electronics

MDPI

Article

PCNet: Leveraging Prototype Complementarity to Improve Prototype Affinity for Few-Shot Segmentation

Jing-Yu Wang, Shang-Kun Liu, Shi-Cheng Guo, Cheng-Yu Jiang and Wei-Min Zheng *

College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China; wjy@sdust.edu.cn (J.-Y.W.); liushangkun@sdust.edu.cn (S.-K.L.); guoshicheng@sdust.edu.cn (S.-C.G.); 202182060076@sdust.edu.cn (C.-Y.J.)
* Correspondence: zhengweimin@sdust.edu.cn

Abstract: With the advent of large-scale datasets, significant advancements have been made in image semantic segmentation. However, the annotation of these datasets necessitates substantial human and financial resources. Therefore, the focus of research has shifted towards few-shot semantic segmentation, which leverages a small number of labeled samples to effectively segment unknown categories. The current mainstream methods are to use the meta-learning framework to achieve model generalization, and the main challenges are as follows. (1) The trained model will be biased towards the seen class, so the model will misactivate the seen class when segmenting the unseen class, which makes it difficult to achieve the idealized class agnostic effect. (2) When the sample size is limited, there exists an intra-class gap between the provided support images and the query images, significantly impacting the model's generalization capability. To solve the above two problems, we propose a network with prototype complementarity characteristics (PCNet). Specifically, we first generate a self-support query prototype based on the query image. Through the self-distillation, the query prototype and the support prototype perform feature complementary learning, which effectively reduces the influence of the intra-class gap on the model generalization. A standard semantic segmentation model is introduced to segment the seen classes during the training process to achieve accurate irrelevant class shielding. After that, we use the rough prediction map to extract its background prototype and shield the background in the query image by the background prototype. In this way, we obtain more accurate fine-grained segmentation results. The proposed method exhibits superiority in extensive experiments conducted on the PASCAL-5^i and COCO-20^i datasets. We achieve new state-of-the-art results in the few-shot semantic segmentation task, with an mIoU of 71.27% and 51.71% in the 5-shot setting, respectively. Comprehensive ablation experiments and visualization studies show that the proposed method has a significant effect on small-sample semantic segmentation.

Keywords: few-shot semantic segmentation; few-shot learning; self-distillation; self-support

Citation: Wang, J.-Y.; Liu, S.-K.; Guo, S.-C.; Jiang, C.-Y.; Zheng, W.-M. PCNet: Leveraging Prototype Complementarity to Improve Prototype Affinity for Few-Shot Segmentation. *Electronics* **2024**, *13*, 142. https://doi.org/10.3390/electronics13010142

Academic Editor: Byung-Gyu Kim

Received: 22 November 2023
Revised: 23 December 2023
Accepted: 27 December 2023
Published: 28 December 2023

1. Introduction

As computer technology rapidly advances, various artificial intelligence technologies [1,2] gradually influence human life. Semantic segmentation is an image processing method in the field of computer vision. The goal of this method is to classify each pixel in the image and determine the category of each point, so as to divide the region of the whole image. Unlike object detection, semantic segmentation involves not only identifying objects in an image but also classifying each pixel. Semantic segmentation provides specific category meanings for each part of an image, leading to a better understanding of its semantic information. The semantic segmentation can be applied to autonomous driving, medical image analysis, intelligent video surveillance and many other fields [3–5]. Traditional semantic segmentation [6] methods include region-based semantic segmentation [7], fully convolutional semantic segmentation [8] and weakly supervised semantic segmentation [9,10]. Semantic segmentation models require

a large number of finely labeled data, and it is difficult to quickly segment new classes that do not satisfy this condition. Due to the insufficient amount of labeled data, the training data cannot cover the complete features of the category, and the accuracy of semantic segmentation will gradually decrease. Therefore, how to reduce the use of resources and accurately segment the image has become an important research domain in the field of semantic segmentation. The few-shot semantic segmentation (FSS) [11] is used to solve the disadvantage of traditional methods in this paper.

The meta-learning [12–15] and the metric-learning [16–18] are the two most important paradigms in FSS. Meta-learning necessitates the model to posses the capability of "learning to learn". The metric-learning transforms a classifier guided by parameters into a classifier guided by distance. A model with a strong distance mapping ability can serve as a viable solution to the challenges of FSS. The current FSS methods combine meta-learning and metric-learning to construct reasonable segmentation models, as shown in Figure 1. The support image and query image sample pairs are input into the backbone network, the respective feature maps are generated and the query image is segmented under the guidance of the support mask. These models [19–22] enhance their performance by accumulating experience through multiple learning phases and leveraging this experience to classify the test set through self-learning on the training set. Although these models combining the two paradigms can segment the image, they have low accuracy and slow segmentation efficiency in the segmentation effect. Stable and accurate segmentation of images is very important in the fields of intelligent medical treatment, automatic driving, national defense and so on. A small deviation can lead to unpredictable consequences. Therefore, how to improve the accuracy and speed of model segmentation is the important and difficult problem of current research. To enhance the accuracy of our model, we employ self-adaptation and self-support strategies.

The self-adaptation strategy effectively addresses the knowledge transfer barrier that arises from the substantial feature difference between the support and query prototype. The self-support strategy can significantly improve the prototype quality and enhances the segmentation effect of the model by alleviating the intra-class appearance differences. The self-adaptation strategy employs knowledge distillation to transfer the information held within the support and query prototype. The self-adaptation strategy combines the support and query prototype to create a teacher prototype. With the help of this teacher model, the feature alignment of the two prototypes can be obtained. Utilizing the self-adaptation strategy can enhance the precision of segmentation. The self-support strategy first predicts the query feature image under the guidance of the adaptive self-distillation prototype. Firstly, the rough background mask of the query image is acquired. Subsequently, the self-support strategy employs mask average pooling between the background mask and the query feature image to generate a self-support background prototype. Finally, the self-support strategy generates the final prediction by combining the adaptive self-distillation prototype and the self-support background prototype. The self-adaptation and self-support strategies facilitate the segmentation model in attaining a remarkable level of accuracy. The primary contributions are as follows:

- We propose an adaptive self-distillation module (ASD) to solve the intra-class gap problem in the FSS task. The self-distillation method makes the support prototype and the query prototype supplement each other, and the base learner is introduced to suppress the base class in the query image.
- We propose a self-support background prototype module (SBP). The SBP is used to learn the feature comparison between the irrelevant class prototype and the query feature, which alleviates the adverse impact of the background features on the teacher prototypes generated by the adaptive self-distillation module.
- Combining an adaptive self-distillation module and a self-support background prototype module, we propose Prototype Complementarity Network (PCNet), which achieves new state-of-the-art results on the PASCAL-5^i and COCO-20^i.

The structure of the rest of this paper is as follows. The related work is included in Section 2. The problem setting, motivation and specific implementation details of the model are in Section 3. The performance analysis, comparison results and discussion of PCNet are presented in Section 4. The conclusion is given in Section 5.

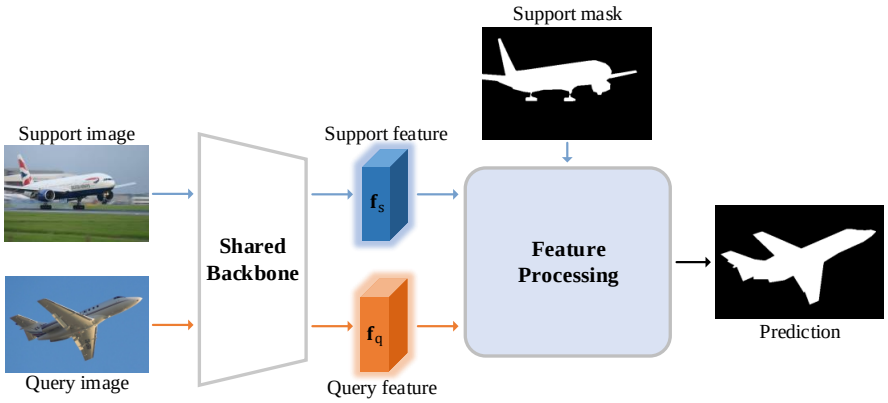


Figure 1. The framework of common few-shot learning methods. The support image and query image sample pairs are input into the backbone network, and the respective feature maps are generated. In feature processing, support feature performs a MAP operation with a support mask to generate the support prototype. The query feature will segment the query image and generate the prediction map under the guidance of the support prototype.

2. Related Work

2.1. Semantic Segmentation

Semantic segmentation is a core computer vision task that involves assigning each pixel in an image to a specific category. Recently, significant advancements have been made in this domain, particularly with the introduction of the fully convolutional network (FCN) [8]. The FCN revolutionized semantic segmentation by substituting the traditional fully connected layer with a convolutional layer, enabling end-to-end semantic segmentation. UNet [23] introduced an encoder–decoder structure to help reconstruct and refine the segmentation step by step. PSPNet [24] integrates the pyramid pooling module (PPM) into multiple baseline architectures, such as ResNet [25], and obtains context information at different scales by using pooling layers with different convolution kernel sizes. This method has also been widely used in few-shot segmentation. DeepLab [26] develops uniform atrous spatial pyramid pooling (ASPP) with filters in different dilation rates. In addition, there are some methods dedicated to attention mechanism. DANet [27] incorporates a position attention module and a channel attention module to capture the relationships between positions and channels. It learns the dependencies between different spatial locations and channels within an image. PSANet [28] introduces a point-wise spatial attention mechanism to investigate the correlation information among pixels. This approach enables the model to effectively leverage the spatial context and improve the accuracy of semantic segmentation. CCNet [29] adopts a cross-attention module to obtain context information from the dependency of the whole image. SAM [30] is a recently proposed revolutionary work. It builds the largest dataset available, the training of the model is promptable and it can transfer zero-shot to new image distributions and image segmentation tasks, contributing significantly to the field of computer vision. However, traditional semantic segmentation methods often rely on a substantial amount of annotated data to achieve a desirable performance. Moreover, they can face challenges when confronted with unseen categories, as they may require fine-tuning or additional training to adapt to these new classes. This limitation can hinder their ability to generalize effectively across a wide range of categories.

2.2. Few-Shot Learning

Few-shot learning hopes to use few annotated samples to identify new objects, and the mainstream approach in this field is to use a meta-learning framework [29] to simulate few-shot scenarios by sampling a set of learning tasks from the underlying datasets. Based on the characteristics and approaches used, few-shot learning (FSL) can be further subdivided into three types: the first is a transfer-learning approach [31,32] in which, through a two-stage fine-tuning process, the features learned from the base class are adapted to the new category. The second is an optimization-based method [19,33–36] in which meta-learning of the optimization process from some samples quicklys update the model. The third is the metric-based method [37,38], which applies a Siamese network [39] to support-query pairs and learn an embedding space. Instances from the same class in this embedding space are closer to each other than instances from different classes. Contrast loss is a classical metric learning method that learns a discriminant metric through the Siamese network. Both Meta-learning LSTM [40] and Model-Agnostic [19] methods employ Recurrent Neural Networks (RNNs) to capture and retain prior information, enabling them to address few-shot learning problems. To leverage the benefits of both approaches, ProtoMAML [41] combines the advantageous aspects of metric learning and gradient-based meta-learning methods. Our approach bears close resemblance to few-shot learning methods that rely on metric learning. In these methods, a prototype network is trained to map input data into a metric space. To overcome the model bias towards the base class, we incorporate a technique where we predict the base class region in the query image, thereby achieving suppression of the base class.

2.3. Few-Shot Semantic Segmentation

Few-shot semantic segmentation refers to the task of performing pixel-level semantic segmentation for novel classes in a query image, using only a limited number of annotated support images. This task is considered as an application of few-shot learning, which deals with the challenge of learning novel classes from a small amount of labeled data. Shaban et al. [22] formally defined few-shot segmentation for the first time and proposed the two-branch architecture OSLSM [42], which generates binary masks for novel classes in a point-similar manner by using classifier weights that support branch generation to predict query branches. In PANet [20], prototypes are utilized to represent characteristic features that indicate the presence of foreground categories within an image. The method involves conducting pixel-level feature comparison and prediction between the support prototype and query feature by means of cosine similarity. This comparison enables the model to determine the similarity between the query image and the support prototype, facilitating accurate pixel-level semantic segmentation. CANet [43] uses convolutions instead of cosine similarity, which may not work well for complex pixel classification tasks. PFENet [44] leverages a multi-scale query feature to further enhance the representation capability of the target class in the query image. By incorporating information from multiple scales, PFENet improves the model's ability to capture and represent fine-grained details of the target class. PGNet [21] incorporates graph attention, which considers each position of the foreground feature in the support image as a distinct entity. It establishes pixel correspondence between the query and support feature, enabling effective information exchange between them. PMMs [45] associates different regions of an image with multiple prototypes with a prototype mixture model. BAM [46] introduces an additional base class learning branch, which suppresses false activation of the base classes by the model during few-shot segmentation. CWT [47] introduces a novel and straightforward transformer architecture that dynamically transfers the weights of the support set classifier to the query set during prediction. This process effectively minimizes the intra-class difference between the support and query images.

However, these methods primarily rely on the foreground features from the support image to guide the model in segmenting the query image, which is relatively weak for few-shot semantic segmentation. Specifically, there is a huge intra-class gap between the

foreground features of the supporting image and the foreground features of the query image, and the model does not fully utilize the global information of the query image, so it is difficult for the model to perform pixel-level matching of the target class. We summarize the previous FSS model methods and limitations, as shown in Table 1. In contrast, in our work, we utilize query images to generate query prototypes, combine query prototypes with support prototypes, generate teacher prototypes for rudeness prediction and generate background prototypes to assist in learning feature alignment.

Table 1. Methods, results and limitations of previous models. These models are tested on both PASCAL-5^i and COCO-20^i datasets, and we report the results on the PASCAL-5^i dataset.

Method	Model	mIoU(PASCAL-5^i)		Limitations
		1-Shot	5-Shot	
Siamese Neural Network-based method	OSLSM [42]	40.80	44.00	Double branch has more parameters, which is computationally expensive and prone to overfitting, and multiple samples are less efficient.
	PFENet [44]	60.80	61.90	
	CANet [43]	55.40	57.10	
	BAM [46]	64.41	68.76	
Prototype Learning-based method	PANet [20]	48.10	55.70	Some spatial information is lost, it is difficult to adapt to the appearance and shape of different images, and the effect of boundary segmentation is not good.
	PMMs [45]	56.30	57.30	
Attention Mechanism-based method	PGNet [21]	56.00	58.50	All input vectors participate in the training, which is computationally expensive and difficult to capture the position information of the image.
	CWT [47]	58.00	64.70	

3. Method

In this section, we provide an explanation of the motivation and framework overview behind our proposed method, as detailed in Section 3.1. We define the few-shot semantic segmentation task in Section 3.2. And we introduce the specific implementation details of the adaptive self-distillation prototype module (ASD) in Section 3.3. The implementation details of the self-supporting background prototype module (SBP) are presented in Section 3.4. Finally, the optimization and inference of the model PCNet are described in Section 3.5.

3.1. Problem Setting

The objective of few-shot segmentation is to accurately segment objects belonging to unseen classes using only a limited number of labeled samples as guidance. The datasets are divided into: training set D_{train} and test set D_{test}. In the standard approach for few-shot learning, the class is used as the basis for splitting the dataset, and we define the class in D_{train} as the base class C_{base} and the class in D_{test} as the new class C_{novel}. Note that the two sets are disjoint, that is, $(C_{base} \cap C_{novel} = \varnothing)$. The episodic training mechanism used in this paper is an effective approach for few-shot learning. In this setup, each set is composed of a support set S and a query set Q, both belonging to the same category. There are K samples in the support set S, denoted by $S = \left\{\left(I_s^1, M_s^1\right), \left(I_s^2, M_s^2\right), \cdots, \left(I_s^K, M_s^K\right)\right\}$. Each $\left(I_s^i, M_s^i\right)$ represents an image-label sample pair in S, called a K-shot, where I_s^i and M_s^i denote the support image and the corresponding ground truth of this image, respectively. The query set has one sample $\left(I_q, M_q\right)$, where I_q and M_q denote the query image and the corresponding ground truth of this image, respectively. The input to the model is a sample pair of query image I_q and support set S, denoted by $\left\{I_q, S\right\} = \left\{I_q, \left(I_s^1, M_s^1\right), \left(I_s^2, M_s^2\right), \cdots, \left(I_s^K, M_s^K\right)\right\}$. During training, the model is guided by the support set to make predictions on the query image, and the model is continuously optimized. The ground truth of the query image is not visible in the test process, which is used to evaluate the performance of the method. After the training is completed, the model can directly predict the segmentation of the new class C_{novel} in the test set D_{test}.

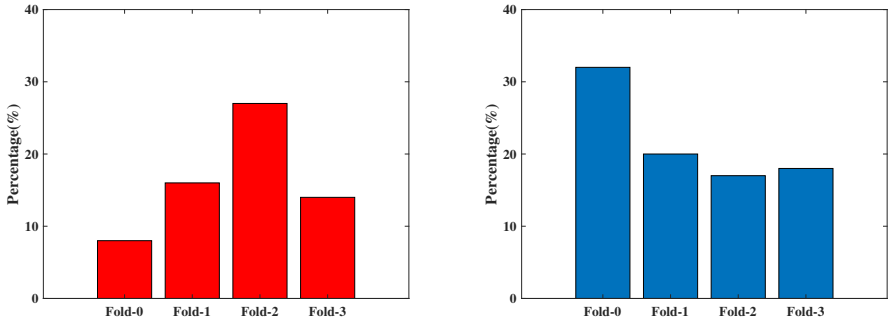
3.2. Motivation and Framework Overview

At present, the mainstream methods of FSS only rely on a single support prototype and perform dense matching on each pixel of the query image through the support prototype.

However, the samples provided by the support set and query set may have great appearance differences due to various realistic factors, such as shooting angles, lighting, etc., which makes the model have to face the problem of intra-class gap. Even though the objects in the support and query image belong to the same class, they may look very different, and the features learned by the model can hardly generate representative prototypes. This problem has a huge impact on the model's ability to generalize. In addition, Qi Fan et al. [48] calculated the cosine similarity for cross/intra object pixels, in Table 2. Through analysis, it is found that the background pixels of the support image and the query image seem to be chaotic, but there are certain similar characteristics between them. This may have something to do with the fact that objects of the same category often appear in similar scenes, such as boats and rivers, and airplanes and skies, which often appear together. In Figure 2, Chen et al. [49] recounted that some new classes would appear in the training images in each fold. In other words, new classes and base classes would co-appear in the training images, so the model would have a certain bias toward the base class. Through the study of previous models, we found that previous methods have limitations, such as too many parameters, difficulty in adapting to shapes from different angles and a lack of the ability to capture image position information, as shown in Table 1. Aiming at the target features in the dataset, we draw the advantages of the previous methods and try to break through the limitations of the previous methods and propose PCNet, as shown in Figure 3.

Table 2. Cosine similarity for cross/intra object pixels.

FG Pixels		**BG Pixels**	
cross-object	intra-object	cross-object	intra-object
0.308	0.416	0.298	0.365



(**a**) (**b**)

Figure 2. Percentage of potential novel-class objects in each fold. It can be seen that there are a large number of novel-class objects in the training images. Therefore, the model will be biased towards base-class objects during the testing process. (**a**) PASCAL-5^i; (**b**) COCO-20^i.

Based on the above problems and observations, we proposes an adaptive self-distillation prototype. The knowledge distillation method introduced by Hinton et al. [50] inspired us to transfer the information contained in the support prototype and the query prototype to knowledge transfer. Therefore, we uses self-distillation to fuse the support prototype and the query prototype into a teacher prototype, so as to realize feature alignment. The adaptive self-distillation prototype can not only express the characteristics of the support prototype, but also be as close to the unique characteristics of the target class in the query image as possible. The base learner is introduced in the module to weaken the bias of the model to the base class. The adaptive ability of the model is improved to a great extent. Aiming at the observation that there is a certain similarity between the background pixels of the support image and query image, we proposes a self-support background prototype. Under the guidance of the adaptive self-distillation prototype, the model first makes a

prediction on the query feature map to obtain a rough background mask of the query image. We further perform mask average pooling of the background mask with the query feature map to generate a self-support background prototype. Finally, we generate the final prediction map by combining the adaptive self-distillation prototype and self-support background prototype to densely match the feature maps of the query image.

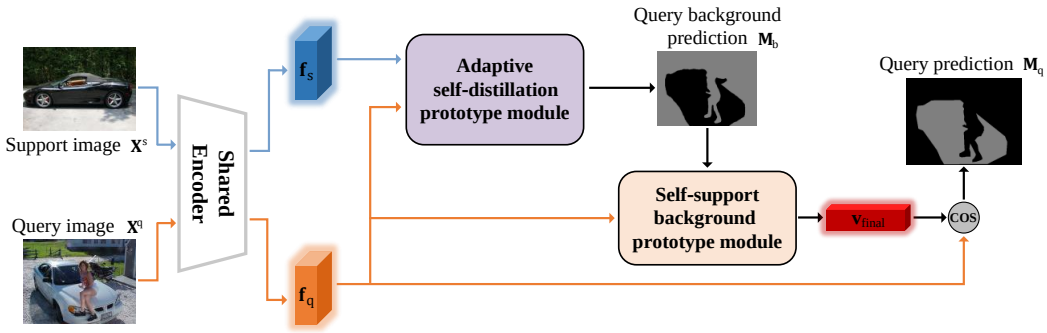


Figure 3. Architecture of PCNet. In PCNet, CNN is used as the backbone network to extract a feature map of the support image and query image. Firstly, the feature map of both are input into the adaptive self-distillation prototype module (ASD). The ASD module fuses the support prototype and the query prototype through the self-distillation process and uses the obtained teacher prototype to segment the query image. The background prediction map $\mathbf{M}_b$ to be enhanced is obtained. Then, $\mathbf{M}_b$ and the query feature map are input into the self-support background prototype module (SBP). The SBP module will use $\mathbf{M}_b$ to generate a background prototype and fuse the background prototype with the teacher prototype to generate a high-quality final prototype. The cosine distance between the final prototype and the query feature map is calculated to generate the final prediction map.

3.3. Adaptive Self-Distillation Prototype Module

Current mainstream prototype learning methods, including PFENet, outperform prior work by a large degree on the PASCAL-5^i and COCO-20^i datasets. The prototypes generated by these methods can effectively transfer the target features to the query branch and have a certain guiding effect. However, there can often be notable differences between the support target and the query target in few-shot segmentation. It is difficult to extend the feature prototypes provided by the support image to the complex and diverse query image. Therefore, we want to generate a kind of prototype, called a query prototype, for the query image during the training process, which carries the unique characteristics of the target class in the query image. In this way, the two prototypes obtained can adaptively express more complete feature information. To address the challenge of aligning the features of the support prototype and the query prototype, we propose a self-distillation method. This approach facilitates the transfer of knowledge between the two prototypes, resulting in the generation of an adaptive self-distillation prototype. By leveraging this knowledge transfer, the model can effectively align the features of the support and query prototypes. After generating the prediction map under the guidance of the adaptive self-distillation prototype, we further introduce the base learner to suppress the base class in the prediction map, which is proved to be effective in BAM [46].

As shown in Figure 4, we combine the query prototype with unique features and the support prototype with common features. They form a teacher prototype, and the loss is calculated between the teacher prototype and the support prototype. This allows continuous adjustment and optimization of the teacher prototype. Finally, we apply self-distillation to generate the self-support adaptive prototype. $\mathbf{V}_s$ denotes the support prototype. $\mathbf{V}_q$ denotes the query prototype. The equation is as follows:

$$\mathbf{V}_s = \mathcal{F}_p(\mathbf{f}_s, M_s), \mathbf{V}_q = \mathcal{F}_p(\mathbf{f}_q, M_q), \tag{1}$$

where $\mathbf{f}_s$ and $\mathbf{f}_q$ represent the support and query feature map obtained from the support and query image through the shared network CNN, respectively. M_s and M_q denote the

support mask and query mask, respectively. $\mathcal{F}_p$ represents the masked GAP operation. The mask M and the feature map $\mathbf{f}$ have the same height and width. Y_s and Y_q are the outputs of softmax operation on $\mathbf{V}_s$ and $\mathbf{V}_q$. On the basis of knowledge distillation, we use the Kullback Leibler divergence to calculate loss to adjust and optimize the support prototype. $\mathcal{L}_{KD}$ represents the loss of the support and teacher prototype in the knowledge distillation process:

$$Y_s = \text{Softmax}(\mathbf{V}_s), Y_q = \text{Softmax}(\mathbf{V}_q) \tag{2}$$

$$\mathcal{L}_{KD} = KL(\mathbf{V}_t \| Y_s) = \sum_{i=1}^{c} \mathbf{V}_t^i \log \frac{Y_s^i}{\mathbf{V}_t^i}, \tag{3}$$

where $\mathbf{V}_t$ denotes the teacher prototype and is equal to $\mathbf{V}_t = \frac{Y_s + Y_q}{2}$. KL$(\cdot)$ represents the KL divergence function.

The support and query prototype can be seen as consisting of two parts: common features and unique features, namely, $\mathbf{V}_s = (f_c, f_s)$, $\mathbf{V}_q = (f_s, f_c)$. In the knowledge distillation process, the goal is to improve the prototype's expression of common features while preserving the unique characteristics of a specific query target. This enhances the prototype's consistency and adaptability. Therefore, this method will lead to two prototypes close to the common features, that is, $\mathbf{V}_s(f_c, f_s) \rightarrow \mathbf{V}_s(f_c)$. The effectiveness of this module is demonstrated in our experiments.

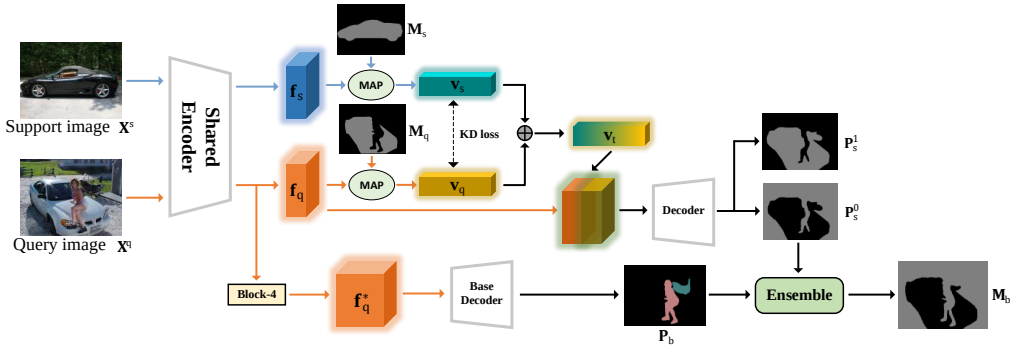


Figure 4. The ASD module receives the support feature MAP and query feature map generated by the encoder, generates the support prototype and the query prototype using the map operation, and uses the self-distillation method between the two prototypes. The generated teacher prototype not only retains the common features of the two prototypes but also integrates some unique features of the query prototype, which has a higher affinity with the query image. The rough prediction map is generated through the decoder under the guidance of the teacher prototype. The base class learner is a PSPNet trained separately. The Block-4 feature map generated by the backbone network is input into the base class learner, and the base class prediction map is fused with the rough background prediction map to achieve the masking effect on the base class.

To extend the method to the K-shot segmentation task, we make some modifications to the teacher prototype generation process. Since the K support images are different, the generated support prototypes will also be different. The teacher prototype needs to supervise each support prototype. The respective teacher prototype is then generated for each support prototype. $\mathcal{L}_{KD}^S$ is used as the loss between each support prototype and the teacher prototype. The equation is as follows:

$$\mathcal{L}_{KD}^S = \frac{1}{K} \sum_{i=1}^{K} KL(\mathbf{V}_t^i \| \mathbf{V}_s^i), \tag{4}$$

where $\mathbf{V}_s^i$ denotes the prototype of the i-th support sample, and $\mathbf{V}_t^i$ denotes the teacher prototype generated when the i-th support image is input. Then, we activate the target region in $\mathbf{f}_q$ under the guidance of $\mathbf{V}_t$ and generate a prediction result $\mathbf{P}_s$ through the decoder.

$$\mathbf{P}_s = \text{softmax}\left(\mathcal{D}_m\left(\mathcal{F}_{\text{guidance}}\left(\mathbf{V}_t, \mathbf{f}_q\right)\right)\right) \in \mathbb{R}^{2 \times h \times w}, \tag{5}$$

where $\mathcal{F}_{\text{guidance}}$ represents the process of archetypal guidance for query feature map segmentation, and $\mathcal{D}_m$ represents the decoder network for meta-learning. We also need to compute the loss for $\mathbf{P}_s$ and M_q to update the parameters:

$$\mathcal{L}_{\text{s}} = \frac{1}{e}\sum_{i=1}^{e} \text{BCE}\left(\mathbf{P}_{s_{,i}}, M_{q;i}^{1}\right), \tag{6}$$

where M_q^1 denotes the foreground mask of the query image, and e denotes the number of training samples in each batch.

In the training process, the model will be activated by the error of the seen class, that is, the base class bias. We introduce a base learner to realize the inhibition of the base class, and the learner uses PSPNet [24] to segment the base class target. $\mathbf{P}_b$ is the prediction mask of the base class by the PSPNet [24] network, which is expressed as:

$$\mathbf{P}_b = \text{softmax}\left(\mathcal{D}_b\left(\mathbf{f}_q^*\right)\right) \in \mathbb{R}^{(1+N_{\text{b}})\times h\times w}, \tag{7}$$

where $\mathbf{f}_q^*$ is the Block-4 query feature map generated by the encoder. $\mathcal{D}_b$ represents the decoder of the standard semantic segmentation. N_{b} denotes the number of base categories. The cross-entropy loss $\mathcal{L}_{\text{bs}}$ is used to update the optimization parameters, which is expressed as:

$$\mathcal{L}_{\text{bs}} = \frac{1}{n_{\text{bs}}}\sum_{i=1}^{n_{\text{bs}}} \text{CE}\left(\mathbf{P}_{b;i}, M_{q;i}^{\text{g}}\right), \tag{8}$$

where M_q^{g} is the ground truth of all base classes in the query image. n_{bs} represents the number of training samples in each batch.

We first combine all the class prediction maps to obtain the base class region prediction relative to the few-shot task, that is, the irrelevant class that is likely to be misactivated. Then, we aggregate the predicted base class region with the predicted background mask to obtain the full background prediction map:

$$\mathbf{P}_b^{\text{f}} = \sum_{i=1}^{N_{\text{b}}} \mathbf{P}_b^i \tag{9}$$

$$\mathbf{M}_b = \mathcal{F}_{\text{ensemble}}\left(\mathbf{P}_s^0, \mathbf{P}_b^{\text{f}}\right), \tag{10}$$

where $\mathbf{P}_s^0$ represents the background mask of the query image generated under the guidance of the adaptive self-distillation prototype. $\mathbf{P}_b^{\text{f}}$ represents the irrelevant class set mask output by the base learners. $\mathcal{F}_{\text{ensemble}}$ represents the aggregation function, which is a 1*1 convolution operation with specific initial parameters. $\mathbf{M}_b$ represents the prediction of the background mask to be augmented with respect to the target class of the query image. It is also the output of the whole adaptive self-distillation module. We need to update the optimization parameters by calculating the loss $\mathcal{L}_b$ between the predicted background map to be augmented and the groundtruth background mask:

$$\mathcal{L}_b = \frac{1}{e}\sum_{i=1}^{e} \text{BCE}\left(\mathbf{M}_{b,i}, M_{q;i}^{0}\right), \tag{11}$$

where M_q^0 is the background mask of groundtruth, and e represents the number of training samples in each batch.

3.4. Self-Support Background Prototype Module

Through the adaptive self-distillation module, we make the prototype used more suitable for the target class in the query image, and then we introduce the branch of the base learner to realize the inhibition of the base class target. In summary, we obtain a relatively accurate background mask prediction map of the target class. To enhance the fine-grained accuracy, we add the generation of the self-support background prototype module, as shown in Figure 5.

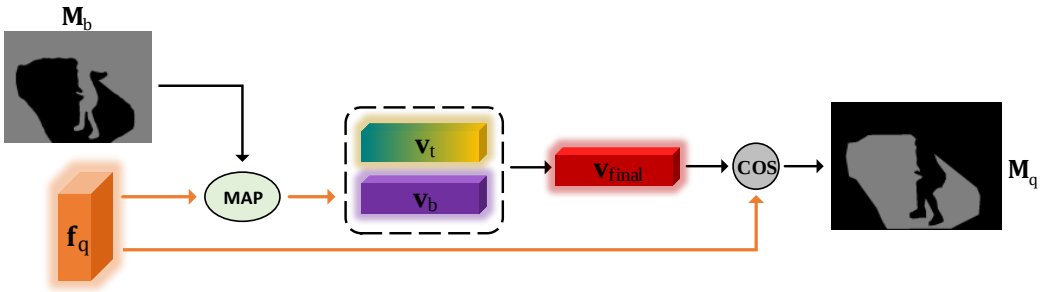


Figure 5. SBP receives the background prediction mask to be enhanced from ASD and uses the MAP operation and the query image to generate the self-support background prototype. In order to achieve fine-grained correction of background pixels, the teacher prototype and the self-support background prototype are fused to generate the final prototype, and then the cosine distance between the final prototype and the query feature map is calculated to generate the final prediction mask.

For foreground pixels, they possess similar semantic information and appearance, making the prototype learning method applicable. However, background pixels are disorganized, as the background may contain numerous categories. Their semantic similarity is limited to a local scope, and finding shared semantic similarity for the entire background becomes challenging. This may require us to generate multiple targeted background prototypes and perform pixel grouping matching for each query pixel. For the problem of generating prototypes from the background, the clustering algorithm has been mentioned many times in the previous work, and experiments have been carried out. Although the problem is solved to some extent, the clustering algorithm has the disadvantage of instability.

We use the background prediction map of the query image generated by the adaptive self-distillation module to generate a background prototype with the query image itself after the masked GAP operation, which is shown as follows.

$$\mathbf{V}_b = \mathcal{F}_p\left(\mathbf{f}_q, \mathbf{M}_b\right), \tag{12}$$

where $\mathbf{V}_b$ denotes the self-support background prototype. This way, we update the final prototype as: $\mathbf{V}_{final} = \{\mathbf{V}_b, \mathbf{V}_t\}$. Finally, we obtain the final prediction map $\mathbf{M}_q$ by computing the cosine distance between the final prototype $\mathbf{V}_{final}$ and the query image$\mathbf{f}_p$.

$$\mathbf{M}_q = \text{softmax}\Big(\text{cosine}\Big(\mathbf{V}_{final}, \mathbf{f}_q\Big)\Big) \tag{13}$$

We compute the loss function $\mathcal{L}_f$ on the cosine distance map generated during the final training phase.

$$\mathcal{L}_f = \text{BCE}\Big(\text{cosine}\Big(\mathbf{V}_{final}, \mathbf{f}_q\Big), M_q^1\Big), \tag{14}$$

where BCE is the binary cross entropy loss, and M_q^1 is the groundtruth of the query image.

3.5. Optimization

Based on the adaptive self-distillation module and the self-support background prototype module, we propose the PCNet model, as shown in Figure 3. For the whole model, we calculate the $\mathcal{L}_{KD}$ loss function of the support and teacher prototype in the self-distillation process. The query prediction mask $\mathbf{P}_s$ generated by the self-distillation process and the ground truth are supervised by the binary cross entropy loss function $\mathcal{L}_s$. The cross-entropy loss function $\mathcal{L}_{bs}$ is used for the base class prediction mask generated by the base learner. Since the base learner needs to be trained separately, the loss function of the base learner does not participate in the optimization process of the model. We also calculate the loss function $\mathcal{L}_b$ between the predicted background map to be augmented, generated by the adaptive self-distillation module and the background mask of the groundtruth, which is used to supervise the segmentation effect of this module. Finally, the query prediction

mask and the ground truth generated after the correction of the self-support background prototype module are trained and supervised by the Binary Cross Entropy loss function $\mathcal{L}_f$. Finally, we realize the end-to-end training process of the model through the joint optimization of all the above losses:

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{KD} + \mathcal{L}_b + \mathcal{L}_s + \beta \cdot \mathcal{L}_f, \tag{15}$$

where α and β are the coefficients of $\mathcal{L}_{KD}$ and $\mathcal{L}_f$, which are used to balance the supervision effects of the four losses. In the ablation experiment section, we discuss the values of the two coefficients.

4. Experiments

4.1. Setup

Datasets. To assess the performance of the proposed method, we conduct evaluations on two widely used datasets for few-shot segmentation tasks: PASCAL-5^i [22] and COCO-20^i [51]. Each image in the dataset corresponds to a mask labeled with only one type of object. PASCAL-5^i is composed of extended annotations from PASCAL VOC 2012 [52] and SDS [53] datasets. The dataset contains 20 object categories, and in order to make the training set and test set disjoint, we divide all categories into four folds with five categories each. Three of these folds are used as the training set, and the other fold is used as the test set. The objects of the test set are not visible during the training process. We set the test set separately for each fold, so the experimental results will contain mIoU of four folds. Following OSLSM [42], we randomly sample 1000 query support pairs in each test fold. Following [51], we evaluate the proposed method on the more challenging dataset COCO-20^i built by MSCOCO [54], with more samples and categories per image. The dataset contains 80 object categories; again, we split into 4 folds with 20 categories per fold. The experimental setup is the same as PASCAL-5^i. Since there are far more images in COCO-20^i than there are in PASCAL-5^i, in our tests, we scale the number of randomly sampled query-support pairs to 20000.

Evaluation metrics. Based on previous studies [44,45], we use the class mean intersection over union (mIoU) and the foreground-background IoU (FBIoU) as evaluation indicators. IoU is calculated by $\frac{\mathrm{TP}}{\mathrm{TP+FP+FN}}$, where TP represents true positive, FP represents false positive and FN represents false negative, and mIoU is the mean IoU of each category. For comparison with the previous method, we also report the results of FBIoU, and the average of the results is also the final FBIoU. It is important to mention that FBIoU treats all object classes as a single foreground class. This metric is biased towards and influenced by the background classes, as foreground classes typically occupy a small portion of the entire image. Therefore, mIoU becomes a crucial evaluation metric for FSS tasks. mIoU provides a more comprehensive assessment of the segmentation performance by considering the overlap between predicted and ground truth masks for all object classes individually.

Implementation Details. Our framework is built on PyTorch11.7. We choose a modified version of ResNet50 as the backbone network for a fair comparison with previous methods. The backbone network is initialized with ImageNet [55] pre-trained weights. Our model was trained on PASCAL-5^i for 200 rounds using an initial learning rate of 0.0025 and the training batch size set to 4. It is trained on COCO-20^i for 50 rounds using an initial learning rate of 0.0025 and training batch size set to 8, following the data augmentation technique in [44]. Similar to BAM [46], the base learner branch we introduced utilizes PSPNet [24]. It was trained for 100 epochs on PASCAL-5^i and 20 epochs on COCO-20^i. SGD optimizer was employed with an initial learning rate of 0.0025 to update the parameters. The batch size was set to 12 during training, and all parameters in the base learner were frozen after training completion.

4.2. Comparison with Previous Work

We extensively compare our method with the state-of-the-art approach using VGG16 and ResNet50 backbone networks in a few-shot setting to demonstrate its effectiveness.

PASCAL-5^i. As shown in Table 3, we report the mIoU of each fold and the average of the four folds in the PASCAL-5^i dataset. It can be seen that the quantitative results of the proposed model PCNet on the VGG16 [56] and ResNet50 [25] backbone networks are better than other methods. Compared with BAM [46], our method achieves a 1.4% and 0.43% mIoU improvement on 1-shot, and 2.34% and 0.36% mIoU improvement on 5-shot, respectively. Under the backbone network of ResNet50, PCNet reaches 68.24% and 71.27%, respectively, which are higher than the segmentation results under the backbone network of VGG16, which is related to the fact that ResNet50 can provide more useful information for segmentation, so that the two modules in this paper can more effectively play the expression ability of the prototype. The fold-0 of PCNet under the ResNet50 backbone network is 1.13% lower than the mIoU of BAM. We believe that the objects in fold-0 are images with a relatively single background, such as boat and airplane, while our model is sensitive to the background, so a small amount of deviation may be caused. In most cases, the performance of PCNet is still more outstanding. We also report FBIoU results on the PASCAL-5^i dataset, as shown in Table 4. Since FBIoU is biased towards the background and covers most foreground region classes, and our model predicts background pixels as well as foreground pixels, our method improves on previous methods. On the RseNet50 backbone network, our results reach 80.36% and 82.6%.

Table 3. Performance comparison on PASCAL-5^i in terms of mIoU. "Baseline" is the model that replaces the self-distillation process with the basic meta-learning framework and removes the SBP module. The results in **bold** indicate the optimal performance.

Backbone	Method	1-Shot					5-Shot				
		Fold-0	Fold-1	Fold-2	Fold-3	Mean	Fold-0	Fold-1	Fold-2	Fold-3	Mean
VGG16	OSLSM [42]	33.60	55.30	40.90	33.50	40.80	35.90	58.10	42.70	39.10	44.00
	PANet [20]	42.30	58.00	51.10	41.20	48.10	51.80	64.60	59.80	46.50	55.70
	FWB [51]	47.00	59.60	52.60	48.30	51.90	50.90	62.90	56.50	50.10	55.10
	PFENet [44]	56.90	68.20	54.40	52.40	58.00	59.00	69.10	54.80	52.90	59.00
	HSNet [57]	59.60	65.70	59.60	54.00	59.70	64.90	69.00	64.10	58.60	64.10
	APANet [49]	58.00	68.90	57.00	52.20	59.00	59.80	70.00	62.70	57.70	62.60
	BAM [46]	63.18	70.77	66.14	57.57	64.41	67.36	73.05	70.61	64.00	68.76
	Baseline	59.54	68.36	65.55	54.73	62.04	64.12	70.41	69.74	63.62	66.97
	PCNet (ours)	**64.93**	**72.21**	**66.81**	**59.29**	**65.81**	**69.97**	**74.68**	**72.01**	**67.75**	**71.10**
ResNet50	CANet [43]	52.50	65.90	51.30	51.90	55.40	55.50	67.80	51.90	53.20	57.10
	PGNet [21]	56.00	66.90	50.60	50.40	56.00	57.70	68.70	52.90	54.60	58.50
	RPMM [45]	55.20	66.90	52.60	50.70	56.30	56.30	67.30	54.50	51.00	57.30
	PFENet [44]	61.70	69.50	55.40	56.30	60.80	63.10	70.70	55.80	57.90	61.90
	HSNet [57]	64.30	70.70	60.30	60.50	64.00	70.30	73.20	67.40	67.10	69.50
	SAGNN [58]	64.70	69.60	57.00	57.20	62.10	64.90	70.00	57.90	59.30	62.80
	APANet [49]	62.20	70.50	61.10	58.10	63.00	63.30	72.00	68.40	60.20	66.00
	BAM [46]	**68.97**	73.59	67.55	61.13	67.81	70.59	75.05	70.79	67.20	70.91
	Baseline	64.52	71.88	66.01	58.31	65.18	67.55	72.92	70.23	66.74	69.36
	PCNet (ours)	67.84	**74.32**	**67.70**	**63.11**	**68.24**	**70.81**	**75.46**	**71.35**	**67.47**	**71.27**

Table 4. Averaged FBIoU over 4 folds on PASCAL-5^i. The results in **bold** indicate the optimal performance.

Backbone	Method	FBIoU (%)	
		1-Shot	5-Shot
ResNet50	ASGNet [12]	60.40	67.00
	PGNet [21]	69.90	70.50
	PPNet [59]	69.19	75.76
	PFENet [44]	73.30	73.90
	HSNet [57]	76.70	80.60
	BAM [46]	79.71	82.18
	PCNet (ours)	**80.36**	**82.60**

COCO-20^i. COCO-20^i is a very challenging dataset. It contains larger number of classes and more complex and diverse samples. We present the results in Table 5. When VGG16 is used as the backbone network, our method achieves higher or equivalent results, outperforming the state-of-the-art method BAM [46] by 0.74% and 1.52% in 1-shot and 5-shot settings, respectively. When utilizing ResNet50 as the backbone network, our method also demonstrates promising results, which are 0.43% and 0.36% higher than BAM. PCNet performs worse than SSP on fold-0. We look at the fold-0 contains, and, similar to PASCAL-5^i, a large number of single background images are included in fold-0. In addition, SSP performs a complex fusion operation on the background prototype and the foreground prototype, which proves the feasibility of the background prototype, but the mIoU on other folds is far lower than our results. The performance of PCNet is higher than the baseline model, which is 45.93% and 51.71% on the ResNet50 backbone network, respectively, which is related to the purification of target class features by the adaptive self-distillation module. The utilization of the background in the self-support background prototype module also benefits the model.

Table 5. Performance comparison on COCO-20^i in terms of mIoU. "Baseline" is the model that replaces the self-distillation process with the basic meta-learning framework and removes the SBP module. The results in **bold** indicate the optimal performance.

Backbone	Method	1-Shot					5-Shot				
		Fold-0	Fold-1	Fold-2	Fold-3	Mean	Fold-0	Fold-1	Fold-2	Fold-3	Mean
VGG16	FWB [51]	18.35	16.72	19.59	25.43	20.02	20.94	19.24	21.94	28.39	22.63
	PFENet [44]	35.40	38.10	36.80	34.70	36.30	38.20	42.50	41.80	38.90	40.40
	PRNet [60]	27.46	32.99	26.70	28.98	29.03	31.18	36.54	31.54	32.00	32.82
	SAGNN [58]	35.00	40.50	37.60	36.00	37.30	37.20	45.20	40.40	40.00	40.70
	PANet [20]	-	-	-	-	20.90	-	-	-	-	29.70
	APANet [49]	35.60	40.00	36.00	37.10	37.20	40.10	48.70	43.30	40.70	43.20
	BAM [46]	38.96	47.04	46.41	41.57	43.50	**47.02**	52.62	48.59	49.11	49.34
	Baseline	38.34	46.32	43.62	39.75	41.96	44.21	48.57	44.26	47.32	46.09
	PCNet(ours)	**39.11**	**49.16**	**46.90**	**42.51**	**44.24**	46.74	**54.19**	**52.72**	**49.79**	**50.86**
ResNet50	ASGNet [12]	-	-	-	-	34.50	-	-	-	-	42.40
	HSNet [57]	36.30	43.10	38.70	38.70	39.20	43.30	51.30	48.20	45.00	46.90
	PPNet [59]	34.50	25.40	24.30	18.60	25.70	48.30	30.90	35.70	30.20	36.20
	RPMM [45]	29.50	36.80	29.00	27.00	30.60	33.80	42.00	33.00	33.30	35.50
	APANet [49]	37.50	43.90	39.70	40.70	40.50	39.80	46.90	43.10	42.20	43.00
	CWT [47]	32.20	36.00	31.60	31.60	32.90	40.10	43.80	39.00	42.40	41.30
	SSP [48]	**46.40**	35.20	27.30	25.40	33.60	**53.80**	41.50	36.00	33.70	41.30
	BAM [46]	43.41	50.59	47.49	43.42	46.23	49.26	54.20	51.63	49.55	51.16
	Baseline	41.72	44.15	49.36	41.27	42.50	44.02	47.42	46.34	45.96	45.93
	PCNet(ours)	43.57	**52.08**	**47.71**	**44.13**	**46.79**	48.56	**54.73**	**52.61**	**50.93**	**51.71**

Qualitative results. To analyze and understand the performance of the proposed model more directly, as shown in Figure 6, we present partial results on the PASCAL-$\mathbf{5}^i$ dataset under the 5-shot setting. Firstly, in the first eight columns, the query images all contain base objects, indicating that the model has a clear suppression effect on base class objects and achieves more accurate segmentation of the targets. For example, in the seventh column, even with occlusion from base class objects, our model performs well in segmenting the car. But the baseline model misclassified a cow as a car. Secondly, there may be significant intra-class differences between the support images and query images. For instance, in the ninth column, the boat in the query image is in a highly complex environment, and the baseline model is greatly affected, while our model still demonstrates good generalization ability.

Figure 6. Example results for different classes on the PASCAL-$\mathbf{5}^i$ dataset. Each row from top to bottom represents the support images with ground truth masks, query images with ground truth masks, baseline results and our results, respectively.

4.3. Ablation Studies

To investigate the performance impact of each module component on the model, we conducted a series of ablation experiments, and the experiments in this section are conducted on the PASCAL-$\mathbf{5}^i$ dataset using the ResNet50 backbone network.

Ablation study of ASD and SBP. In order to analyze the effect and influence of ASD and SBP modules, we conduct an experiment about the performance of the model with and without the ASD and SBP. The experiment was conducted in a 1-shot setting. As shown in Table 6, the proposed SBP module improves the model's performance by 1.16% compared to the Baseline. This indicates that the self-support background prototype we introduced can extract similar semantic information from a cluttered background. It assists in the self-correct of prototypes generated during the meta-learning process, effectively enhancing the model's predictive ability. We propose that the ASD module improves the performance of the model by 2.31%, which is related to the unique characteristics of the query image and the support image target class in the teacher prototype, and effectively solves the problem of the intra-class gap. Finally, we introduce the two modules into Baseline at the same time, and the performance of the model is improved by 3.06% compared with Baseline. The performance is improved to 68.24%. Through ablation experiments on two specific modules, we provide further evidence of the effectiveness of the proposed method.

Table 6. Ablation study of our proposed SBP and ASD on PASCAL-5^i for 1-shot segmentation. "PCNet" means the model with both SBP and ASD. The results in **bold** indicate the optimal performance.

Methods	Fold-0	Fold-1	Fold-2	Fold-3	Mean
Baseline	64.52	71.88	66.01	58.31	65.18
Baseline + SBP	65.73	72.82	66.34	60.48	66.34
Baseline + ASD	66.94	73.45	66.93	62.65	67.49
PCNet	**67.84**	**74.32**	**67.70**	**63.11**	**68.24**

Ablation experiments about the values of α and β. In Optimization, we set the coefficients α and β for $\mathcal{L}_{KD}$ and $\mathcal{L}_f$, respectively, and explore the influence of the value relationship between the loss function coefficients α and β on the performance of the model. In order to keep all the loss functions on the same scale, we set a reasonable value range for α and β based on the experience of previous methods and performed ablation experiments for each value within this value range, $\alpha \in [\mathbf{30, 70}]$, $\beta \in [\mathbf{0.2, 1.0}]$. The result is shown in Figure 7. The experiment was performed in Fold-0 with a 5-shot setting. $\mathcal{L}_{KD}$ calculated the loss between the teacher prototype and the support prototype. We infer that when the value of α is too large, the teacher prototype may be very close to the teacher prototype, which is difficult in absorbing the unique characteristics of the query image, and the model is difficult to achieve the desired effect of the self-distillation process. When the value of α is too small, $\mathcal{L}_{KD}$ will lose the supervision effect on the teacher prototype implementation. $\mathcal{L}_f$ calculates the loss between the final prediction mask and the groundtruth. The background prototype in this process is generated by the MAP operation of the prediction mask to be enhanced and the query image of the ASD module. The self-support background prototype is not accurate, but it can help the model to correct the background pixels in a fine-grained way. So, the coefficients of $\mathcal{L}_f$ have a great influence on the performance of the model. According to the matrix relationship, it can be seen that when α = 50 and β = 0.4, the optimal result of the model is 70.81%.

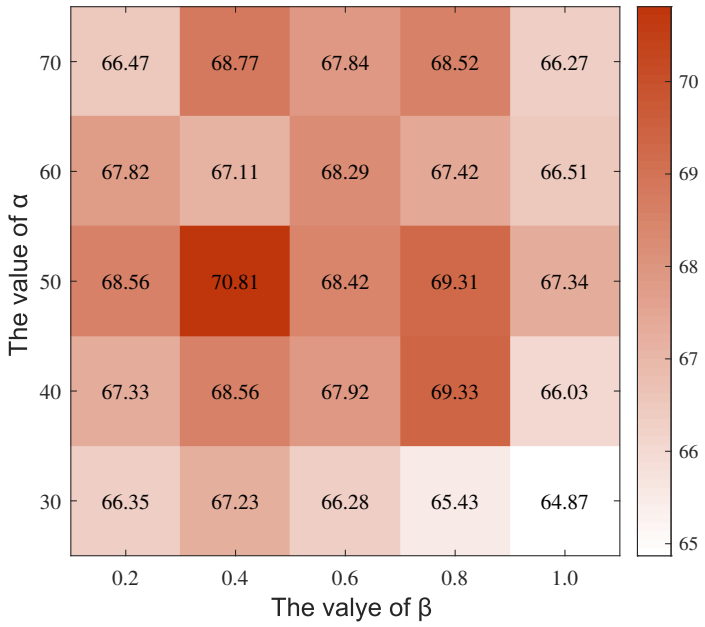


Figure 7. Ablation experiments with the values of the two loss function coefficients α and β, and the color shades represent the performance of the model.

4.4. Discussion

In this study, we propose a few shot segment segmentation method, which is experimented on PASCAL-$\mathbf{5}^{i}$ and COCO-$\mathbf{20}^{i}$ datasets and achieves decent results. Compared with the previous work, our method uses the idea of prototype complementation and fuses the query and support prototype through the self-distillation method to generate the teacher prototype, which has the characteristics of a higher affinity with the query image. Subsequently, the base class learner and background prototype are added to further correct the base class and background of the prediction map. Thanks to the ASD module and the SBP module, we have made considerable progress in our results. Although these studies reveal important findings, they also have limitations. In order to pursue the suppression of base class pixels and background pixels, our model needs to revise the prediction map again during the training process, which leads to the relatively slow training speed of the model. When the model background is single or there are many base objects in the image, PCNet may be sensitive to base class pixels and background pixels, resulting in a slight decline in the segmentation effect. Therefore, we need to continue to study the model structure to reduce parameter redundancy and balance between the novel class and background class. The experimental findings are expected to provide useful reference information for FSS tasks, and future studies may demonstrate greater power.

5. Conclusions

We propose PCNet, a new few-shot segmentation method based on the proposed adaptive self-distillation prototype module and self-support background prototype module. To solve the problem of an intra-class gap, we propose an adaptive self-distillation prototype. It combines shared features from query and support images, while accommodating unique features from the query image. This enables effective feature complementarity and generates a teacher prototype aligned with the query image. The teacher prototype guides target class segmentation in the query feature map. On this basis, we introduce a base learner to help the model adaptively mask the base pixels in the query image to avoid the model misactivating the base targets in the query image. We further construct a self-support background prototype, which uses the similar semantic information in the background pixels to generate a self-support background prototype for targeted correction of fine-grained boundary matching in the query image. It has been proved by experiments that our proposed PCNet achieves advanced results on the PASCAL-$\mathbf{5}^{i}$ and COCO-$\mathbf{20}^{i}$ datasets, which verifies the effectiveness of the proposed method.

Author Contributions: Conceptualization, J.-Y.W. and W.-M.Z.; Methodology, J.-Y.W., S.-K.L. and S.-C.G.; Validation, S.-C.G.; Formal analysis, C.-Y.J.; Writing—original draft, J.-Y.W.; Writing—review & editing, S.-K.L. and S.-C.G.; Visualization, S.-K.L. and C.-Y.J.; Supervision, W.-M.Z.; Funding acquisition, W.-M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Natural Science Foundation of Shandong Province under Grant ZR2022LZH014.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pan, J.S.; Liang, Q.; Chu, S.C.; Tseng, K.K.; Watada, J. A multi-strategy surrogate-assisted competitive swarm optimizer for expensive optimization problems. *Appl. Soft Comput.* **2023**, *147*, 110733. [CrossRef]
2. Yang, Q.; Chu, S.C.; Hu, C.C.; Kong, L.; Pan, J.S. A Task Offloading Method Based on User Satisfaction in C-RAN With Mobile Edge Computing. *IEEE Trans. Mob. Comput.* **2023**, 1–15. [CrossRef]
3. Liu, S.; Li, Y.; Chai, Q.w.; Zheng, W. Region-scalable fitting-assisted medical image segmentation with noisy labels. *Expert Syst. Appl.* **2024**, *238*, 121926. [CrossRef]
4. Zhou, L.; Liu, S.; Zheng, W. Automatic Analysis of Transverse Musculoskeletal Ultrasound Images Based on the Multi-Task Learning Model. *Entropy* **2023**, *25*, 662. [CrossRef] [PubMed]

5. Xu, X.; Du, J.; Song, J.; Xue, Z. InfoMax Classification-Enhanced Learnable Network for Few-Shot Node Classification. *Electronics* **2023**, *12*, 239. [CrossRef]
6. Lin, D.; Dai, J.; Jia, J.; He, K.; Sun, J. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3159–3167.
7. Caesar, H.; Uijlings, J.; Ferrari, V. Region-based semantic segmentation with end-to-end training. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part I 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 381–397.
8. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
9. Wei, Y.; Liang, X.; Chen, Y.; Shen, X.; Cheng, M.M.; Feng, J.; Zhao, Y.; Yan, S. Stc: A simple to complex framework for weakly-supervised semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 2314–2320. [CrossRef] [PubMed]
10. Han, H.; Huang, Y.; Wang, Z. Collaborative Self-Supervised Transductive Few-Shot Learning for Remote Sensing Scene Classification. *Electronics* **2023**, *12*, 3846. [CrossRef]
11. Guo, S.C.; Liu, S.K.; Wang, J.Y.; Zheng, W.M.; Jiang, C.Y. CLIP-Driven Prototype Network for Few-Shot Semantic Segmentation. *Entropy* **2023**, *25*, 1353. [CrossRef]
12. Li, G.; Jampani, V.; Sevilla-Lara, L.; Sun, D.; Kim, J.; Kim, J. Adaptive prototype learning and allocation for few-shot segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 8334–8343.
13. Liu, B.; Ding, Y.; Jiao, J.; Ji, X.; Ye, Q. Anti-aliasing semantic reconstruction for few-shot semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9747–9756.
14. Siam, M.; Oreshkin, B.N.; Jagersand, M. Amp: Adaptive masked proxies for few-shot segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5249–5258.
15. Chen, J.; Yuan, W.; Chen, S.; Hu, Z.; Li, P. Evo-MAML: Meta-Learning with Evolving Gradient. *Electronics* **2023**, *12*, 3865. [CrossRef]
16. Kulis, B. Metric learning: A survey. *Found. Trends® Mach. Learn.* **2013**, *5*, 287–364. [CrossRef]
17. Li, H.; Eigen, D.; Dodge, S.; Zeiler, M.; Wang, X. Finding task-relevant features for few-shot learning by category traversal. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1–10.
18. Snell, J.; Swersky, K.; Zemel, R. Prototypical networks for few-shot learning. *Adv. Neural Inf. Process. Syst.* **2017**, *30* , 4080–4090.
19. Finn, C.; Abbeel, P.; Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, NSW, Australia, 6–11 August 2017; pp. 1126–1135.
20. 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 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1199–1208.
21. Zhang, C.; Lin, G.; Liu, F.; Guo, J.; Wu, Q.; Yao, R. Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9587–9595.
22. Shaban, A.; Bansal, S.; Liu, Z.; Essa, I.; Boots, B. One-shot learning for semantic segmentation. *arXiv* **2017**, arXiv:1709.03410.
23. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; Proceedings, Part III 18; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
24. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
26. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef] [PubMed]
27. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual attention network for scene segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3146–3154.
28. 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 European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 267–283.
29. 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, Republic of Korea, 27 October–2 November 2019; pp. 603–612.
30. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.Y.; et al. Segment anything. *arXiv* **2023**, arXiv:2304.02643.
31. Chen, W.Y.; Liu, Y.C.; Kira, Z.; Wang, Y.C.F.; Huang, J.B. A closer look at few-shot classification. *arXiv* **2019**, arXiv:1904.04232.

32. Qi, H.; Brown, M.; Lowe, D.G. Low-shot learning with imprinted weights. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5822–5830.
33. Lee, Y.; Choi, S. Gradient-based meta-learning with learned layerwise metric and subspace. In Proceedings of the International Conference on Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018; pp. 2927–2936.
34. Gordon, J.; Bronskill, J.; Bauer, M.; Nowozin, S.; Turner, R.E. Meta-learning probabilistic inference for prediction. *arXiv* **2018**, arXiv:1805.09921.
35. Grant, E.; Finn, C.; Levine, S.; Darrell, T.; Griffiths, T. Recasting gradient-based meta-learning as hierarchical bayes. *arXiv* **2018**, arXiv:1801.08930.
36. Rusu, A.A.; Rao, D.; Sygnowski, J.; Vinyals, O.; Pascanu, R.; Osindero, S.; Hadsell, R. Meta-learning with latent embedding optimization. *arXiv* **2018**, arXiv:1807.05960.
37. Hou, R.; Chang, H.; Ma, B.; Shan, S.; Chen, X. Cross attention network for few-shot classification. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 4003–4014.
38. Doersch, C.; Gupta, A.; Zisserman, A. Crosstransformers: Spatially-aware few-shot transfer. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 21981–21993.
39. Koch, G.; Zemel, R.; Salakhutdinov, R. Siamese neural networks for one-shot image recognition. In Proceedings of the ICML Deep Learning Workshop, Lille, France, 6–11 July 2015; Volume 2.
40. Ravi, S.; Larochelle, H. Optimization as a model for few-shot learning. In Proceedings of the International Conference on Learning Representations, San Juan, Puerto Rico, 2–4 May 2016.
41. Triantafillou, E.; Zhu, T.; Dumoulin, V.; Lamblin, P.; Evci, U.; Xu, K.; Goroshin, R.; Gelada, C.; Swersky, K.; Manzagol, P.A.; et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv* **2019**, arXiv:1903.03096.
42. Rakelly, K.; Shelhamer, E.; Darrell, T.; Efros, A.; Levine, S. Conditional networks for few-shot semantic segmentation. In Proceedings of the Workshop Track-ICLR 2018, Vancouver, BC, Canada, 30 April–3 May 2018.
43. Wang, X.; Ye, Y.; Gupta, A. Zero-shot recognition via semantic embeddings and knowledge graphs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6857–6866.
44. Tian, Z.; Zhao, H.; Shu, M.; Yang, Z.; Li, R.; Jia, J. Prior guided feature enrichment network for few-shot segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 1050–1065. [CrossRef] [PubMed]
45. Yang, B.; Liu, C.; Li, B.; Jiao, J.; Ye, Q. Prototype mixture models for few-shot semantic segmentation. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part VIII 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 763–778.
46. Lang, C.; Cheng, G.; Tu, B.; Han, J. Learning what not to segment: A new perspective on few-shot segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 8057–8067.
47. Lu, Z.; He, S.; Zhu, X.; Zhang, L.; Song, Y.Z.; Xiang, T. Simpler is better: Few-shot semantic segmentation with classifier weight transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 8741–8750.
48. Fan, Q.; Pei, W.; Tai, Y.W.; Tang, C.K. Self-support few-shot semantic segmentation. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 701–719.
49. Chen, J.; Gao, B.B.; Lu, Z.; Xue, J.H.; Wang, C.; Liao, Q. Apanet: Adaptive prototypes alignment network for few-shot semantic segmentation. *IEEE Trans. Multimed.* **2022**, *25*, 4361–4373. [CrossRef]
50. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
51. Nguyen, K.; Todorovic, S. Feature weighting and boosting for few-shot segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 622–631.
52. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
53. Hariharan, B.; Arbeláez, P.; Bourdev, L.; Maji, S.; Malik, J. Semantic contours from inverse detectors. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 991–998.
54. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Proceedings, Part V 13; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755.
55. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
56. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
57. Min, J.; Kang, D.; Cho, M. Hypercorrelation squeeze for few-shot segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 6941–6952.
58. Xie, G.S.; Liu, J.; Xiong, H.; Shao, L. Scale-aware graph neural network for few-shot semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 5475–5484.

59. Liu, Y.; Zhang, X.; Zhang, S.; He, X. Part-aware prototype network for few-shot semantic segmentation. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part IX 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 142–158.
60. Liu, J.; Qin, Y. Prototype refinement network for few-shot segmentation. *arXiv* **2020**, arXiv:2002.03579.