Supplementary Material for
Adaptive Prototype Learning and Allocation for Few-Shot Segmentation

Gen Li[1,3], Varun Jampani[2], Laura Sevilla-Lara[1], Deqing Sun[2], Jonghyun Kim[3], Joongkyu Kim[3*]
[1]University of Edinburgh [2]Google Research [3]Sungkyunkwan University

Thank you for reading the supplementary material, in which we introduce more experimental details in Section 1, and provide more qualitative visualization examples on Pascal-5^i and COCO-20^i in Section 2.

1. Additional Experimental Details

1.1. Detailed mean IoU results on COCO-20^i

In Table 1, we present the detailed per-split results in terms of mean IoU. As can be seen in the table, we achieve the best performance in every split, which demonstrates the superiority of our method.

1.2. Calculation of FLOPs

In the ablation study, we use floating point operations (FLOPs) to evaluate the amount of computation and model complexity. Here, we describe the calculation in detail. For a general convolution layer, the operations of one pixel in the output feature map are calculated as follows:

$$F = \begin{cases} (C_{in} \cdot K^2) + (C_{in} \cdot K^2 - 1) & \text{bias=False} \\ (C_{in} \cdot K^2) + (C_{in} \cdot K^2) & \text{bias=True} \end{cases} \quad (1)$$

where the first item is multiplication, and the second one denotes addition. C_{in} is the number of input channels and K is the kernel size. Then, extending to the whole feature map, we get the number of FLOPs as:

$$FLOPs = F \times H \times W \times C_{out}, \quad (2)$$

where H, W is the size of output feature, and C_{out} is the number of output feature channels. For example, the FLOPs are 0.9G when using 256 1×1 convolution filters to process the merged feature $F_Q^{'} \in \mathbb{R}^{513 \times 60 \times 60}$.

1.3. Ablation Study on Iteration Number

To explore the effect of the number of iterations, we implement experiments of 1-shot setting with different iteration numbers on Pascal-5^0. As shown in Figure 1, both FB-IoU and mIoU increase monotonically with more iterations, and it takes around 5 iterations to obtain the converged result.

*Corresponding author

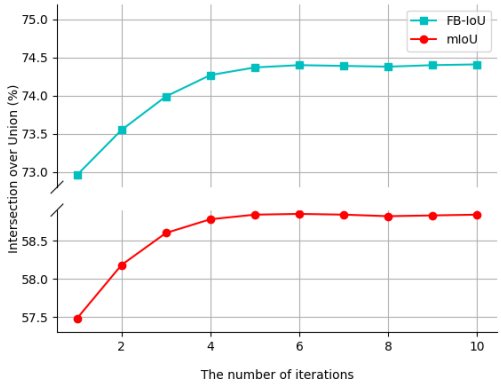

Figure 1. Ablation study on evaluation iterations.

2. Additional Qualitative Results

2.1. Visual Results on Pascal-5^i and COCO-20^i

In Figure 2, we present more qualitative results in comparison to the single prototype baseline. These qualitative results demonstrate that our model is capable of handling large variations in appearance, scale and shape between support and query images. Compared with the baseline, we perform particularly better in occluded cases, e.g, column 3-6 of Figure 2.

2.2. Visualizations of Simiarity Map

To better understand the proposed method, we visualize each similarity map, which is obtained by computing the cosine distance between each prototype and query feature. As presented in Figure 3, prototypes represent parts of the object with similar characteristics, which make the network more adaptive and discriminative.

Backbone	Methods	1-shot					5-shot				
		s-0	s-1	s-2	s-3	mean	s-0	s-1	s-2	s-3	mean
ResNet101	FWB	16.98	17.98	20.96	28.85	21.19	19.13	21.46	23.93	30.08	23.65
	DAN	-	-	-	-	24.20	-	-	-	-	29.60
	PFENet	34.30	33.00	32.30	30.10	32.40	38.50	38.60	38.20	34.30	37.40
ResNet50	RPMMs	29.53	36.82	28.94	27.02	30.58	33.82	41.96	32.99	33.33	35.52
	ASGNet	**34.89**	**36.94**	**34.33**	**32.08**	**34.56**	**40.99**	**48.28**	**40.10**	**40.54**	**42.48**

Table 1. Comparison with state-of-the-arts on COCO-20^i with per-split results.

Figure 2. Qualitative visualization of baseline (single prototype learning) and the proposed ASGNet. On the left are examples from Pascal-5^i and the right ones are from COCO-20^i. Best viewed in color and zoom in.

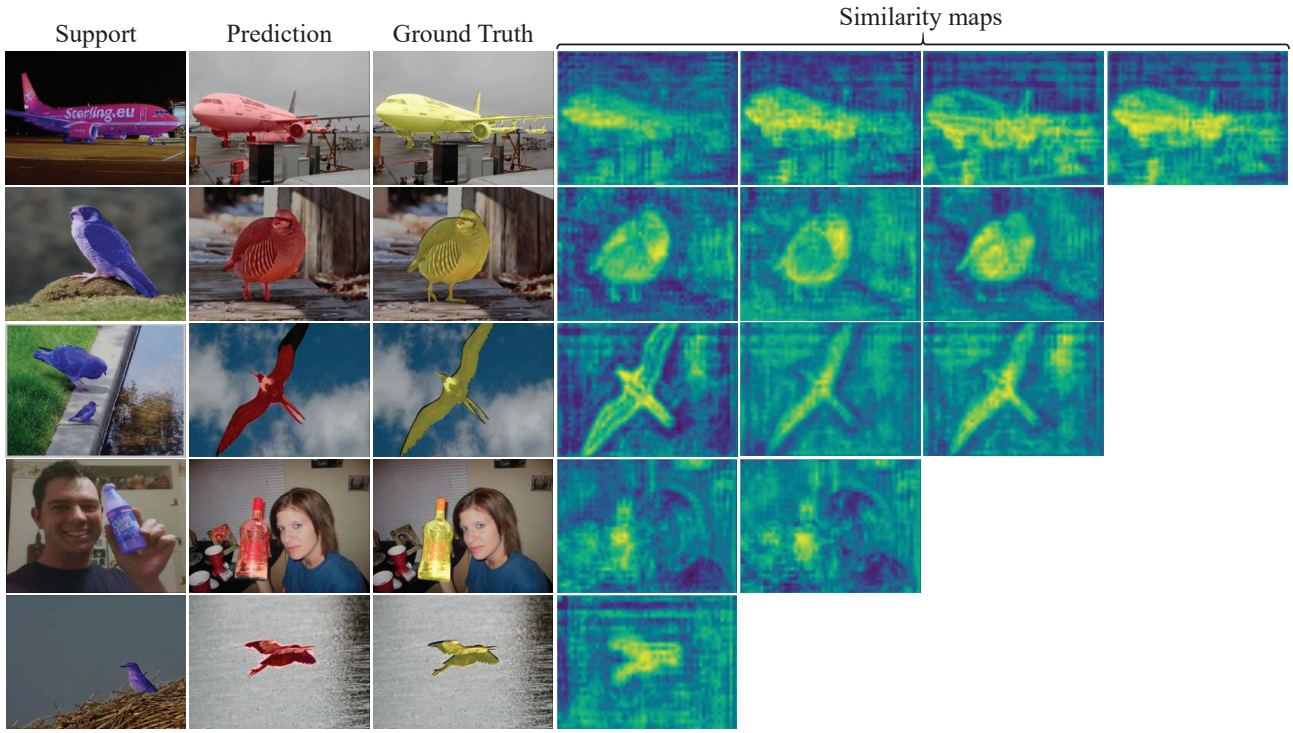

Figure 3. Visualization of similarity maps on Pascal-5^i. The number of prototypes is determined by the size of support object. Best viewed in color and zoom in.