Self-Calibrated Cross Attention Network for Few-Shot Segmentation

Qianxiong Xu[1]   Wenting Zhao[2]   Guosheng Lin[1*]   Cheng Long[1*]

[1]S-Lab, Nanyang Technological University   [2]Nanjing University of Science and Technology

{qianxiong.xu,gslin,c.long}@ntu.edu.sg, wtingzhao@njust.edu.cn

Abstract

The key to the success of few-shot segmentation (FSS) lies in how to effectively utilize support samples. Most solutions compress support foreground (FG) features into prototypes, but lose some spatial details. Instead, others use cross attention to fuse query features with uncompressed support FG. Query FG could be fused with support FG, however, query background (BG) cannot find matched BG features in support FG, yet inevitably integrates dissimilar features. Besides, as both query FG and BG are combined with support FG, they get entangled, thereby leading to ineffective segmentation. To cope with these issues, we design a self-calibrated cross attention (SCCA) block. For efficient patch-based attention, query and support features are firstly split into patches. Then, we design a patch alignment module to align each query patch with its most similar support patch for better cross attention. Specifically, SCCA takes a query patch as Q, and groups the patches from the same query image and the aligned patches from the support image as K&V. In this way, the query BG features are fused with matched BG features (from query patches), and thus the aforementioned issues will be mitigated. Moreover, when calculating SCCA, we design a scaled-cosine mechanism to better utilize the support features for similarity calculation. Extensive experiments conducted on PASCAL-5^i and COCO-20^i demonstrate the superiority of our model, e.g., the mIoU score under 5-shot setting on COCO-20^i is 5.6%+ better than previous state-of-the-arts. The code is available at https://github.com/Sam1224/SCCAN.

1. Introduction

With the rapid development of deep learning, semantic segmentation has made tremendous progress [21, 48, 1], but such success requires massive time and human efforts
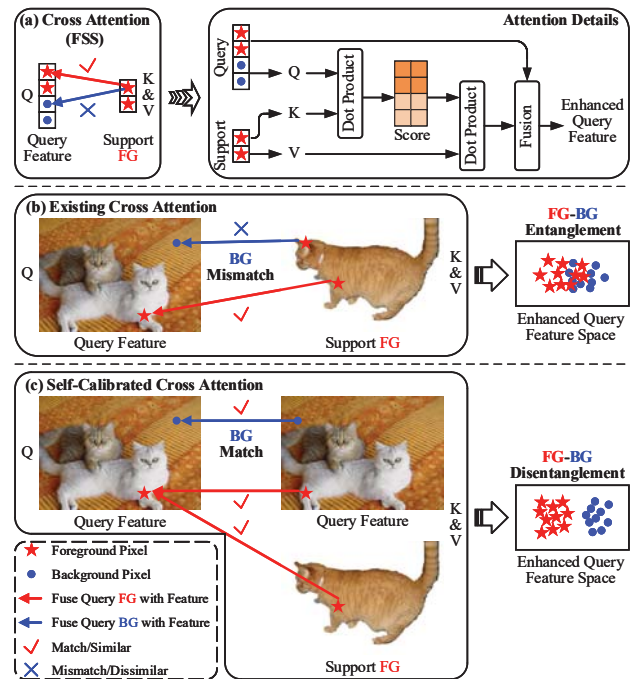
---
*Co-corresponding authors

Figure 1. **(a) Details of cross attention for FSS. (b)&(c) Existing and our proposed cross attentions.** In (**b**), query FG features are correctly fused with matched support FG features, but query BG features are combined with mismatched support FG features, and they get entangled. In (**c**), query BG is correctly fused with matched query BG instead, so query features are disentangled.

for annotating the pixel-wise masks. To save annotation cost, semi-/weakly-supervised segmentation [38, 11, 4] are proposed to take advantages of unlabelled/weakly-labelled data. Nevertheless, they both fail to segment unseen classes (during training), which prevents segmentation from generalizing to novel classes. To tackle this problem, few-shot segmentation (FSS) [28, 35, 45] is introduced to quickly adapt the model to novel classes, with the help of a few annotated samples.

Naturally, human could refer to a few samples of a

novel object, and recognize them in unlabelled images, even without knowing what they actually are, *i.e.*, class-agnostic. Inspired by this phenomenon, researchers develop the learning-to-learn paradigm [26, 30] for FSS models. They learn a segmentation pattern on some base classes during training, and apply the same pattern to segment novel classes during testing. Particularly, this pattern is to use the knowledge of a few annotated support samples to perform segmentation on an unlabelled query image for a class.

There exist many solutions for FSS, where prototype-based methods [45, 35, 15] are the most popular. Specifically, support prototypes [45, 15] are extracted from support FG[1], and they are used to segment query image through, *e.g.*, feature comparison [35]. However, compressing support FG to prototypes leads to information loss, Therefore, some methods [44, 34, 46, 37] employ cross attention instead to fuse query features with uncompressed support FG (as shown in Fig. 1(a)). Unfortunately, two other issues would arise. (1) As shown in Fig. 1(b), query BG (Q, carpet) cannot find matched features in support FG ($K\&V$, cat), *i.e.*, the similarity scores between all query BG and support FG pixel pairs are low. Then, such low scores are amplified after softmax normalization, *e.g.*, 1e-10 to 0.1 for one pair. As a result, query BG features are inevitably fused with mismatched support FG features and get biased. This issue is termed *BG mismatch*. (2) Meanwhile, query FG could effectively integrate the knowledge of matched support FG. As both query FG and BG are fused with support FG, their features get entangled, which is called *FG-BG entanglement*, thereby leading to ineffective segmentation.

To address these two issues, we propose a novel attention module named **self-calibrated cross attention (SCCA)** that calculates self and cross attentions simultaneously. As shown in Fig. 1(c), query features are still taken as Q, but the query features and support FG are grouped as $K\&V$. The rationality is explained as follows: (1) Query BG features are effectively fused with matched query BG features from the same query image in $K\&V$ (via self attention). Hence, the *BG mismatch* issue is solved. (2) Query FG can be enhanced by matched information in both query features (via self attention) and support FG (via cross attention). In this way, query BG features persist to integrate BG information, while query FG is combined with FG information, the *FG-BG entanglement* issue is mitigated.

Unluckily, the attention cost of SCCA is twice as much as that of standard cross attention. Therefore, we incorporate SCCA with the memory efficient swin transformer (ST) [20], and present the **self-calibrated cross attention network (SCCAN)** to boost FSS. To be specific, a pseudo mask aggregation (PMA) module is firstly developed to generate pseudo query masks. PMA takes *all* pairwise similarities to generate the mask value for a query pixel, which

---
[1]**FG** and **BG** are used to represent foreground and background.

is better at alleviating the effects of noises, compared to existing methods [22, 32] that merely use a *single* largest similarity value. Then, **self-calibrated cross attention (SCCA) block** is proposed for effective patch-based cross attention, which mainly consists of (1) patch alignment (PA) module and (2) aforementioned SCCA module. (1) ST-based cross attention may have the patch misalignment issue in local attention calculation. Thus, we design a PA module to align each query patch with its most similar support patch (with FG pixels). (2) Given pairs of query and aligned support patches, SCCA effectively fuses query features with support FG features. Recall that query FG features integrate information both from the same query image patches and the support image patches. Naturally, a query patch is more similar to itself than another support patch, and thus self attention might dominate in SCCA, and the query FG features may not integrate sufficient support FG features, which may weaken the FG feature representations. Hence, we further incorporate SCCA with a scaled-cosine (SC) mechanism to encourage query FG to integrate more information from the support image. In a nutshell, our contributions could be summarized as follows:

- We propose a self-calibrated cross attention network (SCCAN), including pseudo mask aggregation (PMA) module and self-calibrated cross attention (SCCA) blocks, to effectively utilize support information.

- SCCA could tackle the *BG mismatch* and *FG-BG entanglement* issues, which disentangles query FG and BG, thereby leading to effective segmentation.

- Training-agnostic PMA module could roughly locate query FG, and it is better at suppressing the effects of noises, compared to existing methods.

- New state-of-the-art could be set on PASCAL-5^i and COCO-20^i, *e.g.*, the mIoU score under 5-shot setting on COCO-20^i is 5.6%+ better than previous methods.

2. Related work

Few-shot segmentation. To alleviate the generalization problem on unseen classes of semantic segmentation, few-shot segmentation (FSS) is firstly introduced in OSLSM [28], which segments query image with the help of annotated support samples. Recent FSS methods could be roughly divided into four categories. (1) Prototype-based methods [45, 35, 32, 15, 14, 18, 43, 5, 19] compress support FG into single or multiple prototypes, which are then used to help segment the query image through, *e.g.*, cosine similarity or feature concatenation. Based on the intuition that pixels from the same object are more similar than those from different objects, SSP [5] is designed to generate
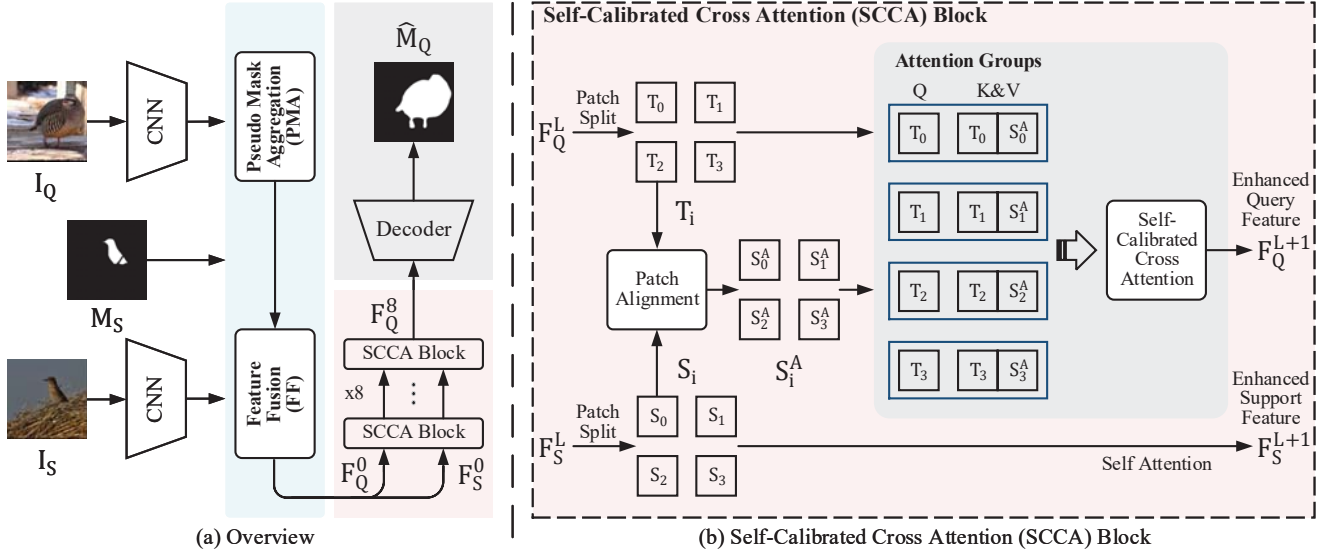
Figure 2. **Overall architecture of (a) self-calibrated cross attention network (SCCAN) and (b) self-calibrated cross attention (SCCA) block. (a) Pseudo mask aggregation** module generates a pseudo mask that could roughly locate query FG. **Feature fusion** module adapts query and support features for better cross attention. **SCCA blocks** could effectively fuse query features with support FG. Finally, the enhanced query features is forwarded to a decoder for segmentation. **(b) SCCA blocks** perform patch-based attentions. **Patch alignment** module aims at finding the most similar support patch S_i^A for each query patch T_i. **SCCA** takes a query patch T_i as Q, and groups the same T_i and its aligned support patch S_i^A as $K\&V$ to address the *BG mismatch* and *FG-BG entanglement* issues.

query FG and BG prototypes to find other similar query features. (2) To prevent from information loss of prototypes, attention-based methods [46, 44, 34, 9, 39, 37, 12] build up per-pixel relationships between query and uncompressed support features, then use cross attention to fuse query features with support FG features. Nevertheless, they suffer from the *BG mismatch* and *FG-BG entanglement* issues as explained in Sec. 1. (3) More recently, correlation-based methods [24, 29, 8, 40] focus on visual correspondence to build 4D correlations upon query and support features, then use expensive 4D operations to perform segmentation. (4) There also exist some other methods, *e.g.*, DPCN [17] generates dynamic kernels from support sample which is then used to process query features, some methods [13, 31, 10] benefit from base classes during testing and generate good results. In this work, we focus on the cross attention-based methods, and contribute effective self-calibrated cross attention (SCCA) for FSS, to solve the aforementioned issues.

Attention. Recently, some works [2, 33, 36, 42, 47, 20] demonstrate that pure transformer architecture could achieve excellent results in computer vision tasks. Particularly, swin transformer (ST) [20] calculates efficient self attention within small windows to reduce computational burden, while achieving good results. In spite of its good performance, ST in its original form does not support effective cross attention that could be utilized to fuse query features with support samples in FSS. In this work, we adapt ST for

SCCA to enable efficient and effective cross attentions.

3. Problem definition

Suppose the sets for training and testing are denoted as $\mathcal{D}_{tr}$ and $\mathcal{D}_{te}$, respectively. $\mathcal{D}_{tr}$ involves some classes $\mathcal{C}_{tr}$, while $\mathcal{D}_{te}$ covers another class set $\mathcal{C}_{te}$. FSS studies a scenario where $\mathcal{C}_{tr}$ and $\mathcal{C}_{te}$ are disjoint, *i.e.*, $\mathcal{C}_{tr} \cap \mathcal{C}_{te} = \emptyset$. Both $\mathcal{D}_{tr}$ and $\mathcal{D}_{te}$ contain numerous *episodes*, which are the basic elements of episodic training. For k-shot setting, each episode consists of a support set $\mathcal{S} = \{I_S^n, M_S^n\}_{n=1}^k$ and a query set $\mathcal{Q} = \{I_Q, M_Q\}$ for a specific class c, where I_S^n and M_S^n represent the n-th support image and its annotated binary mask, I_Q and M_Q indicate the query image and corresponding mask. During training, the model learns to segment I_Q under the guidance of support set $\mathcal{S}$ for classes $\mathcal{C}_{tr}$, and then apply the learned pattern to $\mathcal{C}_{te}$ during testing. For simplicity, we introduce our method under 1-shot setting.

4. Methodology

As shown in Fig. 2(a), we propose self-calibrated cross attention network (SCCAN), which consists of pseudo mask aggregation (PMA) module, feature fusion (FF) module and self-calibrated cross attention (SCCA) blocks. PMA is responsible for generating a pseudo query mask that could roughly locate query FG with minor cost. Then, FF adapts query and support features to close the gap between their FG features for better cross attention. Particularly, SCCA is
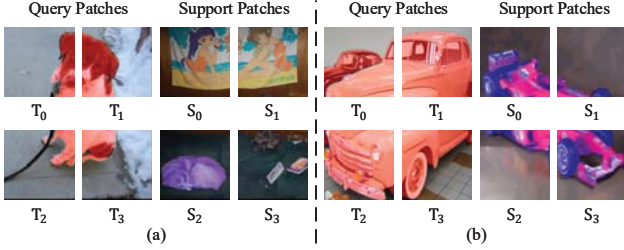
Figure 3. **Issues of swin transformer when used for cross attention.** Support BGs are usually not used in FSS, we preserve them for completeness. **(a)** *Invalid support patch.* **(b)** *Misaligned support patch.*

designed to alleviate the *BG mismatch* and *FG-BG entanglement* issues of existing cross attention-based FSS methods, and to effectively fuse query features with support FG information. Next, we will introduce the details of SCCA first, followed by PMA and FF modules.

4.1. Self-calibrated cross attention block

Swin transformer (ST) [20] is well known for its excellent performance and efficiency for self attention. Specifically, it seamlessly splits features into small windows, and performs window-based self attention. Then, windows are shifted for another self attention to capture long-range dependencies. Nevertheless, merely taking query patch (we denote that pixels in a window form a patch) as Q, and its corresponding support patch as $K\&V$ for cross attention would raise some problems. (1) *BG mismatch* and *FG-BG entanglement* (as explained earlier); (2) *Invalid support patch* (Fig. 3(a)): T_0's corresponding patch S_0 does not contain FG objects, and thus T_0 cannot be enhanced after performing cross attention with S_0; (3) *Misaligned support patch* (Fig. 3(b)): T_2 represents headstock, but in support image, headstock locates in S_3. T_2 would perform cross attention with S_2, and thus its most similar features, *i.e.*, headstock, cannot be directly accessed, which degrades the effectiveness of cross attention.

Therefore, we propose self-calibrated cross attention (SCCA) to adapt ST for effective cross attention and mitigate these issues. We employ 8 consecutive SCCA blocks (as shown in Fig. 2(b)), each of which takes the enhanced query and support features from previous block as inputs, and seamlessly split them into patches. Then, patch alignment (PA) module is proposed to align each query patch with its most similar support patch. After that, each query patch and its aligned support patch are grouped for SCCA to effectively enhance query features with support FG.

The inputs for the first block come from feature fusion (FF) module, which will be introduced later. Both input features of the L-th block $F_Q^L \in \mathbb{R}^{C \times H \times W}$ and $F_S^L \in \mathbb{R}^{C \times H \times W}$ are split into patches $T_i \in \mathbb{R}^{N^2 \times C \times K \times K}$ and
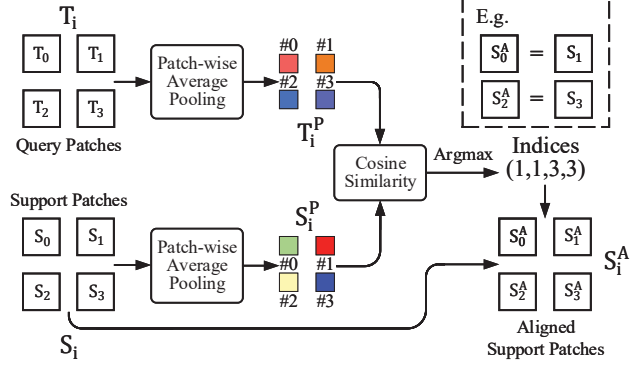

Figure 4. **Details of patch alignment (PA) module.** Firstly, prototype is obtained from each input patch. Then, cosine similarity is calculated among query and support prototypes. Finally, each query patch's most similar support patch is obtained via $\arg \max$ operation, and then they are aligned.

$S_i \in \mathbb{R}^{N^2 \times C \times K \times K}, i \in \{0, \cdots, N^2 - 1\}$, where N^2 is the number of patches, and $K \times K$ denotes window/patch size.

Patch alignment (PA). PA could mitigate the *invalid support patch* and *misaligned support patch* issues by aligning each query patch with its most similar support patch (with FG pixels) for effective cross attention. Specifically, as shown in Fig. 4, the prototype of each patch is obtained via patch-wise average pooling.

$$T_i^P = PAP(T_i), S_i^P = PAP(S_i) \qquad (1)$$

where $T_i^P \in \mathbb{R}^{N^2 \times C \times 1 \times 1}$ and $S_i^P \in \mathbb{R}^{N^2 \times C \times 1 \times 1}$ are the prototypes of query and support patches, and PAP denotes patch-wise average pooling. Note that support features and support prototypes only contain FG information.

Next, we measure the cosine similarity between patch prototypes T_i^P and S_i^P, mask out support patches without FG objects in the similarity score, and perform $\arg \max$ to obtain each query patch's most similar support patch.

$$Indices = \arg \max(Cos(T_i^P, S_i^P) \circ M_S^P) \qquad (2)$$

where $Indices \in \mathbb{R}^{N^2}$ are the indices of aligned support patches S_i^A, Cos means cosine similarity, $\circ$ denotes Hadamard product, $M_S^P \in \{1, 0\}^{1 \times N^2}$ is a mask indicating if each support patch has FG pixels or not. Then, each query and its aligned support patch are grouped for SCCA.

Self-calibrated cross attention (SCCA). Given a pair of query and its aligned support patch, the query patch is taken as Q, while both of them are taken as $K\&V$, and thus self attention (query patch in Q with query patch in $K\&V$) and cross attention (query patch in Q with support patch in $K\&V$) would be calculated simultaneously. We explain the rationale as follows.

(1) For a query BG pixel, it cannot find similar information from support FG in existing cross attention-based
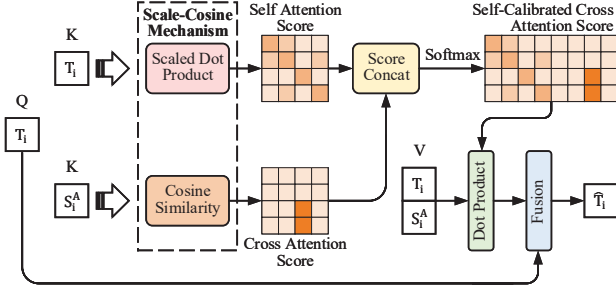
Figure 5. **Details of self-calibrated cross attention.** Fusion module is a feed-forward network. Query patch firstly calculates self and cross attention scores with itself and the aligned support patch. Scores are then concatenated and normalized. Finally, the score is used to obtain useful support information, which is fused with input query patch for query feature enhancement.

FSS methods. During SCCA, query BG could find other matched BG information from query features themselves (via self attention), while ignoring the dissimilar support FG. Thus, the *BG mismatch* problem is solved.

(2) Query FG could find matched FG features not only from support FG but also from query features, and thus it could be fused with reasonable features, and get enhanced.

(3) Since both query FG and BG are correctly fused with their matched features in SCCA, the *FG-BG entanglement* problem could be mitigated.

Recall that the performance of FSS is affected by how the support sample is utilized. Unfortunately, in (2), query FG may not be fused with sufficient support features, because FG pixels in a query patch are more similar to themselves, compared with those in the aligned support patch. As a result, self attention might dominate in SCCA. To address this issue, we further contribute a **scaled-cosine (SC)** mechanism. As illustrated in Fig. 5, when performing SCCA, we use scaled dot product for self attention, and cosine similarity for cross attention. Their attention scores are then concatenated and normalized. Dot product is a more strict similarity-related operator, as it takes both direction and magnitude into consideration. In contrast, cosine similarity only measures the similarity of directions. Hence, we employ cosine similarity for cross attention to encourage to discover more similar regions, and query FG is more likely to integrate sufficient information from support FG.

We formally describe SCCA as follows. Scaled dot product is firstly conducted on query patch T_i and itself to obtain self attention score. Meanwhile, cosine similarity is performed on the same T_i and aligned support patch S_i^A to obtain cross attention score. Their scores are denoted as $A_{QQ} \in \mathbb{R}^{N^2 \times K^2 \times K^2}$ and $A_{QS} \in \mathbb{R}^{N^2 \times K^2 \times K^2}$, where N^2 and K^2 are the number of patches and pixels, respectively.

$$A_{QQ} = \frac{T_i \cdot T_i}{\sqrt{d_k}}, A_{QS} = \frac{T_i \cdot S_i^A}{\|T_i\|\|S_i^A\|} \tag{3}$$
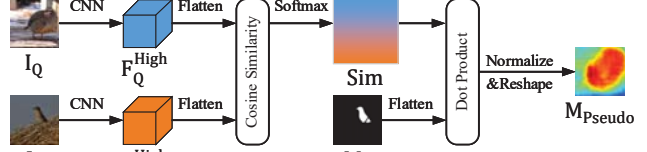

Figure 6. **Details of pseudo mask aggregation (PMA).** Given a query pixel, the weighted summation of its normalized similarity scores with all support pixels and the annotated support mask means the probability of being FG.

Then, two scores are concatenated and normalized by the softmax operation to obtain the final attention score $A \in \mathbb{R}^{N^2 \times K^2 \times 2K^2}$.

$$A = Softmax(Concat(A_{QQ}, A_{QS})) \tag{4}$$

Finally, A is used to aggregate information from T_i and S_i^A, and then be fused with T_i for query feature enhancement.

$$\hat{T}_i = FFN(A \cdot Concat(T_i, S_i^A) + T_i) \tag{5}$$

where FFN is feed-forward network.

Once all enhanced query patches are obtained, they are assembled to be $F_Q^{L+1} \in \mathbb{R}^{C \times H \times W}$. Moreover, F_Q^7 from the last SCCA block is forwarded to the decoder [13] and generate the segmentation $\hat{M}_Q \in \mathbb{R}^{2 \times H \times W}$.

Window shifting. To enable interactions among patches, we adopt window shifting operation in even layers as swin transformer [20]. After window shifting, some irregular patches are obtained at image borders, and we pad them to have regular size for PA and SCCA.

4.2. Pseudo mask aggregation

Pseudo masks [17, 22, 32, 46] are commonly incorporated into FSS models because they could roughly locate the query FG object without learnable parameters. Specifically, they measure the cosine similarity between each pair of high-level query and support FG pixels [32, 46] or patches [17, 22]. Then, query pixel's largest similarity score is normalized and taken as its probability of being FG.

However, existing methods suffer from the following two issues. (1) They only take support FG into consideration, which may not work well when both query FG and BG look dissimilar to support FG, *i.e.*, their scores are similar. (2) As each value in pseudo mask merely corresponds to the largest cosine similarity among pixel or patch pairs, they would be heavily affected by noises. As a result, the locating function of pseudo mask is weakened.

To tackle these issues, we propose a pseudo mask aggregation (PMA) module, which is illustrated in Fig. 6. Concretely, we apply two changes to the original one: (1) Both support FG and BG features are used for reliable comparisons, *e.g.*, human head (query FG) is more similar to body

659

(support FG) than room (support BG), and thus it is classified as FG. (2) To suppress the effects of noises, we use all pairwise similarities to perform weighted summation on binary support mask, instead of merely taking a largest value.

First of all, PMA calculates cosine similarity $Sim \in \mathbb{R}^{HW \times HW}$ between high-level query features $F_Q^{High} \in \mathbb{R}^{C \times H \times W}$ and support features $F_S^{High} \in \mathbb{R}^{C \times H \times W}$. Then, a softmax operation is applied to normalize the score. After that, dot product is conducted on Sim and support mask M_S to obtain the pseudo mask $M_{Pseudo} \in \mathbb{R}^{HW \times 1}$. Finally, normalization and reshape operations are employed to generate the pseudo mask $M_{Pseudo} \in \mathbb{R}^{1 \times H \times W}$.

4.3. Feature fusion

Due to the fact that FG objects in query and support images could be dissimilar, we start from the mid-level query features F_Q^{Mid} and support features F_S^{Mid} (obtained from the 3rd and 4th blocks of the backbone [32]), and use the support prototype P_S (extracted from support FG) to close their gaps. Concretely, F_Q^{Mid} is combined with P_S and its pseudo mask M_{Pseudo} to obtain F_Q. Similarly, F_S^{Mid} is fused with support prototype P_S and support mask M_S to generate F_S. Particularly, F_Q and F_S are inputted to SCCA as F_Q^0 and F_S^0, respectively.

5. Experiments

5.1. Experimental settings

Datasets. We evaluate our method on two public benchmark datasets, including PASCAL-5^i [28] and COCO-20^i [25]. PASCAL-5^i contains 20 classes, and is built upon PASCAL VOC 2012 [3] with additional annotations from SDS [6], while COCO-20^i is created from MSCOCO [16], which is more challenging and has 80 classes. Both PASCAL-5^i and COCO-20^i are evenly split into 4 folds for cross validation, *i.e.*, 5 and 20 classes per fold, respectively. In each fold, the union of other three folds is for training, while the fold itself is preserved for testing. Besides, 1,000 and 4,000 episodes are randomly sampled from PASCAL-5^i and COCO-20^i during testing.

Evaluation metrics. Following existing works [32, 46, 8], mean intersection over union (mIoU) and foreground-background IoU (FB-IoU) are adopted as evaluation metrics. The former measures the mean IoU scores for all FG classes in a fold, while FB-IoU regards them as a single FG class, and reports the average IoU scores for FG and BG.

Implementation details. We take ResNet50/101 [7] pre-trained on ImageNet [27] as backbones, whose weights are frozen. Following CyCTR [46], we use AdamW and SGD optimizers to optimize attention-related (*i.e.*, SCCA) and other parameters, respectively, and Dice loss [23] is taken as the loss function $\mathcal{L}_{Seg}$. The model is trained for 200 epochs on PASCAL-5^i, and 50 epochs on COCO-20^i. For both datasets, the batch size is fixed as 8, and the learning rates of AdamW and SGD are initialized as 6e-5 and 5e-3, respectively. During training, all images are randomly cropped to 473×473 patches as inputs, and we employ the same set of data augmentation techniques as PFENet [32]. During testing, we resize the outputs to compare with the original ground truths. For SCCA, we employ 8 SCCA blocks, and set window size as 8. For other attention-related parameters, number of heads is 8, embedding dimension is 256, and MLP ratio is 1. For k-shot setting, when $k > 1$, we simply follow PFENet [32] to average the support features.

5.2. Comparisons with state-of-the-arts

Quantitative results. The results of baselines, as well as our proposed SCCAN, are shown in Tab. 1 and Tab. 2, and we could draw the following conclusions. (1) SCCAN outperforms all baselines by considerable margins under most of the settings, and achieves new state-of-the-arts. (2) Particularly, the gap between SCCAN and other methods is larger on COCO-20^i. For example, with ResNet50, SCCAN outperforms others by 1.3%+ and 3.0%+ under 1-shot setting on two datasets, in terms of mIoU (averaged from 4 folds). When referring to 5-shot setting (with ResNet50), the gap is more prominent, *e.g.*, mIoU of SCCAN is 5.6%+ better than that of the best baseline DCAMA [29] on COCO-20^i. We believe this can be explained by the differences between two datasets. PASCAL-5^i contains many images with easy BG (*e.g.*, pure sky), while COCO-20^i is a more challenging benchmark, *e.g.*, the images usually contain multiple FG objects from different classes. In each episode, only one class is considered as FG, while other objects will be taken all as BG. Therefore, BG is much more complex in this case. Our SCCAN directly mitigates the *BG mismatch* and *FG-BG entanglement* issues, and could deal with the complex BG better. Thus, SCCAN could gain more improvements on COCO-20^i than on PASCAL-5^i.

Qualitative results. To better understand our SCCAN, we take some episodes from two datasets, then compare SCCAN with a latest baseline VAT [8], and the qualitative results are presented in Fig. 7. We could observe that SCCAN is better at distinguishing FG and BG, *e.g.*, in the third and fifth columns of PASCAL-5^i, VAT mistakenly classifies human as FG (horse and train), but SCCAN could recognize them well. Moreover, SCCAN could sometimes yield more reasonable results than human-annotated labels, *e.g.*, in the second column of COCO-20^i, there exist a hole for plate in the right doughnut, SCCAN could point it out, but VAT and ground truth wrongly consider it as FG.

5.3. Ablation study

Components analysis. To validate the effectiveness of proposed modules, we perform ablation study with ResNet50 on PASCAL-5^i, under 1-shot setting. SCCAN mainly con-

Backbone	Method	1-shot						5-shot					
		5^0	5^1	5^2	5^3	Mean	FB-IoU	5^0	5^1	5^2	5^3	Mean	FB-IoU
ResNet50	PFENet† (TPAMI'20) [32]	61.7	69.5	55.4	56.3	60.8	73.3	63.1	70.7	55.8	57.9	61.9	73.9
	MLC† (ICCV'21) [41]	59.2	71.2	65.6	52.5	62.1	-	63.5	71.6	**71.2**	58.1	66.1	-
	HSNet‡* (ICCV'21) [24]	63.5	68.2	62.4	<u>59.8</u>	63.5	76.5	70.1	72.0	67.9	<u>67.1</u>	69.3	80.6
	CyCTR† (NIPS'21) [46]	65.7	71.0	59.5	59.7	64.0	-	69.3	73.5	63.8	63.5	67.5	-
	NTRENet† (CVPR'22) [18]	65.4	72.3	59.4	<u>59.8</u>	64.2	77.0	66.2	72.8	61.7	62.2	65.7	78.4
	DPCN† (CVPR'22) [17]	-	-	-	-	65.7	<u>77.4</u>	-	-	-	-	-	-
	DCAMA‡* (ECCV'22) [29]	66.1	71.9	59.7	57.5	63.8	75.7	70.7	72.9	63.0	65.0	67.9	79.4
	VAT‡* (ECCV'22) [8]	65.9	70.0	64.9	59.0	65.0	<u>77.4</u>	<u>72.1</u>	<u>74.2</u>	69.7	65.3	<u>70.3</u>	<u>81.1</u>
	SCCAN† (Ours)	<u>67.5</u>	**72.6**	**67.2**	60.5	**67.0**	76.4	69.9	**74.3**	<u>70.1</u>	66.9	<u>70.3</u>	79.7
	SCCAN‡ (Ours)	**68.3**	72.5	66.8	<u>59.8</u>	66.8	**77.7**	**72.3**	74.1	69.1	65.6	**70.3**	**81.8**
ResNet101	PFENet† (TPAMI'20) [32]	60.5	69.4	54.4	55.9	60.1	72.9	62.8	70.4	54.9	57.6	61.4	73.5
	MLC† (ICCV'21) [41]	60.8	71.3	61.5	56.9	62.6	-	65.8	74.9	**71.4**	63.1	68.8	-
	HSNet‡* (ICCV'21) [24]	65.7	70.3	63.2	<u>61.9</u>	65.3	77.2	72.0	73.6	68.7	<u>68.4</u>	70.7	80.9
	CyCTR† (NIPS'21) [46]	67.2	71.1	57.6	59.0	63.7	-	71.0	75.0	58.5	65.0	67.4	-
	NTRENet† (CVPR'22) [18]	65.5	71.8	59.1	58.3	63.7	75.3	67.9	73.2	60.1	66.8	67.0	78.2
	DCAMA‡ (ECCV'22) [29]	62.5	70.8	64.5	56.4	63.5	76.5	70.0	73.8	66.8	65.0	68.9	81.1
	VAT‡* (ECCV'22) [8]	68.1	71.7	64.8	**63.3**	67.0	**78.7**	<u>72.6</u>	74.1	69.5	**69.5**	<u>71.4</u>	<u>82.0</u>
	SCCAN† (Ours)	<u>69.1</u>	**74.0**	<u>66.3</u>	61.6	<u>67.7</u>	77.3	71.6	<u>75.2</u>	69.5	66.5	70.7	79.6
	SCCAN‡ (Ours)	**70.9**	<u>73.9</u>	**66.8**	61.7	**68.3**	<u>78.5</u>	**73.1**	**76.4**	<u>70.3</u>	66.1	**71.5**	**82.1**

Table 1. **Comparison with state-of-the-arts on PASCAL-5^i**. **Bold** results represent the best performance, while the <u>underlined</u> results indicate the second best. ⋆ means a method originally uses different data lists for testing, and is adapted with the uniform ones. † and ‡ indicate that the resize methods from PFENet [32] and HSNet [24] are used (which is explained in supplementary material).

Backbone	Method	1-shot						5-shot					
		20^0	20^1	20^2	20^3	Mean	FB-IoU	20^0	20^1	20^2	20^3	Mean	FB-IoU
ResNet50	MLC† (ICCV'21) [41]	**46.8**	35.3	26.2	27.1	33.9	-	**54.1**	41.2	34.1	33.1	40.6	-
	HSNet‡ (ICCV'21) [24]	36.7	41.4	39.5	39.1	39.2	67.6	44.4	49.7	46.1	45.5	46.4	70.9
	CyCTR† (NIPS'21) [46]	38.9	43.0	39.6	39.8	40.3	-	41.1	48.9	45.2	47.0	45.6	-
	NTRENet† (CVPR'22) [18]	36.8	42.6	39.9	37.9	39.3	68.5	38.2	44.1	40.4	38.4	40.3	69.2
	DPCN† (CVPR'22) [17]	-	-	-	-	41.5	62.7	-	-	-	-	-	-
	DCAMA‡ (ECCV'22) [29]	<u>41.9</u>	45.1	44.4	41.7	43.3	<u>69.5</u>	45.9	50.5	50.7	46.0	48.3	71.7
	VAT‡ (ECCV'22) [8]	39.1	43.5	42.1	39.9	41.1	68.2	45.2	50.1	48.0	45.3	47.2	71.6
	SCCAN† (Ours)	39.5	<u>49.3</u>	<u>47.3</u>	<u>44.3</u>	<u>45.1</u>	68.5	45.7	<u>56.4</u>	<u>56.5</u>	<u>50.7</u>	<u>52.3</u>	<u>72.2</u>
	SCCAN‡ (Ours)	40.4	**49.7**	**49.6**	**45.6**	**46.3**	69.9	<u>47.2</u>	**57.2**	**59.2**	**52.1**	**53.9**	**74.2**
ResNet101	PFENet† (TPAMI'20) [32]	34.3	33.0	32.3	30.1	32.4	58.6	38.5	38.6	38.2	34.3	37.4	61.9
	MLC† (ICCV'21) [41]	**50.2**	37.8	27.1	30.4	36.4	-	**57.0**	46.2	37.3	37.2	44.4	
	HSNet‡ (ICCV'21) [24]	37.6	44.5	44.4	40.7	41.8	69.0	45.1	52.3	48.5	47.9	48.5	72.1
	NTRENet† (CVPR'22) [18]	38.3	40.4	39.5	38.1	39.1	67.5	42.3	44.4	44.2	41.7	43.2	69.6
	DCAMA‡ (ECCV'22) [29]	41.5	46.2	45.2	41.3	43.5	**69.9**	48.0	58.0	54.3	47.1	51.9	73.3
	SCCAN† (Ours)	41.7	<u>51.3</u>	<u>48.4</u>	<u>46.7</u>	<u>47.0</u>	68.5	49.0	<u>59.3</u>	<u>59.4</u>	<u>52.7</u>	<u>55.1</u>	<u>73.4</u>
	SCCAN‡ (Ours)	<u>42.6</u>	**51.4**	**50.0**	**48.8**	**48.2**	<u>69.7</u>	<u>49.4</u>	**61.7**	**61.9**	**55.0**	**57.0**	**74.8**

Table 2. **Comparison with state-of-the-arts on COCO-20^i**. **Bold** results represent the best performance, while the <u>underlined</u> results indicate the second best. † and ‡ indicate that the resize methods from PFENet [32] and HSNet [24] are used.

PMA	Attention	PA	SC	mIoU	FB-IoU
				62.3	73.3
✓				65.2	76.0
✓	ST			65.9	75.5
✓	CST			65.3	75.4
✓	SCCA			66.0	75.3
✓	SCCA	✓		<u>66.5</u>	<u>76.2</u>
✓	SCCA	✓	✓	**67.0**	**76.4**

Table 3. **Ablation study**. mIoU shows the average score of 4 folds. **PMA**: Pseudo mask aggregation; **ST**: Swin transformer; **CST**: Swin transformer with cross attentions; **SCCA**: Self-calibrated cross attention; **PA**: Patch alignment; **SC**: Scaled-cosine.

sists of PMA module and SCCA which further contains PA, and SC (Scaled-cosine) mechanism. Our basic model is obtained when we remove all these components from SC-CAN. As shown in Tab. 3, the usage of PMA could bring 2.9% growth of mIoU, and if we further employ the standard ST for feature enhancement, the mIoU score could reach 65.9%. When we merely replace self attention as cross attention in ST (denoted as CST), the *BG mismatch* and *FG-BG entanglement* issues would arise, and the mIoU score will be 0.6% worse. Fortunately, SCCA could mitigate these issues, but the obtained score is close to that of the standard ST (66.0% vs 65.9%), which is caused by *invalid&misaligned support patch* issues (described in Sec. 4.1). PA is used to tackle the issues, and the score is 66.5%. Finally, SC mechanism aims at encouraging cross attention, and the final mIoU is boosted to 67.0%.

Figure 7. **Qualitative results of our SCCAN and VAT [8] under 1-shot setting on PASCAL-5^i and COCO-20^i.**
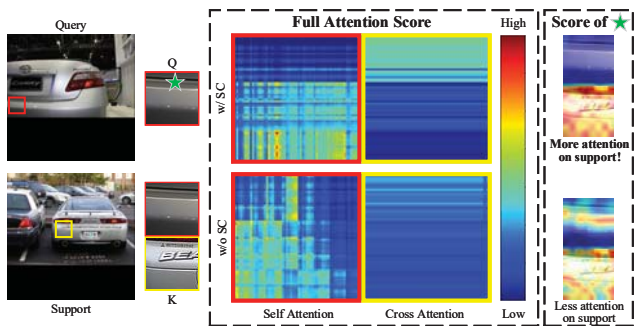

Figure 8. **Visualization of attention scores (after softmax) with/without SC mechanism**. Zoom in for more details. Red rectangle means query patch, while yellow rectangle means support patch.

Patch size	5^0	5^1	5^2	5^3	mIoU	FB-IoU
5	67.1	72.4	66.5	60.0	66.5	76.0
8	67.5	**72.6**	**67.2**	**60.5**	**67.0**	**76.4**
10	67.2	72.3	66.8	60.1	66.6	75.6
15	**67.8**	72.5	66.4	59.8	66.6	75.1

Table 4. **Parameter study of window/patch sizes on PASCAL-5^i**. mIoU shows the average score of 4 folds.

Visualization of scaled-cosine mechanism. To better understand the function of SC mechanism, we refer to the last two rows of Tab. 3, and visualize their attention scores with an example in Fig. 8. The yellow support patch is the aligned patch of the red one. Referring to the rightmost column in Fig. 8, we could observe that for a query FG pixel, SC mechanism could help to focus more on support FG features, *i.e.*, more information could be obtained from support patch. In this way, support information is better utilized.

#SCCA Blocks	mIoU	FB-IoU	#Params (M)	FLOPs (G)
4	66.3	75.8	31.8	447.7
8	**67.0**	**76.4**	35.0	480.9
12	66.3	76.0	38.1	514.1
16	66.2	75.8	41.3	547.3

Table 5. **Parameter study of number of SCCA blocks on PASCAL-5^i**. mIoU shows the average score of 4 folds.

Patch/Window size in SCCA. Patch size plays an important role in swin transformer. With the decrease of patch size, less GPU memory is required, but one pixel could access to less pixels within one attention. In SCCA, patch size would also affect patch alignment. Therefore, we take values from {5, 8, 10, 15} to study its impacts. As shown in Tab. 4, when patch size is 8, best performance is achieved.

Number of SCCA blocks. SCCA blocks are the basis of our proposed SCCAN, and we conduct experiments to study SCCAN with different number of SCCA blocks. As shown in Tab. 5, the best mIoU and FB-IoU scores could be achieved when we use 8 SCCA blocks. The Floating Point Operations (FLOPs) score is calculated based on one query and support image pairs, whose shapes are uniformly set as 473×473.

6. Conclusion

We propose a self-calibrated cross attention network (SCCAN) for accurate FSS, which consists of SCCA blocks and PMA module. SCCA is designed to effectively fuse query features with support FG features. PMA could generate robust pseudo masks for query images with minor cost. Extensive experiments are conducted and show that SCCAN could achieve new state-of-the-arts for FSS.

Limitation. Currently, we follow PFENet [32] to average the features of 5 support samples under 5-shot setting. However, cross attention will be affected, *e.g.*, if there are objects at the top-left and bottom-right corners for 1 and 4 support images, respectively, then their features will roughly multiply 1/5 and 4/5 due to the average operation, and the top-left support features will be less similar to query features, and less used. Thus, a k-shot strategy should be particularly designed for cross attention, and we leave it for future work.

References

[1] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 1

[2] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[3] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 6

[4] Jiashuo Fan, Bin Gao, Huan Jin, and Lihui Jiang. Ucc: Uncertainty guided cross-head co-training for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9947–9956, 2022. 1

[5] Qi Fan, Wenjie Pei, Yu-Wing Tai, and Chi-Keung Tang. Self-support few-shot semantic segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIX*, pages 701–719. Springer, 2022. 2

[6] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *European conference on computer vision*, pages 297–312. Springer, 2014. 6

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[8] Sunghwan Hong, Seokju Cho, Jisu Nam, Stephen Lin, and Seungryong Kim. Cost aggregation with 4d convolutional swin transformer for few-shot segmentation. In *European Conference on Computer Vision*, pages 108–126. Springer, 2022. 3, 6, 7, 8

[9] Tao Hu, Pengwan Yang, Chiliang Zhang, Gang Yu, Yadong Mu, and Cees GM Snoek. Attention-based multi-context guiding for few-shot semantic segmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8441–8448, 2019. 3

[10] Ehtesham Iqbal, Sirojbek Safarov, and Seongdeok Bang. Msanet: Multi-similarity and attention guidance for boosting few-shot segmentation. *arXiv preprint arXiv:2206.09667*, 2022. 3

[11] Peng-Tao Jiang, Yuqi Yang, Qibin Hou, and Yunchao Wei. L2g: A simple local-to-global knowledge transfer framework for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16886–16896, 2022. 1

[12] Siyu Jiao, Gengwei Zhang, Shant Navasardyan, Ling Chen, Yao Zhao, Yunchao Wei, and Humphrey Shi. Mask matching transformer for few-shot segmentation. *arXiv preprint arXiv:2301.01208*, 2022. 3

[13] Chunbo Lang, Gong Cheng, Binfei Tu, and Junwei Han. 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*, pages 8057–8067, 2022. 3, 5

[14] Chunbo Lang, Binfei Tu, Gong Cheng, and Junwei Han. Beyond the prototype: Divide-and-conquer proxies for few-shot segmentation. *arXiv preprint arXiv:2204.09903*, 2022. 2

[15] Gen Li, Varun Jampani, Laura Sevilla-Lara, Deqing Sun, Jonghyun Kim, and Joongkyu Kim. Adaptive prototype learning and allocation for few-shot segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8334–8343, 2021. 2

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

[17] Jie Liu, Yanqi Bao, Guo-Sen Xie, Huan Xiong, Jan-Jakob Sonke, and Efstratios Gavves. Dynamic prototype convolution network for few-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11553–11562, 2022. 3, 5, 7

[18] Yuanwei Liu, Nian Liu, Qinglong Cao, Xiwen Yao, Junwei Han, and Ling Shao. Learning non-target knowledge for few-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11573–11582, 2022. 2, 7

[19] Yuanwei Liu, Nian Liu, Xiwen Yao, and Junwei Han. Intermediate prototype mining transformer for few-shot semantic segmentation. *arXiv preprint arXiv:2210.06780*, 2022. 2

[20] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 2, 3, 4, 5

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1

[22] Xiaoliu Luo, Zhuotao Tian, Taiping Zhang, Bei Yu, Yuan Yan Tang, and Jiaya Jia. Pfenet++: Boosting few-shot semantic segmentation with the noise-filtered context-aware prior mask. *arXiv preprint arXiv:2109.13788*, 2021. 2, 5

[23] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016. 6

[24] Juhong Min, Dahyun Kang, and Minsu Cho. Hypercorrelation squeeze for few-shot segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6941–6952, 2021. 3, 7

[25] Khoi Nguyen and Sinisa Todorovic. Feature weighting and boosting for few-shot segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 622–631, 2019. 6

[26] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018. 2

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 6

[28] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. *arXiv preprint arXiv:1709.03410*, 2017. 1, 2, 6

[29] Xinyu Shi, Dong Wei, Yu Zhang, Donghuan Lu, Munan Ning, Jiashun Chen, Kai Ma, and Yefeng Zheng. Dense cross-query-and-support attention weighted mask aggregation for few-shot segmentation. In *European Conference on Computer Vision*, pages 151–168. Springer, 2022. 3, 6, 7

[30] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017. 2

[31] Yanpeng Sun, Qiang Chen, Xiangyu He, Jian Wang, Haocheng Feng, Junyu Han, Errui Ding, Jian Cheng, Zechao Li, and Jingdong Wang. Singular value fine-tuning: Few-shot segmentation requires few-parameters fine-tuning. *arXiv preprint arXiv:2206.06122*, 2022. 3

[32] Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 2, 5, 6, 7, 9

[33] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 3

[34] Haochen Wang, Xudong Zhang, Yutao Hu, Yandan Yang, Xianbin Cao, and Xiantong Zhen. Few-shot semantic segmentation with democratic attention networks. In *European Conference on Computer Vision*, pages 730–746. Springer, 2020. 2, 3

[35] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. Panet: Few-shot image semantic segmentation with prototype alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9197–9206, 2019. 1, 2

[36] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 568–578, 2021. 3

[37] Yuan Wang, Rui Sun, Zhe Zhang, and Tianzhu Zhang. Adaptive agent transformer for few-shot segmentation. In *European Conference on Computer Vision*, pages 36–52. Springer, 2022. 2, 3

[38] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7268–7277, 2018. 1

[39] Guo-Sen Xie, Jie Liu, Huan Xiong, and Ling Shao. Scale-aware graph neural network for few-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5475–5484, 2021. 3

[40] Zhitong Xiong, Haopeng Li, and Xiao Xiang Zhu. Doubly deformable aggregation of covariance matrices for few-shot segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XX*, pages 133–150. Springer, 2022. 3

[41] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. Mining latent classes for few-shot segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8721–8730, 2021. 7

[42] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 558–567, 2021. 3

[43] Bingfeng Zhang, Jimin Xiao, and Terry Qin. Self-guided and cross-guided learning for few-shot segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8312–8321, 2021. 2

[44] Chi Zhang, Guosheng Lin, Fayao Liu, Jiushuang Guo, Qingyao Wu, and Rui Yao. Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9587–9595, 2019. 2, 3

[45] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5217–5226, 2019. 1, 2

[46] Gengwei Zhang, Guoliang Kang, Yi Yang, and Yunchao Wei. Few-shot segmentation via cycle-consistent trans-

former. *Advances in Neural Information Processing Systems*, 34:21984–21996, 2021. 2, 3, 5, 6, 7

[47] Xiaosong Zhang, Yunjie Tian, Wei Huang, Qixiang Ye, Qi Dai, Lingxi Xie, and Qi Tian. Hivit: Hierarchical vision transformer meets masked image modeling. *arXiv preprint arXiv:2205.14949*, 2022. 3

[48] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 1