computer sciences and mathematics forum

MDPI

Proceeding Paper

Dual Complementary Prototype Learning for Few-Shot Segmentation †

Qian Ren and Jie Chen *

School of Electronic and Computer Engineering, Peking University, Shenzhen 518055, China; renq2019@pku.edu.cn

* Correspondence: chenj@pcl.ac.cn

† Presented at the AAAI Workshop on Artificial Intelligence with Biased or Scarce Data (AIBSD), Online, 28 February 2022.

Abstract: Few-shot semantic segmentation aims to transfer knowledge from base classes with sufficient data to represent novel classes with limited few-shot samples. Recent methods follow a metric learning framework with prototypes for foreground representation. However, they still face the challenge of segmentation of novel classes due to inadequate representation of foreground and lack of discriminability between foreground and background. To address this problem, we propose the Dual Complementary prototype Network (DCNet). Firstly, we design a training-free Complementary Prototype Generation (CPG) module to extract comprehensive information from the mask region in the support image. Secondly, we design a Background Guided Learning (BGL) as a complementary branch of the foreground segmentation branch, which enlarges difference between the foreground and its corresponding background so that the representation of novel class in the foreground could be more discriminative. Extensive experiments on PASCAL-5^i and COCO-20^i demonstrate that our DCNet achieves state-of-the-art results.

Keywords: few-shot; semantic segmentation

check for updates

Citation: Ren, Q.; Chen, J. Dual Complementary Prototype Learning for Few-Shot Segmentation. *Comput. Sci. Math. Forum* **2022**, *3*, 8. https://doi.org/10.3390/cmsf2022003008

Academic Editors: Kuan-Chuan Peng and Ziyan Wu

Published: 29 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Attributed to the development of convolutional neural networks (CNNs) with its strong representation ability and the access of large-scale datasets, semantic segmentation and object detection have developed tremendously. However, it is worth to point out that annotating a large number of object masks is time-consuming, expensive, and sometimes infeasible in some scenarios, such as computer-aided diagnosis systems. Moreover, without massive annotated data, the performance of deep learning models drops dramatically on classes that do not appear in the training dataset. Few-shot segmentation (FSS) is a promising field to tackle this issue. Unlike conventional semantic segmentation, which merely segments the classes appearing in the training set, few-shot segmentation utilizes one or a few annotated samples to segment new classes.

They firstly extract features from both query and support images, and then the support features and their masks are encoded into a single prototype [1] to represent foreground semantics or a pair of prototypes [2,3] to represent the foreground and background. Finally, they conduct dense comparison between prototype(s) and query feature. Feature comparison methods are usually performed in one of two ways: explicit metric function, (e.g., cosine-similarity [3]) and implicit metric function (e.g., relationNet [4]).

As shown in Figure 1a, it is common-sense [2,5,6] that using a single prototype generated by masked average pooling is unable to carry sufficient information. Specifically, due to variant appearance and poses, using masked average pooling only retains the information of discriminative pixels and ignores the information of plain pixels. To overcome this problem, multi-prototype strategy [2,5,6] is proposed by dividing foreground regions into several pieces.

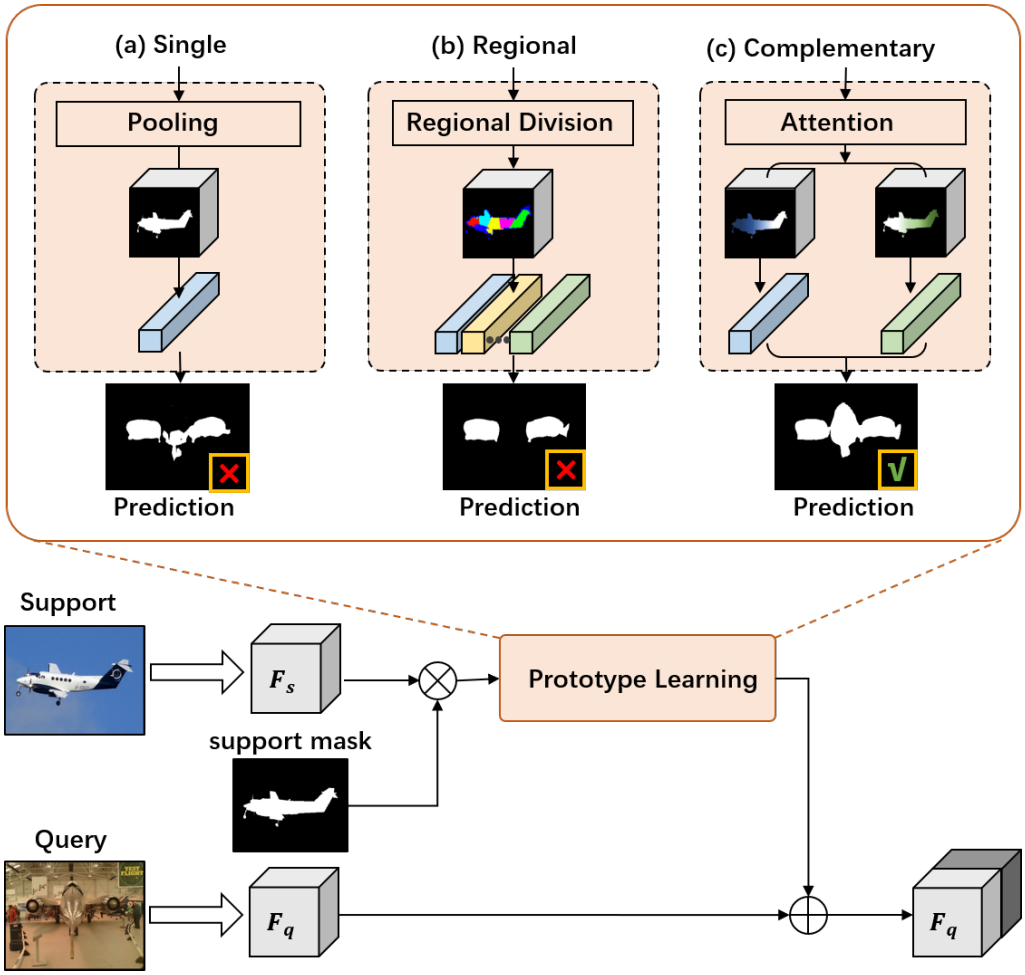

Figure 1. Illustration of difference in prototype learning for 1-shot segmentation. (**a**) Single prototype methods [1,7] tend to lose information as plain pixels. (**b**) Multi-prototype methods [2,5,8] based on regional division may damage the representation for the whole object. (**c**) Our Complementary Prototype Generation module retains the information of discriminative pixels and plain pixels adaptively.

However, as shown in Figure 1b, these multi-prototype methods still suffer from two drawbacks. Firstly, the whole representation of foreground region is weakened, since existing methods split regions into several pieces and damage the correlation among the generated prototypes. Moreover, current methods often ignore inter-class similarity between foreground and background, and their training strategy in the context of segmenting the main foreground objects leads to underestimating the discrimination between the foreground and background. As a result, existing multi-prototype methods tend to misclassify background pixels into foreground.

In this paper, we propose a simple yet effective method, called Dual Complementary prototype Network (DCNet), to overcome the above mentioned drawbacks. Specifically, it is composed of two branches to segment the foreground and background in a complementary manner, and both segmentation branches rely on our proposed Complementary Prototype Generation (CPG) module. The CPG module is proposed to extract comprehensive support information from the support set. Through global average pooling with support mask, we extract the average prototype at first, and we obtain its attention weight on the support image by calculating the cosine distance between the foreground feature and the average prototype iteratively. In this way, we can easily figure out which part of the information is focused and which part of the information is ignored without segmentation on support image. Then we use this attention weight to generate a pair of prototypes to represent

the focused and the ignored region. By using a weight map to generate prototypes for comparison, we can preserve the correlation among the generated prototypes and avoid the information loss to a certain extent.

Furthermore, we introduce background guided learning to pay additional attention on the inter-class similarity between the foreground and background. Considering that the background in support images is not always the same as that in a query image, we adopt a different training manner from foreground segmentation, where the query background mask is used as guidance for query image background segmentation. In this way, our model could learn a more discriminative representation for distinguishing foreground and background. The proposed method effectively and efficiently improves the performance on FSS benchmarks without extra inference cost.

The main contributions of this work are summarized as follows.

1. We propose Complementary Prototype Generation (CPG) to learn powerful prototype representation without extra parameters costs;
2. We propose Background Guided Learning (BGL) to increase the feature discrimination between foreground and background. Besides, BGL is merely applied in the training phase so that it would not increase the inference time;
3. Our approach achieves the state-of-the-art results on both PASCAL-5^i and COCO-20^i datasets and improves the performance of the baseline model by 9.1% and 12.6% for 1-shot and 5-shot setting on COCO-20^i.

2. Related Work

2.1. Semantic Segmentation

Semantic segmentation, which aims to perform classification for each pixel, has been extensively investigated. Following Fully Convolution Network (FCN) [9], which uses fully convolutional layers instead of fully connected layers as a classifier for semantic segmentation, large numbers of network frameworks have been designed. For example, Unet adopted a multi-scale strategy and a encoder-decoder architecture to improve the performance of FCN, and PSPNet was proposed to use the pyramid pooling module (PPM) to generate object details. Deeplab [10,11] designed an Atrous Spatial Pyramid Pooling (ASPP) module, conditional random field (CRF) module, and dilated convolution to FCN architecture. Recently, attention mechanism has been introduced, PSANet [12] was proposed to use point-wise spatial attention with a bi-directional information propagation paradigm. Channel-wise attention [13] and non-local attention [14–17] are also effective for segmentation. These methods have managed to succeed in large-scale datasets but they are not designed to deal with rare and unseen classes and cannot be accommodated without fine-tuning.

2.2. Few-Shot Learning

Few-shot learning focuses on the generalization ability of models, so that they can learn to predict novel classes with a few annotated examples [4,18–21]. Matching networks [19] were proposed for 1-shot learning to exploit a special kind of mini-batches called episodes to match the training and testing environments, enhancing the generalization on the novel classes. Prototypical network [20] was introduced to compute the distances between the representation cluster centers for few-shot classification. Finn et al. [21] proposed an algorithm for meta-learning that is model-agnostic. Even though few shot learning has been extensively studied for classification task, it is still hard to adopt few-shot learning directly on segmentation due to the dense prediction.

2.3. Few-Shot Segmentation

As the extension of few-shot learning, few-shot semantic segmentation has also received considerable attention very recently. Shaban et al. first proposed the few-shot segmentation problem with a two-branch conditional network that learned the parameters on support images. Different from [22], later works [1–3,23,24] follow the idea of metric learning. Zhang et al. generates the foreground object segmentation of the support class by measuring the embedding similarity between query and supports, where their embeddings are extracted by the same backbone model. Generally, metric learning based methods can be divided into two groups: one group is inspired by ProtoNet [20], e.g., PANet [3] first embeds different foreground objects and the background into different prototypes via a shared feature extractor, and then measures the similarity between the query and the prototypes. The other group is inspired by relationNet [4], which learns a metric function to measure the similarity, e.g., Refs. [1,7,8] use an FPN-like structure to perform dense comparison with affinity alignment. Then, considering the incomplete representation of a single prototype, Li et al. [5] divide the masked region into pieces, the number of which is decided by the area of the masked region and then conducts masked average pooling for each piece to generate the numbers of the prototypes.Zhang et al. [6] utilize the uncovered foreground region and covered foreground region through segmentation on support images to generate a pair of prototypes to retrieve the loss information. However, compared to self-segmentation mechanism [6], our CPG does not need to segment on support images and utilization of CPG obtains competitive performance with few cost.s Compared to cluster methods [5,8], the experiment in the ablation study shows that our method can avoid over-fitting and generate stable performance in each setting.

Moreover, recent methods such as MLC [25] and SCNet [26] start to make use of knowledge hidden in the background. By exploiting the pre-training knowledge for the discovery of the latent novel class in the background, their methods bring huge improvements to the few-shot segmentation task. However, we argue that such a method is difficult to apply in realistic scenarios, since a novel class object is not only unlabelled but also unseen in the training set. Instead, we propose background guided learning to enhance the feature discriminability between the foreground and the background, which also improves the performance of the model.

3. Proposed Methods

3.1. Problem Setting

The aim of few-shot segmentation is to obtain a model that can learn to perform segmentation from only a few annotated support images in novel classes. The few-shot segmentation model should be trained on a dataset D_{train} and evaluated on a dataset D_{test}. Given the classes set in D_{train} is C_{train} and classes set in D_{test} is C_{test}, there is no overlap between training classes and test classes, e.g., $C_{train} \cap C_{test} = \emptyset$.

Following a previous definition [22], we divide the images into two non-overlapping sets of classes C_{train} and C_{test}. The training set D_{train} is built on C_{train} and the test set is built on C_{test}. We adopt the episode training strategy, which has been demonstrated as an effective approach for few-shot recognition. Each episode is composed of a shot support set $S = \{I_k^s, M_k^s\}_{k=1}^K$ and a query set $Q = I^q, M^q$ to form a K-shot episode $\{S, I^q\}$, where I^* andM^* are the image and its corresponding mask label, respectively. Then, the training set and test set are denoted by $D_{train} = \{S\}^{N_{train}}$ and $D_{test} = \{Q\}^{N_{test}}$, where N_{train} and N_{test} is the number of episodes for the training and test set. Note that both the mask M^s of the support set and the mask M^q of the query set are provided in the training phase, but only the support image mask M^s is included in the test phase.

3.2. Overview

As shown in Figure 2, our Dual Complementary prototype Network (DCNet) is trained via the episodic scheme on the support-query pairs. In episodic training, supports images and a query image are input to the share-weight encoder for feature extraction. Then,

the query feature is compared with prototypes of the current support class to generate a foreground segmentation mask via a FPN-like decoder. Besides, we propose an auxiliary supervision, named Background Guided Learning (BGL), where our network learns robust prototype representation for a class-agnostic background in an embedding space. In this supervision, the query feature is compared with prototypes of the query background to make a prediction on its own background. With this joint training strategy, our model can learn discriminative representation for foreground and background.

Thus, the overall optimization target can be briefly formulated as:

$$\mathcal{L}_{overall} = \mathcal{L}_{fg} + \gamma \mathcal{L}_{bg}, \tag{1}$$

where $\mathcal{L}_{fg}$ and $\mathcal{L}_{bg}$ denote the foreground segmentation loss and background segmentation loss, respectively, and γ is the balance weight, which is simply set as 1.

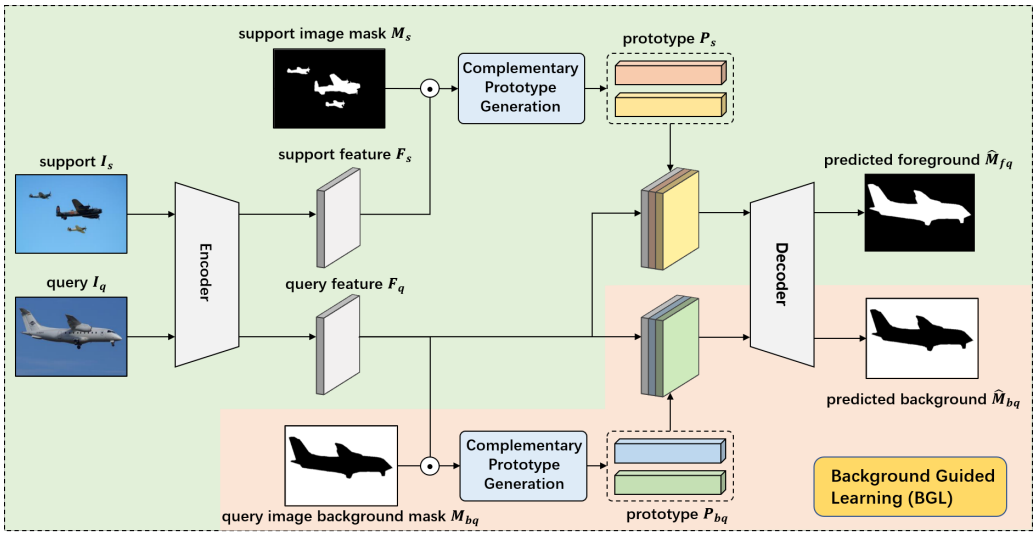

Figure 2. The framework of the proposed DCNet for 1-shot segmentation. At first, the encoder generates feature maps F_s and F_q from the support images and query images. Then, the support image masks M_s and related features are fed into CPG to generate a pair of foreground prototypes P_s. Finally, P_s is expanded and concatenated with the query feature F_q as an input to the decoder to predict the foreground in the query image. In the meantime, in BGL, the query feature F_q and its background mask M_{bq} are fed into CPG to generate a pair of background prototypes P_{bq}. P_{bq} is expanded and concatenated with query feature F_q as an input to the decoder to predict the background in the query image.

In the following subsections, we first elaborate our prototype generation algorithm. Then, background-guided learning on 1-shot setting is introduced, followed by inference.

3.3. Complementary Prototypes Generation

Inspired by SCL [6], we propose a simple and effective algorithm, named Complementary Prototypes Generation (CPG), as shown in Figure 3. This CPG algorithm generates a pair of complementary prototypes and aggregates information hidden in features based on cosine similarity. Specifically, given the support feature $F \in \mathbb{R}^{H \times W \times C}$ with the mask region as $M \in \mathbb{R}^{H \times W}$, we extract a pair of prototypes to fully represent the information in the mask region.

As the first step, we extract the targeted feature $F' \in \mathbb{R}^{H \times W \times C}$ filtered through mask M from F, in Equation (2),

$$F' = F \odot M \tag{2}$$

where $\odot$ represents element-wise multiplication. Then, we initiate prototype P_0 by masked average pooling, in Equation (3),

$$P_0 = \frac{\sum_i^H \sum_j^W F'_{i,j}}{\sum_i^H \sum_j^W M_{i,j}} \tag{3}$$

where i, j represents the coordination of each pixel, H, W denotes the width and height of feature F', respectively. Since $M_{i,j} \in 0, 1$, the sum of M represents the area of the foreground region.

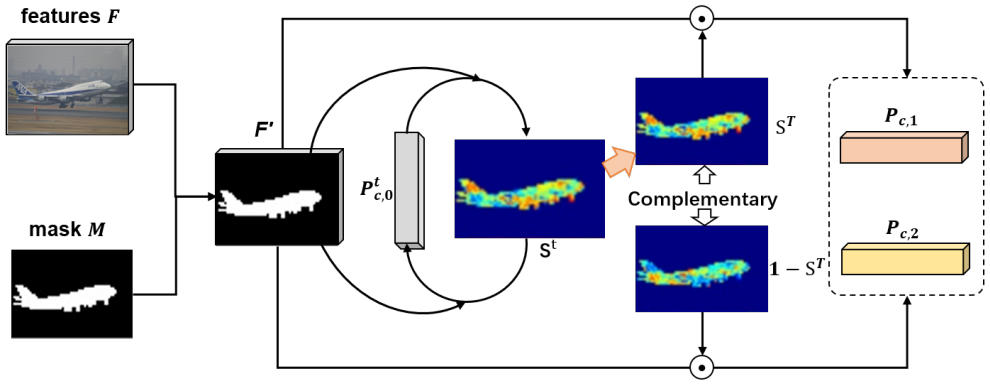

Figure 3. Illustration of the proposed Complementary Prototypes Generation. Similarity S^t and prototype $P^t_{c,0}$ is obtained in t-th iteration. The red arrow indicates the final result S^T after T iterations.

In the next step, we aggregate the foreground features into two complementary clusters. For each iteration t, we first compute the cosine distance matrix $S^t \in \mathbb{R}^{H \times W}$ between the prototype P_0^{t-1} and the targeted features F' as follows,

$$S^t = \text{cosine}(F', P_0^{t-1}) \tag{4}$$

As we keep the relu layer in the encoder layer, the cosine distance is limited in $[0, 1]$. To calculate the weight of target features contributed to P_0^t, we normalize the S matrix as:

$$S^t_{i,j} = \frac{S^t_{i,j}}{\sum_i^H \sum_j^W S^t_{i,j}} \tag{5}$$

Then, after the end of the iteration, based on matrix S^t, we aggregate the features into two complementary prototypes as:

$$P_1 = \sum_i^H \sum_j^W S_{i,j} * F'_{i,j} \tag{6}$$

$$P_2 = \sum_i^H \sum_j^W (1 - S_{i,j}) * F'_{i,j} \tag{7}$$

It is worth noting that these prototypes are not separated like priors and CPG algorithm utilizes a weighted map to generate a pair of complementary prototypes. In this way, we retain the correlation between the prototypes. The whole CPG is delineated in Algorithm 1.

Algorithm 1 Complementary Prototypes Generation (CPG).

Input: targeted feature F', corresponding mask M, the number of iteration T.

init prototype $P^0_{c,0}$ by masked average pooling with F'. $P_0 = \frac{\sum_i^H \sum_j^W F'_{i,j}}{\sum_i^H \sum_j^W M_{i,j}}$

for iteration t in {1, .., T} **do**

Compute association matrix S between targeted feature F' and prototype P_0^{t-1}, $S^t = \text{cosine}(F', P_{c,0}^{t-1})$

Standardize association S^t,
$S^t_{i,j} = S^t_{i,j} / (\sum_i^H \sum_j^W S^t_{i,j})$

Update prototype $P_{c,0}$,
$P_0^t = \sum_i^H \sum_j^W S^t_{i,j} * F'_{i,j}$

end for

generate complementary prototypes P_c from S^T,
$P_1 = \sum_i^H \sum_j^W (S^T_{i,j}) * F'_{i,j}$
$P_2 = \sum_i^H \sum_j^W (1 - S^T_{i,j}) * F'_{i,j}$

return final prototypes P_1, P_2

3.4. Background Guided Learning

In previous works [1,5,6], the background information has not been adequately exploited for few-shot learning. Especially, these methods only use foreground prototypes to make a final prediction on the query image in the training. As a result, the representation on class-agnostic background is the lack of discriminability. To solve this problem, Background Guided Learning (BGL) is proposed via joint training strategy.

As shown in Figure 2, BGL is proposed to segment the background on the query image based on query background mask M_{bq}. As the first step, query feature F_q and its background mask M_{bq} are fed into the CPG module to generate a pair of complementary prototypes $P_{bq} = P_1, P_2$, following Algorithm 1. Next, we concatenate the complementary prototype P_{bq} with all spatial location in query feature map F_q, as Equation (8):

$$F_m = \epsilon(P_1) \oplus \epsilon(P_2) \oplus F_q, \tag{8}$$

where ϵ denotes the expansion operation and $\oplus$ denotes the concatenation operation, P_1 and P_2 are the complementary prototypes P_{bq} as well as F_m, denoting the concatenated feature. Then, concatenate feature F_m is fed into the decoder, generating the final prediction, as shown in Equation (9):

$$\hat{M} = \text{D}(F_m), \tag{9}$$

where $\hat{M}$ is the prediction of the model, D is a decoder. The loss $\mathcal{L}_{bg}$ is computed by:

$$\mathcal{L}_{bg} = \text{CE}(\hat{M}_{bq}, M_{bq}) \tag{10}$$

where $\hat{M}_{bq}$ denotes the background prediction on a query image and CE denotes the cross-entropy loss.

Intuitively, if the model can predict a good segmentation mask for the foreground using a prototype extracted from the foreground mask region, the prototype learned from the background mask region should be able to segment itself well. Thus, our BGL encourages the model to distinguish the background from the foreground better.

3.5. Inference

In the inference phase, we only keep the foreground segmentation branch for the final prediction. For K-shot setting, we following previous works and use the average to generate a pair of complementary prototypes.

4. Experiments

4.1. Dataset and Evaluation Metrics

4.1.1. Datasets

We evaluate our algorithm on two public few-shot datasets: PASCAL-5^i [22] and COCO-20^i [27]. PASCAL-5^i is built from PASCAL VOC 2012 and SBD datasets. COCO-20^i is built from MS-COCO dataset. In PASCAL-5^i, 20 object classes of PASCAL VOC are split into 4 groups, in which each group contains 5 categories. In COCO-20^i, as PASCAL-5^i, we divide MS-COCO into 4 groups, in which each group contains 20 categories. For PASCAL-5^i and COCO-20^i, we evaluate our approach based on PFENet. We use the same categories division and randomly sample 20,000 support-query pairs to evaluate as PFENet.

For both datasets, we adopt 4-fold cross-validation i.e., a training model on three folds (base class) and the inference model on the remaining one (novel class). The experimental results are reported on each test fold, and we also report the average performance of all four test folds.

4.1.2. Evaluation Metrics

Following previous work [7,27], we use the widely adopted class mean intersection over union (mIoU) as our major evaluation metric for the ablation study, since the class mIoU is more reasonable than the foreground-background IoU (FB-IoU), as stated in [7]. For each class, the IoU is calculated by $\frac{TP}{TP+FN+FP}$, where TP denotes the number of true positives, FP denotes the number of false postives and FN denotes the number of false negatives. Then, mIOU is the mean value of all classes IoU in the test set. For FB-IoU, only the foreground and background are considered ($C = 2$). We take the average of the results on all folds as the final mIoU/FB-IoU.

4.2. Implementation Details

Our approach is based on PFENet [1] with ResNet-50 as the backbone to create a fair comparison with the other methods. Following previous work [1,5,6], the parameters of the backbone are initialized with the pre-trained ImageNet, and is kept fixed during training. Other layers are initialized by the default setting of PyTorch. For PASCAL-5^i, the network is trained with an initial learning rate of 2.5×10^{-3}, weight decay of 1×10^{-4}, and a momentum of 0.9 for only 100 epochs. The batch size is 4. For COCO-20^i, the network is trained for 50 epochs with a learning rate of 0.005 and batch size of 8. We use data augmentation during training. Specifically, input images are transformed with random scale, horizontally flipped and rotated from $[-10, 10]$, and then all images are cropped to 473×473 (for PASCAL and COCO) or 641×641 (for COCO) as the training samples, for fair comparison. We implemented our model with 4 RTX2080Ti.

4.3. Comparisons with State-of-the-Art

4.3.1. COCO-20^i Result

COCO-20^i is a very challenging dataset that contains the numbers of objects in realistic scene images. We compare our approach with others on this dataset, and our approach outperforms other approaches by a big margin, as shown in Table 1. It can be seen that our approach achieves state-of-the-art performance on both 1-shot and 5-shot settings with mIOU gain of 0.3% and 0.5%, respectively. Furthermore, compared to our baseline (PFENet with ResNet101), our approach (with ResNet101) obtains 9.1% and 12.6% mIoU increases for 1-shot and 5-shot settings. In Table 2, our method obtains a top-performing 1-shot result and competitive 5-shot result with respect to FB-IoU. Once again, these results demonstrate that the proposed method is able to deal with more complex cases, since MSCOCO is a much more challenging dataset with diverse samples and categories.

Table 1. Comparison with other state-of-the-art methods on COCO-20^i for 1-shot and 5-shot settings. † denotes the model using size 641 × 641 as the training samples. All methods are tested on the original size. **Bold** denotes the best performance and red denotes the second best performance.

Method	Backbone	1-Shot					5-Shot				
		Fold-1	Fold-2	Fold-3	Fold-4	Mean	Fold-1	Fold-2	Fold-3	Fold-4	Mean
PFENet (TPAMI'20)	ResNet101	34.3	33.0	32.3	30.1	32.4	38.5	38.6	38.2	34.3	37.4
SCL (CVPR'21)	ResNet101	36.4	38.6	37.5	35.4	37.0	38.9	40.5	41.5	38.7	39.9
RePRI (CVPR'21)	ResNet101	36.8	41.8	38.7	36.7	38.5	40.4	46.8	43.2	40.5	42.7
FWB (ICCV'19)	ResNet101	17.0	18.0	21.0	28.9	21.2	19.1	21.5	23.9	30.1	23.7
CWT (ICCV'21)	ResNet101	30.3	36.6	30.5	32.2	32.4	38.5	46.7	39.4	43.2	42.0
HSNet (ICCV'21)	ResNet101	37.2	**44.1**	**42.4**	**41.3**	41.2	45.9	**53**	**51.8**	47.1	49.5
SCNet (2021)	ResNet101	38.3	43.1	40.0	39.1	40.1	44.0	47.7	45.0	42.8	44.8
MLC (ICCV'21)	ResNet101	**50.2**	37.8	27.1	30.4	36.4	**57.0**	46.2	37.3	37.2	44.4
SST (IJCAI'20)	ResNet50	-	-	-	-	22.2	-	-	-	-	31.3
DAN (ECCV'20)	ResNet50	-	-	-	-	24.4	-	-	-	-	29.6
PPNet (ECCV'20)	ResNet50	34.5	25.4	24.3	18.6	25.7	48.3	30.9	35.7	30.2	36.2
RPMMs (ECCV'20)	ResNet50	29.5	36.8	28.9	27.0	30.6	33.8	42.0	33.0	33.3	35.5
ASR (CVPR'21)	ResNet50	29.9	35.0	31.9	33.5	32.6	31.3	37.9	33.5	35.2	34.4
ASGNet † (CVPR'21)	ResNet50	-	-	-	-	34.6	-	-	-	-	42.5
CWT (ICCV'21)	ResNet50	32.2	36.0	31.6	31.6	32.9	40.1	43.8	39.0	42.4	41.3
Ours †	ResNet50	37.1	42.8	39.4	37.7	39.3	41.9	49.0	46.3	44.0	45.3
Ours	ResNet101	40.6	**44.1**	40.6	40.2	**41.5**	49.0	52.9	50.5	**47.7**	**50.0**

Table 2. Comparison of FB-IoU on COCO-20^i.

Methods	Backbone	1-Shot	5-Shot
PFENet (TPAMI'20)	ResNet101	58.6	61.9
DAN (ECCV'20)	ResNet101	62.3	63.9
Ours	ResNet101	64.0	68.8

4.3.2. PASCAL-5^i Result

In Table 3, we compare our method with other state-of-the-art methods on PASCAL-5^i. It can be seen that our method achieves on par state-of-the-art performance on 1-shot setting and 5-shot setting. Additionally, our method significantly improves the performance of PFENet on 1-shot and 5-shot segmentation settings, with an mIOU increase of 1.6% and 4%, respectively. In Table 4, our method obtains competitive 1-shot results and top-performing 5-shot results with respect to FB-IoU. In Figure 4, we report some qualitative results generated by our approach with PFENet [1] as the baseline. Our method is capable of making correct predictions and each part of our method could independently improve the performance of the model.

Table 3. Comparison with state-of-the-art methods on PASCAL-5^i for 1-shot and 5-shot settings. For fair comparison, all methods are evaluated with backbone ResNet50 and tested on labels with original sizes. **Bold** denotes the best performance and red denotes the second best performance.

Method	1-Shot					5-Shot				
	Fold-1	Fold-2	Fold-3	Fold-4	Mean	Fold-1	Fold-2	Fold-3	Fold-4	Mean
PGNet (ICCV'19)	56.0	66.9	50.6	50.4	56.0	57.7	68.7	52.9	54.6	58.5
CANet (CVPR'19)	52.5	65.9	51.3	51.9	55.4	55.5	67.8	51.9	53.2	57.1
CRNet (CVPR'20)	-	-	-	-	55.7	-	-	-	-	58.8
SimPropNet (IJCAI'20)	54.9	67.3	54.5	52.0	57.2	57.2	68.5	58.4	56.1	60.0
DAN (ECCV'20)	-	-	-	-	57.1	-	-	-	-	59.5
PPNet (ECCV'20)	47.8	58.8	53.8	45.6	51.5	58.4	67.8	64.9	56.7	62.0

Table 3. *Cont.*

Method	1-Shot					5-Shot				
	Fold-1	**Fold-2**	**Fold-3**	**Fold-4**	**Mean**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Mean**
RPMMs (ECCV'20)	55.2	66.9	52.6	50.7	56.3	56.3	67.3	54.5	51.0	57.3
PFENet (TPAMI'20)	61.7	69.5	55.4	56.3	60.7	63.1	70.7	55.8	57.9	61.9
ASR (CVPR'21)	53.8	69.6	51.6	52.8	56.9	56.2	70.6	53.9	53.4	58.5
ASGNet (CVPR'21)	58.8	67.9	56.8	53.8	59.3	63.7	70.6	64.2	57.4	63.9
SCL (CVPR'21)	63.0	70.0	56.5	57.7	61.8	64.5	70.9	57.3	58.7	62.9
RePRI (CVPR'21)	59.8	68.3	62.1	48.5	59.7	64.6	71.4	71.7	59.3	66.6
CWT (ICCV'21)	56.3	62.0	59.9	47.2	56.4	61.3	68.5	**68.5**	56.6	63.7
MLC (ICCV'21)	59.2	**71.2**	**65.6**	52.5	62.1	63.5	71.6	71.2	58.1	66.1
HSNet (ICCV'21)	**64.3**	70.7	60.3	**60.5**	**64.0**	**70.3**	**73.2**	67.4	**67.1**	**69.5**
Ours	63.6	70.2	57.1	58.2	62.3	67.7	72.3	59.3	64.1	65.9

Table 4. Comparison of FB-IoU on PASCAL-5^i for 1-shot and 5-shot settings. We used ResNet50 as the backbone.

Methods	1-Shot	5-Shot
PFENet(TPAMI'20)	73.3	73.9
PANet (ICCV'19)	66.5	70.7
CANet (CVPR'19)	66.2	69.6
PGNet (ICCV'19)	69.9	70.5
CRNet (CVPR'20)	66.8	71.5
PPNet (ECCV'20)	69.2	75.8
DAN (ECCV'20)	71.9	72.3
SCL (CVPR'21)	71.9	72.8
ASGNet (CVPR'21)	69.2	74.2
ASR (ICCV'21)	71.3	72.5
Ours	72.5	76.0

Figure 4. Qualitative examples of 5-shot segmentation on the PASCAL-5^i. (**a**) The ground-truth of the query images. (**b**) Results of baseline (PFENet) . (**c**) Results of BGL. (**d**) Results of CPG. (**e**) Results of the combination of BGL and CPG. Best viewed in color and zoomed in.

4.4. Ablation Study

To verify the effectiveness of out proposed methods, we conduct extensive ablation studies with a ResNet-50 backbone on PASCAL-5^i.

4.4.1. The Effectiveness of CPG

To verify the effectiveness of CPG, we conduct several experiments on prototype generation and compare it with other prototype generation algorithms. As a kind of soft cluster algorithm, we first compare our method with Adaptive K-means Algorithm (AK) provided by ASGNet [5], and a traditional algorithm, Expectation-Maximization Algorithm (EM), as shown in Table 5. Compared to the baseline, both AK and EM degenerate the performance of segmentation in a 1-shot setting while our CPG offers 0.6% improvement on the baseline. Compared to SCL [6] which needs to segment both support images and query images, our approach uses less computation cost and inference times (in Table 6) with competitive results on both 1-shot and 5-shot settings. These indicated the superiority of CPG on the few-shot segmentation task.

Table 5. Ablation study on prototype generation in a 1-shot setting on PASCAL-5^i.

Methods	Fold-1	Fold-2	Fold-3	Fold-4	Mean
baseline	61.7	69.5	55.4	56.3	60.8
AK [5]	60.5	68	55	54.2	59.4
EM	56.9	67.7	54.2	53.6	58.1
CPG	62.9	69.6	56.8	56.4	61.4

Table 6. Ablation study on the effectiveness of different components, evaluated on PASCAL-5^i. We report the mIoU and Frames (number of episodes) per second (FPS) for 1-shot and 5-shot. CPG: Complementary Prototypes Generation. BGL: Background Guided Learning.

CPG	BGL	1-Shot	FPS	5-Shot	FPS
-	-	60.7	50	61.9	12.5
√	-	61.4	50	63.6	11.11
-	√	62.1	50	65.1	12.5
√	√	62.3	50	65.9	11.11

4.4.2. The Effectiveness of BGL

To demonstrate the effectiveness of our proposed BGL, we conduct both qualitative and quantitative analysis on BGL. We assume the BGL has two sides of effectiveness on feature representation. The first one is the enhancement of feature representation for the novel classes and the second one is discrimination between the class-specific (foreground) feature and the class-agnostic (background) feature. Following [28], we measure the inter-class variance, intra-class variance, and discriminative function ϕ. Here ϕ is defined as inter-class variance divided by the intra-class variance.

As shown in Figure 5a,b,d, BGL not only enlarges the inter-class variance for novel classes but also increases intra-class variance for novel classes. In other words, BGL does not improve the representation discriminability for novel classes. However, as shown in Figure 5c,e, BGL enlarges the inter-class distance and increases the discriminative function ϕ between the foreground and the background. Therefore, the effectiveness of BGL is in the promotion of discrimination between the foreground and background.

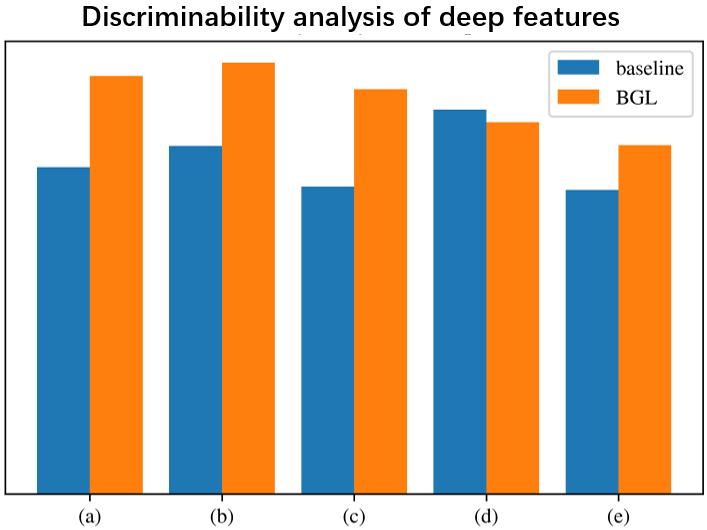

Figure 5. Discriminability analysis. (**a**) intra-class variance on novel classes. (**b**) Inter-class variance on novel classes. (**c**) Inter-class variance on the foreground/background. (**d**) Discriminative function ϕ on the novel class. (**e**) Discriminative function ϕ on the foreground/background.

4.4.3. The Effectiveness of BGL and CPG

To demonstrate the effectiveness of both CPG and BGL, ablation studies are conducted on PASCAL-5^i, as shown in Table 6. Compared with the baseline, using CPG and BGL alone improves the performance by a large margin, 1.7% and 2.6% for mIoU on 5-shot setting, respectively. In addition, we show that using CPG alone could achieve the current SOTA performance provided by SCL [6], and using BGL could surpass thestate-of-the-art performance with a 2.2% mIoU score. Then, combining both CPG and BGL achieves higher performance than the aforementioned one, with 4% improvement in total. In Figure 4, we show that using CPG and BGL alone may generate wrong segmentations on the background, but a combination of them could improve the results. In Figure 6, we show some representative heatmap examples, which further shows how the combination of CPG and BGL helps the model segment precisely and accurately.

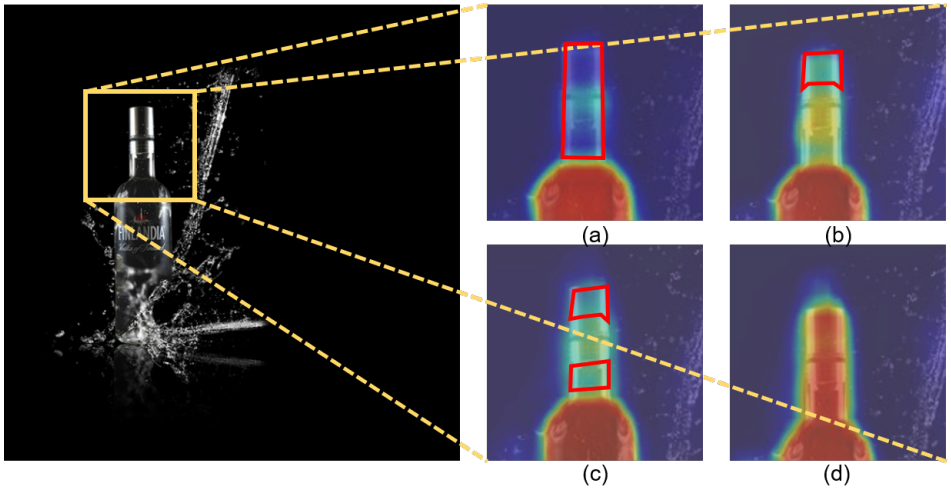

Figure 6. Heatmap examples on PASCAL-5^i in a 5-shot setting. (**a**) Result of baseline. (**b**) Result of CPG . (**c**) Result of BGL. (**d**) Result of the combination of BGL and CPG.

5. Conclusions

In this paper, we propose a novel few-shot semantic segmentation method named DCNet, which is composed of CPG and BGL. Our approach is able to extract comprehensive

support information through our proposed CPG module and generate discriminative feature representation for background pixels by BGL. Extensive experiments demonstrate the effectiveness of our proposed method.

Author Contributions: Main contribution: Q.R.; supervision: J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Nature Science Foundation of China (No.61972217, No. 62081360152), Natural Science Foundation of Guangdong Province in China (No. 2019B15 15120049, 2020B1111340056).

Institutional Review Board Statement: Ethical review and approval were waived for this study, due to no humans or animals were involved.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

References

1. Tian, Z.; Zhao, H.; Shu, M.; Yang, Z.; Li, R.; Jia, J. Prior Guided Feature Enrichment Network for Few-Shot Segmentation. *arXiv* **2020**, arXiv:2008.01449.
2. Liu, Y.; Zhang, X.; Zhang, S.; He, X. Part-Aware Prototype Network for Few-Shot Semantic Segmentation. *arXiv* **2020**, arXiv:2007.06309.
3. Wang, K.; Liew, J.H.; Zou, Y.; Zhou, D.; Feng, J. PANet: Few-Shot Image Semantic Segmentation With Prototype Alignment. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 9196–9205.
4. Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P.H.; Hospedales, T.M. Learning to Compare: Relation Network for Few-Shot Learning. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1199–1208.
5. Li, G.; Jampani, V.; Sevilla-Lara, L.; Sun, D.; Kim, J.; Kim, J. Adaptive Prototype Learning and Allocation for Few-Shot Segmentation. *arXiv* **2021**, arXiv:2104.01893.
6. Zhang, B.; Xiao, J.; Qin, T. Self-Guided and Cross-Guided Learning for Few-Shot Segmentation. *arXiv* **2021**, arXiv:2103.16129.
7. Zhang, C.; Lin, G.; Liu, F.; Yao, R.; Shen, C. CANet: Class-Agnostic Segmentation Networks With Iterative Refinement and Attentive Few-Shot Learning. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5212–5221.
8. Yang, B.; Liu, C.; Li, B.; Jiao, J.; Ye, Q. Prototype mixture models for few-shot semantic segmentation. In *Computer Vision—ECCV 2020*, Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2020; Volume 12353, pp. 763–778.
9. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; p. 10.
10. Yang, M.; Yu, K.; Zhang, C.; Li, Z.; Yang, K. DenseASPP for Semantic Segmentation in Street Scenes. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3684–3692.
11. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *arXiv* **2018**, arXiv:1802.02611.
12. Zhao, H.; Zhang, Y.; Liu, S.; Shi, J.; Loy, C.C.; Lin, D.; Jia, J. PSANet: Point-Wise Spatial Attention Network for Scene Parsing. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 267–283.
13. Zhang, H.; Dana, K.; Shi, J.; Zhang, Z.; Wang, X.; Tyagi, A.; Agrawal, A. Context Encoding for Semantic Segmentation. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7151–7160.
14. Zhang, H.; Zhang, H.; Wang, C.; Xie, J. Co-Occurrent Features in Semantic Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 548–557.
15. Yuan, Y.; Huang, L.; Guo, J.; Zhang, C.; Chen, X.; Wang, J. OCNet: Object Context Network for Scene Parsing. *arXiv* **2021**, arXiv:1809.00916.
16. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual Attention Network for Scene Segmentation. *arXiv* **2019**, arXiv:1809.02983.
17. Huang, Z.; Wang, X.; Huang, L.; Huang, C.; Wei, Y.; Liu, W. CCNet: Criss-Cross Attention for Semantic Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 603–612.
18. Chen, W.Y.; Liu, Y.C.; Kira, Z.; Wang, Y.C.F.; Huang, J.B. A Closer Look at Few-Shot Classification. *arXiv* **2020**, arXiv:1904.04232.
19. Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; Wierstra, D. Matching Networks for One Shot Learning. *arXiv* **2017**, arXiv:1606.04080.
20. Snell, J.; Swersky, K.; Zemel, R.S. Prototypical Networks for Few-Shot Learning. *arXiv* **2017**, arXiv:1703.05175.

21. Finn, C.; Abbeel, P.; Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. *arXiv* **2017**, arXiv:1703.03400.
22. Shaban, A.; Bansal, S.; Liu, Z.; Essa, I.; Boots, B. One-Shot Learning for Semantic Segmentation. *arXiv* **2017**, arXiv:1709.03410.
23. Liu, W.; Zhang, C.; Lin, G.; Liu, F. CRNet: Cross-Reference Networks for Few-Shot Segmentation. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 4164–4172. [CrossRef]
24. Zhang, X.; Wei, Y.; Yang, Y.; Huang, T. SG-One: Similarity Guidance Network for One-Shot Semantic Segmentation. *arXiv* **2020**, arXiv:1810.09091.
25. Yang, L.; Zhuo, W.; Qi, L.; Shi, Y.; Gao, Y. Mining Latent Classes for Few-Shot Segmentation. *arXiv* **2021**, arXiv:2103.15402.
26. Chen, J.; Gao, B.B.; Lu, Z.; Xue, J.H.; Wang, C.; Liao, Q. SCNet: Enhancing Few-Shot Semantic Segmentation by Self-Contrastive Background Prototypes. *arXiv* **2021**, arXiv:2104.09216.
27. Nguyen, K.; Todorovic, S. Feature Weighting and Boosting for Few-Shot Segmentation. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 622–631. [CrossRef]
28. Liu, B.; Cao, Y.; Lin, Y.; Li, Q.; Zhang, Z.; Long, M.; Hu, H. Negative Margin Matters: Understanding Margin in Few-Shot Classification. In *Computer Vision—ECCV 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M., Eds.; Springer International Publishing: Cham, Switzerland, 2020; Volume 12349, pp. 438–455. [CrossRef]