforms other values for fold 0, 1, and the mean result. Hence, we use 640 channels in the BG prototype in our network setting.

5. Conclusion

We address the few-shot semantic segmentation from a new perspective and propose a novel NTRE framework

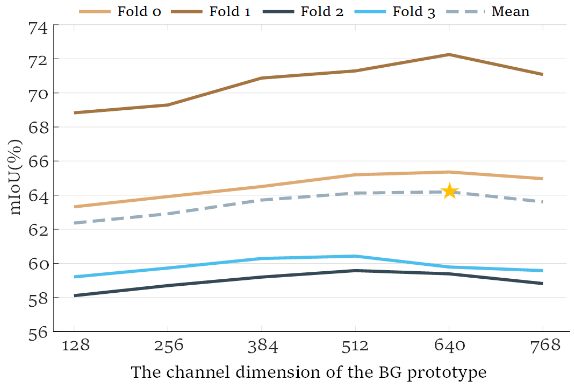


Figure 6. Ablation study on the channel dimension of the BG prototype in the 1-shot setting on the PASCAL-5^i dataset. ★ indicates the best result of the averaged class mIoU.

to pay attention to BG and DO regions. We propose the BGMM, BGEM, and DOEM for effectively implementing the mining and eliminating to the BG and DOs. Particularly, the BG mining loss is proposed to supervise the learning of the BGMM and a BG prototype without using BG ground truth. Besides, PCL is proposed to improve the model ability for better distinguishing target objects from DOs. Extensive experiments on two benchmark datasets demonstrate the performance superiority of our method over the previous methods.

Acknowledgments: This work was supported in part by the Key-Area Research and Development Program of Guangdong Province under Grant 2019B010110001, and the National Natural Science Foundation of China under Grants 62071388, 62136007, U20B2065 and 62036005, the Key R&D Program of Shaanxi Province under Grant 2021ZDLGY01-08, and the National Key R&D Program of China under Grant 2020AAA0105701.

References

[1] Malik Boudiaf, Hoel Kervadec, Ziko Imtiaz Masud, Pablo Piantanida, Ismail Ben Ayed, and Jose Dolz. Few-shot segmentation without meta-learning: A good transductive inference is all you need? In *CVPR*, pages 13979–13988, 2021. 7

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

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. pages 1597–1607. PMLR, 2020. 3

[4] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 3

[5] Sungha Choi, Joanne T Kim, and Jaegul Choo. Cars can't fly up in the sky: Improving urban-scene segmentation via height-driven attention networks. In *CVPR*, pages 9373–9383, 2020. 2

[6] Nanqing Dong and Eric P Xing. Few-shot semantic segmentation with prototype learning. In *BMVC*, volume 3, 2018. 1

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 5

[8] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *CVPR*, pages 3146–3154, 2019. 2

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

[10] Junjun He, Zhongying Deng, Lei Zhou, Yali Wang, and Yu Qiao. Adaptive pyramid context network for semantic segmentation. In *CVPR*, pages 7519–7528, 2019. 2

[11] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 3

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 6

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 2

[14] Gen Li, Varun Jampani, Laura Sevilla-Lara, Deqing Sun, Jonghyun Kim, and Joongkyu Kim. Adaptive prototype learning and allocation for few-shot segmentation. In *CVPR*, pages 8334–8343, 2021. 1, 3, 7

[15] Xia Li, Zhisheng Zhong, Jianlong Wu, Yibo Yang, Zhouchen Lin, and Hong Liu. Expectation-maximization attention networks for semantic segmentation. In *ICCV*, pages 9167–9176, 2019. 2

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014. 6

[17] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *CVPR*, pages 3089–3098, 2018. 3

[18] Nian Liu, Ni Zhang, Kaiyuan Wan, Ling Shao, and Junwei Han. Visual saliency transformer. In *ICCV*, pages 4722–4732, 2021. 3

[19] Weide Liu, Chi Zhang, Guosheng Lin, and Fayao Liu. Crnet: Cross-reference networks for few-shot segmentation. In *CVPR*, pages 4165–4173, 2020. 6

[20] Yongfei Liu, Xiangyi Zhang, Songyang Zhang, and Xuming He. Part-aware prototype network for few-shot semantic segmentation. In *ECCV*, pages 142–158. Springer, 2020. 1, 6, 7

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015. 1, 2

[22] Khoi Nguyen and Sinisa Todorovic. Feature weighting and boosting for few-shot segmentation. In *ICCV*, pages 622–631, 2019. 5

[23] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 5

[24] 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. *Advances in neural information processing systems*, 32:8026–8037, 2019. 6

[25] Kate Rakelly, Evan Shelhamer, Trevor Darrell, Alyosha Efros, and Sergey Levine. Conditional networks for few-shot semantic segmentation. 2018. 7

[26] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. *arXiv preprint arXiv:1709.03410*, 2017. 2, 5, 6, 7

[27] Mennatullah Siam, Boris N Oreshkin, and Martin Jagersand. Amp: Adaptive masked proxies for few-shot segmentation. In *ICCV*, pages 5249–5258, 2019. 6

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6

[29] Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017. 1

[30] Andrew Tao, Karan Sapra, and Bryan Catanzaro. Hierarchical multi-scale attention for semantic segmentation. *arXiv preprint arXiv:2005.10821*, 2020. 2

[31] Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *IEEE TPAMI*, (01):1–1, 2020. 1, 2, 5, 6, 7

[32] Haochen Wang, Xudong Zhang, Yutao Hu, Yandan Yang, Xianbin Cao, and Xiantong Zhen. Few-shot semantic segmentation with democratic attention networks. In *ECCV*, pages 730–746. Springer, 2020. 7

[33] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. Panet: Few-shot image semantic segmentation with prototype alignment. In *ICCV*, pages 9197–9206, 2019. 1, 3, 6

[34] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. *arXiv preprint arXiv:2101.11939*, 2021. 3

[35] Zhonghua Wu, Xiangxi Shi, Guosheng Lin, and Jianfei Cai. Learning meta-class memory for few-shot semantic segmentation. In *ICCV*, pages 517–526, 2021. 7

[36] Guo-Sen Xie, Jie Liu, Huan Xiong, and Ling Shao. Scale-aware graph neural network for few-shot semantic segmentation. In *CVPR*, pages 5475–5484, 2021. 7

[37] Boyu Yang, Chang Liu, Bohao Li, Jianbin Jiao, and Qixiang Ye. Prototype mixture models for few-shot semantic segmentation. In *ECCV*, pages 763–778. Springer, 2020. 7

[38] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. Mining latent classes for few-shot segmentation. *arXiv preprint arXiv:2103.15402*, 2021. 3, 6, 7

[39] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *CVPR*, pages 3684–3692, 2018. 2

[40] Yuhui Yuan, Lang Huang, Jianyuan Guo, Chao Zhang, Xilin Chen, and Jingdong Wang. Ocnet: Object context network for scene parsing. *arXiv preprint arXiv:1809.00916*, 2018. 2

[41] Bingfeng Zhang, Jimin Xiao, and Terry Qin. Self-guided and cross-guided learning for few-shot segmentation. In *CVPR*, pages 8312–8321, 2021. 7

[42] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *CVPR*, pages 5217–5226, 2019. 1, 2, 7

[43] Fan Zhang, Yanqin Chen, Zhihang Li, Zhibin Hong, Jingtuo Liu, Feifei Ma, Junyu Han, and Errui Ding. Acfnet: Attentional class feature network for semantic segmentation. In *ICCV*, pages 6798–6807, 2019. 2

[44] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Thomas S Huang. Sg-one: Similarity guidance network for one-shot semantic segmentation. *IEEE transactions on cybernetics*, 50(9):3855–3865, 2020. 1, 2

[45] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, pages 2881–2890, 2017. 2

[46] Zhen Zhu, Mengde Xu, Song Bai, Tengteng Huang, and Xiang Bai. Asymmetric non-local neural networks for semantic segmentation. In *ICCV*, pages 593–602, 2019. 2