PSEUDO-INTERACTING GUIDED NETWORK FOR FEW-SHOT SEGMENTATION

Xiaoliu Luo[1], *Jing Luo*[2], *Zhao Duan*[1], *Jin Tan*[1], *Taiping Zhang*[1*]

[1]Chongqing University, [2]Zhengzhou University

ABSTRACT

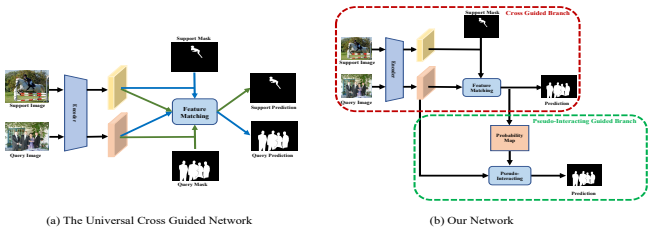
Few-shot segmentation has got a lot of concerns recently. Existing methods mainly locate and recognize the target object based on a cross-guided way that applies masked target object features of support(query) images to make a feature matching with query(support) images. However, there are some differences between support images and query images because of large appearance and scale variation, which will lead to inaccurate and incomplete segmentation. This problem inspired us to explore the local coherence of the image to guide the segmentation. We try to get some target pixels in the query image and apply these pixels to search for more target pixels in the query image. In this work, we propose a novel network that combines a universal cross-guided branch with a new pseudo-interacting guided branch. Specifically, we first employ the universal cross-guided branch to produce a pseudo-labeling that represents the probability of each pixel belonging to the target object. Then we design a pseudo-interacting guided branch, which applies some pixels with high probabilities based on generated pseudo-labeling to segment the target object in the query image and revises the results of the cross-guided branch simultaneously. Extensive experiments show that our approach outperforms state-of-the-art methods on both PASCAL-5^i and COCO-20^i datasets.

Index Terms— few-shot segmentation, semantic segmentation, few-shot learning

1. INTRODUCTION

Semantic segmentation has made substantial progress owing to advanced deep convolutional neural networks. Nevertheless, these approaches based deep on learning require a large amount of pixel-wise labeled images, which are expensive and laborious to obtain. And their performance drops sharply while dealing with unseen classes or insufficient labeled data. These issues impel the development of few-shot segmentation. Few-shot segmentation aims to segment images from the query set given few labeled samples from the support set, which requires the model trained with seen classes to generalize well on unseen classes with only a few labeled samples.

Current works for few-shot segmentation mainly make a feature matching between support and query images to seek the target object. And common methods for feature matching are prototypical learning and affinity learning. At present, both prototypical learning and affinity learning are only making a cross-guided feature matching between support and query images. As shown in Figure 1(a), they only apply the target information in support(query) images to make a feature matching with query(support) images, which can lead to inaccurate and incomplete segmentation because of the large appearance and scale variation between support and query images, just like in Figure 2. As we know, natural images have a local coherence,



(a) The Universal Cross Guided Network (b) Our Network

Fig. 1. Illustration of (a) the universal cross-guided network and (b) our network(PIGNet).

which inspired us to explore whether we can apply some target pixels to search more target regions. To this end, we have to face the following problems: (1) how to get some pixels belonging to the target object? (2) how to apply these selected pixels to search more target regions?

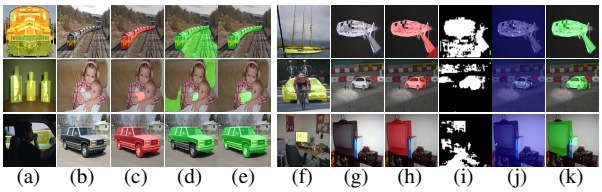
To solve the above problems, we propose a novel network that consists of two branches: the cross-guided branch and the pseudo-interacting guided branch, as shown in Figure 1(b). Although we can't directly get some pixels belonging to the target object in the query image, we find that most universal cross-guided methods can localize the most relevant regions according to the support image, even though not complete. So we attempt to get some target pixels from the most relevant regions based on the cross-guided branch. We produce a pseudo-labeling by a universal cross-guided branch, and we pick out some confident seed points among the pixels with high probabilities based on the pseudo-labeling.

Then we design a pseudo-interacting guided branch to apply these selected points to guide the segmentation of the query image. These selected seed points can serve as a location indication and global information guidance for the target object. We first design a position encoding module(PEM) to focus on the area around these selected seed points, because the area around these selected points is also part of the target object to a large extent. Next, the pseudo-labeling is also good prior information, but there will inevitably be some segmentation errors, especially at the edge of the object or in the area where the background is similar to the foreground, just like in Figure 2. Thus we introduce a mask encoding module(MEM) that can reduce the influence of wrong initial segmentation greatly. The whole network is end-to-end trainable. Extensive experiments show that our approach achieves new state-of-the-art performances both on PASCAL-5^i [1] and COCO-20^i [2]. The main contributions of this paper are summarized as follows:

- We propose a simple yet efficient network that combines a cross-guided branch with a pseudo-interacting guided branch. And results show that our approach is robust to image-pairs with large appearance and scale variations.
- This is the first work that employs the target information in the query image to locate the target object in the query image. And results show that this way can significantly improve the

*Corresponding Author.

This work was supported by the National Natural Science Foundation of China (No.62076043).



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k)

Fig. 2. The left samples are some segmentation results about appearance and scale variations. The right samples are with wrong initial segmentation. (a) and (f) are support images; (b) and (g) are query images; (c) and (h) are ground truth of query images; (d) are the results of PFENet; (e) and (k) are the results of our approach; (i) are the initial masks; (j) are the selected points on query images.

integrity of target segmentation and accuracy.

- Experiments on PASCAL-5^i and COCO-20^i demonstrate that our approach significantly outperforms baseline models and achieves new state-of-the-art performance.

2. RELATED WORK

2.1. Semantic Segmentation

Semantic segmentation that aims to make a pixel-wise prediction for an image has made great progress based on deep convolutional neural networks. Fully convolutional networks(FCNs) [3] that apply fully convolutional layers to replace fully connected layers make a basis for the behind work. And later works [4, 5, 6, 7, 8] have a significant improvement in performance by introducing some useful techniques, such as the pyramid pooling module [4], dilated convolution [5] and deformable convolution [6] and non-local module [7, 8]. However, these approaches require massive annotated data and trained models cannot generalize well to unseen classes during the inference stage.

2.2. Few-shot Segmentation

Few-shot segmentation was first proposed by Shaban *et al.* [1] that trained a linear classifier based on the support set and used this classifier to perform dense pixel-level prediction on a query image, which was improved in works [9, 10]. Later on, prototype-based methods aim to learn the prototypes based on the support set and then compute the similarity between prototypes and all pixels in the query image are proposed. SG-One [11] adopted masked average pooling to acquire the object-related prototype. PANet [12] proposed an alignment regularization to fully employ the information of support image. CANet [13] expanded the prototype to the same size of the query feature maps to make a dense comparison with the query feature maps. Besides, affinity-based methods that generate dense correspondence between support images and query images to maintain the spatial structure of image are proposed. These works mainly rely on attention mechanism. PGNet [14] established a correspondence between object parts by attention mechanisms. PFENet [15] generated a prior mask and then enriched it based on query features. However, all of them only consider recognizing and locating the target object through a cross-guided way, the local coherence of the target object is ignored.

3. PROBLEM SETTING

Suppose there are two image sets $\mathcal{D}_{train}$ with known classes and $\mathcal{D}_{test}$ with unseen classes, where $\mathcal{D}_{train} \cap \mathcal{D}_{test} = \emptyset$. We employ the episode paradigm, and each episode is comprised of a support set $\mathcal{S}$ and a query set $\mathcal{Q}$ from the same class c. For k-shot setting, $\mathcal{S} = \{(x_i^s, y_i^s)\}_{i=1}^K$ includes k images x^s and the corresponding binary masks y^s, and $\mathcal{Q} = (x^q, y^q)$ contains the query image x^q to be segmented and the corresponding ground truth mask y^q. We aim to learn a model $\mathcal{M}(x^q, S)$ that trained by episodically sampling support and query pairs from $\mathcal{D}_{train}$. While testing, given the inputs (x^q, S) from $\mathcal{D}_{test}$, we predict the query mask $\widetilde{y}^q$ based on the trained model.

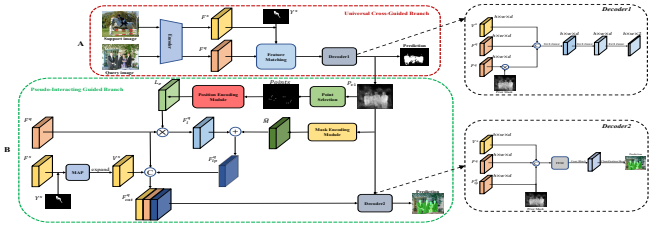


Fig. 3. Illustration of our network. Red dotted box A denotes the universal cross-guided branch and green dotted box B denotes the pseudo-interacting guided branch. *MAP* [11] denotes the masked average pooling.

4. OUR APPROACH

In this section, we first describe the cross-guided branch simply. Then we describe in detail the pseudo-interacting guided branch. The architecture of the two branches is shown in Figure 3.

4.1. Cross-Guided Branch

Although we can't directly get the target pixels from the query image, we find that most methods based on the cross-guided way can localize the most relevant regions according to the support image, so we try to get some target pixels through the cross-guided branch. We first make an initial segmentation through the cross-guided branch. Suppose the initial probability map is $P_c \in \mathcal{R}^{h \times w \times 2}$, which represents the probability of each pixel belonging to the background or target object. We try to pick out some confident seed pixels among the pixels with high probabilities in the target object channel and regard these selected seed points as the target object pixels. An auxiliary loss in the training phase is provided to facilitate the training of the cross-guided branch. The cross-guided branch can be accomplished by any universal cross-guided method.

4.2. Pseudo-Interacting Guided Branch

Below, we describe how to select some confident seed points and how to apply selected points to detect the target object. This branch is shown in Figure 3(B).

Point Selection. We directly select the confident seed points among the pixels with high probabilities in the target object probability map. With the probability map $P_c \in \mathcal{R}^{h \times w \times 2}$ from the cross-guided branch, suppose the probability map of each pixel belonging to the target object is $P_{c_1} \in \mathcal{R}^{h \times w}$. To ensure the selected points belonging to the target object as much as possible, we first set a threshold τ_1 to filter out pixels with low probabilities. Next, we observe that there are always some outliers with high probabilities. To eliminate the interference of outliers, we employ a mean filter to the probability map with high probabilities. And then we set the other threshold τ_2 to filter out outliers, which is formulated as follows:

$$\hat{P}_{c_1} = \{MeanFilter(\{P_{c_1}(i,j) >= \tau_1\}) >= \tau_2\} \quad (1)$$

, where $i \in \{0, ..., h-1\}, j \in \{0, ..., w-1\}$, and $MeanFilter$ denotes the mean filter operation, and we set the filter size is 3×3. Finally, we select some confident seed points from $\hat{P}_{c_1}$ randomly, we

Fig. 4. Illustration of point selection. *Initial Mask* represents the prediction from the cross-guided branch; *Probability Map* represents the probability of each pixel belonging to the target object; τ_1 and τ_2 are the corresponding threshold; *MeanFilter(3,3)* represents the operation of mean filter with the size 3×3; *Points* represents the randomly selected points.

use $Points = \{p_1, p_2, ..., p_n\}$ to denote selected points set, where n denotes the number of selected pixels. We visualize this process in Figure 4.

Position Encoding Module. Selected confident seed points can be regarded as a location indiction and global information guidance for the target object. To fully exploit the guidance information of these selected seed points, we propose a position encoding module to focus on the area around these selected seed points, where the area around these points is belonging to the target object to a large extent. Suppose the set $Points$ are the selected confident seed points. We try to encode their position information with a simple convolutional layer directly. Specifically, we first produce a position mask $L \in \mathcal{R}^{h \times w}$, where the value is one on the coordinates of all selected points and the value of all other points is zero. This can be is formulated as follows:

$$L(p_{l_x}, p_{l_y}) = 1, l = 1, ..., n \tag{2}$$

, where p_{l_x} and p_{l_y} denote the row and column coordinate of the selected point p_l, respectively. Then we employ a convolutional layer with kernel size 3×3 and channels d to L that aims to encode its position information to high dimensional space, suppose the encoded position maps is:

$$L_e = Conv_{3\times3}(L) \in \mathcal{R}^{h \times w \times d} \tag{3}$$

, where the area around these selected points can be expanded adaptively. Finally, suppose we have the transformed query feature maps $F^q \in \mathcal{R}^{h \times w \times d}$, we merge the position maps and query feature maps F^q by element-wise multiplication, which is formulated as follows:

$$F_l^q = F^q \times L_e \tag{4}$$

, where $F_l^q \in \mathcal{R}^{h \times w \times d}$ denotes the final position encoded maps.

Mask Encoding Module. The initial mask is good prior information, which indicates the target area to a certain degree. However, there will inevitably be some segmentation errors, especially at the edge of the object or in the area where the background is similar to the foreground, just like in Figure 2, the initial mask exists some errors. To exploit this prior information and reduce the influence of initial mask errors, we make an encoding for the initial mask. Based on the probability map P_{c_1}, we get a new mask $M \in \mathcal{R}^{h \times w}$ by filtering the probability map P_{c_1} with a threshold τ_3, which means $M\{P_{c_1} > \tau_3\} = 1$, others are set to zero. Then we also apply a convolutional layer with kernel size 3×3 and channels d to M that aims to encode the information of the initial mask to high dimensional space, suppose the encoded initial mask is:

$$\hat{M} = Conv_{3\times3}(M) \in \mathcal{R}^{h \times w \times d} \tag{5}$$

, which can revise the information of the initial mask adaptively. We concatenate the results of the position encoding module and mask encoding module together, which is formulated as follows:

$$F_{lp}^q = F_l^q + \hat{M} \tag{6}$$

Methods	*Backbone*	1-Shot						5-Shot						Params
		Fold-0	Fold-1	Fold-2	Fold-3	Mean	FB-IoU	Fold-0	Fold-1	Fold-2	Fold-3	Mean	FB-IoU	
OSLM [16]	VGG-16	33.6	55.3	40.9	33.5	40.8	61.3	35.9	58.1	44.1	33.9	41.4	61.5	272.6M
co-FCN [17]		36.7	50.6	44.9	32.4	41.1	60.1	37.5	50.0	44.1	33.9	41.4	60.2	34.2M
AMP [18]		41.9	50.2	46.7	34.7	43.4	62.2	41.8	55.5	50.3	39.9	46.9	63.8	15.8M
SG-One [11]		40.2	58.4	48.4	38.4	46.3	63.9	41.9	58.6	48.6	39.4	47.1	65.9	19.0M
FWB [2]		47.0	59.6	52.6	48.3	51.9	-	47.0	59.6	52.6	48.3	51.9	-	-
PANet [12]		42.3	58.0	51.1	41.2	48.1	66.5	51.8	64.6	59.8	46.5	55.7	70.7	14.7M
CRNet [19]		-	-	-	-	55.2	66.4	-	-	-	-	58.5	71.0	-
PFENet [15]		56.9	68.2	54.4	52.4	58.0	72.0	59.0	69.1	54.8	52.9	59.0	72.3	10.4M
CANet [13]	ResNet-50	52.5	65.9	51.3	51.9	55.4	66.2	55.5	67.8	51.9	53.2	57.1	69.6	19.0M
PGNet[14]		56.0	66.9	50.6	50.4	56.0	69.9	57.7	68.7	52.9	54.6	58.5	70.5	17.2M
CRNet [19]		-	-	-	-	55.7	66.8	-	-	-	-	58.8	71.5	-
BriNet [20]		56.54	67.2	51.56	53.02	57.08	-	-	-	-	-	-	-	-
RPMMs [21]		55.15	66.91	52.61	50.68	56.34	-	56.28	67.34	54.52	51.0	57.3	-	-
SimProgNet [22]		54.86	67.33	54.52	52.02	57.19	72.96	57.2	68.50	58.4	56.05	60.04	72.90	-
PPNet [23]		48.58	60.58	55.71	46.47	52.84	69.2	58.85	68.28	66.77	57.98	62.97	75.8	31.5M
PFENet [15]		61.7	69.5	55.4	56.3	60.8	73.3	63.1	70.7	55.8	57.9	61.9	73.9	10.8M
ASGNet [24]		58.84	67.86	56.79	53.66	59.29	69.2	63.66	70.55	64.17	57.38	63.94	74.2	10.4M
SCL [25]		63.0	70.0	56.5	57.7	61.8	71.9	64.5	70.9	57.3	58.7	62.9	72.8	-
DAN [26]	ResNet-101	54.7	68.6	**57.8**	51.6	58.2	71.9	57.9	69.0	60.1	54.9	60.5	72.3	-
FWB [2]		51.3	64.5	56.7	52.2	56.2	-	54.8	67.4	62.2	55.3	59.9	-	43.0M
PPNet [23]		52.71	62.82	57.38	47.74	55.16	70.9	60.25	70.00	**69.41**	60.72	65.1	**77.45**	50.5M
PFENet [15]		60.5	69.4	54.4	55.9	60.1	72.9	62.8	70.4	54.9	57.6	61.4	73.5	10.8M
ASGNet [24]		59.84	67.43	55.59	54.39	59.31	71.7	64.55	71.32	64.24	57.33	64.36	75.2	10.4M
Ours(PFENet)	ResNet-50	**63.7**	**70.7**	56.4	**60.3**	**62.8**	**73.7**	**67.9**	**72.8**	60.6	**66.1**	**66.9**	76.1	12.8M
Ours(ASGNet)		60.0	68.7	57.2	55.1	60.3	70.8	64.7	70.8	65.4	58.1	64.8	75.0	11.9M

Table 1. Results of class mIoU and FB-IoU on PASCAL-5^i under the 1-shot setting and 5-shot setting. The result of FB-IoU is the average of four folds. Params: number of learnable parameters. Best results in bold.

, where F_{lp}^q denotes the merged feature maps.

Finally, suppose $F^s \in \mathcal{R}^{h \times w \times d}$ is the transformed support feature maps, where d denotes the number of channels and $Y^s \in \mathcal{R}^{h \times w}$ is the corresponding support mask. We adopt the masked average pooling(MAP) [11] operation on the support feature maps to get a masked global average vector $v^s \in \mathcal{R}^{1 \times d}$, which is formulated as follows:

$$v^s = \frac{\sum_{x=0,y=0}^{h,w} Y_{x,y}^s * F_{x,y}^s}{\sum_{x=0,y=0}^{h,w} Y_{x,y}^s} \tag{7}$$

. Here, we expand the global average vector v^s to the same size with F^s to get the feature maps $V^s \in \mathcal{R}^{h \times w \times d}$. Then we concatenate F_{lp}^q with the query feature maps $F^q \in \mathcal{R}^{h \times w \times d}$ and the expanded feature maps V^s to get a new feature map F_{cat}^q, which is formulated as follows:

$$F_{cat}^q = Concat([F^q, F_{lp}^q, V^s]) \tag{8}$$

. Then we apply concatenated feature maps F_{cat}^q and initial mask to make a segmentation with the $Decoder$, we can get the final prediction P^q:

$$P^q = softmax(\mathcal{D}ecoder2(Concat(F_{cat}^q, P_{c1}))) \tag{9}$$

. In our work, we use the cross-entropy loss as the loss function.

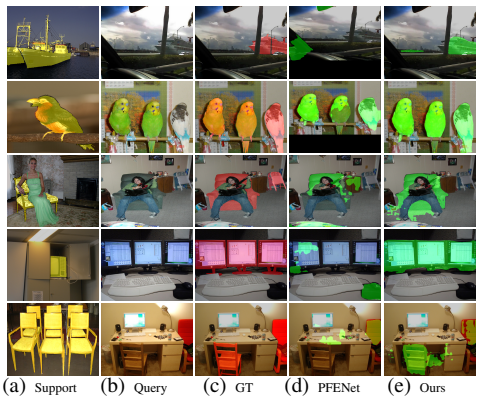
5. EXPERIMENTS

5.1. Implementation Details

To evaluate the performance of our network, we choose two baselines: PFENet and ASGNet. One is based on affinity learning and the other is based on prototype learning. In Figure 3, we illustrate the architectures of *Decoder1* and *Decoder2* while taking the PFENet as the baseline, where the prior mask, FEM, Conv Block, and Classification Head are the same as in PFENet. And while taking the ASGNet as the baseline, we only need to replace the V^s and the prior mask with the guide feature G^s and the probability map, respectively, where the guide feature G^s and the probability map are the same as in ASGNet. To reduce the numbers of the parameters, we set $d = 256$, and the transformed support(query) feature maps $F^s(F^q)$ are obtained by using 1×1 convolutional layer on the feature maps that obtained by concatenating the feature maps of layer2 and layer3 of ResNet50. All training settings are the same as that in the corresponding baseline. All experiments are completed with Nvidia Titan RTX GPUs.

Methods	*Backbone*	1-Shot		5-Shot	
		Mean	FB-IoU	Mean	FB-IoU
FWBF [2]	VGG-16	20.0	-	22.6	-
PANet [12]		20.9	59.2	29.7	63.5
PFENet [15]		34.1	60.0	37.7	61.6
BriNet [20]	ResNet-50	34.36	-	-	-
PPNet [23]		29.03	-	38.53	-
RPMMs [21]		30.58	-	35.52	-
ASGNet [24]		34.56	60.39	42.48	66.96
FWBF [2]	ResNet-101	21.2	-	23.7	-
PPNet [23]		21.19	-	23.05	-
DAN [26]		24.4	62.3	29.6	63.9
PFENet [15]		32.4	58.6	37.4	61.9
Ours(PFENet)	ResNet-50	**40.0**	**62.7**	**45.6**	**68.7**
Ours(ASGNet)		37.2	61.8	44.1	67.5

Table 2. Results on COCO-20^i under the 1-shot setting and 5-shot setting. Best results in bold.



(a) Support (b) Query (c) GT (d) PFENet (e) Ours

Fig. 5. Qualitative results of our approach on Pascal-5^i under the 1-shot setting. *GT* denotes the ground truth of query images; *PFENet* denotes the prediction of PFENet; *Ours* denotes the prediction of our approach based on the PFENet.

5.2. Datasets and Evaluation metrics

We choose PASCAL-5^i [16] and COCO-20^i [27] to verify our approach. For PASCAL-5^i, following [1], 1000 support-query pairs are randomly sampled in the test split. For COCO-20^i, following [2], we divide 80 classes into 4 folds and each fold contains 20 classes to evaluate our model. Two evaluation metrics are used in our work. One is proposed in [1] that measures the per-class foreground Intersection-over-Union(IoU), and the average IoU over all classes(mIoU) is used to report the results. The other is used in [17] that computes the mean of foreground IoU and background IoU over all test images(FB-IoU).

5.3. Comparison to State-of-the-art

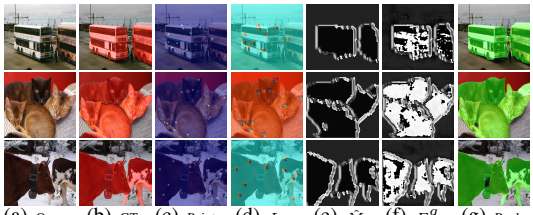
In Table 1, our approach achieves new state-of-the-art performances on PASCAL-5^i under 1-shot and 5-shot tasks with few trainable parameters increased. In Table 2, we present the performance comparison of mean IoU and FB-IoU on COCO-20^i. Our approach outperforms two baselines with a considerable margin of 5.9% and 2.6% in 1-shot segmentation, respectively. For the 5-shot segmentation task, our approach outperforms two baselines with a considerable margin of 7.9% and 1.6%, respectively. Besides, we show some qualitative results of our approach while taking the PFENet as the baseline in Figure 5.

5.4. Ablation study

We conduct ablation experiments on the PASCAL-5^i dataset using the PFENet as the baseline. Here we propose two ways to get the initial mask. One is with training, and the other is no training. Specifically, the initial segmentation obtained by training is described above, where the *Decoder1* is used to produce the initial mask. And the way without training is directly taking the prior mask [15] as the initial mask. Besides, we propose another simple prototype-based way to build the pseudo-interacting guide branch and we use *Proto* to denote it. Specifically, we directly take the initial mask filtered with a threshold $\tau_3 = 0.7$ as a pseudo-labeling, then we adopt the masked average pooling(MAP) [11] operation on the query feature maps F^q to get a prototype $v^q \in \mathcal{R}^{1\times d}$ and expand it to the same size with F^q to get the feature maps $V^q \in \mathcal{R}^{h\times w\times d}$, just like getting the V^s. Finally, we replace F^q_{lp} with V^q to make the segmentation. For all experiments, we set two thresholds: $\tau_1 = 0.8$ and $\tau_2 = 0.6$ to filter some points with low probabilities and outliers. Two thresholds are fixed and all points are randomly selected and repeatable. And considering that each class may include multiple objects in an image, we fix the number of points $n = 10$, where we hope selected points can locate all objects of some class in an image as possible. Table 3 studies the influence of our proposed PEM and MEM with/without a trainable initial mask, where the *Base* denotes the baseline PFENet results directly obtained from the original paper. As we can see, our approach can improve the performance of the baseline under two settings. We visualize each module feature in the pseudo-interacting guided branch in Figure 6. It clearly shows that the PEM determines the location of the object, the MEM determines the boundary of each object and the addition of both determines the target object.

Shot	Methods				Training		No-training	
	Base	Proto	PEM	MEM	Mean	FB-IoU	Mean	FB-IoU
1	✓				60.8	73.3	60.8	**73.3**
1	✓	✓			62.4	72.5	**62.3**	72.8
1	✓		✓		61.6	71.9	60.9	70.4
1	✓			✓	62.1	73.0	61.3	73.1
1	✓		✓	✓	**62.8**	**73.7**	61.7	72.9
5	✓				61.9	73.9	61.9	73.9
5	✓	✓			63.5	74.0	63.1	73.7
5	✓		✓		65.4	74.7	62.1	73.3
5	✓			✓	66.0	75.2	63.4	73.8
5	✓		✓	✓	**66.9**	**76.1**	**64.0**	**74.5**

Table 3. Ablation study of our proposed PEM and MEM with/without a trainable initial mask on PASCAL-5^i in the 1-shot setting and 5-shot setting.



(a) Query (b) GT (c) Points (d) L_e (e) $\hat{M}$ (f) F^q_{lp} (g) Pred

Fig. 6. Visualization of the pseudo-interacting branch. *GT* represents the ground truth; *Points* represents the selected points showed on the query image; L_e, $\hat{M}$ and F^q_{lp} are described in the pseudo-interacting branch; *Pred* represents the prediction with our approach.

6. CONCLUSION

In this paper, we propose a pseudo-interacting guided network for few-shot segmentation that combines the universal cross-guided branch with the pseudo-interacting guided branch together in an end-to-end way. We adopt the features of the target object in the query image to exploit more target regions. Extensive experiments have demonstrated the superiority of PIGNet and our approach achieves state-of-the-art performance on PASCAL-5^i and COCO-20^i.

7. REFERENCES

[1] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots, "One-shot learning for semantic segmentation," *CoRR*, vol. abs/1709.03410, 2017.

[2] Khoi Nguyen and Sinisa Todorovic, "Feature weighting and boosting for few-shot segmentation," *International Conference on Computer Vision*, pp. 622–631, 2019.

[3] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2015.

[4] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, "Pyramid scene parsing network," *Computer Vision and Pattern Recognition*, pp. 6230–6239, 2017.

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Computer Vision – ECCV 2018*, Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, Eds., Cham, 2018, pp. 833–851, Springer International Publishing.

[6] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei, "Deformable convolutional networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 764–773.

[7] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He, "Non-local neural networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7794–7803.

[8] Zhen Zhu, Mengdu Xu, Song Bai, Tengteng Huang, and Xiang Bai, "Asymmetric non-local neural networks for semantic segmentation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 593–602.

[9] Mennatullah Siam, Boris N. Oreshkin, and Martin Jagersand, "Amp: Adaptive masked proxies for few-shot segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[10] Lizhao Liu, Junyi Cao, Minqian Liu, Yong Guo, Qi Chen, and Mingkui Tan, "Dynamic extension nets for few-shot semantic segmentation," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020.

[11] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Thomas S Huang, "Sg-one: Similarity guidance network for one-shot semantic segmentation.," *arXiv: Computer Vision and Pattern Recognition*, 2018.

[12] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng, "Panet: Few-shot image semantic segmentation with prototype alignment," *International Conference on Computer Vision*, pp. 9197–9206, 2019.

[13] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen, "Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning," *Computer Vision and Pattern Recognition*, pp. 5217–5226, 2019.

[14] 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," *International Conference on Computer Vision*, pp. 9587–9595, 2019.

[15] Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia, "Prior guided feature enrichment network for few-shot segmentation," *TPAMI*, 2020.

[16] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots, "One-shot learning for semantic segmentation," *Computer Vision and Pattern Recognition*, 2017.

[17] Kate Rakelly, Evan Shelhamer, Trevor Darrell, Alyosha Efros, and Sergey Levine, "Conditional networks for few-shot semantic segmentation," *International Conference on Learning Representations*, 2018.

[18] Boris Oreshkin, "ADAPTIVE MASKED WEIGHT IMPRINTING FOR FEW-SHOT SEGMENTATION," *In International Conference on Learning Representations (ICLR)*, 2019.

[19] Weide Liu, Chi Zhang, Guosheng Lin, and Fayao Liu, "Crnet: Cross-reference networks for few-shot segmentation," *Computer Vision and Pattern Recognition*, 2020.

[20] Xianghui Yang, Bairun Wang, Kaige Chen, Xinchi Zhou, Shuai Yi, Wanli Ouyang, and Luping Zhou, "Brinet: Towards bridging the intra-class and inter-class gaps in one-shot segmentation," 2020.

[21] Boyu Yang, Chang Liu, Bohao Li, Jianbin Jiao, and Ye Qixiang, "Prototype mixture models for few-shot semantic segmentation," in *ECCV*, 2020.

[22] Siddhartha Gairola, Mayur Hemani, Ayush Chopra, and Balaji Krishnamurthy, "Simpropnet: Improved similarity propagation for few-shot image segmentation," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, Christian Bessiere, Ed. 7 2020, pp. 573–579, International Joint Conferences on Artificial Intelligence Organization, Main track.

[23] Yongfei Liu, Xiangyi Zhang, Songyang Zhang, and Xuming He, "Part-aware prototype network for few-shot semantic segmentation," *ECCV*, 2020.

[24] Gen Li, Varun Jampani, Laura Sevilla-Lara, Deqing Sun, Jonghyun Kim, and Joongkyu Kim, "Adaptive prototype learning and allocation for few-shot segmentation," in *CVPR*, 2021.

[25] Bingfeng Zhang, Jimin Xiao, and Terry Qin, "Self-guided and cross-guided learning for few-shot segmentation," *CoRR*, vol. abs/2103.16129, 2021.

[26] Haochen Wang, Xudong Zhang, Yutao Hu, Yandan Yang, Xianbin Cao, and Xiantong Zhen, "Few-Shot Semantic Segmentation with Democratic Attention Networks," *ECCV*, 2020.

[27] 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 *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Eds., Cham, 2014, pp. 740–755, Springer International Publishing.