HM: Hybrid Masking for Few-Shot Segmentation

Seonghyeon Moon[1(✉)], Samuel S. Sohn[1], Honglu Zhou[1], Sejong Yoon[2], Vladimir Pavlovic[1], Muhammad Haris Khan[3], and Mubbasir Kapadia[1]

[1] Rutgers University, Newark, NJ, USA
{sm2062,samuel.sohn,honglu.zhou,vladimir,mubbasir.kapadia}@rutgers.edu
[2] The College of New Jersey, Trenton, NJ, USA
[3] Mohamed Bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE
muhammad.haris@mbzuai.ac.ae

Abstract. We study few-shot semantic segmentation that aims to segment a target object from a query image when provided with a few annotated support images of the target class. Several recent methods resort to a feature masking (FM) technique to discard irrelevant feature activations which eventually facilitates the reliable prediction of segmentation mask. A fundamental limitation of FM is the inability to preserve the fine-grained spatial details that affect the accuracy of segmentation mask, especially for small target objects. In this paper, we develop a simple, effective, and efficient approach to enhance feature masking (FM). We dub the enhanced FM as hybrid masking (HM). Specifically, we compensate for the loss of fine-grained spatial details in FM technique by investigating and leveraging a complementary basic input masking method. Experiments have been conducted on three publicly available benchmarks with strong few-shot segmentation (FSS) baselines. We empirically show improved performance against the current state-of-the-art methods by visible margins across different benchmarks. Our code and trained models are available at: https://github.com/moonsh/HM-Hybrid-Masking

Keywords: Few-shot segmentation · Semantic segmentation · Few-shot learning

1 Introduction

Deep convolutional neural networks (DCNNs) have enabled remarkable progress in various important computer vision (CV) tasks, such as image recognition [8, 10, 13, 31], object detection [17, 27, 28], and semantic segmentation [3, 20, 43]. Despite proving effective for various CV tasks, DCNNs require a large amount of labeled training data, which is quite cumbersome and costly to acquire for

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-20044-1_29.

S. Avidan et al. (Eds.): ECCV 2022, LNCS 13680, pp. 506–523, 2022.
https://doi.org/10.1007/978-3-031-20044-1_29

dense prediction tasks, such as semantic segmentation. Furthermore, these models often fail to segment novel (unseen) objects when provided with very few annotated training images. To counter the aforementioned problems, few shot segmentation (FSS) methods, that rely on a few annotated support images, have been actively studied [1,14,18,22,24–26,30,33,35–38,40,42].

After the pioneering work of OSLSM [29], many few-shot segmentation methods have been proposed in recent years [1,14,18,22,24–26,30,33,35–38,40,42]. Among others, an important challenge in few-shot segmentation is how to use support images towards capturing more meaningful information. Many recent state-of-the-art methods [9,11,24,33,36,39,40,42] rely on feature masking (FM) [42] to discard irrelevant feature activations for reliable segmentation mask prediction. However, when masking a feature map, some crucial information in support images, such as the target object boundary, is partially lost. In particular, when the size of the target object is relatively small, this lost fine-grained spatial information renders it rather difficult to obtain accurate segmentation (see Fig. 1).

In this paper, we propose a simple, effective, and efficient technique to enhance feature masking (FM) [42]. We dub the enhanced FM as hybrid masking (HM). In particular, we compensate for the loss of target object details in FM technique through leveraging a simple input masking (IM) technique [29]. We note that IM is capable of preserving the fine details, especially around object boundaries, however, it lacks discriminative information, as such, after the removal of background information. To this end, we investigate the possibility of transferring object details in the IM to enrich FM technique. We instantiate the proposed hybrid masking (HM) into two recent strong baselines: HSNet [24] and VAT [9]. Results reveal more accurate segmentation masks by recovering the fine-grained details, such as target boundaries and target textures (see Fig. 1). Following are the main contributions of this paper:

- We propose a simple, effective, and efficient way to enhance a de-facto feature masking technique (FM) in several recent few-shot segmentation methods.
- We perform extensive experiments to validate the effectiveness of proposed hybrid masking across two strong FSS baselines, namely HSNet [24] and VAT [9] on three publicly available datasets: Pascal-5^i [29], COCO-20^i [16], and FSS-1000 [15]. Results show notable improvements against the state-of-the-art methods in all datasets.
- We note that our HM facilitates improving the training efficiency. When integrated into HSNet [24] with ResNet101 [8], it speeds up its training convergence by around 11x times on average on COCO-20^i [16].

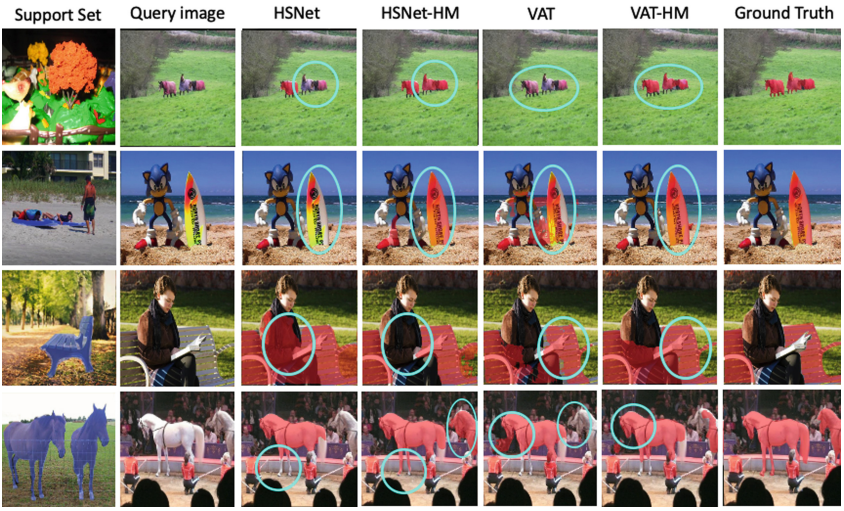

Fig. 1. Our HM allows generation of more accurate segmentation masks by recovering the fine-grained details (marked in cyan ellipse) when integrated into the current state-of-the-art methods, HSNet [24] and VAT [9] on COCO-20^i [16].

2 Related Work

Few-Shot Segmentation. The work of Shaban *et al.* [29] is believed to introduce the few shot segmentation task to the community. It generated segmentation parameters by using the conditioning branch on the support set. Later, we observe steady progress in this task, and so several methods were proposed [1,14,18,22,24–26,30,33,35–38,40,42]. CANet [40] modified the cosine similarity with the additive alignment module and enhanced the performance by performing various iterations. To improve segmentation quality, PFENet [33] designed a pyramid module and used a prior map. Inspired by prototypical networks [32], PANet [36] leveraged novel prototype alignment network. Along similar lines, PPNet [19] utilized part-aware prototypes to get the detailed object features and PMM [38] used the expectation-maximization (EM) algorithm to generate multiple prototypes. ASGNet [14] proposed two modules, superpixel-guided clustering (SGC) and guided prototype allocation (GPC) to extract and allocate multiple prototypes. In pursuit of improving correspondence between support and query images, DAN [35] democratized graph attention. Yang *et al.* [39] introduced a method to mine latent classes in the background, and CWT [22] designed a simple classifier with transformer. CyCTR [41] mined information from the whole support image using transformer. ASNet [11] proposed the integrative few-shot learning framework (iFSL) overcoming limitations of few-shot classification and few-shot segmentation. HSNet [24] utilized efficient 4D convolution to analyze deeply accumulated features and achieved remarkable performance. Recently, VAT [9] proposed a cost aggregation network, based on

transformers, to model dense semantic correspondence between images and capture intra-class variations. We validate the effectiveness of our hybrid masking approach by instantiating it in two strong FSS baselines: HSNet [24] and VAT [9].

Feature Masking. Zhang *et al.* [42] proposed Masked Average Pooling (MAP) to eliminate irrelevant feature activations which facilitates reliable mask prediction. In MAP, feature masking (FM) was introduced and utilized before average pooling. Afterward, FM was widely adopted as the de-facto technique to achieve feature masking [9,11,24,33,36,39,40,42]. We note that the FM method loses information about the target object in the process of feature masking. Specifically, it is prone to losing the fine-grained spatial details, which can be crucial for generating a precise segmentation mask. In this work, we compensate for the loss of target object details in FM technique via leveraging a simple input masking (IM) technique [29].

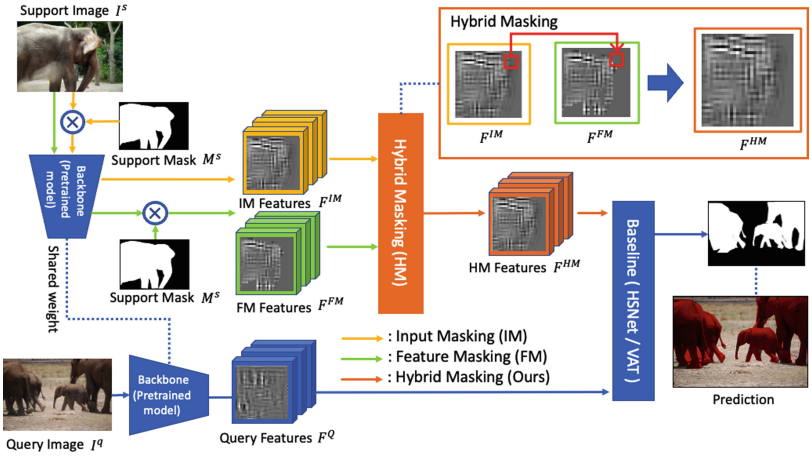

Fig. 2. The overall architecture when our proposed hybrid masking (HM) approach is integrated into FSS baselines. At its core, it contains a feature backbone, a feature masking (FM) technique. After extracting support and query features, the feature masking suppresses irrelevant activations in the support features. We introduce a simple, effective, and efficient way to enhance feature masking (FM), termed as *hybrid masking* (HM). It compensates for the loss of target object details in the FM technique by leveraging a simple input masking (IM) technique [29].

Input Masking. Input masking (IM) [29] is a technique to eliminate background pixels by multiplying the support image with its corresponding support mask. There were two key motivations behind erasing the background pixels. First, the largest object in the image has a tendency to dominate the network. Second, the variance of the output parameters increased when the background information was included in the input. We observe that IM can preserve the

fine details, however, it lacks target discriminative information, important for distinguishing between the foreground and the background. In this work, we investigate the possibility of transferring object details present in the IM to enrich the FM technique, thereby exploiting the complementary strengths of both.

3 Methodology

Figure 2 displays the overall architecture when our proposed *hybrid masking* (HM) approach is introduced into the FSS baselines, such as HSNet [24] and VAT [9]. Fundamentally, it comprises of a feature backbone for extracting support and query features, a feature masking (FM) technique for suppressing irrelevant support activations, and FSS model (i.e. HSNet/VAT) for predicting the segmentation mask from the relevant activations. In this work, we propose a simple, effective, and efficient way to enhance feature masking (FM), termed as hybrid masking (HM). It compensates for the loss of target object details in the FM technique by leveraging a simple input masking (IM) technique [29]. In what follows, we first lay out the problem setting (sect. 3.1), next we describe feature masking technique (sect. 3.2), and finally we detail the proposed hybrid masking for few-shot segmentation (sect. 3.3).

3.1 Problem Setting

Few-shot segmentation's objective is to train a model that can recognize the target object in a query image given a small sample of annotated images from the target class. We tackle this problem using the widely adopted episodic training scheme [9,24,34,36], which has been shown to reduce overfitting. We have the disjoint sets of training classes C_{train} and testing classes C_{test}. The training data D_{train} belongs to C_{train} and the testing data D_{test} is from C_{test}. Multiple episodes are constructed using the D_{train} and D_{test}. A support set, $S = (I^s, M^s)$, and a query set, $Q = (I^q, M^q)$, are the two components that make up each episode. I and M represent an image and its mask. We have N_{train} episodes for training $D_{train} = \{(S_i, Q_i)\}_{i=1}^{N_{train}}$ and N_{test} episodes for testing $D_{test} = \{(S_i, Q_i)\}_{i=1}^{N_{test}}$. Sampled episodes from D_{train} are used to train a model to predict query mask M^q. Afterward, the learned model is evaluated by randomly sampling episodes from the testing data D_{test} in the same manner and comparing the predicted query masks to the ground truth.

3.2 Feature Masking

Zhang *et al.* [42] argued that IM greatly increases the variance of the input data for a unified network, and according to Long *et al.* [21], the relative positions of input pixels can be preserved by fully convolutional networks. These two ideas motivated Masked Average Pooling (MAP), which ideally extracts the features of a target object while eliminating the background content. Although MAP falls short of this in practice, it remains helpful for learning better object features [3] while keeping the input structure of the network unchanged. In particular, the feature masking part is still widely used.

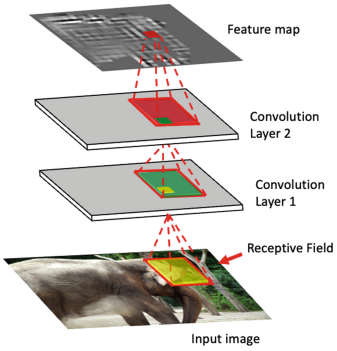

Fig. 3. Impact of growing receptive field on feature masking. The input image's elephant is a target object and the other pixels are background. One pixel at the feature is generated from lots of pixels' information from previous layer. The background and target object information both can be present in one feature map pixel.

Given a support RGB image $I^s \in \mathbb{R}^{3\times w\times h}$ and a support mask $\mathbf{M}^s \in \{0,1\}^{w\times h}$, where w and h are the width and height of the image, the support feature maps of I^s are $F^s \in \mathbb{R}^{c\times w'\times h'}$, where c is the number of channels, w' and h' are the width and height of the feature maps. Feature masking is performed after matching the mask to the feature size using the bilinear interpolation. We denote a function resizing the mask as $\tau(\cdot) : \mathbb{R}^{w\times h} \rightarrow \mathbb{R}^{c\times w'\times h'}$. Then, the feature masking features $F^{FM} \in \mathbb{R}^{c\times w'\times h'}$ are computed according to Eq. 1,

$$F^{FM} = F^s \odot \tau(M^s), \tag{1}$$

where $\odot$ denotes the Hadamard product. Zhang *et al.* [42] fit the feature size to the mask size, but we conversely fit the mask to the feature size.

Feature masking (FM), which forms the core part of Masked Average Pooling (MAP) [42], is utilized to eliminate background information from support features and has become the de facto technique for masking feature maps, appearing in several recent few-shot segmentation methods [33,36,39,40,42] even in current state-of-the-art [24]. However, FM inadvertently eliminates both the background and the target object information because one pixel from the last layer's feature map corresponds to many pixels in the input image.

Figure 3 shows that one pixel of the feature map could contain background and target object information together [23]. This is further analyzed in Fig. 5, which clearly shows that FM loses useful information through its masking and progressively worsens with deeper layers. If a target object in the support set appears very small, the segmentation of the query image becomes even more challenging because the features are fed into network with relatively large proportion of undesired background features. Figure 4 shows the limitation of FM.

3.3 Hybrid Masking for Few-shot Segmentation (HM)

We aim to maximize target information from the support set so that the network can efficiently learn to provide more accurate segmentation of a target object.

The overall architecture when our proposed HM is integrated into a FSS baseline is shown in Fig. 2. We obtain two feature maps, F^{IM} and F^{FM}, using IM and FM respectively. These two feature maps are merged by hybrid masking (Algorithm 1) to generate HM feature map F^{HM}. This HM feature map will be used as input for HSNet and VAT, which takes full advantage of the given features to predict the target object mask.

Input Masking (IM). IM [29] eliminates background pixels by multiplying the support image with its corresponding support mask because of two empirical reasons. (1) The network has a tendency to favor the largest object in the image, which is not the object we want to segment. (2) The background information will result in an increase in the variance of the output parameters.

Suppose we have the RGB support image $I^s \in \mathbb{R}^{3\times w\times h}$ and a support mask $\mathbf{M}^s \in \{0,1\}^{w\times h}$ in the image space, where w and h are the width and height of the image. IM, computed as

$$I^{s'} = I^s \odot \tau(M^s), \tag{2}$$

contains the target object alone. We use the function $\tau(\cdot)$ for resizing the mask M^s to fit the image I^s.

Hybrid Masking (HM). We propose an alternative masking approach, which takes advantage of the features generated by both FM and IM. First, FM and IM features are computed according to the existing methods. The unactivated values in the FM features are then replaced with IM features. Other activated values remain without replacing to maintain FM features. We name this process as hybrid masking (Algorithm 1). HM prioritizes the information from FM features and supplements the lacking information, such as the precise target boundaries and fine-grained texture information, from IM features, which is superior for delineating the boundaries of target objects and the missing texture information. The method is as follows even if feature maps are stacked to have a sequence of intermediate feature maps. Only one more loop is needed for the deep feature maps.

Algorithm 1: Hybrid Masking

Input : IM feature maps F^{IM} and FM features maps F^{FM}
Each channel i, $f_i^{IM} \in F^{IM}$ and $f_i^{FM} \in F^{FM}$
for i =1, ..., c **do**
  Set $f_i^{HM} = f_i^{FM}$
  for *Entire pixels* $\in f_i^{HM}$ **do**
    Find an inactive pixel, $p \in f_i^{HM}$
    if $p < 0$ **then**
      Replace the pixel, p, with corresponding pixel $\in f_i^{IM}$
    end
  end
end
Output: HM feature maps F^{HM}

The generated HM feature maps are used as inputs to two strong FSS models (HSNet and VAT). These two models are best-suited for fully utilizing the features generated by hybrid masking, because they build multi-level correlation maps taking advantage of the rich semantics that are provided at different feature levels.

4 Experiment

4.1 Setup

Datasets. We evaluate the efficacy of our hybrid masking technique on three publicly available segmentation benchmarks: PASCAL-5^i [29] , COCO-20^i [16], and FSS-1000 [15]. PASCAL-5^i was produced from PASCAL VOC 2012 [6] with additional mask annotations [7]. PASCAL-5^i contains 20 types of object classes, COCO-20^i contains 80 classes, and FSS-1000 contains 1000 classes. The PASCAL and COCO data sets were divided into four folds following the training and evaluation methods of other works [9,11,19,24,25,33,35,38], where each fold of PASCAL-5^i consisted of 5 classes, and each fold of COCO-20^i had 20. We conduct cross-validation using these four folds. When evaluating a model on fold^i, all other classes not belonging to fold^i are used for training. 1000 episodes are sampled from the other fold^i to evaluate the trained model. For FSS-1000, the training, validation, and test datasets are divided into 520, 240, and 240 classes.

Implementation Details. We integrate our hybrid masking technique into two FSS baselines: HSNet [24] and VAT [9], and the resulting methods are denoted by HSNet-HM and VAT-HM. We use ResNet50 [8] and ResNet101 [8] backbone networks pre-trained on ImageNet [5] with their weights frozen to extract features, following HSNet [24] and VAT [9]. From conv3_x to conv5_x of ResNet (i.e., the three layers before global average pooling), the features in the bottleneck part before the ReLU activation of each layer were stacked up to create deep

features. We follow the HSNet [24] and the VAT [9] default settings for optimizer [12] and learning rate. A batch size of 20 is used for HSNet-HM training for all benchmarks. For VAT-HM training, 8, 4, and 4 batch sizes are utilized for COCO-20^i, PASCAL-5^i and FSS-1000 respectively. We used data augmentation for HSNet-HM training on PASCAL-5^i following [2,4,9]. For COCO-20^i and FSS-1000 benchmarks, no data augmentation was employed when training HSNet-HM.

Evaluation Metrics. Following [9,24,33,35], we adopt two evaluation metrics, mean intersection over union (mIoU) and foreground-background IoU (FB-IoU) for model evaluation. The mIoU averages the IoU values for all classes in each fold. FB-IoU calculates the foreground and background IoU values ignoring object classes and averages them. Note that, mIoU is a better indicator of model generalization than FB-IoU [24].

4.2 Comparison with the State-of-the-Art (SOTA)

PASCAL-5^i. Table 1 compares our methods, HSNet-HM and VAT-HM, with other methods on PASCAL-5^i datasets. In the 1-shot test, HSNet-HM provides a gain of 0.7% mIoU compared to HSNet [24] with ResNet50 backbone and performs on par with HSNet [24] using ReNet101 backbone. In the 5-shot test, HSNet-HM shows slightly inferior performance in mIoU and FB-IoU. VAT-HM shows a similar pattern to HSNet-HM. In the 1-shot test, VAT-HM shows a gain of 0.5% mIoU with ResNet50 and a gain of 0.3% mIoU with ResNet101. In the 5-shot test, VAT-HM provides an improvement of 0.8% mIoU with ResNet50.

COCO-20^i. Table 2 reports results on the COCO-20^i dataset. In 1-shot test, HSNet-HM, VAT-HM, and ASNet-HM show a significant improvement over HSNet [24], VAT [9], and ASNet [11]. HSNet-HM delivers a gain of 5.1% and 5.3% in mIoU with ResNet50 and ResNet101 backbones, respectively. VAT-HM provides a gain of 2.9% mIoU with ResNet50. ASNet-HM provides a gain of 2.5% and 2.8% mIoU with ResNet50 and ResNet101. Similarly, in 5-shot test, HSNet-HM outperforms HSNet [24] by 3.5% and 1.1% with ResNet50 and ResNet101 backbones, respectively. VAT-HM provides 0.4% mIoU improvement with ResNet50 in 5-shot test. ASNet-HM shows slightly worse performance with ResNet50 but ASNet-HM delivers a gain of 1.1% with ResNet101. Figure 1 draws visual comparison with HSNet [24] and VAT [9] under several challenging segmentation instances. Note that, compared to HSNet and VAT, HSNet-HM and VAT-HM produce more accurate segmentation masks that recover fine-grained details under appearance variations and complex backgrounds.

Table 1. Performance comparison with the existing methods on Pascal-5^i [6]. Superscript asterisk denotes that data augmentation was applied during training. Best results are bold-faced and the second best are underlined.

Backbone feature	Methods	1-shot						5-shot					
		5^0	5^1	5^2	5^3	mIoU	FB-IoU	5^0	5^1	5^2	5^3	mIoU	FB-IoU
ResNet50 [8]	PANet [36]	44.0	57.5	50.8	44.0	49.1	–	55.3	67.2	61.3	53.2	59.3	–
	PFENet [33]	61.7	69.5	55.4	56.3	60.8	73.3	63.1	70.7	55.8	57.9	61.9	73.9
	ASGNet [14]	58.8	67.9	56.8	53.7	59.3	69.2	63.4	70.6	64.2	57.4	63.9	74.2
	CWT [22]	56.3	62.0	59.9	47.2	56.4	–	61.3	68.5	68.5	56.6	63.7	–
	RePRI [1]	59.8	68.3	62.1	48.5	59.7	–	64.6	71.4	**71.1**	59.3	66.6	–
	CyCTR [41]	67.8	**72.8**	58.0	58.0	64.2	–	71.1	73.2	60.5	57.5	65.6	–
	HSNet [24]	64.3	70.7	60.3	60.5	64.0	76.7	70.3	73.2	67.4	**67.1**	69.5	80.6
	HSNet*	63.5	70.9	61.2	60.6	64.3	**78.2**	70.9	73.1	68.4	65.9	69.6	80.6
	VAT [9]	67.6	71.2	**62.3**	60.1	65.3	77.4	72.4	**73.6**	68.6	65.7	**70.0**	**80.9**
	HSNet*-HM	**69.0**	70.9	59.3	61.0	65.0	76.5	69.9	72.0	63.4	63.3	67.1	77.7
	VAT-HM	68.9	70.7	61.0	**62.5**	**65.8**	77.1	**71.1**	72.5	62.6	66.5	68.2	78.5
ResNet101 [8]	FWB [25]	51.3	64.5	56.7	52.2	56.2	–	54.8	67.4	62.2	55.3	59.9	–
	DAN [35]	54.7	68.6	57.8	51.6	58.2	71.9	57.9	69.0	60.1	54.9	60.5	72.3
	PFENet [33]	60.5	69.4	54.4	55.9	60.1	72.9	62.8	70.4	54.9	57.6	61.4	73.5
	ASGNet [14]	59.8	67.4	55.6	54.4	59.3	71.7	64.6	71.3	64.2	57.3	64.4	75.2
	CWT [22]	56.9	65.2	61.2	48.8	58.0	–	62.6	70.2	**68.8**	57.2	64.7	–
	RePRI [1]	59.6	68.6	62.2	47.2	59.4	–	66.2	71.4	67.0	57.7	65.6	–
	CyCTR [41]	69.3	**72.7**	56.5	58.6	64.3	72.9	73.5	74.0	58.6	60.2	66.6	75.0
	HSNet [24]	67.3	72.3	62.0	63.1	66.2	77.6	71.8	74.4	67.0	68.3	70.4	80.6
	HSNet*	67.5	**72.7**	63.5	63.2	66.7	77.7	71.7	74.8	68.2	68.7	70.8	80.9
	VAT [9]	68.4	72.5	**64.8**	64.2	67.5	78.8	73.3	75.2	68.4	**69.5**	**71.6**	**82.0**
	HSNet*-HM	69.8	72.1	60.4	64.3	66.7	77.8	72.2	73.3	64.0	67.9	69.3	79.7
	VAT-HM	**71.2**	**72.7**	62.7	**64.5**	**67.8**	**79.4**	**74.0**	**75.5**	65.4	68.6	70.9	81.5

FSS-1000. Table 3 compares HSNet-HM, VAT-HM, and competing methods on the FSS-1000 dataset [15]. In the 1-shot test, HSNet-HM yields a gain of 1.6% and 1.3% in mIoU over [24] with ResNet50 and ResNet101 backbones, respectively. In the 5-shot test, we observe an improvement of 0.2% in mIoU over [24] with the ResNet50 backbone. In the 1-shot test, VAT-HM shows slightly inferior mIoU compared to VAT [9] with ResNet50 but it performs a little better than VAT [9] with ResNet101.

Generalization Test. Following previous works [1,24], we perform a domain shift test to evaluate the generalization capability of the proposed method. We trained HSNet-HM and VAT-HM on the COCO-20^i dataset and tested this model on the PASCAL-5^i dataset. The training/testing folds were constructed following [1,24]. The objects in training classes do not overlap with the object in the testing classes. As shown in Table 4, HSNet-HM outperforms the current state-of-the-art approaches under both 1-shot and 5-shot tests. In 1-shot test, it delivers a 2% mIoU gain over RePRI [1] and a 2.4% mIoU gain over HSNet [24] with ResNet50 and ResNet101 backbones, respectively. In the 5-shot test,

Table 2. Performance comparison on COCO-20^i [16] in mIoU and FB-IoU. Best results are bold-faced and the second best are underlined.

Backbone feature	Methods	1-shot						5-shot					
		20^0	20^1	20^2	20^3	mIoU	FB-IoU	20^0	20^1	20^2	20^3	mIoU	FB-IoU
ResNet50 [8]	PMM [38]	29.3	34.8	27.1	27.3	29.6	–	33.0	40.6	30.3	33.3	34.3	–
	RPMM [38]	29.5	36.8	28.9	27.0	30.6	–	33.8	42.0	33.0	33.3	35.5	–
	PFENet [33]	36.5	38.6	34.5	33.8	35.8	–	36.5	43.3	37.8	38.4	39.0	–
	ASGNet [14]	–	–	–	–	34.6	60.4	–	–	–	–	42.5	67.0
	RePRI [1]	32.0	38.7	32.7	33.1	34.1	–	39.3	45.4	39.7	41.8	41.6	–
	HSNet [24]	36.3	43.1	38.7	38.7	39.2	68.2	43.3	51.3	48.2	45.0	46.9	70.7
	CyCTR [41]	38.9	43.0	39.6	39.8	40.3	–	41.1	48.9	45.2	47.0	45.6	–
	VAT [9]	39.0	43.8	42.6	39.7	41.3	68.8	44.1	51.1	50.2	46.1	47.9	72.4
	ASNet [11]	41.5	44.1	42.8	40.6	42.2	69.4	**48.0**	52.1	49.7	48.2	**49.5**	**72.7**
	HSNet-HM	41.0	45.7	**46.9**	43.7	44.3	**70.8**	45.3	**53.1**	**52.1**	47.0	49.4	72.2
	VAT-HM	42.2	43.3	45.0	42.2	43.2	70.0	45.2	51.0	50.7	46.4	48.3	71.8
	ASNet-HM	**42.8**	**46.0**	44.8	**45.0**	**44.7**	70.4	46.3	50.2	48.4	**48.6**	48.4	72.2
ResNet101 [8]	FWB [25]	17.0	18.0	21.0	28.9	21.2	–	19.1	21.5	23.9	30.1	23.7	–
	DAN [35]	–	–	–	–	24.4	62.3	–	–	–	–	29.6	63.9
	PFENet [33]	36.8	41.8	38.7	36.7	38.5	63.0	40.4	46.8	43.2	40.5	42.7	65.8
	HSNet [24]	37.2	44.1	42.4	41.3	41.2	69.1	45.9	53.0	51.8	47.1	49.5	72.4
	ASNet [11]	41.8	45.4	43.2	41.9	43.1	69.4	**48.0**	52.1	49.7	48.2	49.5	72.7
	HSNet-HM	41.2	**50.0**	**48.8**	45.9	**46.5**	**71.5**	46.5	**55.2**	51.8	48.9	**50.6**	72.9
	ASNet-HM	**43.5**	46.4	47.2	**46.4**	45.9	71.1	47.7	51.6	**52.1**	**50.8**	**50.6**	**73.3**

Table 3. Performance comparison with other methods on FSS-1000 [15] dataset. Best results are bold-faced and the second best are underlined.

Backbone feature	Methods	mIoU		Backbone feature	Methods	mIoU	
		1-shot	5-shot			1-shot	5-shot
ResNet50 [8]	FSOT [18]	82.5	83.8	ResNet101 [8]	DAN [35]	85.2	88.1
	HSNet [24]	85.5	87.8		HSNet [24]	86.5	88.5
	VAT [9]	**89.5**	**90.3**		VAT [9]	90.0	**90.6**
	HSNet-HM	87.1	88.0		HSNet-HM	87.8	88.5
	VAT-HM	89.4	89.9		VAT-HM	**90.2**	90.5

Table 4. Comparison of generalization performance with domain shift test. A model was trained on COCO-20^i [16] and then evaluated on PASCAL-5^i [6].

Backbone feature	Methods	1-shot					5-shot				
		5^0	5^1	5^2	5^3	mIoU	5^0	5^1	5^2	5^3	mIoU
ResNet50 [8]	RPMM [38]	36.3	55.0	52.5	54.6	49.6	40.2	58.0	55.2	61.8	53.8
	PFENet [33]	43.2	65.1	66.5	69.7	61.1	45.1	66.8	68.5	73.1	63.4
	RePRI [1]	**52.2**	64.3	64.8	71.6	63.2	56.5	68.2	70.0	76.2	67.7
	HSNet [24]	45.4	61.2	63.4	75.9	61.6	56.9	65.9	71.3	80.8	68.7
	VAT	52.1	64.1	67.4	74.2	64.5	**58.5**	68.0	72.5	79.9	**69.7**
	HSNet-HM	43.4	**68.2**	**69.4**	**79.9**	**65.2**	50.7	**71.4**	**73.4**	**83.1**	**69.7**
	VAT-HM	48.3	64.9	67.5	79.8	65.1	55.6	68.1	72.4	82.8	**69.7**
ResNet101 [8]	HSNet [24]	**47.0**	65.2	67.1	77.1	64.1	**57.2**	69.5	72.0	82.4	70.3
	HSNet-HM	46.7	**68.6**	**71.1**	**79.7**	**66.5**	53.7	**70.7**	**75.2**	**83.9**	**70.9**

HSNet-HM outperforms HSNet [24] by 1.0% and 0.6% in mIoU with ResNet50 and ResNet101 backbones, respectively.

4.3 Ablation Study and Analysis

Comparison of the Three Different Masking Approaches. We compare all three masking approaches, IM [29], FM [42], and the proposed HM after incorporating them into HSNet [24] and evaluate them on the COCO-20^i dataset (Table 5). We can see that in both 1-shot and 5-shot tests, the proposed HM approach provides noticeable gains over either individual FM and IM techniques.

Table 5. Ablation study of the three different masking methods on COCO-20^i [16].

Backbone feature	Masking methods	1-shot						5-shot					
		20^0	20^1	20^2	20^3	mIoU	FB-IoU	20^0	20^1	20^2	20^3	mIoU	FB-IoU
ResNet50 [8]	HSNet-FM [24]	36.3	43.1	38.7	38.7	39.2	68.2	43.3	51.3	48.2	45.0	46.9	70.7
	HSNet-IM	39.8	45.0	46.0	43.2	43.5	70.0	43.4	50.9	49.5	**48.0**	47.6	71.7
	HSNet-HM	**41.0**	45.7	46.9	**43.7**	44.3	70.8	**45.3**	53.1	**52.1**	47.0	49.4	72.2
ResNet101 [8]	HSNet-FM [24]	37.2	44.1	42.4	41.3	41.2	69.1	45.9	53.0	**51.8**	47.1	49.5	72.4
	HSNet-IM	41.0	48.3	47.3	44.5	45.2	70.9	**46.6**	54.5	50.4	47.7	49.8	72.7
	HSNet-HM	**41.2**	**50.0**	**48.8**	**45.9**	**46.5**	**71.5**	46.5	**55.2**	**51.8**	**48.9**	**50.6**	**72.9**

Table 6. Ablation study of the three different merging methods on COCO-20^i

Feature backbone	Methods	1-shot				
		20^0	20^1	20^2	20^3	mIoU
ResNet50	HSNet-HM (Simple Add.)	40.0	43.5	43.4	43.2	42.5
	HSNet-HM (Reverse)	39.4	45.2	42.3	41.6	42.1
	HSNet-HM	**41.0**	**45.7**	**46.9**	**43.7**	**44.3**

Table 7. Run-time comparison at inference stage on COCO-20^i [16].

Inference Time	Secs/ Image	Additional Overhead in %
HSNet-FM	0.27	–
HSNet-HM	0.34	25.9

Figure 4 shows the qualitative results from the three masking methods. The blue objects in the support set are the target objects for segmentation. The red pixels are the segmentation results. FM can coarsely segment the objects from the background but fails to precisely recover target details, such as target boundaries. IM is capable of recovering precise object boundaries, but struggles in distinguishing objects from the background. The proposed approach, HM, clearly distinguishes between the target objects and the background and also recovers precise details such as, target boundaries.

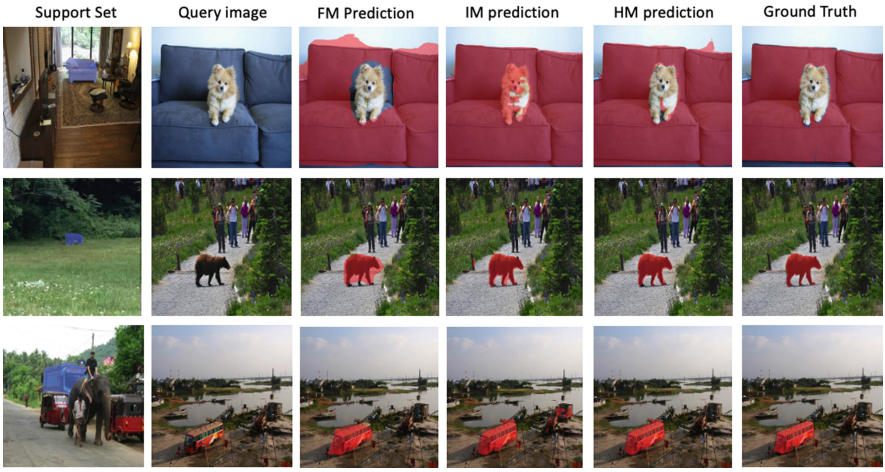

Fig. 4. Qualitative comparison of three different masking approaches on COCO-20^i [16] with HSNet. The blue objects in the support set are the target objects for segmentation. The red pixels are the segmentation results. HSNet-FM can coarsely segment the objects from the background but fails to precisely recover target details, such as target boundaries. HSNet-IM is capable of recovering precise object boundaries, but struggles in distinguishing objects from the background. The proposed approach, HSNet-HM, clearly distinguishes between the target objects and the background and also recovers precise details such as, target boundaries.

Figure 5 shows the visual comparison between the feature maps of IM and FM features. The feature maps inside the red rectangles reveal that the two features produced from the two masking approaches are different. Looking at the area where activations occur in the IM feature map at layer 50, we can see it is more indicative of the target object boundaries than the FM feature. Additionally, looking at the IM feature map at layer 34, we observe that there is a strong signal around the edge and even in side of the target object. This happens because FM performs masking after extracting features, and so this results in less precise target boundaries and loss of texture information.

Other Combination Proposals for Obtaining HM. We apply various masking methods to create HM features. Various mask sizes are tested by applying dilation to the mask of the support set, but the most plausible result is obtained with the IM masking method. Also, the method in [42] obtains a mask using the bounding box and applies the average pooling method, but fails to achieve better performance than FM. We provide two ablation studies to understand the effectiveness of replacement operation (see Table 6). First, we simply add the corresponding feature maps of IM and FM, denoted as HM(Simple Add). Second, we perform the reverse procedure of the proposed HM. We initialize the HM by IM, and supplement the inactivated features with FM features, denoted

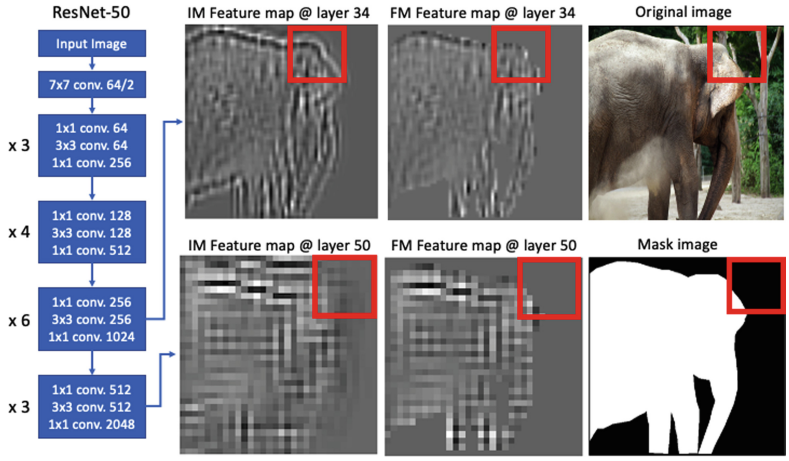

Fig. 5. Visual comparison between the feature maps of IM and FM. These are from ResNet50 at layer34 and layer50. We visualize the first channel of the feature map in grayscale. The feature maps inside the red rectangles reveal that the features from the two feature masking methods are different. Observe activations in the IM feature map at layer 50, it is more indicative of the target object boundaries than the FM feature. Additionally, the IM feature map at layer 34 displays a strong signal around the edge.

as HM(Reverse). Note that, our proposed HM is more effective for FSS compared to HM(Simple Add) and HM(Reverse).

Training Efficiency. Figure 6 shows the training profiles of HSNet-HM on COCO-20^i. We see that HM results in faster training convergence compared to HSNet, reducing the training time by a factor of 11x on average. To reach the best model with ResNet101 on COCO-20^3, 296.5 epochs are required for HSNet [24] but HSNet-HM only needs 26.8 epochs on average. A similar trend was observed in the PASCAL-5^i and FSS-1000 datasets, for which the results are reported in the supplementary material.

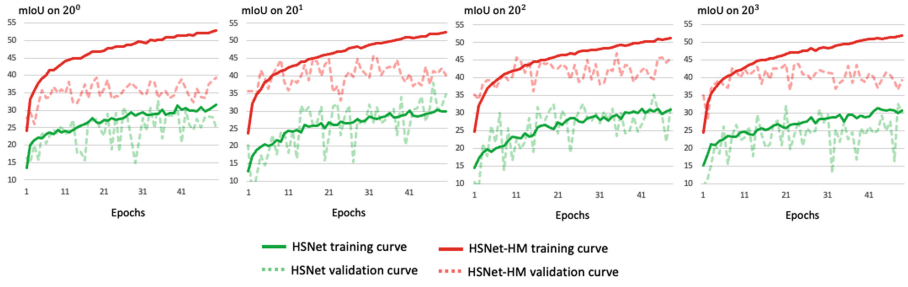

Fig. 6. Training profiles of HSNet [24] and HSNet-HM on COCO-20^i.

Runtime Comparison. Hybrid masking takes an additional pass over the pixel values to choose between FM and IM. We measure the computation time of the HM method and other methods for comparison. IM/FM take 0.05 secs/image, and their throughput is 20 images/sec. Whereas HM takes 0.07 secs/image. Therefore, HM induces 40% more computational time when compared to IM/FM. However, in terms of the model's inference time, our HM adds a relatively less extra overhead (25.9%) on top of HSNet (with FM) (see Table 7).

Limitations. We found that HSNet-HM performance in PASCAL-5^i [6] was inferior to the performance of the COCO-20^i [16] dataset. A potential reason is that HSNet-HM quickly enters the over-fitting phase due to abundance of information about the target object. The following data augmentation method [2, 4, 9] was able to alleviate this problem to some extent, but it did not solve the problem completely. Further, we identify some failure cases for HM (Fig. 7). HM struggles when the target is occluded due to small objects. Also, when the appearance/shape of the target image of the support set and the target image of the query image are radically different.

Fig. 7. Although HSNet-HM/VAT-HM improves mIoU compared to baselines on COCO-20^i [16], its performance can be further improved. We identify cases where it struggles to produce accurate segmentation masks (shown as cyan ellipse).

5 Conclusion

We proposed a new effective masking approach, termed as hybrid masking. It aims to enhance the feature masking (FM) technique, that is commonly used in existing SOTA methods. We instantiate HM in strong baselines and the results reveal that utilizing HM surpasses the existing SOTA by visible margins and also improves training efficiency.

Acknowledgement. This work was supported in part by NSF IIS Grants #1955404 and #1955365.

References

1. Boudiaf, M., Kervadec, H., Masud, Z.I., Piantanida, P., Ben Ayed, I., Dolz, J.: Few-shot segmentation without meta-learning: a good transductive inference is all you need? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13979–13988 (2021)
2. Buslaev, A., Iglovikov, V., Khvedchenya, E., Parinov, A., Druzhinin, M., Kalinin, A.: Albumentations: fast and flexible image augmentations. Information **11**(2), 125 (2020). https://doi.org/10.3390/info11020125
3. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. **40**(4), 834–848 (2018). https://doi.org/10.1109/TPAMI.2017.2699184
4. Cho, S., Hong, S., Jeon, S., Lee, Y., Sohn, K., Kim, S.: Cats: cost aggregation transformers for visual correspondence. In: Thirty-Fifth Conference on Neural Information Processing Systems (2021)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 248–255 (2009). https://doi.org/10.1109/CVPR.2009.5206848
6. Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. Int. J. Comput. Vis. **88**, 303–338 (2010). https://doi.org/10.1007/s11263-009-0275-4
7. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8695, pp. 297–312. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10584-0_20
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.90
9. Hong, S., Cho, S., Nam, J., Kim, S.: Cost aggregation is all you need for few-shot segmentation. arXiv preprint arXiv:2112.11685 (2021)
10. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4700–4708 (2017)
11. Kang, D., Cho, M.: Integrative few-shot learning for classification and segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
12. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015, Conference Track Proceedings (2015). https://arxiv.org/abs/1412.6980
13. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems 25 (2012)
14. Li, G., Jampani, V., Sevilla-Lara, L., Sun, D., Kim, J., Kim, J.: Adaptive prototype learning and allocation for few-shot segmentation. In: CVPR (2021)
15. Li, X., Wei, T., Chen, Y.P., Tai, Y.W., Tang, C.K.: Fss-1000: a 1000-class dataset for few-shot segmentation. In: CVPR (2020)
16. Lin, T.Y., et al.: Microsoft COCO: Common objects in context (2015)

17. Liu, W., et al.: SSD: Single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
18. Liu, W., Zhang, C., Ding, H., Hung, T.Y., Lin, G.: Few-shot segmentation with optimal transport matching and message flow. arXiv preprint arXiv:2108.08518 (2021)
19. Liu, Y., Zhang, X., Zhang, S., He, X.: Part-aware prototype network for few-shot semantic segmentation (2020)
20. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3431–3440. IEEE Computer Society, Los Alamitos, CA, USA (2015). https://doi.org/10.1109/CVPR.2015.7298965
21. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3431–3440. IEEE Computer Society, Los Alamitos, CA, USA (2015). https://doi.org/10.1109/CVPR.2015.7298965
22. Lu, Z., He, S., Zhu, X., Zhang, L., Song, Y.Z., Xiang, T.: Simpler is better: few-shot semantic segmentation with classifier weight transformer. In: ICCV (2021)
23. Luo, W., Li, Y., Urtasun, R., Zemel, R.: Understanding the effective receptive field in deep convolutional neural networks. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, pp. 4905–4913. NIPS2016, Curran Associates Inc., Red Hook, NY, USA (2016)
24. Min, J., Kang, D., Cho, M.: Hypercorrelation squeeze for few-shot segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6941–6952 (2021)
25. Nguyen, K., Todorovic, S.: Feature weighting and boosting for few-shot segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
26. Rakelly, K., Shelhamer, E., Darrell, T., Efros, A.A., Levine, S.: Conditional networks for few-shot semantic segmentation. In: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, 30 April - 3 May 2018, Workshop Track Proceedings. OpenReview.net (2018). https://openreview.net/forum?id=SkMjFKJwG
27. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp. 779–788 (2016)
28. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. Advances in neural information processing systems 28 (2015)
29. Shaban, A., Bansal, S., Liu, Z., Essa, I., Boots, B.: One-shot learning for semantic segmentation. In: Kim, T.-K., Zafeiriou, G.B.S., Mikolajczyk, K. (eds.) Proceedings of the British Machine Vision Conference (BMVC), pp. 1–13. BMVA Press (2017). https://doi.org/10.5244/C.31.167
30. Siam, M., Oreshkin, B.N., Jägersand, M.: AMP: adaptive masked proxies for few-shot segmentation. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), 27 Oct - 2 Nov 2019, pp. 5248–5257. IEEE (2019). https://doi.org/10.1109/ICCV.2019.00535
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015, Conference Track Proceedings (2015). https://arxiv.org/abs/1409.1556

32. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. Advances in neural information processing systems 30 (2017)
33. Tian, Z., Zhao, H., Shu, M., Yang, Z., Li, R., Jia, J.: Prior guided feature enrichment network for few-shot segmentation. In: TPAMI (2020)
34. Vinyals, O., Blundell, C., Lillicrap, T., kavukcuoglu, k., Wierstra, D.: Matching networks for one shot learning. In: Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 29. Curran Associates, Inc. (2016). https://proceedings.neurips.cc//paper/2016/file/90e1357833654983612fb05e3ec9148c-Paper.pdf
35. Wang, H., Zhang, X., Hu, Y., Yang, Y., Cao, X., Zhen, X.: Few-Shot Semantic Segmentation with Democratic Attention Networks. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12358, pp. 730–746. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58601-0_43
36. Wang, K., Liew, J.H., Zou, Y., Zhou, D., Feng, J.: PANet: few-shot image semantic segmentation with prototype alignment. In: The IEEE International Conference on Computer Vision (ICCV) (2019)
37. Xie, G.S., Liu, J., Xiong, H., Shao, L.: Scale-aware graph neural network for few-shot semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5475–5484 (2021)
38. Yang, B., Liu, C., Li, B., Jiao, J., Ye, Q.: Prototype mixture models for few-shot semantic segmentation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12353, pp. 763–778. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58598-3_45
39. Yang, L., Zhuo, W., Qi, L., Shi, Y., Gao, Y.: Mining latent classes for few-shot segmentation. In: ICCV (2021)
40. Zhang, C., Lin, G., Liu, F., Yao, R., Shen, C.: CANet: class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
41. Zhang, G., Kang, G., Yang, Y., Wei, Y.: Few-shot segmentation via cycle-consistent transformer (2021)
42. Zhang, X., Wei, Y., Yang, Y., Huang, T.: SG-One: similarity guidance network for one-shot semantic segmentation. IEEE Trans. Cybern. **50**, 3855–3865 (2020)
43. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6230–6239 (2017). https://doi.org/10.1109/CVPR.2017.660