Prior Guided Feature Enrichment Network for Few-Shot Segmentation

Zhuotao Tian, *Student Member, IEEE*, Hengshuang Zhao, *Member, IEEE*, Michelle Shu, *Student Member, IEEE*, Zhicheng Yang, *Member, IEEE*, Ruiyu Li, *Member, IEEE*, and Jiaya Jia, *Fellow, IEEE*

Abstract—State-of-the-art semantic segmentation methods require sufficient labeled data to achieve good results and hardly work on unseen classes without fine-tuning. Few-shot segmentation is thus proposed to tackle this problem by learning a model that quickly adapts to new classes with a few labeled support samples. Theses frameworks still face the challenge of generalization ability reduction on unseen classes due to inappropriate use of high-level semantic information of training classes and spatial inconsistency between query and support targets. To alleviate these issues, we propose the Prior Guided Feature Enrichment Network (PFENet). It consists of novel designs of (1) a training-free prior mask generation method that not only retains generalization power but also improves model performance and (2) Feature Enrichment Module (FEM) that overcomes spatial inconsistency by adaptively enriching query features with support features and prior masks. Extensive experiments on PASCAL-5^i and COCO prove that the proposed prior generation method and FEM both improve the baseline method significantly. Our PFENet also outperforms state-of-the-art methods by a large margin without efficiency loss. It is surprising that our model even generalizes to cases without labeled support samples.

Index Terms—Few-shot segmentation, few-shot learning, semantic segmentation, scene understanding

1 Introduction

Rapid development of deep learning has brought significant improvement to semantic segmentation. The iconic frameworks [2], [52] have profited a wide range of applications of automatic driving, robot vision, medical image, etc. The performance of these frameworks, however, worsens quickly without sufficient fully-labeled data or when working on unseen classes. Even if additional data is provided, fine-tuning is still time- and resource-consuming.

To address this issue, few-shot segmentation was proposed [29] where data is divided into a support set and a query set. As shown in Fig. 1, images from both support and query sets are first sent to the backbone network to extract features. Feature processing can be accomplished by generating weights for the classifier [29], cosine-similarity calculation [4], [40], or convolutions [13], [46] to generate the final prediction.

The support set provides information about the target class that helps the model to make accurate segmentation prediction on the query images. This process mimics the scenario where a model makes the prediction of unseen classes on testing images (query) with few labeled data (support). Therefore, a few-shot model needs to quickly adapt to the new classes. However, the common problems of existing few-shot segmentation methods include generalization loss due to misuse of high-level features and spatial inconsistency between the query and support samples. In this paper, we mainly tackle these two difficulties.

Generalization Reduction & High-Level Features. Common semantic segmentation models rely heavily on high-level features with semantic information. Experiments of CANet [46] show that simply adding high-level features during feature processing in a few-shot model causes performance drop. Thus the way to utilize semantic information in the few-shot setting is not straightforward. Unlike previous methods, we use ImageNet [28] pre-trained high-level features of the query and support images to produce 'priors' for the model. These priors help the model to better identify targets in query images. Since the prior generation process is training-free, the resulting model does not lose the generalization ability to unseen classes, despite the frequent use of high-level information of seen classes during training.

Spatial Inconsistency. Besides, due to the limited samples, scale and pose of each support object may vary greatly from its query target, which we call spatial inconsistency. To tackle this problem, we propose a new module named Feature Enrichment Module (FEM) to adaptively enrich query features with the support features. Ablation study in Section 4.3 shows that merely incorporating the multi-scale scheme to tackle the spatial inconsistency is sub-optimal by showing that FEM provides conditioned feature selection that helps retain essential information passed across different scales. FEM achieves superior performance than other multi-scale

- *Zhuotao Tian, Hengshuang Zhao, and Jiaya Jia are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. E-mail: {tianzhuotao, hengshuangzhao}@gmail.com, leojia@cse.cuhk.edu.hk.*
- *Michelle Shu is with Johns Hopkins University, Baltimore, MD 21218 USA. E-mail: mshu1@jhu.edu.*
- *Zhicheng Yang and Ruiyu Li are with SmartMore, Shenzhen, Guangdong 518057, China. E-mail: {cosnozc, royliruiyu}@gmail.com.*

Manuscript received 21 Apr. 2020; revised 14 July 2020; accepted 28 July 2020. Date of publication 3 Aug. 2020; date of current version 7 Jan. 2022. (Corresponding author: Hengshuang Zhao.) Recommended for acceptance by J. Wang. Digital Object Identifier no. 10.1109/TPAMI.2020.3013717

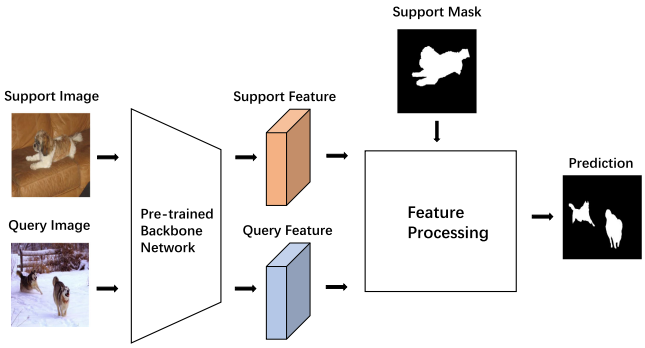


Fig. 1. Summary of recent few-shot segmentation frameworks. The backbone method used to extract support and query features can be either a single shared network or two Siamese networks.

structures, such as HRNet [39], PPM [52], ASPP [3] and GAU [45].

Finally, based on the proposed prior generation method and Feature Enrichment Module, we build a new network–Prior Guided Feature Enrichment Network (PFENet). The ResNet-50 based PFENet only contains 10.8 M learnable parameters, and yet achieves new state-of-the-art results on both PASCAL-5^i [29] and COCO [19] benchmark with 15.9 and 5.1 FPS with 1-shot and 5-shot settings respectively. Moreover, we manifest the effectiveness by applying our model to the zero-shot scenario where no labeled data is available. The result is surprising–PFENet sill achieves decent performance without major structural modification.

Our contribution in this paper is threefold:

- We leverage high-level features and propose training-free prior generation to greatly improve prediction accuracy and retain high generalization.
- By incorporating the support feature and prior information, our FEM helps adaptively refine the query feature with the conditioned inter-scale information interaction.
- PFENet achieves new state-of-the-art results on both PASCAL-5^i and COCO datasets without compromising efficiency.

2 Related Work

2.1 Semantic Segmentation

Semantic segmentation is a fundamental topic to predict the label for each pixel. The Fully Convolutional Network (FCN) [30] is developed for semantic segmentation by replacing the fully-connected layer in a classification framework with convolutional layers. Following approaches, such as DeepLab [2], DPN [21] and CRF-RNN [54], utilize CRF/MRF to help refine coarse prediction. The receptive field is important for semantic segmentation; thus DeepLab [2] and Dilation [43] introduce the dilated convolution to enlarge the receptive field. Encoder-decoder structures [8], [18], [27] are adopted to help reconstruct and refine segmentation in steps.

Contextual information is vital for complex scene understanding. ParseNet [20] applies global pooling for semantic segmentation. PSPNet [52] utilizes a Pyramid Pooling Module (PPM) for context information aggregation over different regions, which is very effective. DeepLab [2] develops atrous spatial pyramid pooling (ASPP) with filters in different dilation rates. Attention models are also introduced. PSANet [53] develops point-wise spatial attention with a bi-directional information propagation paradigm. Channel-wise attention [47] and non-local style attention [7], [14], [44], [48] are also effective for segmentation. These methods work well on large-sample classes. They are not designed to deal with rare and unseen classes. They also cannot be easily adapted without fine-tuning.

2.2 Few-Shot Learning

Few-shot learning aims at image classification when only a few training examples are available. There are meta-learning based methods [1], [6], [9] and metric-learning ones [33], [36], [38]. Data is essential to deep models; therefore, several methods improve performance by synthesizing more training samples [11], [42], [49]. Different from few-shot learning where prediction is at the image-level, few-shot segmentation makes pixel-level predictions, which is much more challenging.

Our work closely relates to metric-learning based few-shot learning methods. Prototypical network [33] is trained to map input data to a metric space where classes are represented as prototypes. During inference, classification is achieved by finding the closest prototype for each input image, because data belonging to the same class should be close to the prototype. Another representative metric-based work is the relation network [36] that projects query and support images to 1×1 vectors and then performs classification based on the cosine similarity between them.

2.3 Few-Shot Segmentation

Few-shot segmentation places the general semantic segmentation in a few-shot scenario, where models perform dense pixel labeling on new classes with only a few support samples. OSLSM [29] first tackles few-shot segmentation by learning to generate weights of the classifier for each class. PL [4] applies prototyping [33] to the segmentation task. It learns a prototype for each class and calculates the cosine similarity between pixels and prototypes to make the prediction. More recently, PANet [40] introduces prototype alignment regularization that encourages the model to learn consistent embedding prototypes for better performance, and CANet [46] uses the iterative optimization module on the merged query and support feature to iteratively refine results.

Similar to CANet [46], we use convolution to replace the cosine similarity that may not well tackle complex pixel-wise classification in the segmentation task. However, different from CANet, our baseline model uses fewer convolution operations and still achieves decent performance.

As discussed before, these few-shot segmentation methods do not sufficiently consider generalization loss and spatial inconsistency. Unlike PGNet [45] that uses a graph-based pyramid structure to refine results via Graph Attention Unit (GAU) followed by three residual blocks and an ASPP [3], we instead incorporate a few basic convolution operations with the proposed prior masks and FEM in a multi-scale structure to accomplish decent performance.

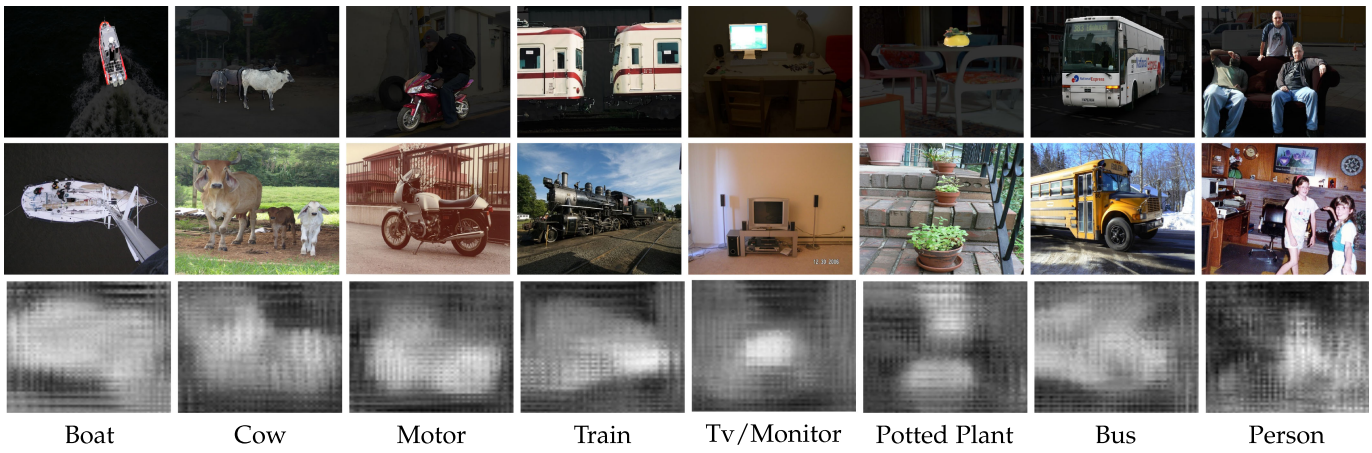


Fig. 2. Illustration of the training-free prior generation. *Top*: support images with the masked area in the target class. *Middle*: query images. *Bottom*: prior masks of query images where the regions of interest are highlighted.

3 Our Method

In this section, we first briefly describe the few-shot segmentation task in Section 3.1. Then, we present the prior generation method and the Feature Enrichment Module in Sections 3.2 and 3.3 respectively. Finally, in Section 3.4, details of our proposed Prior Guided Feature Enrichment Network are discussed.

3.1 Task Description

A few-shot semantic segmentation system has two sets, i.e., the query set Q and support set S. Given K samples from support set S, the goal is to segment the area of unseen class C_{test} from each query image I_Q in the query set.

Models are trained on classes C_{train} (base) and tested on previously unseen classes C_{test} (novel) in episodes ($C_{train} \cap C_{test} = \varnothing$). The episode paradigm was proposed in [38] and was first applied to few-shot segmentation in [29]. Each episode is formed by a support set S and a query set Q of the same class c. The support set S consists of K samples $S = \{S_1, S_2, \ldots, S_K\}$ of class c, which we call 'K-shot scenario'. The ith support sample S_i is a pair of $\{I_{S_i}, M_{S_i}\}$ where I_{S_i} and M_{S_i} are the support image and label of c respectively. For the query set, $Q = \{I_Q, M_Q\}$ where I_Q is the input query image and M_Q is the ground truth mask of class c. The query-support pair $\{I_Q, S\} = \{I_Q, I_{S_1}, M_{S_1}, I_{S_2}, M_{S_2}, \ldots, I_{S_K}, M_{S_K}\}$ forms the input data batch to the model. The ground truth M_Q of the query image is invisible to the model and is used to evaluate the prediction on the query image in each episode.

3.2 Prior for Few-Shot Segmentation

3.2.1 Important Observations

CANet [46] outperforms previous work by a large margin on the benchmark PASCAL-5^i dataset by extracting only middle-level features from the backbone (e.g., conv3_x and conv4_x of ResNet-50). Experiments in CANet also show that the high-level (e.g., conv5_x of ResNet-50) features lead to performance reduction. It is explained in [46] that the middle-level feature performs better since it constitutes object parts shared by unseen classes, but our alternative explanation is that the *semantic information contained in the high-level feature is more class-specific than the middle-level feature*, indicating that the former is more likely to negatively affect model's generalization power to unseen classes. In addition, higher-level feature directly provides semantic information of the training classes C_{train}, contributing more in identifying pixels belonging to C_{train} and reducing the training loss than the middle-level information. Consequently, such behavior results in a preference for C_{train}. The lack of generalization and the preference for the training classes are both harmful for evaluation on unseen test classes C_{test}.

It is noteworthy that contrary to the finding that high-level feature adversely affects performance in few-shot segmentation, prior segmentation frameworks [27], [51] exploit these features to provide semantic cues for final prediction. This contradiction motivates us to find a way to make use of high-level information in a training-class-insensitive way to boost performance in few-shot segmentation.

3.2.2 Prior Generation

In our work, we transform the ImageNet [28] pre-trained high-level feature containing semantic information into a prior mask that tells the probability of pixels belonging to a target class as shown in Fig. 2. During training, the backbone parameters are fixed as those in [40], [46]. Therefore, the prior generation process does not bias towards training classes C_{train} and upholds class-insensitivity during the evaluation on unseen test classes C_{test}. Let I_Q, I_S denote the input query and support images, M_S denote the binary support mask, $\mathcal{F}$ denote the backbone network, and X_Q, X_S denote the high-level query and support features. We have

$$X_Q = \mathcal{F}(I_Q), \quad X_S = \mathcal{F}(I_S) \odot M_S, \tag{1}$$

where $\odot$ is the Hadamard product – the sizes of X_Q and X_S are both $[h, w, c]$. Note that the output of $\mathcal{F}$ is processed with a ReLU function. So the binary support mask M_S removes the background in support feature by setting it to zero.

Specifically, we define the prior Y_Q of query feature X_Q as the mask that reveals the pixel-wise correspondence between X_Q and X_S. A pixel of query feature X_Q with a high value on Y_Q means that this pixel has a high correspondence with at least one pixel in support feature. Thus, it is very likely to be in the target area of the query image. By setting the background on support feature to zero, pixels of

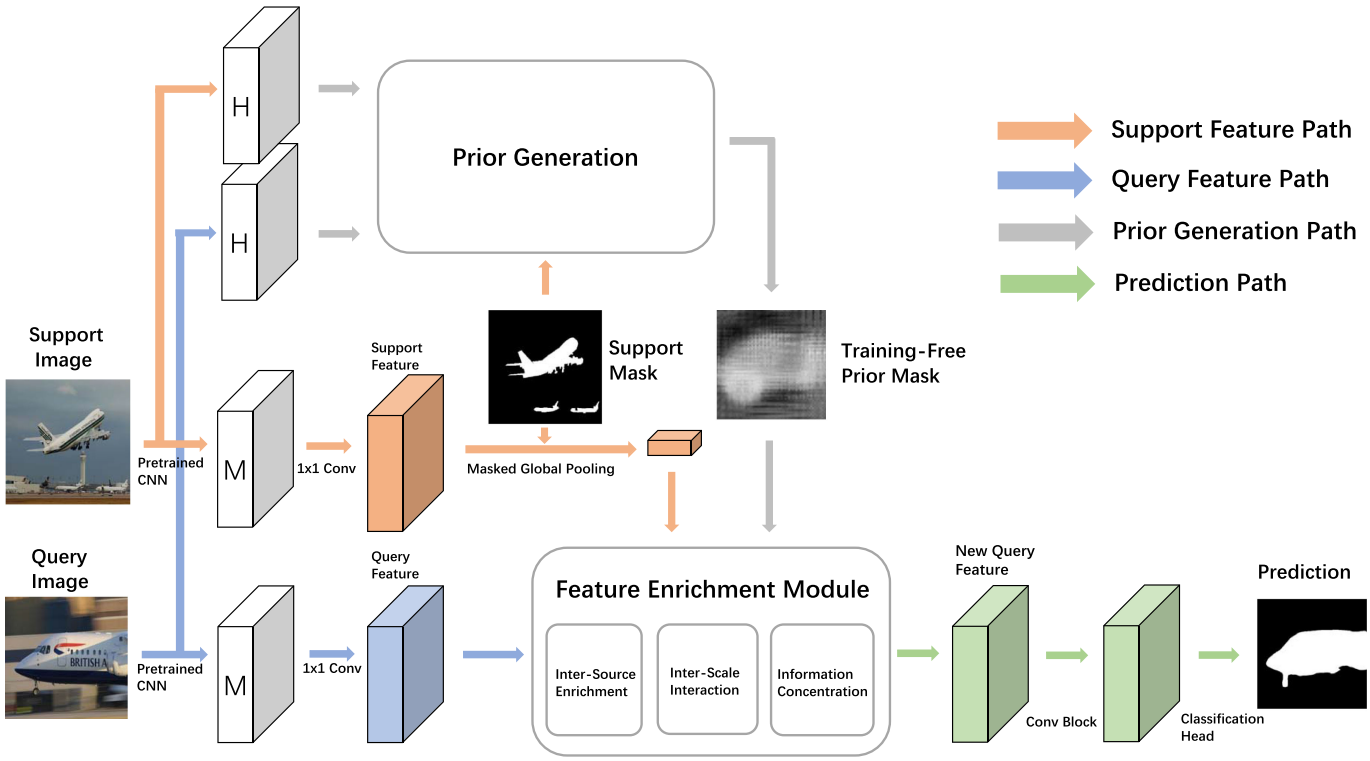


Fig. 3. Overview of our Prior Guided Feature Enrichment Network with the prior generation and Feature Enrichment Module. White blocks marked with H and M represent the high- and middle-level features extracted from backbone respectively.

query feature yield no correspondence with the background on support feature–they only correlate with the foreground target area. To generate Y_Q, we first calculate the pixel-wise cosine similarity $cos(x_q, x_s) \in \mathbb{R}$ between feature vectors of $x_q \in X_Q$ and $x_s \in X_S$ as

$$cos(x_q, x_s) = \frac{x_q^T x_s}{\|x_q\| \|x_s\|} \quad q, s \in \{1, 2, \ldots, hw\}. \tag{2}$$

For each $x_q \in X_Q$, we take the maximum similarity among all support pixels as the correspondence value $c_q \in \mathbb{R}$ as

$$c_q = \max_{s \in \{1,2,\ldots,hw\}} (cos(x_q, x_s)), \tag{3}$$

$$C_Q = [c_1, c_2, \ldots, c_{hw}] \in \mathbb{R}^{hw \times 1}. \tag{4}$$

Then we produce the prior mask Y_Q by reshaping $C_Q \in \mathbb{R}^{hw \times 1}$ into $Y_Q \in \mathbb{R}^{h \times w \times 1}$. We process Y_Q with a min-max normalization (Eq. (5)) to normalize the values to between 0 and 1, as shown in Fig. 2. In Eq. (5), ϵ is set to $1e-7$ in our experiments.

$$Y_Q = \frac{Y_Q - \min(Y_Q)}{\max(Y_Q) - \min(Y_Q) + \epsilon}. \tag{5}$$

The key point of our proposed prior generation method lies in the use of fixed high-level features to yield the prior mask by taking the maximum value from a similarity matrix of size $hw \times hw$ as given in Eqs. (2) and (3), which is rather simple and effective. Ablation study comparing other alternative methods used in [24], [40], [50] in Section 4.4 demonstrates the superiority of our method.

3.3 Feature Enrichment Module

3.3.1 Motivation

Existing few-shot segmentation frameworks [4], [13], [24], [26], [29], [31], [40], [46] use masked global average pooling for extracting class vectors from support images before further processing. However, global pooling on support images results in spatial information inconsistency since the area of query target may be much larger or smaller than support samples. Therefore, using a global pooled support feature to directly match each pixel of the query feature is not ideal.

A natural alternative is to add PPM [52] or ASPP [3] to provide multi-level spatial information to the feature. PPM and ASPP help the baseline model yield better performance (as demonstrated in our later experiments). However, these two modules are suboptimal in that: 1) they provide spatial information to merged features without specific refinement process within each scale; 2) the hierarchical relations across different scales are ignored.

To alleviate these issues, we disentangle the multi-scale structure and propose the feature enrichment module to 1) horizontally interact the query feature with the support features and prior masks in each scale, and 2) vertically leverage the hierarchical relations to enrich coarse feature maps with essential information extracted from the finer feature via a top-down information path. After horizontal and vertical optimization, features projected into different scales are then collected to form the new query feature. Details of FEM are as follows.

3.3.2 Module Structure

As shown in Fig. 3, the feature enrichment module takes the query feature, prior mask and support feature as input. It outputs the refined query feature with enriched information

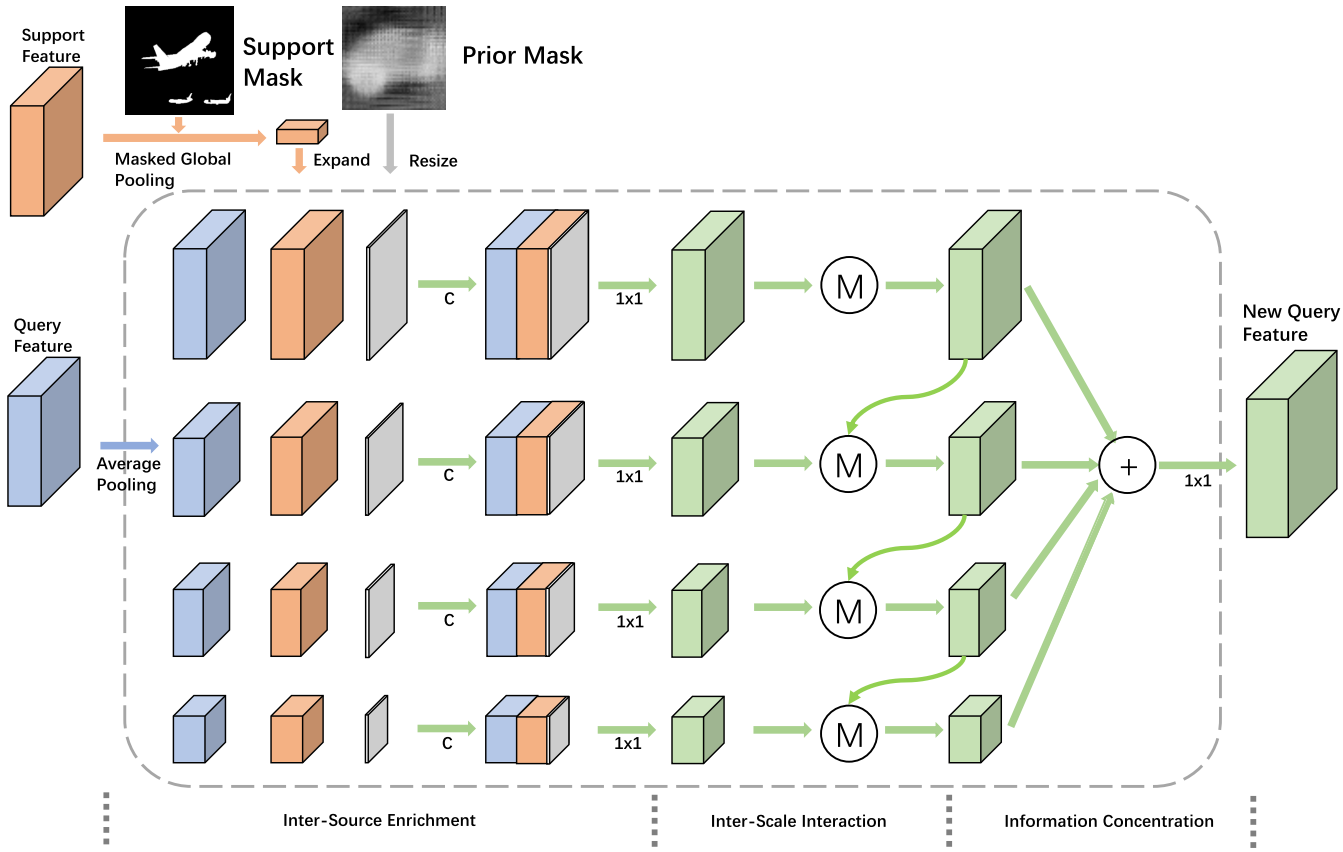


Fig. 4. Visual illustration of FEM (dashed box) with four scales and a top-down path. C, 1 x 1 and Circled M represent concatenation, 1×1 convolution and inter-scale merging module respectively. Activation functions are ReLU.

from the support feature. The enrichment process can be divided into three sub-processes of 1) inter-source enrichment that first projects input to different scales and then interacts the query feature with support feature and prior mask in each scale independently; 2) inter-scale interaction that selectively passes essential information between merged query-support features across different scales; and 3) information concentration that merges features in different scales to finally yield the refined query feature. An illustration of FEM with four scales and a top-down path for inter-scale interaction is shown in Fig. 4.

Inter-Source Enrichment. In FEM, $B = [B^1, B^2, \ldots, B^n]$ denotes n different spatial sizes for average pooling. They are in the descending order $B^1 > B^2 > \ldots > B^n$. The input query feature $X_Q \in \mathbb{R}^{h \times w \times c}$ is first processed with adaptive average pooling to generate n sub-query features $X_Q^{FEM} = [X_Q^1, X_Q^2, \ldots, X_Q^n]$ of n different spatial sizes $X_Q^i \in \mathbb{R}^{B^i \times B^i \times c}$. n spatial sizes make the global-average pooled support feature $X_S \in \mathbb{R}^{1 \times 1 \times c}$ be expanded to different n feature maps $X_S^{FEM} = [X_S^1, X_S^2, \ldots, X_S^n]$ ($X_S^i \in \mathbb{R}^{B^i \times B^i \times c}$), and the prior $Y_Q \in \mathbb{R}^{h \times w \times 1}$ is accordingly resized to $Y_Q^{FEM} = [Y_Q^1, Y_Q^2, \ldots, Y_Q^n]$ ($Y_Q^i \in \mathbb{R}^{B^i \times B^i \times 1}$).

Then, for $i \in \{1, 2, \ldots, n\}$, we concatenate X_Q^i, X_S^i and Y_Q^i, and process each concatenated feature with convolutions to generate the merged query features $X_{Q,m}^i \in \mathbb{R}^{B^i \times B^i \times c}$ as

$$X_{Q,m}^i = \mathcal{F}_{1\times1}(X_Q^i \oplus X_S^i \oplus Y_Q^i), \tag{6}$$

where $\mathcal{F}_{1\times1}$ represents the 1×1 convolution that yields the merged feature with $c = 256$ output channels.

Inter-Scale Interaction. It is worth noting that tiny objects may not exist in the down-sampled feature maps. A top-down path adaptively passing information from finer features to the coarse ones is conducive to building a hierarchical relationship within our feature enrichment module. Now the interaction is between not only the query and support features in each scale (horizontal), but also the merged features of different scales (vertical), which is beneficial to the overall performance.

The circled M in Fig. 4 represents the inter-scale merging module $\mathcal{M}$ that interacts between different scales by selectively passing useful information from the auxiliary feature to the main feature to generate the refined feature $X_{Q,new}^i$. This process can be written as

$$X_{Q,new}^i = \mathcal{M}(X_{Q,m}^{Main,i}, X_{Q,m}^{Aux,i}), \tag{7}$$

where $X_{Q,m}^{Main,i}$ is the main feature and $X_{Q,m}^{Aux,i}$ is the auxiliary feature for the ith scale B^i. For example, in an FEM with a top-down path for inter-scale interaction, finer feature (auxiliary) $X_{Q,m}^{i-1}$ needs to provide additional information to the coarse feature (main) $X_{Q,m}^i (B^{i-1} > B^i, i \geqslant 2)$. In this case, $X_{Q,m}^{Aux,i} = X_{Q,m}^{i-1}$ and $X_{Q,m}^{Main,i} = X_{Q,m}^i$. Other alternatives for inter-scale interaction include the bottom-up path that enriches finer features (main) with information coming from the coarse ones (auxiliary), and the bi-directional variants, i.e., a top-down path followed by a bottom-up path, and a bottom-up path followed by a top-down path. The top-down path shows its superiority in Section 4.3.1.

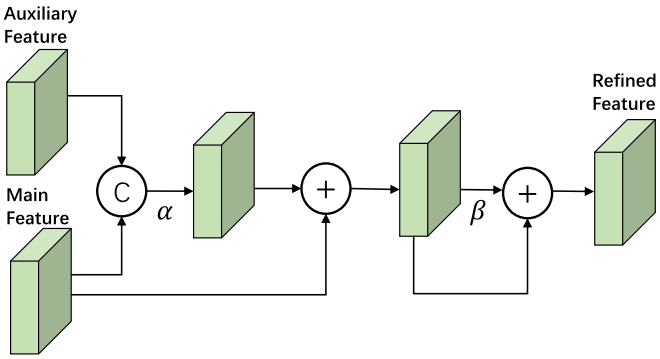


Fig. 5. Visual illustration of the inter-scale merging module $\mathcal{M}$. C is concatenation and $+$ is pixel-wise addition. α means 1×1 convolution and β represents two 3×3 convolutions. Activation functions are ReLU. For features that do not have auxiliary features, there is no concatenation with the auxiliary feature and the refined feature is produced only by the main feature with α and β.

The specific structure of the inter-scale merging module $\mathcal{M}$ is shown in Fig. 5. We first resize the auxiliary feature to the same spatial size as the main feature. Then we use a 1×1 convolution α to extract useful information from the auxiliary feature conditioned on the main feature. Two 3×3 convolutions β followed are used to finish the interaction and output the refined feature. The residual link within the inter-scale merging module $\mathcal{M}$ is used for keeping the integrity of the main feature in the output feature $X^i_{Q,new}$. For those features that do not have auxiliary features (e.g., the first merged feature $X^1_{Q,m}$ in the top-down path and the last merged feature $X^n_{Q,m}$ in the bottom-up path), we simply ignore the concatenation with the auxiliary feature in $\mathcal{M}$ – the refined feature is produced only by the main feature.

Information Concentration. After inter-scale interaction, n refined feature maps are obtained as $X^i_{Q,new}, i \in \{1, 2, \ldots, n\}$. Finally, the output query feature $X_{Q,new} \in \mathbb{R}^{h \times w \times c}$ is formed by interpolation and concatenation of n refined feature maps $X^i_{Q,new} \in \mathbb{R}^{h \times w \times c}$ followed by an 1×1 convolution $\mathcal{F}_{1\times1}$ as

$$X_{Q,new} = \mathcal{F}_{1\times1}(X^1_{Q,new} \oplus X^2_{Q,new} \ldots \oplus X^n_{Q,new}). \tag{8}$$

The visual illustration of the baseline model without FEM ($B^1 = h = w$) is shown in Fig. 6. To encourage better feature enrichment, we add intermediate supervision by attaching classification head (Fig. 7b) to each $X^i_{Q,new}$.

In summary, by incorporating the pooled support features and prior masks to query features with different spatial sizes, the model learns to adaptively enrich the query feature with information coming from the support feature at each location under the guidance of prior mask and

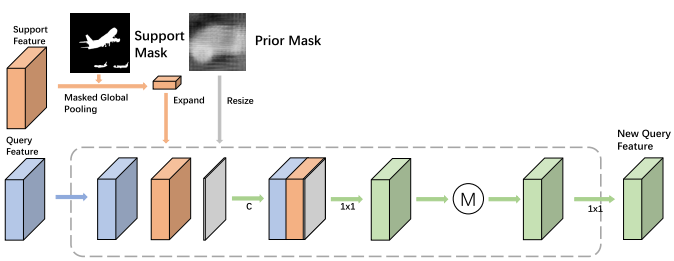


Fig. 6. Visual illustration of the baseline structure that processes features in the original spatial size of the input features.

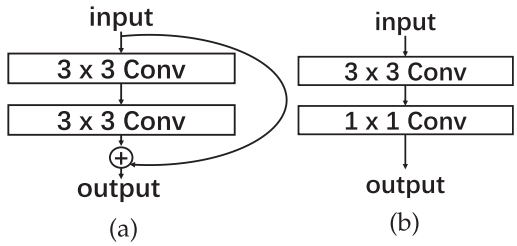


Fig. 7. Structures of (a) convolution block and (b) classification head.

supervision of ground-truth. Moreover, the vertical inter-scale interaction supplements the main feature with the conditioned information provided by the auxiliary feature. Therefore, FEM yields greater performance gain on baseline than other feature enhancement designs (e.g., PPM [52], ASPP [3] and GAU [45]). Experiments in Section 4.3 provide more details.

3.4 Prior Guided Feature Enrichment Network

3.4.1 Model Description

Based on the proposed prior generation method and the feature enrichment module, we propose the Prior Guided Feature Enrichment Network as shown in Fig. 3. The ImageNet [28] pre-trained CNN is shared by support and query images to extract features. The extracted middle-level support and query features are processed by 1×1 convolution to reduce the channel number to 256.

After feature extraction and channel reduction, the feature enrichment module enriches the query feature with the support feature and prior mask. On the output feature of FEM, we apply a convolution block (Fig. 7a) followed by a classification head to yield the final prediction. Classification head is composed of one 3×3 convolution and 1×1 convolution with Softmax function as shown in Fig. 7b. For all backbone networks, we use the outputs of the last layers of conv3_x and conv4_x as middle-level features M to generate the query and support features by concatenation, and take the output of the last layer of conv5_x as high-level features H to produce the prior mask.

In the 5-shot setting, we simply take the average of 5 pooled support features as the new support feature before concatenation with the query feature. Similarly, the final prior mask before the concatenation in FEM is also obtained by averaging five prior masks produced by one query feature with different support features.

3.4.2 Loss Function

We select the cross entropy loss as our loss function. As shown in Section 3.3.2 and Fig. 3, for a FEM with n different spatial sizes, the intermediate supervision on $X^i_{Q,new}$ ($i \in \{1, 2, \ldots, n\}$) generates n losses $\mathcal{L}^i_1$ ($i \in \{1, 2, \ldots, n\}$). The final prediction of PFENet generates the second loss $\mathcal{L}_2$. The total loss $\mathcal{L}$ is the weighted sum of $\mathcal{L}^i_1$ and $\mathcal{L}_2$ as

$$\mathcal{L} = \frac{\sigma}{n} \sum_{i=1}^{n} \mathcal{L}^i_1 + \mathcal{L}_2, \tag{9}$$

where σ is used to balance the effect of intermediate supervision. We empirically set σ to 1.0 in all experiments.

TABLE 1
Class mIoU Results on Four Folds of PASCAL-5^i

Methods	1-Shot					5-Shot					Params
	Fold-0	Fold-1	Fold-2	Fold-3	Mean	Fold-0	Fold-1	Fold-2	Fold-3	Mean	
VGG-16 Backbone											
$OSLSM_{2017}$ [29]	33.6	55.3	40.9	33.5	40.8	35.9	58.1	42.7	39.1	44.0	276.7M
co-FCN_{2018} [25]	36.7	50.6	44.9	32.4	41.1	37.5	50.0	44.1	33.9	41.4	34.2M
SG-One_{2018} [50]	40.2	58.4	48.4	38.4	46.3	41.9	58.6	48.6	39.4	47.1	19.0M
AMP_{2019} [31]	41.9	50.2	46.7	34.7	43.4	41.8	55.5	50.3	39.9	46.9	34.7M
$PANet_{2019}$ [40]	42.3	58.0	51.1	41.2	48.1	51.8	64.6	**59.8**	46.5	55.7	14.7M
$FWBF_{2019}$ [24]	47.0	59.6	52.6	48.3	51.9	50.9	62.9	56.5	50.1	55.1	-
Ours	**56.9**	**68.2**	**54.4**	**52.4**	**58.0**	**59.0**	**69.1**	54.8	**52.9**	**59.0**	**10.4M**
ResNet-50 Backbone											
$CANet_{2019}$ [46]	52.5	65.9	51.3	51.9	55.4	55.5	67.8	51.9	53.2	57.1	19.0M
$PGNet_{2019}$ [45]	56.0	66.9	50.6	50.4	56.0	54.9	67.4	51.8	53.0	56.8	17.2M
Ours	**61.7**	**69.5**	**55.4**	**56.3**	**60.8**	**63.1**	**70.7**	**55.8**	**57.9**	**61.9**	**10.8M**
ResNet-101 Backbone											
$FWBF_{2019}$ [24]	51.3	64.5	**56.7**	52.2	56.2	54.8	67.4	**62.2**	55.3	59.9	-
Ours	**60.5**	**69.4**	54.4	**55.9**	**60.1**	**62.8**	**70.4**	54.9	**57.6**	**61.4**	**10.8M**

Params: *number of learnable parameters.*

4 Experiments

4.1 Implementation Details

Datasets. We use the datasets of PASCAL-5^i [29] and COCO [19] in evaluation. PASCAL-5^i is composed of PASCAL VOC 2012 [5] and extended annotations from SDS [10] datasets. 20 classes are evenly divided into 4 folds $i \in \{0, 1, 2, 3\}$ and each fold contains 5 classes. Following OSLSM [29], we randomly sample 1,000 query-support pairs in each test.

Following [24], we also evaluate our model on COCO by splitting four folds from 80 classes. Thus each fold has 20 classes. The set of class indexes contained in fold i is written as $\{4x - 3 + i\}$ where $x \in \{1, 2, \ldots, 20\}, i \in \{0, 1, 2, 3\}$. Note that the COCO validation set contains 40,137 images (80 classes), which are much more than the images in PASCAL-5^i. Therefore, 1,000 randomly sampled query-support pairs used in previous work are not enough for producing reliable testing results on 20 test classes. We instead randomly sample 20,000 query-support pairs during the evaluation on each fold, making the results more stable than testing on 1,000 query-support pairs used in previous work. Stability statistics are shown in Section 4.7.

For both PASCAL-5^i and COCO, when testing the model on one fold, we use the other three folds to train the model for cross-validation. We take the average of five testing results with different random seeds for comparison as shown in Tables 9 and 10.

Experimental Setting. Our framework is constructed on PyTorch. We select VGG-16 [32], ResNet-50 [12] and ResNet-101 [12] as our backbones for fair comparison with other methods. The ResNet we use is the dilated version used in previous work [13], [24], [46]. The VGG we use is the original version [32]. All backbone networks are initialized with ImageNet [28] pretrained weights. Other layers are initialized by the default setting of PyTorch. We use SGD as our optimizer. The momemtum and weight decay are set to 0.9 and 0.0001 respectively. We adopt the 'poly' policy [2] to decay the learning rate by multiplying $(1 - \frac{current_{iter}}{max_{iter}})^{power}$ where $power$ equals to 0.9.

Our models are trained on PASCAL-5^i for 200 epochs as that of [46] with learning rate 0.0025 and batch size 4. For experiments on COCO, models are trained for 50 epochs with learning rate 0.005 and batch size 8. Parameters of the backbone network are not updated. During training, samples are processed with mirror operation and random rotation from -10 to 10 degrees. Finally, we randomly crop 473×473 patches from the processed images as training samples. During the evaluation, following [31], [40], [46], each input sample is resized to the training patch size but with respect to its original aspect ratio by padding zero. We directly output the single-scale results without fine-tuning and any additional post-processing (such as multi-scale testing and DenseCRF [16]). Our experiments are conducted on an NVIDIA Titan V GPU and Intel Xeon CPU E5-2620 v4 @ 2.10 GHz. The code and trained models will be made publicly available.

Evaluation Metrics. Following [24], [46], we adopt the class mean intersection over union (mIoU) as our major evaluation metric for ablation study since the class mIoU is more reasonable than the foreground-background IoU (FB-IoU) as stated in [46]. The formulation follows $mIoU = \frac{1}{C}\sum_{i=1}^{C} IoU_i$, where C is the number of classes in each fold (e.g., $C = 20$ for COCO and $C = 5$ for PASCAL-5^i) and IoU_i is the intersection-over-union of class i. We also report the results of FB-IoU for comparison with other methods. For FB-IoU calculation on each fold, only foreground and background are considered ($C = 2$). We take average of results on all folds as the final mIoU/FB-IoU.

4.2 Results

As shown in Tables 1, 2 and 3, we build our models on three backbones VGG-16, ResNet-50 and ResNet-101 and report the mIoU/FB-IoU results respectively. By incorporating the proposed prior mask and FEM, our model significantly outperforms previous methods, reaching new state-of-the-art on both PASCAL-5^i and COCO datasets. The PFENet can even outperform other methods on COCO with more than 10 points in terms of class mIoU. Our performance advantage on

TABLE 2
FB-IoU Results on PASCAL-5^i

Methods	1-Shot	5-Shot	Params
VGG-16 Backbone			
$OSLSM_{2017}$ [29]	61.3	61.5	272.6M
co-FCN_{2018} [25]	60.1	60.2	34.2M
PL_{2018} [4]	61.2	62.3	-
SG-One_{2018} [50]	63.9	65.9	19.0M
$PANet_{2019}$ [40]	66.5	70.7	14.7M
Ours	**72.0**	**72.3**	**10.4M**
ResNet-50 Backbone			
$CANet_{2019}$ [46]	66.2	69.6	19.0M
$PGNet_{2019}$ [45]	69.9	70.5	17.2M
Ours	**73.3**	**73.9**	**10.8M**
ResNet-101 Backbone			
A-MCG_{2019} [13]	61.2	62.2	86.1M
Ours	**72.9**	**73.5**	**10.8M**

Our results are single-scale ones without additional post-processing like DenseCRF [16]. As many other methods do not report the specific result of each fold, we present the comparison of the average FB-IoU results in this table.

FB-IoU compared to PANet is relatively smaller than class mIoU on COCO, because FB-IoU is biased towards the background and classes that cover a large part of the foreground area. It is worth noting that our PFENet achieves the best performance with the fewest learnable parameters (10.4M for VGG based model and 10.8M for ResNet based models). Qualitative results are shown in Fig. 8.

4.3 Ablation Study of FEM

The proposed feature enrichment module adaptively enriches the query feature by merging with support features in different scales and utilizes an inter-scale path to vertically transfer useful information from the auxiliary features to the main features. To verify the effectiveness of FEM, we first compare different strategies for inter-scale interaction. It shows that the top-down information path brings a decent performance gain to the baseline without compromising the model size much. Then experiments with different designs for inter-source enrichment are presented followed by comparison with the other feature enrichment designs of HRNet [39], ASPP [3] and PPM [52]. We also compare the Graph Attention Unit used in the recent state-of-the-art few-shot segmentation method PGNet [46] to refine the query feature. In these experiments, since our input images are resized to 473 $\times$ 473, the input feature map of the module (e.g., FEM, GAU) has the spatial size 60 $\times$ 60.

4.3.1 Inter-Scale Interaction Strategies

In this section, we show experimental results and analysis on different vertical inter-scale interaction strategies to manifest the rationales behind our designs of FEM.

As mentioned in Section 3.3, there are four alternatives for the inter-scale interaction: top-down path (TD), bottom-up path (BU), top-down + bottom-up path (TD+BU), and bottom-up + top-down path (BU+TD). Our experimental results in Table 4 show that TD and TD+BU help the basic FEM structure without (W/O) the information path accomplish better results than both BU and BU+TD. The model with TD+BU contains more learnable parameters (16.0M) than TD (10.8M), and yet yields comparable performance. We thus choose TD for inter-scale interaction.

These experiments prove that using the finer feature (auxiliary) to provide additional information to the coarse feature (main) is more effective than using the coarse feature (auxiliary) to refine the finer feature (main). It is because the coarse features are not sufficient for targeting the query classes during the later information concentration stage if the target object disappears in small scales.

Different from common semantic segmentation where contextual information is the key for good performance, the way of representation and acquisition of query information is more important in few-shot segmentation. Our motivation for designing FEM is to match the query and support features in different scales to tackle the spatial inconsistency between the query and support samples. Thus, a downsampled coarse query feature without target information is less helpful for improving the quality of the final prediction as shown in the experiments comparing TD and BU.

4.3.2 Comparison With Other Designs

PPM [52] and ASPP [3] are two popular feature enrichment modules for semantic segmentation by providing multi-resolution context, and HRNet [34], [35], [39] provides a new

TABLE 3
Class mIoU / FB-IoU Results on COCO

Methods	*Backbone*	1-Shot					5-Shot				
		Fold-0	Fold-1	Fold-2	Fold-3	Mean	Fold-0	Fold-1	Fold-2	Fold-3	Mean
Class mIoU Evaluation											
$FWBF_{2019}$ [24]	VGG-16	18.4	16.7	19.6	25.4	20.0	20.9	19.2	21.9	28.4	22.6
$PANet_{2019}$ [40]	VGG-16	-	-	-	-	20.9	-	-	-	-	29.7
Ours	VGG-16	**35.4**	**38.1**	**36.8**	**34.7**	**36.3**	**38.2**	**42.5**	**41.8**	**38.9**	**40.4**
$FWBF_{2019}$ [24]	ResNet-101	19.9	18.0	21.0	28.9	21.2	19.1	21.5	23.9	30.1	23.7
Ours	ResNet-101	**36.8**	**41.8**	**38.7**	**36.7**	**38.5**	**40.4**	**46.8**	**43.2**	**40.5**	**42.7**
FB-IoU Evaluation											
$PANet_{2019}$ [40]	VGG-16	-	-	-	-	59.2	-	-	-	-	63.5
Ours	VGG-16	**53.3**	**66.1**	**66.6**	**67.1**	**63.3**	**53.5**	**68.3**	**68.2**	**70.1**	**65.0**
A-MCG_{2019} [13]	ResNet-101	-	-	-	-	52.0	-	-	-	-	54.7
Ours	ResNet-101	**51.6**	**65.9**	**66.6**	**66.0**	**63.0**	**52.3**	**70.0**	**69.5**	**71.3**	**65.8**

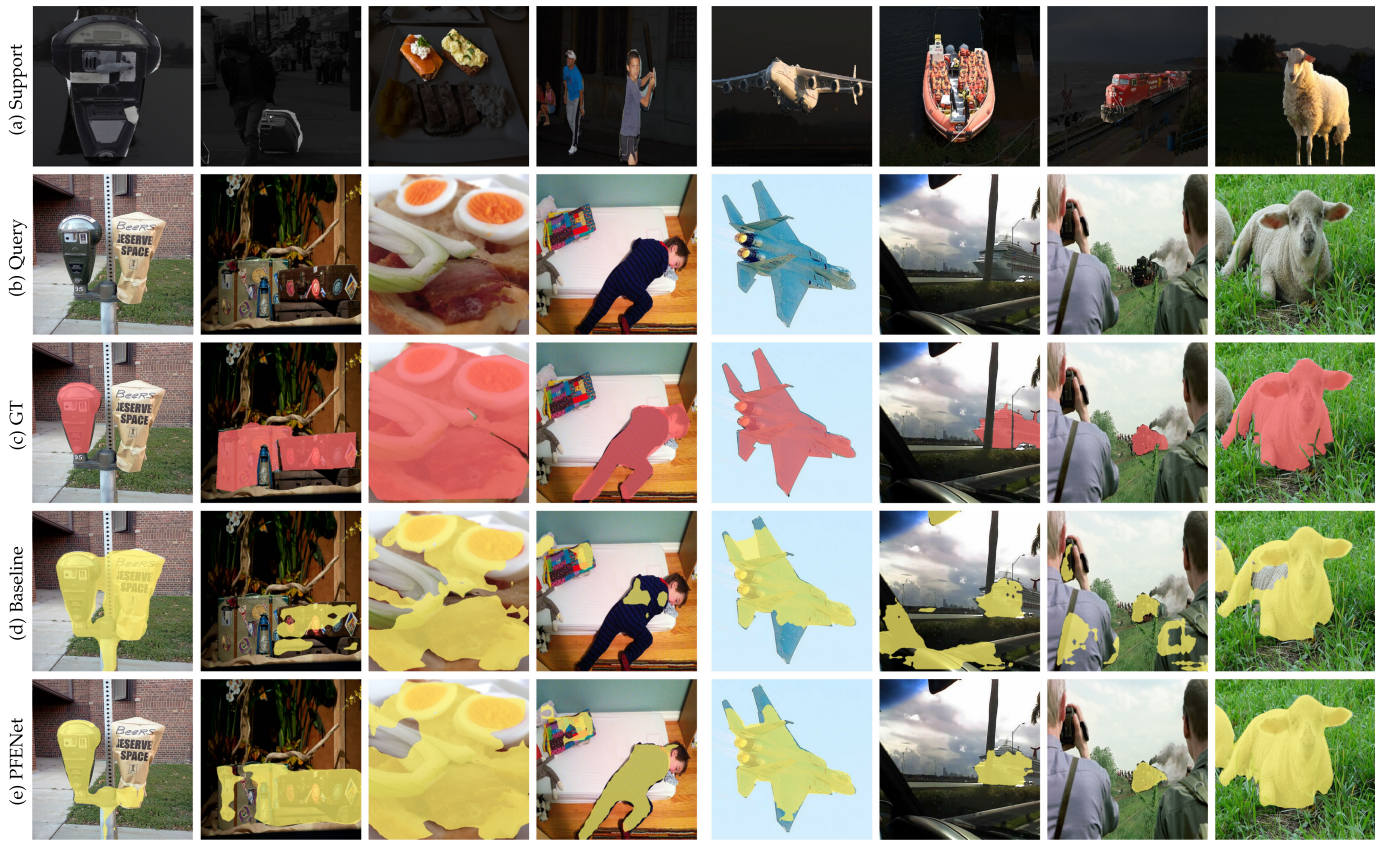


Fig. 8. Qualitative results of the proposed PFENet and the baseline. The left samples are from COCO and the right ones are from PASCAL-5^i. From top to bottom: (a) support images, (b) query images, (c) ground truth of query images, (d) predictions of baseline, (e) predictions of PFENet.

feature enrichment module for the segmentation task–it achieved SOTA results on semantic segmentation benchmarks. In few-shot segmentation, the Graph Attention Unit has been used in PGNet [45] to refine the query feature with contextual information. We note the proposed FEM module yields even better few-shot segmentation performance.

The improvement brought by FEM stems from: 1) the fusions of query and support features in different spatial sizes (inter-source enrichment) since it encourages the following convolution blocks to process the concatenated features independently in different spatial resolutions, which is beneficial to predicting query targets in various scales; 2) the inter-scale interaction that selectively passes useful information from the auxiliary feature to supplement the main feature. The model without the vertical top-down information path (marked with WO) yields worse results in Table 5.

We implement the ASPP with dilation rates $\{1, 6, 12, 18\}$ and it achieves close results to PPM. The dilated convolution is less effective than adaptive average pooling for few-shot segmentation [45]. In the following, we mainly make comparisons with PPM and GAU first since they both use the adaptive pooling to provide multi-scale information. Then, we make a discussion with the module proposed by HRNet.

Pyramid Pooling Module. As shown in Table 5, the model with spatial sizes $\{60, 30, 15, 8\}$ achieves better performance than the baseline (original size with spatial size $\{60\}$) and models that replace FEM with PPM and ASPP. Experiments of PSPNet [52] show that the Pyramid Pooling Module with spatial sizes $\{6, 3, 2, 1\}$ yields the best performance. When small spatial sizes are applied to FEM, it still outperforms PPM. But small spatial sizes are not optimal in FEM because

TABLE 4
Class mIoU Results of Different Ways for Inter-Scale Interaction on PASCAL-5^i

Methods	1-Shot					5-Shot				
	Fold-0	Fold-1	Fold-2	Fold-3	Mean	Fold-0	Fold-1	Fold-2	Fold-3	Mean
W/O	60.5	68.4	55.4	54.9	59.8	62.8	68.9	55.6	56.5	61.0
TD	61.7	69.5	55.4	56.3	**60.8**	**63.1**	**70.7**	55.8	57.9	61.9
BU	**62.4**	69.2	53.9	55.9	60.4	**63.1**	70.1	53.7	56.0	60.7
TD+BU	61.0	**69.7**	**55.6**	**57.0**	**60.8**	62.4	70.4	**56.4**	**58.9**	**62.0**
BU+TD	61.0	68.9	54.8	56.0	60.2	62.4	69.8	54.5	56.7	60.8

All models in this table are based on ResNet-50 and are trained and tested with prior masks. W/O*: FEM without the information path for inter-scale interaction.* TD*: FEM with top-down information path.* BU*: FEM with bottom-up information path.* TD+BU*: FEM with top-down + bottom-up information path.* BU+TD*: FEM with bottom-up + top-down information path.*

TABLE 5
Class mIoU of FEM With Different Spatial Sizes and the Comparison With PPM [52] and ASPP [3] on PASCAL-5^i

Methods	1-Shot					5-Shot				
	Fold-0	Fold-1	Fold-2	Fold-3	Mean	Fold-0	Fold-1	Fold-2	Fold-3	Mean
{60} (Baseline)	54.3	67.3	53.3	50.4	56.3	57.1	68.0	53.8	52.9	58.0
{60} + PPM [52]	55.4	68.4	53.2	51.4	57.1	58.3	68.9	53.5	50.8	57.9
{60} + ASPP [3]	57.6	68.4	52.8	49.0	56.9	59.5	69.3	52.6	50.7	58.0
{60, 6, 3, 2, 1}	58.8	68.0	54.1	51.2	58.0	59.8	68.4	53.8	52.1	58.5
{60, 30}	55.3	67.8	**54.7**	51.2	57.3	58.4	68.7	54.5	53.1	58.7
{60, 30, 15}	56.6	68.0	54.6	52.9	58.0	59.0	68.7	55.0	54.0	59.2
{60, 30, 15, 8}	**59.4**	**68.9**	**54.7**	53.6	**59.2**	**61.5**	**69.5**	**55.4**	55.3	**60.4**
{60, 30, 15, 8, 4}	58.7	68.5	54.1	**54.5**	58.9	60.3	69.3	54.9	**56.4**	60.2
{60, 30, 15, 8}-WO	57.9	67.4	53.7	53.6	58.2	60.5	68.0	54.2	53.8	59.1

The backbone is ResNet-50. '$\{60, 30, 15, 8\}$*': the input query feature is average-pooled into four scales* $\{60, 30, 15, 8\}$ *and concatenate with the expanded support features respectively as shown in Fig. 4.* WO*: without inter-scale interaction.*

the features pooled to spatial sizes like $\{6, 3, 2, 1\}$ are too coarse for interaction and fusion of query and support features. Similarly, with small spatial size 4, the FEM with $\{60, 30, 15, 8, 4\}$ yields inferior performance compared to using the model with spatial sizes $\{60, 30, 15, 8\}$. Hence, we select $\{60, 30, 15, 8\}$ as the feature scales for the inter-source enrichment of FEM.

Graph Attention Unit. GAU [45] uses the graph attention mechanism to establish the element-to-element correspondence between the query and support features in each scale. Pixels of the support feature are weighed by the GAU and the new support feature is the weighted sum of the original support feature. Then the new support feature is concatenated with the query feature for further processing.

We directly replace the FEM with GAU on our baseline and keep other settings for a fair comparison. GAU is implemented with the code provided by the authors. Our baseline with GAU achieves class mIoU 55.4 and 56.1 in 1- and 5-shot evaluation respectively. Noticing the original feature scales in GAU are $\{60, 8, 4\}$, we also implement it with scales $\{60, 30, 15, 8\}$ (denoted as GAU+) used in our FEM. GAU+ yields smaller mIoU than GAU (54.9 in 1-shot and 55.4 in 5-shot). Though GAU also forms a pyramid structure via adaptive pooling to capture the multi-level semantic information, it is less competitive than the proposed FEM (59.2 in 1-shot and 60.4 in 5-shot) because it misses the hierarchical inter-scale relationship that adaptively provides information extracted from other levels to help refine the merged feature.

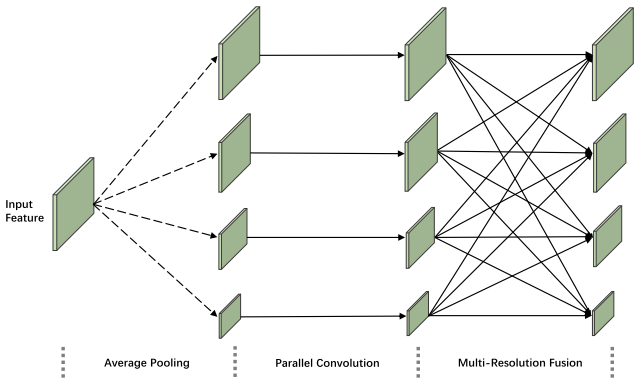


Fig. 9. Modularized block of HRNet (HRB) that applies dense multi-resolution fusions.

High-Resolution Network (HRNet). HRNet has shown its superiority on many vision tasks by maintaining a high-resolution feature through all the networks and gradually fusing multi-scale features to enrich the high-resolution features. The proposed FEM can be deemed as a variant of HRB to tackle the few-shot segmentation problem. The inter-source enrichment of FEM is analogous to the multi-resolution parallel convolution in HRB as shown in Fig. 9. But the inter-scale interaction in FEM passes conditioned information from large to small scales rather than dense interaction among all scales without selection in HRB.

For comparison, we experiment with replacing the FEM in PFENet with HRB and generate feature maps in HRB with the same scales of those in FEM ($\{60, 30, 15, 8\}$). Results are listed in Table 6. Directly applying HRB to the baseline (Baseline + HRB) does yield better results than PPM and ASPP. Densely passing information without selection causes redundancy to

TABLE 6
Class mIoU on PASCAL-5^i and Efficiency of Models With/Without the Proposed Prior and FEM

Methods	1-Shot	5-Shot	Params	Speed
Baseline	56.3	58.0	4.5 M	17.7 FPS
Baseline + PPM [52]	57.1	57.9	5.7 M	17.6 FPS
Baseline + ASPP [3]	56.9	58.0	7.9 M	17.5 FPS
Baseline + HRB [39]	58.3	59.4	14.4 M	15.7 FPS
Baseline + HRB-Cond	59.2	60.0	23.0 M	14.5 FPS
Baseline + HRB-TD	58.9	60.0	14.0 M	16.1 FPS
Baseline + HRB-TD-Cond	59.3	60.4	18.3 M	15.6 FPS
Baseline + FEM	59.2	60.4	10.8 M	17.3 FPS
Baseline + FEM‡	58.9	60.2	12.9 M	16.1 FPS
Baseline + Prior	58.2	59.6	4.5 M	16.5 FPS
Baseline + FEM + Prior	**60.8**	**61.9**	10.8 M	15.9 FPS
Baseline†	48.8	50.1	28.2 M	17.7 FPS
Baseline† + FEM	50.2	52.3	34.5 M	16.1 FPS
Baseline† + Prior†	49.7	53.1	28.2 M	16.5 FPS
Baseline† + FEM + Prior†	51.9	55.3	34.5 M	15.9 FPS

Models are based on ResNet-50. Params*: The number of learnable parameters.* Speed*: Average frame-per-second (FPS) of 1-shot evaluation.* HRB*: Modularized block of HRNet [39].* -TD*: Only top-down feature enrichment paths are enabled.* -Cond*: The inter-scale enrichment modules are implemented to pass the conditioned information.* FEM*: Feature enrichment module with* $\{60, 30, 15, 8\}$*.* FEM‡*: FEM with spatial sizes* $\{60, 30, 15, 8, 4\}$*.* Prior*: Prior masks got by fixed high-level features (conv5_x).* Baseline†*: Models trained with all backbone parameters.* Prior†*: Prior masks got by learnable high-level features.*

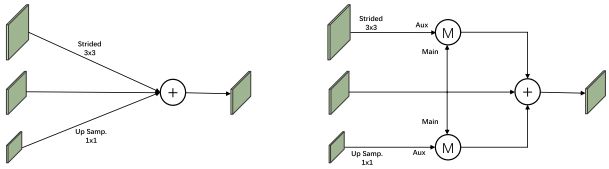


Fig. 10. Comparison between feature fusion strategies of (left) HRB and (right) HRB-Cond. Features from different scales are directly added to the main feature in (left), while in (right), essential information is selected from auxiliary features conditioned on the main features by the inter-scale merging module $\mathcal{M}$.

the target feature and yields suboptimal results. Our solution is, in the multi-resolution fusion stage of HRB, to apply the proposed inter-scale merging module $\mathcal{M}$ to extract essential information from the auxiliary features as shown in Fig. 10. The model with conditioned feature selection (HRB-Cond) accomplishes better performance.

As shown in Table 4, passing features from coarse to fine levels (in a bottom-up order) adversely affects inter-scale interaction. We accordingly remove all bottom-up paths in HRB and only allow top-down ones (denoted as HRB-TD). It is not surprising that HRB-TD achieves better performance than HRB, and adding conditioned feature selection (HRB-TD-Cond) brings even further improvement.

The best variant of HRB (i.e., HRB-TD-Cond) yields comparable results with FEM, and yet it brings much more learnable parameters (7.5M). Therefore, for few-shot segmentation, the conditioned feature selection mechanism of the proposed inter-scale merging module $\mathcal{M}$ is essential for improving the performance of the multi-resolution structures.

4.4 Ablation Study of the Prior Generation

Experimental results in Table 6 show that the prior improves models w/ and wo/ FEM. The cosine-similarity is widely used for tackling few-shot segmentation. PANet [40] uses the cosine-similarity to yield the intermediate and the final prediction masks; SG-One [50] and [24] both utilize the cosine-similarity mask from the mask pooled support feature to provide additional guidance. However, these methods overlooked two factors. First, the mask generation process contains trainable components and the generated mask is thus biased towards the base classes during training.

Second, the discrimination loss is led by the masked average pooling on support features, since the most relevant information in the support feature may be overwhelmed by the irrelevant ones during the pooling operation. For example, the discriminative regions for "cat & dog" are mainly around their heads. The main bodies share similar characteristics (e.g., tailed quadrupeds), making representation produced by masked global average pooling lose the discriminative information contained in the support samples.

In the following, we first show the rationale behind our prior generation using the fixed high-level feature and taking the maximum pixel-wise correspondence value from the similarity matrix. Then we make a comparison with other methods to demonstrate the superiority of our strategy. We also include the analysis of the generalization ability on the unseen objects out of the ImageNet [28] dataset to further manifest the robustness of our method.

4.4.1 Feature Selection

In our design, we select the fixed high-level feature for the prior generation because it can provide sufficient semantic information for accurate segmentation without sacrificing the generalization ability. The proposed prior generation is independent of the training process. So it does not lead to loss of generalization power. The prior masks provide the bias-free prior information from high-level features for both seen and unseen data during the evaluation, while masks produced by learnable feature maps (e.g., [24], [40], [50]) are affected by parameter learning during training. As a result, the preference for the training classes is inevitable for these later masks during the inference. To show the superiority of our choice, we conduct experiments on different sources of features for generating prior masks.

Quantitative Analysis. Table 7 shows that the mask generated by either learnable or fixed middle-level features (Prior$_{LM}$ or Prior$_{FM}$) is less improved than our Prior$_{FH}$ since the middle-level feature is less effective to reveal the semantic correspondence between the query and support features. However, the results of mask got by learnable high-level feature (Prior$_{LH}$) are even significantly worse than that of our baseline due to the fact that the learnable high-level feature severely overfits to the base classes: the model relies on

TABLE 7
Class mIoU Results of Different Prior Masks on PASCAL-5^i

Methods	1-Shot					5-Shot				
	Fold-0	Fold-1	Fold-2	Fold-3	Mean	Fold-0	Fold-1	Fold-2	Fold-3	Mean
Baseline	49.4	64.6	53.3	46.0	53.3	51.5	65.5	52.5	47.0	54.1
Baseline + Prior$_{LM}$	50.3	54.5	53.0	46.2	53.5	51.9	65.7	52.9	47.2	54.4
Baseline + Prior$_{LH}$	37.8	60.8	53.5	43.4	48.9	42.5	64.2	57.8	47.6	53.0
Baseline + Prior$_{FM}$	51.2	64.4	53.9	45.7	53.8	52.8	65.1	53.2	47.5	54.7
Baseline + Prior$_{FH}$	**53.5**	**65.6**	53.6	**48.8**	**55.4**	**55.7**	**66.4**	53.8	49.8	**56.4**
Baseline + Prior-A$_{FH}$	52.2	65.4	54.5	48.5	55.1	54.8	66.0	54.3	50.2	56.3
Baseline + Prior-P$_{FH}$	52.4	65.8	53.1	47.6	54.7	54.9	67.0	53.5	48.8	56.1
Baseline + Prior-FW$_{LM}$	50.6	64.9	52.4	42.9	52.7	53.4	65.5	51.7	43.2	53.5
Baseline + Prior-FW$_{LH}$	37.5	60.3	**54.8**	43.9	49.1	44.2	62.8	**58.5**	47.0	53.1
Baseline + Prior-FW$_{FM}$	50.6	64.7	54.4	47.0	54.2	52.5	65.4	53.7	47.8	54.9
Baseline + Prior-FW$_{FH}$	51.0	65.1	53.9	**48.8**	54.7	52.7	66.1	53.8	**50.4**	55.8

All models in this table are based on VGG-16. LM*: Learnable middle-level features.* LH*: Learnable high-level features.* FM*: Fixed middle-level features.* FH*: Fixed high-level features.* Prior*: Prior mask got by taking the maximum similarity value.* Prior-A*: Prior mask got by the average similarity value.* Prior-P*: Prior mask generated with the mask-pooled support feature.* Prior-FW*: Prior mask got by the feature weighting mechanism proposed in [24].*

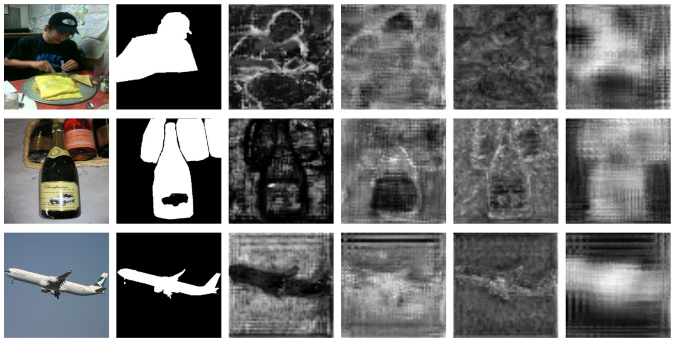


Fig. 11. Visual comparison between priors generated by different sources. Prior values are normalized to 0-1, which implies the probability of being the target region. *GT*: Ground truth. *L-M*: Learnable middle-level features. *L-H*: Learnable high-level features. *F-M*: Fixed middle-level features. *F-H*: Fixed high-level features.

the accurate prior masks produced by the learnable high-level feature for locating the target region of base classes during training and therefore it hardly generalizes to the previously unseen classes during inference.

Qualitative Analysis. Generated prior masks are shown in Fig. 11. Masks of unseen classes generated by learnable high-level feature maps (L-H) cannot reveal the potential region-of-interest clearly while using the fixed high-level feature maps (F-H) keeps the general integrity of the target region. Compared to high-level features, prior masks produced by middle-level ones (L-M and F-M) are more biased towards the background region.

To help explain the quantitative results and those in Fig. 11, embedding visualization is shown in Fig. 12 where 1,000 samples of base classes (gray) and 1,000 samples of novel classes (colored in green, red, purple, blue and orange) are processed by the backbone followed by t-SNE [37]. Based on the overlapping area between the clusters of the base and novel classes, we draw two conclusions. First, the middle-level features in Figs. 12a & 12c are less discriminative than the high-level features as shown in Figs. 12b & 12d. Second, learnable features lose discrimination ability as shown in (a) & (b) because embeddings of novel classes bias towards that of the base classes, which is detrimental to the generalization on unseen classes.

4.4.2 Discrimination Ability

In our model, the prior mask acts as a pixel-wise indicator for each query image. As given in Eq. (3), taking the maximum correspondence value from the pixel-wise similarity between the query and support features indicates that there exists at least one pixel/area in the support image that has close semantic relation to the query pixel with a high prior value. It is beneficial to reveal most of the potential targets on query images. Other alternatives include using mask pooled support feature to generate the similarity mask as [24], [40], [50], and taking the average value rather than the maximum value from the pixel-wise similarity.

To verify the effectiveness of our design, we train two additional models in Table 7: one with prior masks generated by averaging similarities (Prior-A_{FH}), and another whose prior masks are obtained by the mask-pooled support feature (Prior-P_{FH}). They both perform less satisfyingly than the proposed strategy (Prior$_{FH}$).

We note the following fact. Our prior generation method takes the maximum value from a similarity matrix of size $hw \times hw$ to generate the prior mask of size $h \times w$ (Eq. (3)), in contrast to Prior-P forming the mask from the similarity matrix of size $hw \times 1$, the difference of speed is rather small because computational complexities of the two mask generation methods are much smaller than that of the rest of network. The FPS values of Prior$_{FH}$, Prior-A_{FH}, Prior-P_{FH} and Prior-FW_{FH} based on VGG-16 baseline are both around 23.1 FPS because the output features only contain 512 channels. The FPS values of Prior$_{FH}$, Prior-A_{FH}, Prior-P_{FH} and Prior-FW_{FH} based on ResNet-50 baseline whose output features have 2,048 channels are 16.5, 16.5, 17.4 and 17.0 respectively.

4.4.3 Comparison With Other Designs

Some other methods also use the similarity mask as an intermediate guidance for improving performance (e.g., [24], [40], [50]). Their masks are obtained by the learnable mask-pooled support and learnable query feature that is then used for further processing the making final prediction. The strategy of this type of method is similar to Prior-P_{LM}.

In [24], the good discrimination ability of features makes activation high on the foreground and low elsewhere. We follow Eqs. (3), (4), (5), and (6) in [24] to implement the feature weighting mechanism on both the query and support features used for prior mask generation. In [24], the weighting mechanism is directly applied to learnable features, and we offer two choices in our model: the learnable middle- and high-level features. However, it does not perform better for Prior-FW_{LM} and Prior-FW_{LH}. Results of Prior-FW_{FH} demonstrates the effectiveness of our feature selection

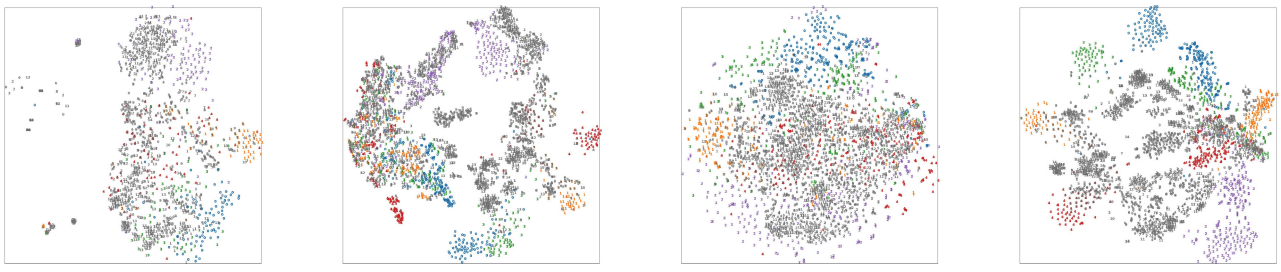


Fig. 12. Visual comparison between t-SNE results of different feature sources. 1,000 features in gray color are from base classes and 1,000 features in other colors are from novel classes. *L-M*: Learnable middle-level features. *L-H*: Learnable high-level features. *F-M*: Fixed middle-level features. *F-H*: Fixed high-level features.

TABLE 8
Foreground IoU Results on Totally Unseen Classes of FSS-1000 [17]

Methods	1-Shot	5-Shot
Baseline	79.7	80.1
Baseline + Prior	**80.8**	**81.4**

strategy (with fixed high-level features) for prior generation. Our feature selection strategy is complementary to the weighting mechanism of [24].

4.4.4 Generalization on Totally Unseen Objects

Many objects of PASCAL-5^i and COCO have been included in ImageNet [28] for backbone pre-training. For those previously unseen objects, the backbone still provides strong semantic cues to help identify the target area in query images with the information provided by the support images. The class 'Person' in PASCAL-5^i is not contained in ImageNet, and the baseline with the prior mask achieves 15.81 IoU, better than that without the prior mask (14.38). However, the class 'Person' is not rare in ImageNet samples even if their labels are not 'Person'.

To further demonstrate our generalization ability to totally unseen objects, we conduct experiments on the recently proposed FSS-1000 [17] dataset where the foreground IoU is used as the evaluation metric. FSS-1000 is composed of 1,000 classes, among which 486 classes are not included in any other existing datasets [17]. We train our models with ResNet-50 backbone on the seen classes for 100 epochs with batch size 16 and initial learning rate 0.01, and then test them on the unseen classes. The number of query-support pairs sampled for testing is equal to five times the number of unseen samples.

As shown in Table 8, the baseline with the prior mask achieves 80.8 and 81.4 foreground IoU in 1- and 5-shot evaluations respectively that outperform the vanilla baseline (79.7 and 80.1) by more than 1.0 foreground IoU in both settings. The visual illustration is given in Fig. 13 where the target regions can still be highlighted in the prior masks even if these objects were not witnessed by the ImageNet pre-trained backbone.

4.5 Backbone Training

In OSLSM [29], two backbone networks are trained to achieve few-shot segmentation. However, backbone parameters in recent work [40], [46] are kept to prevent overfitting. There is no experiment to show what effect the backbone training has. To reach a better understanding of how the backbone affects our method, the results of four models trained with all parameters in the backbone are shown in the last four rows of Table 6.

The additional trainable backbone parameters cause significant performance reduction due to the overfitting of training classes. Moreover, the backbone training nearly doubles the training time of each batch because an additional parameter update is required. It does not, however, affect the inference speed. As shown in the results, the improvement that FEM and prior mask bring to models with trainable backbones is less significant than on those with fixed backbones. We note that the prior masks in this section are produced by learnable high-level features because the whole backbone is trainable. The learnable high-level features bring worse performance to the fixed backbone as shown in Table 7, but they are beneficial to the trainable backbone. On 5-shot evaluation, the prior yields higher performance gain compared to FEM, because the prior is averaged over five support samples, providing a more accurate prior mask than 1-shot for query images to combat overfitting. Finally, the model with both FEM and the prior still outperforms the baseline model, which demonstrates the robustness of our proposed design even with all learnable parameters.

4.6 Model Efficiency

Parameters. The parameters of our backbone network are fixed as those in [40], [45], [46]. Four parts in the baseline model are learnable: two 1×1 convolutions for reducing dimension number of the query and support features, FEM, one convolution block and one classification head. As shown in Table 6, our best model (Baseline + FEM + Prior) only has 10.8M trainable parameters that are much fewer than other methods shown in Table 1. The prior generation does not bring additional parameters to the model, and FEM with spatial sizes $\{60, 30, 15, 8\}$ only brings 6.3M

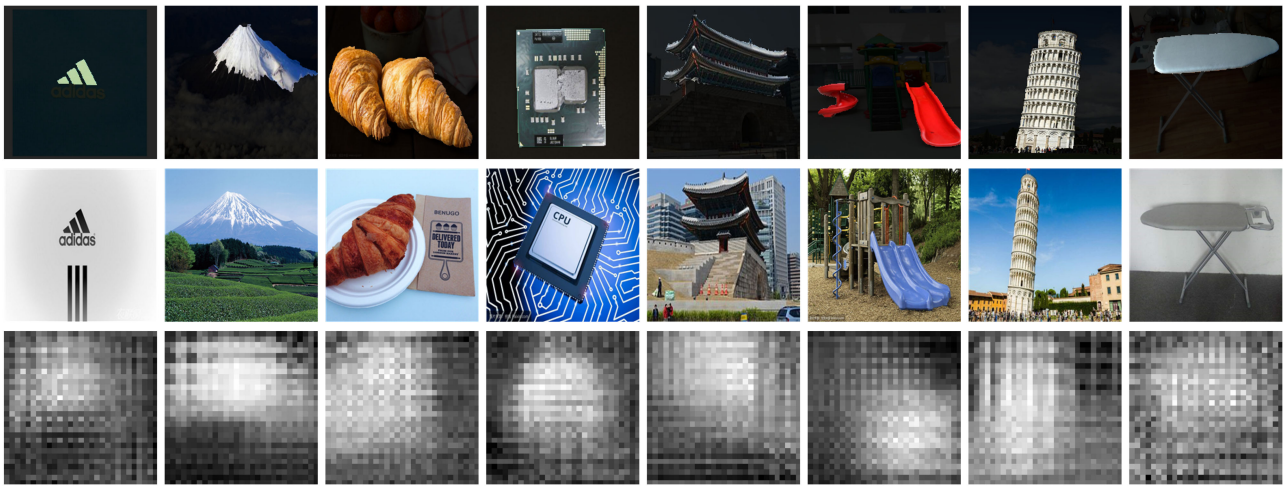


Fig. 13. Visual illustrations of prior masks for totally unseen objects in FSS-1000 dataset. *Top*: support images with the masked area in the target class. *Middle*: query images. *Bottom*: prior masks of query images where the regions of interest are highlighted.

TABLE 9
Mean and Std. of Five Test Results (class mIoU) on PASCAL-5^i

Fold - Shot	1	2	3	4	5	Mean	Std.
F0 - S1	61.1	61.9	62.2	61.6	61.7	61.7	0.406
F0 - S5	63.1	63.2	63.3	63.1	63.3	63.1	0.148
F1 - S1	69.5	69.7	69.1	69.5	69.7	69.5	0.245
F1 - S5	70.7	70.8	70.9	70.6	70.5	70.7	0.158
F2 - S1	55.3	55.2	55.6	55.4	55.1	55.4	0.230
F2 - S5	55.2	56.3	55.5	55.9	56.0	55.8	0.432
F3 - S1	56.0	56.2	56.2	56.7	56.3	56.3	0.259
F3 - S5	57.9	58.1	57.9	58.0	57.6	57.9	0.187

'Fm - Sn' means the n-shot results of Fold-m. Each row shows five test results with the values of mean and standard deviation (Std.).

additional learnable parameters to the baseline (4.5M → 10.8M). To prove that the improvement brought by FEM is not due to more learnable parameters, we show results of the model with FEM‡ that has more parameters (12.9M) but it yields even worse results than FEM (10.8M).

Speed. PFENet based on ResNet-50 yields the best performance with 15.9 and 5.1 FPS in 1- and 5-shot setting respectively on an NVIDIA Titan V GPU. During evaluation, test images are resized to 473 × 473. As shown in Table 6, FEM does not affect the inference speed much (from 17.7 to 17.3 FPS). Though the proposed prior generation process slows down the baseline from 17.7 to 16.5 FPS, the final model is still efficient with 15+ FPS. Note that we include the processing time of the last block of ResNet in these experiments for a fair comparison.

4.7 Analysis on Result Stability

As mentioned in the implementation details, evaluating 1,000 query-support pairs on PASCAL-5^i and COCO may cause instability on results. In this section, we show the analysis of result stability by conducting multiple experiments with different support samples.

PASCAL-5^i. Results in Table 9 show that the values of standard deviation are lower than 0.5 in both 1-shot and 5-shot setting, which shows the stability of our results on PASCAL-5^i with 1,000 pairs for evaluation.

COCO. However, 1,000 pairs are not sufficient to provide reliable results for comparison as shown in Table 10, since the COCO validation set contains 40,137 images and 1,000 pairs could not even cover the entire 20 test classes. Based on this observation, we instead randomly sample 20,000 query-support pairs to evaluate our models on four folds, and the results in Table 10 show that 20,000 pairs bring much more stable results than 1,000 pairs.

TABLE 11
Experimental Results in the Zero-Shot Setting

Methods	Shot	Fold-0	Fold-1	Fold-2	Fold-3	Mean
$OSLSM_{2017}$ [29]	5	35.9	58.1	42.7	39.1	44.0
co-FCN_{2018} [25]	5	37.5	50.0	44.1	33.9	41.4
$SG\text{-}One_{2018}$ [50]	5	41.9	58.6	48.6	39.4	47.1
AMP_{2019} [31]	5	41.8	55.5	50.3	39.9	46.9
Kato *et al.*$_{2019}$ [15]	0	39.6	52.6	41.0	35.6	42.2
Baseline	0	49.4	67.1	50.3	46.0	53.2
Baseline + FEM	0	50.0	68.5	51.7	46.6	54.2

Models shown in this table are based on VGG-16.

4.8 Extension to Zero-Shot Segmentation

Zero-shot learning aims at learning a model that is robust even when no labeled data is given. It is an extreme case of few-shot learning. To further demonstrate the robustness of our proposed PFENet in the extreme case, we modify our model by replacing the pooled support features with class label embeddings. Note that our proposed prior generation method requires support features. Therefore the prior is not applicable and we only verify FEM on the baseline with VGG-16 backbone in the zero-shot setting.

Structural Change. Embeddings of Word2Vec [23] and FastText [22] are trained on Google News [41] and Common Crawl [22] respectively. The concatenated feature of Word2Vec and FastText embeddings directly replaces the pooled support feature in the original model without normalization. Therefore the structural change on the model structure is the first learnable 1×1 convolution for reducing the support feature channel. Its input channel number 768 (512 + 256) in the original few-shot model (VGG-16 backbone) is updated to 600 (300 + 300) in the zero-shot model.

Results. As shown in Table 11, our base structure achieves 53.2 class mIoU on unseen classes without support samples, which even outperforms some models with five support samples on PASCAL-5^i in the few-shot setting of OSLSM [29]. Also, the proposed FEM tackles the spatial inconsistency in

TABLE 10
Analysis on Values of Mean and Std. of Five Test Results (Class mIoU) on COCO With Different Numbers of Test Query-Support Pairs (1,000 and 20,000)

Folds	*Pairs*	1-Shot							5-Shot						
		Test-1	Test-2	Test-3	Test-4	Test-5	Mean	Std	Test-1	Test-2	Test-3	Test-4	Test-5	Mean	Std
Fold-0	1,000	35.0	36.8	35.4	37.8	34.6	35.9	1.339	38.6	35.5	37.8	38.4	38.9	37.8	1.369
Fold-0	20,000	35.5	35.6	35.5	35.3	35.2	35.4	0.164	38.2	37.7	38.4	38.5	38.2	38.2	0.308
Fold-1	1,000	36.4	36.9	38.9	34.7	36.1	36.6	1.523	41.8	41.8	39.2	42.8	40.1	41.4	1.455
Fold-1	20,000	38.3	37.8	38.2	38.2	38.2	38.1	0.195	42.2	42.6	42.3	42.6	42.8	42.5	0.245
Fold-2	1,000	36.9	35.1	37.0	34.3	32.0	35.1	2.067	39.8	40.8	41.7	40.4	38.3	40.2	1.267
Fold-2	20,000	37.0	36.4	36.9	36.4	37.2	36.8	0.363	42.1	41.5	41.9	41.6	41.8	41.8	0.239
Fold-3	1,000	34.9	35.1	36.5	34.7	35.9	35.4	0.756	38.2	38.4	39.5	36.9	38.7	38.3	0.945
Fold-3	20,000	34.6	34.8	34.5	34.6	34.9	34.7	0.164	38.8	38.8	39.2	38.9	38.7	38.9	0.192

The model is based on VGG-16 [32]. 20,000 query-support pairs yield more stable results with a lower standard deviation than 1,000 query-support pairs.

the zero-shot setting and brings 1.0 points mIoU improvement (from 53.2 to 54.2) to the baseline.

5 Conclusion

We have presented the prior guided feature enrichment network with the proposed prior generation method and the feature enrichment module. The prior generation method boosts the performance by leveraging the cosine-similarity calculation on pre-trained high-level features. The prior mask encourages the model to localize the query target better without losing generalization power. FEM helps solve the spatial inconsistency by adaptively merging the query and support features at multiple scales with intermediate supervision and conditioned feature selection. With these modules, PFENet achieves new state-of-the-art results on both PASCAL-5^i and COCO datasets without much model size increase and notable efficiency loss. Experiments in the zero-shot scenario further demonstrate the robustness of our work. Possible future work includes extending these two designs to few-shot object detection and few-shot instance segmentation.

References

[1] Q. Cai, Y. Pan, T. Yao, C. Yan, and T. Mei, "Memory matching networks for one-shot image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4080–4088.

[2] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[3] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, 2017, *arXiv: abs/1706.05587*.

[4] N. Dong and E. P. Xing, "Few-shot semantic segmentation with prototype learning," in *Proc. Brit. Mach. Vis. Conf.*, Newcastle, UK, BMVA Press, 2018, p. 79.

[5] M. Everingham, L. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, pp. 303–338, 2010.

[6] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 1126–1135.

[7] J. Fu *et al.*, "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3141–3149.

[8] G. Ghiasi and C. C. Fowlkes, "Laplacian pyramid reconstruction and refinement for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 519–534.

[9] S. Gidaris and N. Komodakis, "Dynamic few-shot visual learning without forgetting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4367–4375.

[10] B. Hariharan, P. A. Arbeláez, R. B. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 297–312.

[11] B. Hariharan and R. B. Girshick, "Low-shot visual recognition by shrinking and hallucinating features," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3037–3046.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[13] T. Hu, P. Yang, C. Zhang, G. Yu, Y. Mu, and C. G. M. Snoek, "Attention-based multi-context guiding for few-shot semantic segmentation," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 8441–8448.

[14] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 603–612.

[15] N. Kato, T. Yamasaki, and K. Aizawa, "Zero-shot semantic segmentation via variational mapping," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop*, 2019, pp. 1363–1370.

[16] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with Gaussian edge potentials," in *Proc. 24th Int. Conf. Neural Inf. Process. Syst.*, 2011, pp. 109–117.

[17] X. Li, T. Wei, Y. P. Chen, Y.-W. Tai, and C.-K. Tang, "FSS-1000: A 1000-class dataset for few-shot segmentation," *CoRR*, 2020, *arXiv: abs/1907.12347*.

[18] G. Lin, A. Milan, C. Shen, and I. D. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5168–5177.

[19] T. Lin *et al.*, "Microsoft COCO: Common objects in context," in *13th Proc. Eur. Conf. Comput. Vis.*, Springer, Zurich, Switzerland, 2014, vol. 8693, pp. 740–755.

[20] W. Liu, A. Rabinovich, and A. C. Berg, "ParseNet: Looking wider to see better," *CoRR*, 2015, *arXiv: abs/1506.04579*.

[21] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang, "Semantic image segmentation via deep parsing network," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1377–1385.

[22] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," in *Proc. 11th Int. Conf. Lang. Resour. Eval.*, Eur. Language Resources Assoc., Miyazaki, Japan, 2018.

[23] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. 26th Int. Conf. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.

[24] K. Nguyen and S. Todorovic, "Feature weighting and boosting for few-shot segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 622–631.

[25] K. Rakelly, E. Shelhamer, T. Darrell, A. A. Efros, and S. Levine, "Conditional networks for few-shot semantic segmentation," in *6th Proc. Int. Conf. Learn. Representations Workshop*, Vancouver, BC, Canada, 2018.

[26] K. Rakelly, E. Shelhamer, T. Darrell, A. A. Efros, and S. Levine, "Few-shot segmentation propagation with guided networks," *CoRR*, 2018, *arXiv: abs/1806.07373*.

[27] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.

[28] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, pp. 211–252, 2015.

[29] A. Shaban, S. Bansal, Z. Liu, I. Essa, and B. Boots, "One-shot learning for semantic segmentation," in *Proc. Brit. Mach. Vis. Conf.*, 2017, pp. 167.1–167.13.

[30] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[31] M. Siam and B. N. Oreshkin, "Adaptive masked weight imprinting for few-shot segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 5248–5257.

[32] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd Proc. Int. Conf. Learn. Representations*, Y. Bengio and Y. LeCun, Eds., San Diego, CA, USA, 2015.

[33] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4080–4090.

[34] K. Sun *et al.*, "Bottom-up human pose estimation by ranking heatmap-guided adaptive keypoint estimates," *CoRR*, 2020, *arXiv: abs/2006.15480*.

[35] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5686–5696.

[36] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1199–1208.

[37] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2018.

[38] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 3637–3645.

[39] J. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell., CoRR*, 2019, *arXiv: abs/1908.07919*.

[40] K. Wang, J. Liew, Y. Zou, D. Zhou, and J. Feng, "PANet: Few-shot image semantic segmentation with prototype alignment," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9196–9205.

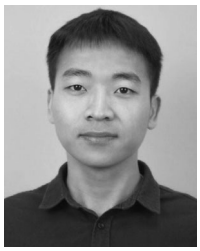
[41] X. Wang, Y. Ye, and A. Gupta, "Zero-shot recognition via semantic embeddings and knowledge graphs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6857–6866.

[42] Y. Wang, R. B. Girshick, M. Hebert, and B. Hariharan, "Low-shot learning from imaginary data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7278–7286.
[43] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *4th Proc. Int. Conf. Learn. Representations*, Y. Bengio and Y. LeCun, Eds., San Juan, Puerto Rico, 2016.
[44] Y. Yuan and J. Wang, "OCNet: Object context network for scene parsing," *CoRR*, 2018, *arXiv: abs/1809.00916*.
[45] C. Zhang, G. Lin, F. Liu, J. Guo, Q. Wu, and R. Yao, "Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9586–9594.
[46] C. Zhang, G. Lin, F. Liu, R. Yao, and C. Shen, "CANet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5212–5221.
[47] H. Zhang *et al.*, "Context encoding for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7151–7160.
[48] H. Zhang, H. Zhang, C. Wang, and J. Xie, "Co-occurrent features in semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 548–557.
[49] H. Zhang, J. Zhang, and P. Koniusz, "Few-shot learning via saliency-guided hallucination of samples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2765–2774.
[50] X. Zhang, Y. Wei, Y. Yang, and T. Huang, "SG-One: Similarity guidance network for one-shot semantic segmentation," *CoRR*, 2018, *arXiv: abs/1810.09091*.
[51] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 418–434.
[52] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6230–6239.
[53] H. Zhao *et al.*, "PSANet: Point-wise spatial attention network for scene parsing," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 270–286.
[54] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1529–1537.

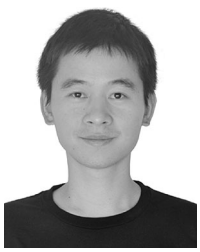
Zhuotao Tian (Student Member, IEEE) received the BEng degree in computer science and technology from the Honors School, Harbin Institute of Technology (HIT), China, in 2018. He is currently working toward the PhD degree at the Chinese University of Hong Kong (CUHK), Hong Kong, under the supervision of Prof. Jiaya Jia. His research interests include few-shot learning, segmentation, and scene text detection.

Hengshuang Zhao (Member, IEEE) received the BEng degree in information engineering from the Huazhong University of Science and Technology (HUST), China, in 2015, and the PhD degree in computer science and engineering from the Chinese University of Hong Kong (CUHK), Hong Kong, in 2019. His team won champions of ImageNet Scene Parsing Challenge, LSUN Semantic Segmentation Challenge and WAD Drivable Area Segmentation Challenge at ECCV'16, CVPR'17, and CVPR'18 respectively. He serves as a reviewer for the *IEEE Transactions on Pattern Analysis and Machine Intelligence*, the *IEEE Transactions on Image Processing*, CVPR, ICCV, ECCV, NeurIPS, ICML, ICLR, AAAI, etc. His general research interests cover the broad area of computer vision and deep learning, with special emphasis on high-level scene recognition and pixel-level scene understanding.

Michelle Shu (Student Member, IEEE) received the BS degree from Hopkins, Baltimore, Maryland, in 2019. She is currently working toward the combined bachelor's/master's degree at Johns Hopkins University, Baltimore, Maryland. During her time studying at Hopkins and interning at Tencent, she has built a wide range of interests including language and vision, scene text detection and human object interaction.



Zhicheng Yang (Member, IEEE) received the BSc and MSc degrees in pattern recognition and intelligent system from the Harbin Institute of Technology, China, in 2016 and 2019 respectively. His research interests include a range of topics including scene text recognition and text detection. From 2018 to present, he is the algorithm researcher of Smartmore, China.



Ruiyu Li (Member, IEEE) received the BS degree in computer science and technology from Sun Yat-sen University, China, in 2014, and the PhD degree in computer science and engineering from The Chinese University of Hong Kong, Hong Kong, in 2018. He is currently a senior researcher in Smartmore, China, working with Prof. Jiaya Jia. His research interests include unsupervised domain adaptation, scene text recognition, and the interplay between vision and language.

Jiaya Jia (Fellow, IEEE) received the PhD degree in computer science from the Hong Kong University of Science and Technology, Hong Kong, in 2004 and is currently a full professor with the Department of Computer Science and Engineering, Chinese University of Hong Kong (CUHK), Hong Kong. He is in the editorial boards of the *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* and the *International Journal of Computer Vision (IJCV)*. He continuously served as area chairs for ICCV, CVPR, AAAI, ECCV, and several other conferences for the organization. He was on program committees of major conferences in graphics and computational imaging, including ICCP, SIGGRAPH, and SIGGRAPH Asia.