Blessing Few-Shot Segmentation via Semi-Supervised Learning with Noisy Support Images

Runtong Zhang[a], Hongyuan Zhu[b], Hanwang Zhang[c], Chen Gong[d], Joey Tianyi Zhou[e], Fanman Meng[a,*]

[a] *University of Electronic Science and Technology of China, Chengdu, China*
[b] *Institute for Infocomm Research (I^2R) & Centre for Frontier AI Research (CFAR), A*STAR, Singapore*
[c] *Nanyang Technological University, Singapore*
[d] *Nanjing University of Science and Technology, Nanjing, China*
[e] *Centre for Frontier AI Research (CFAR), A*STAR, Singapore*

Abstract

Mainstream few-shot segmentation methods meet performance bottleneck due to the data scarcity of novel classes with insufficient intra-class variations, which results in a biased model primarily favoring the base classes. Fortunately, owing to the evolution of the Internet, an extensive repository of unlabeled images has become accessible from diverse sources such as search engines and publicly available datasets. However, such unlabeled images are not a free lunch. There are noisy inter-class and intra-class samples causing severe feature bias and performance degradation. Therefore, we propose a semi-supervised few-shot segmentation framework named **F4S**, which incorporates a ranking algorithm designed to eliminate noisy samples and select superior pseudo-labeled images, thereby fostering the improvement of few-shot segmentation within a semi-supervised paradigm. The proposed F4S framework can not only enrich the intra-class variations of novel classes during the test phase, but also enhance meta-learning of the network during the training phase. Furthermore, it can be readily implemented with ease on any off-the-shelf few-shot segmentation methods. Additionally, based on a Structural Causal Model (SCM), we further theoretically explain why the proposed method can solve the noise problem: the severe noise effects are removed by

*Corresponding author
Email address: fmmeng@uestc.edu.cn (Fanman Meng)

cutting off the backdoor path between pseudo labels and noisy support images via causal intervention. On PASCAL-5^i and COCO-20^i datasets, we show that the proposed F4S can boost various popular few-shot segmentation methods to new state-of-the-art performances.

Keywords: Few-shot segmentation, Semi-supervised learning, Noisy images, Causal inference

1. Introduction

Few-shot segmentation (FSS) [1] aims to segment the object regions in query images of novel classes using a minimal number (N-shot) of annotated support images. The most common experimental settings for FSS use 1-shot and 5-shot annotated support samples, as shown in Fig. 1 (a) and (b). The primary challenge for FSS is how to effectively utilize the information provided by the N-shot support images. Prototype-based approaches [2, 3, 4, 5, 6] focus on generating representative prototypes from the N-shot support images to accurately characterize the novel classes. In contrast, the metric-based approaches [7, 8, 9] focus on learning a class-agnostic similarity metric that can precisely measure the regions similar to the N-shot support regions in the query image. However, the most significant challenge of few-shot learning is how to maximize the exploration of data distributions under data scarcity [10]. Increasing manually annotated data is the most direct and effective method, but it is extremely time and labor-consuming.

Thanks to semi-supervised learning (SSL), the pseudo-labeling methods have provided a practical solution for the data scarcity issue in few-shot learning tasks, and there is already relevant research work published on this. For example, the method in [11] combines semi-supervised learning with few-shot classification and proposes the PLCM network, which generates and selects good pseudo labels based on loss distribution to enrich the dataset. the method in [12] proposes a semi-supervised few-shot segmentation method in remote sensing cases, which generates pseudo labels on super-pixels of backgrounds for mining latent features to enhance the network's generalization capacity. The method in [13] combines semi-supervised learning with few-shot object detection and proposes the APLDet network, which utilizes a teacher model adaptively generating pseudo labels to guide the training of a student model.

In this study, we combine semi-supervised learning (SSL) with few-shot

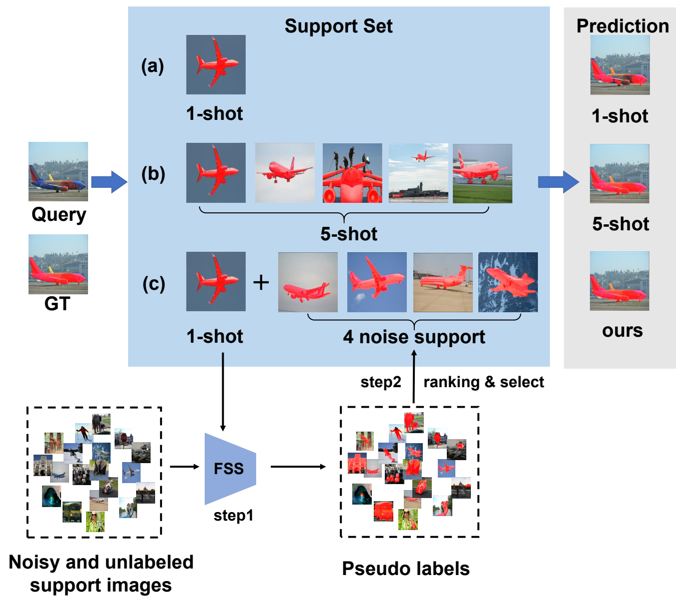

Figure 1: (a) 1-shot setting. (b) 5-shot setting. (c) 1-shot with additional 4 noise support images with pseudo labels. There is a large performance gap between 1-shot and 5-shot. Using 1-shot and 4 noise support can achieve comparable performance to 5-shot without increasing annotation cost.

segmentation (FSS) and propose a novel semi-supervised few-shot segmentation framework named **F4S**. Different from existing method [12] that introduces super-pixels to generate pseudo labels and only enhances the training phase of FSS, the proposed F4S framework generates pseudo labels of unlabeled images directly, and quantitatively evaluates the quality of pseudo labels based on a novel ranking algorithm, and finally enhance both the training and test phases of any off-the-shelf FSS models. A brief pipeline of F4S is shown in Fig. 1 (c) , which consists of three steps. Firstly, pseudo labels are generated using a pre-trained FSS model for noisy and unlabeled support images. Secondly, pseudo labels with high confidence scores are selected as ground truth to augment the support set. Thirdly, the augmented support set is utilized to enhance the FSS model in both the training and test phases.

However, unlabeled support images are not a free lunch, as there are two problems that complicate pseudo-label selection (as shown in Fig. 2). 1) **Noisy Intra-Class Samples:** The noisy intra-class samples contain ambiguous objects that may strengthen the background and weaken the foreground, e.g., noisy "background" dominates the image as shown in Fig. 2 (a).

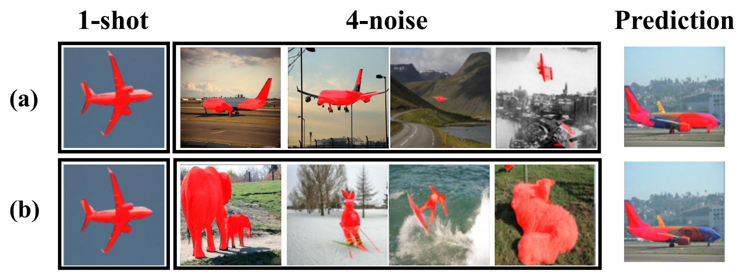

Figure 2: Examples of two basic problems. (a) Noisy intra-class samples as support samples. (b) Noisy inter-class samples as support samples.

2) **Noisy Inter-Class Samples:** The noisy inter-class samples introduce irrelevant features to the task, which may cause feature bias and thus confuse the FSS model, e.g., the FSS model is confused by “elephant”, “person” and “sheep” when segmenting “aeroplane” as shown in Fig. 2 (b). We need to eliminate the two types of samples.

To solve the two basic problems, we propose a ranking algorithm in F4S to automatically eliminate the noisy intra-class samples and inter-class samples. This ranking algorithm consists of two terms: an intra-class confidence term R and an inter-class confidence term T. The term R aims to identify the noisy intra-class samples by calculating three terms: E_{sc}, E_{imc} and E_{cyc}. Specifically, E_{sc} measures prediction uncertainty based on binary entropy, E_{imc} identifies different types of errors based on the co-teaching framework [14, 15], and E_{cyc} measures object features completeness based on the cycle-consistency strategy [16, 17]. Besides, the term T aims to identify the noisy inter-class samples. It calculates the feature similarities between the support prototypes and the pseudo labels of noisy images. Finally, a ranking score E is calculated by weighting R and T, and the top-scored pseudo labels are treated as new support samples.

In order to theoretically explain the effectiveness of the ranking algorithm, we design a Structural Causal Model (SCM), which models the relevance of input support samples, noisy support set, and query labels. The SCM proves that the proposed ranking algorithm can successfully remove the confounding bias in the noisy support set (*cf.* Sect. 5). We also evaluate the proposed F4S framework on two popular FSS benchmarks: PASCAL-5^i [1], and COCO-20^i [18] in Sect. 6. Extensive quantitative and qualitative studies show that the F4S achieves new SOTA performance compared with existing fully supervised FSS methods.

This paper represents a very substantial extension of our previous conference paper [19]. The main improvements compared with [19] lie in threefold: (i) We have improved the F4S framework by integrating a new term, E_{cyc}, derived from the cycle-consistency strategy, into the proposed ranking algorithm. This enhancement notably boosts the model's ability to identify noisy samples without increasing its learnable parameters or memory cost, achieving improved performances. (ii) We have added a justification section (Sect. 5), where we theoretically explain why the proposed method can work successfully based on a Structural Causal Model (SCM), which models the causal relevance of input data, generated pseudo labels, and output predictions. (iii) We have conducted more comprehensive experiments to evaluate the proposed method thoroughly. These experiments include extensive evaluations on the PASCAL-5^i dataset, along with additional comparisons with both inductive and transductive FSS methods, as well as recent semi-supervised FSS methods. Furthermore, we have included visualization results, conducted more comprehensive ablation studies, and performed additional experimental analysis.

Our main contributions are as follows:

- We incorporate semi-supervised learning into the few-shot segmentation task and propose the **F4S** framework. It can benefit any off-the-shelf few-shot segmentation models by solving the data scarcity problem via introducing pseudo-labeled images, which has less been studied.

- We design a ranking algorithm including an intra-class confidence score R and an inter-class confidence score T to automatically identify and eliminate the noisy samples in pseudo labels. The designing of R and T are based on the underlying mechanism of FSS models. To the best of our knowledge, this is the first work that quantitatively evaluates the quality of pseudo labels in semi-supervised few-shot segmentation.

- We offer a theoretical explanation of the ranking algorithm grounded in a Structural Causal Model (SCM). This analysis proves that the proposed method has the capability to mitigate confounding bias within the noisy support set through causal intervention.

2. Related Work

2.1. Few-shot Segmentation

Few-shot segmentation performs semantic segmentation in the few-shot scenario, where only a few support images are given for a new class. Two types of FSS methods, i.e., the prototype-based approaches [20, 2, 3, 4, 21, 5, 6] and the metric-based approaches [7, 8, 9], are mainly used to achieve accurate segmentation.

The prototype-based approaches try to generate prototypes that describe the class well from the limited training samples. For example, the method in [20] generates foreground and background prototypes via a classifier trained by support images with image-level labels. The method in [2] uses a prototype alignment strategy to make the prototypes more consistent. Seeing the fact that one single prototype is hard to fully describe the class, some methods [3, 4] try to generate multiple prototypes for each class. For example, the methods in [3] and [4] decompose the single class representation into a set of part-aware prototypes that can describe diverse fine-grained object features more precisely. The methods in [21] and [5] propose a parameter-free based prototype generation method via feature clustering.

The metric-based approaches try to learn a class-agnostic similarity metric that measures the similarity of region pairs, by which the query region similar to the support region can be obtained. For example, the method in [7] proposes a dense comparison module to calculate the similarity between support features and query features under multiple levels. The method in [8] proposes a multi-scale decoder with attention prior masks to achieve better measurement. Besides, the methods in [22] and [23] provide a fresh insight into the FSS task. The proposed BAM network incorporates an auxiliary base learner into the conventional FSS meta learner to identify and remove the feature-biased problem caused by base-class objects, and thus learn a better class-agnostic metric function. Moreover, the method in [24] introduces a divide-and-conquer strategy in FSS, which divides coarse results into small regions and conquers the segmentation failures by leveraging the information derived from support image-mask pairs.

Different from these existing methods, we generalize few-shot segmentation with more noisy and unlabeled images in both the training and testing phases. Furthermore, we propose a new quality ranking algorithm that can select good support samples from noisy samples accurately.

2.2. Semi-Supervised Learning

Semi-supervised learning [25, 26, 27, 28, 29, 30, 31] trains neural networks on partially labeled datasets, including both labeled and unlabeled data. The labeled data provides discriminative information about classes, while the unlabeled data provides the underlying structure of the input data. Recent works based on semi-supervised learning not only improve the performance of deep neural networks, but also significantly reduce the cost associated with data labeling. For example, the method in [25] generates and selects pseudo labels for unlabeled data that exhibit high confidence above a specific threshold to enhance image classification. The method in [29] utilizes the teacher-student framework, where the teacher model learns to generate good pseudo labels from unlabeled data to benefit the student model for object detection. The method in [28] proposes a new confidence score based on the loss distribution to select good pseudo labels and benefit few-shot classification. The method in [27] generates and retains pseudo-labeled samples with high confidence of the target domain for adversarial learning to solve the domain adaptation problem. The method in [30] proposes a transfer network, which is trained by pseudo labels and learns to exploit beneficial feature representation knowledge in the extractor to enhance the training of semantic segmentation network. In this paper, we propose a semi-supervised FSS framework to expand the support image set with unlabeled images and their pseudo labels.

2.3. Few-shot Learning with Noisy Samples

Few-shot learning with noisy samples [32, 33, 34, 35, 36] represents a more realistic scenario, where support sets are susceptible to mislabeled samples. Robustness to noisy samples is crucial for practical few-shot learning methods. Some existing works [32, 33] focus on feature similarity to identify and eliminate the noisy samples. For instance, the method proposed in [32] employs soft k-means clustering to detect noise within the support samples, given that the features of noisy samples deviate significantly from the current support set. The method described in [33] utilizes a feature-level similarity assessment to reveal the heterogeneity and homogeneity within support samples.

Additionally, designing attention mechanisms is widely utilized for suppressing noise. For example, the method in [34] introduces a semantically-conditioned attention mechanism to estimate the importance of training instances and bolster the model's resilience to noise. Similarly, the method outlined in [35] introduces an attention mechanism based on a novel transformer architecture, to effectively weigh mislabeled samples against correct ones. Moreover, the method described in [36] presents an attention-based contrastive learning model incorporating discrete cosine transform input. This model utilizes transformed frequency domain representations obtained through discrete cosine transform as input, effectively removing high-frequency components to suppress input noise.

Furthermore, recent research effort [37] extends the handling of noisy samples to the few-shot segmentation task. It proposes a noise suppression module to eliminate noisy activations by analyzing the correlation distribution between query and support features. However, [37] only considers the inter-class noisy samples and cannot be generalized to a semi-supervised scenario, where both intra-class and inter-class noisy samples abound. Therefore, semi-supervised few-shot segmentation with noisy samples is a more crucial scenario and remains largely unexplored. In this study, we introduce a novel quality ranking algorithm designed to select high-quality support samples from noisy pseudo-labeled data. This approach enhances few-shot segmentation models in a semi-supervised way during both the training and testing phases.

2.4. Causal Inference

Causal inference [38, 39] aims to formulate tasks in the view of causalities and makes the network benefit from causal effects by removing the *confounder*. Recently, a growing number of methods combing with causal inference are proposed [40, 41, 42, 43, 44] in computer vision. For example, the method in [40] uses causal inference to solve the semi-supervised semantic segmentation, where the co-occurrence context is considered as a *confounder* making the model hard to distinguish the category boundaries. A context adjustment method with causal intervention is proposed to remove the confounding bias. The method in [41] treats the pre-trained knowledge as a *confounder* in few-shot learning, and uses causal intervention to remove the negative effect of the pre-trained knowledge. The method in [42] tackles the out-of-distribution (OOD) generalization problem with causality. A causal invariant transformation is proposed to keep the causal features from non-causal features. Similarly, the method in [43] designs a meta-causal learner to capture common causal features from multiple tasks and realize out-of-distribution generalization. In this paper, we propose a structural causal model in Sect. 5.1 to analyze the causalities among support samples, noisy support set, and query labels in our F4S framework, and aim at improving the FSS performance.

3. Formulation

We mathematically formulate the conventional few-shot segmentation methods and the proposed F4S for better understanding.

Conventional few-shot segmentation methods: ① In the training phase, a support set S^{base} including images I_S^{base} and pixel-level annotations M_S^{base} of base classes is given. A few-shot segmentation network N_θ parameterized by θ need to be trained on $\{I_S^{base}, M_S^{base}\}$ to segment objects from a query set Q^{base} within the meta-learning paradigm. The ground truth M_Q^{base} of Q^{base} is given for loss calculation and backward propagation. ② In the test phase, $\{I_S^{novel}, M_S^{novel}\}$ of novel classes is given, which provides support features to help network N_θ predict segmentation masks M_Q^{novel} of novel objects from Q^{novel}. Then, an evaluation metric, e.g. mIoU, is adopted to evaluate the performance of N_θ, i.e. $mIoU(\hat{M}_Q^{novel}, M_Q^{novel})$.

The proposed method F4S: ① Before training, $\{I_S^{base}, M_S^{base}\}$ and a set of noisy unlabeled images $I_{unlabel}$ are given. Pseudo labels P of $I_{unlabel}$ are generated by the pretrained network N_θ based on the support features of $\{I_S^{base}, M_S^{base}\}$. ② A ranking algorithm is proposed here to obtain $\{I_{unlabel}^{base}, P^{base}\}$, where the noisy pseudo-labeled samples are eliminated and superior pseudo-labeled samples of base classes are retained. ③ In the training phase, based on $\{I_S^{base}, M_S^{base}, I_{unlabel}^{base}, P^{base}\}$, the network N_θ is retrained within the meta-learning paradigm. ④ Before test, we implement ① and ② again based on $\{I_S^{novel}, M_S^{novel}\}$ to obtain $\{I_{unlabel}^{novel}, P^{novel}\}$ of novel classes. ⑤ In the test phase, based on $\{I_S^{novel}, M_S^{novel}, I_{unlabel}^{novel}, P^{novel}\}$, the network N_θ outputs the predictions $\hat{M}_Q^{novel}$ of the query set Q^{novel}. Then, an evaluation metric $mIoU(\hat{M}_Q^{novel}, M_Q^{novel})$ is utilized to evaluate the performance.

4. Method

4.1. Overview

Fig. 3 (a) shows the proposed F4S framework, which consists of three phases. In phase I, a pretrained FSS network N_θ is used to obtain the pseudo labels of the noisy and unlabeled support images. Various existing FSS models can be employed here.

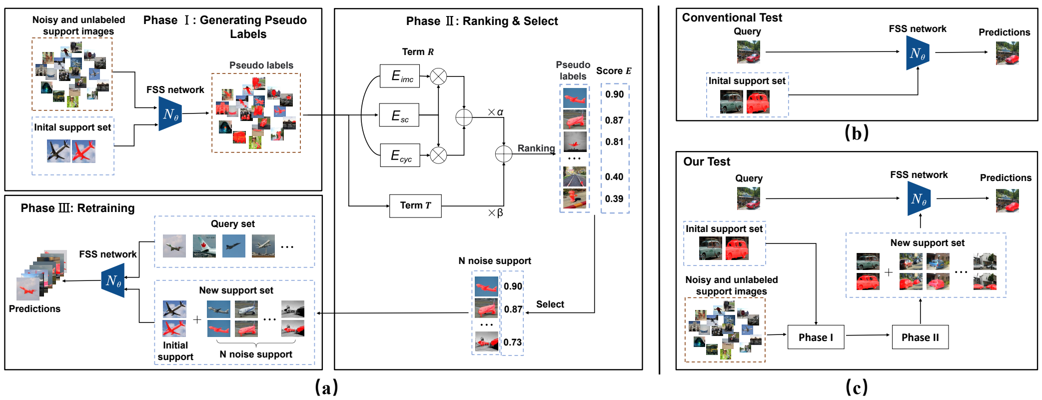

Figure 3: (a) The pipeline of the proposed F4S framework, which consists of three phases. In phase I, a pretrained FSS network N_θ is used to obtain the pseudo labels. Then, in phase II, a ranking algorithm is utilized to calculate quality scores E of pseudo labels and rank them. Finally, in phase III, top-scored pseudo labels are selected as new support samples to retrain N_θ. (b) The pipeline of the conventional FSS test. After retraining N_θ, it is tested on novel classes, e.g., "car", with an annotated initial support set. (c) The pipeline of our FSS test based on the proposed semi-supervised framework. N_θ is tested on novel classes with a new support set, which is expanded following phase I and phase II.

In phase II, the ranking algorithm is utilized to evaluate the pseudo labels. Specifically, an intra-class confidence term R and an inter-class confidence term T are calculated for each pseudo label. Then, a final ranking score E is obtained by simply calculating the weighted sum of R and T:

$$E = \alpha \cdot R + \beta \cdot T \tag{1}$$

where α and β are weighting coefficients. Afterwards, the top k scored pseudo labels are selected to form a new annotation set:

$$\mathcal{S}_{new}^{base} \leftarrow \mathcal{S}^{base} + \{(X_1, \hat{Y}_{X_1}), (X_2, \hat{Y}_{X_2}), ..., (X_k, \hat{Y}_{X_k})\} \tag{2}$$

where $\mathcal{S}^{base}$ indicates the initial annotation set of base classes in the training phase, $\hat{Y}_X$ indicates the pseudo label of image X.

Finally, in phase III, the new annotation set $\mathcal{S}_{new}^{base}$ is used to retrain N_θ and get better predictions. More details of the intra-class confidence term R and the inter-class confidence term T are introduced in Sect. 4.2 and Sect. 4.3, respectively. Besides, in order to enhance the inference of FSS models, we further propose a new test process based on F4S in Sect. 4.4.

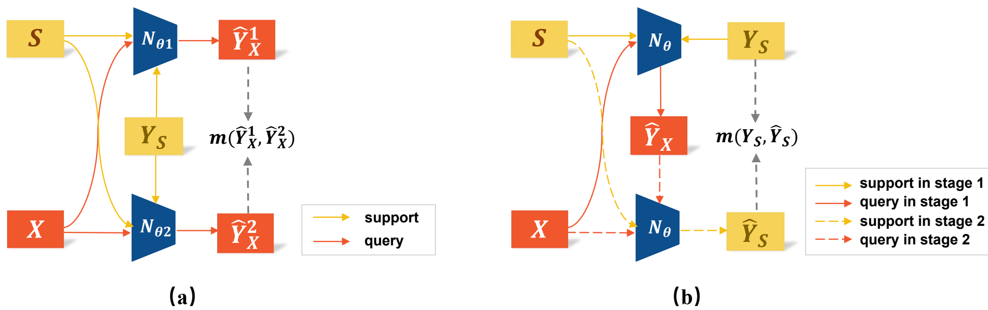

Figure 4: (a) The pipeline of E_{imc}. The unlabeled image X is processed by two FSS models $N_{\theta 1}$, $N_{\theta 2}$, with a given support sample $\{S, Y_S\}$. Then, a metric $m(\cdot,\cdot)$ is calculated between the two output $\hat{Y}_X^1$, $\hat{Y}_X^2$. (b) The pipeline of E_{cyc}, which consists of two stages. In stage 1, a FSS model N_θ makes prediction $\hat{Y}_X$ of the unlabeled image X based on a given support sample $\{S, Y_S\}$. In stage 2, N_θ makes prediction $\hat{Y}_S$ of S based on $\{X, \hat{Y}_X\}$. Finally, a metric $m(\cdot,\cdot)$ is calculated between Y_S and $\hat{Y}_S$.

4.2. Intra-Class Confidence Term R

The term R aims to identify the noisy intra-class samples. The calculation of R is shown in Eq. 3:

$$R = E_{sc} \times (E_{imc} + E_{cyc}) \tag{3}$$

where the segmentation confidence term E_{sc} estimates the prediction uncertainty of pseudo labels, the instance mask consensus term E_{imc} identifies different types of errors in pseudo labels, and the cyclic mask consensus term E_{cyc} identifies pseudo labels with incomplete object features. Now, we introduce the three terms E_{sc}, E_{imc}, and E_{cyc} in detail.

Segmentation Confidence Term E_{sc}. This term is calculated by adopting a binary-entropy-based function to measure the prediction uncertainty:

$$E_{sc} = -\frac{1}{N}\sum_i H(i) + B \tag{4}$$

where i indicates a pixel position, $H(\cdot)$ is the binary entropy function, N is the total number of pixels, and B is a bias term to ensure $E_{sc} \in [0, 1]$. The formulation of $H(x)$ is shown in Eq. 5, where $p(i)$ is the logit at pixel position i.

$$H(x) = -p(i)log(p(i)) - (1 - p(i))log(1 - p(i)) \tag{5}$$

Instance Mask Consensus Term E_{imc}. This term is motivated by the co-teaching theory [14, 15], which proves that two diverged networks can filter different types of errors. Therefore, if two diverged few-shot segmentation networks output similar predictions to the same wild image, the predictions contain less error and have high confidence. The pipeline of getting E_{imc} is shown in Fig. 4 (a) and its calculation is:

$$E_{imc} = m(\hat{Y}_X^1, \hat{Y}_X^2) \tag{6}$$

where $\hat{Y}_X^1$ and $\hat{Y}_X^2$ are predictions of the same unlabeled image X from two diverged networks $N_{\theta 1}$ and $N_{\theta 2}$. $m(\cdot, \cdot)$ indicates a segmentation metric score, e.g., mIoU.

Cyclic Mask Consensus Term E_{cyc}. Inspired by the cycle-consistency strategy of [16], we design a cyclic pipeline in FSS to estimate the segmentation confidence. The detailed pipeline is shown in Fig. 4 (b). Specifically, it consists of two stages: in stage 1, a FSS model N_θ makes a prediction $\hat{Y}_X$ of the unlabeled image X based on the annotated support sample $\{S, Y_S\}$; in stage 2, based on $\{X, \hat{Y}_X\}$, N_θ makes a prediction $\hat{Y}_S$ of the support image S. Finally, the E_{cyc} can be calculated by:

$$E_{cyc} = m(Y_S, \hat{Y}_S) \tag{7}$$

4.3. Inter-Class Confidence Term T

The term T aims to identify the noisy inter-class samples based on the feature similarities between the support prototypes and the pseudo labels. First, the prototype of class c of the initial support set $\mathcal{S}^c = \{S_1^c, S_2^c, ..., S_n^c\}$ are calculated by:

$$\mathcal{P}^c = \frac{1}{n}\sum_{i=1}^{n} \sigma(\mathcal{F}_{S_i^c}, Y_{S_i^c}) \tag{8}$$

where $\mathcal{F}_{S_i^c} \in \mathbb{R}^{C\times H\times W}$ is the feature map of support S_i^c of class c, $Y_{S_i^c}$ is the manual annotation, $\sigma(\cdot)$ is the masked global average pooling, and $\mathcal{P}^c \in \mathbb{R}^C$ is the prototype of class c. Then, the term T can be calculated by:

$$T = s(\mathcal{P}^c, \sigma(\mathcal{F}_X, \hat{Y}_X)) \tag{9}$$

where $\mathcal{F}_X \in \mathbb{R}^{C\times H\times X}$ is the feature map of X, $\hat{Y}_X$ is the generated pseudo label, $s(\cdot, \cdot)$ is a similarity metric, e.g., cosine similarity.

4.4. A New Test Process based on F4S

To enhance the inference of FSS models, we can further expand the initial support set of novel classes via F4S in the test phase, of which the pipeline is shown in Fig. 3 (c). Specifically, different from the conventional FSS test (Fig. 3 (b)), where only a small annotated support set $\mathcal{S}^{novel}$ of novel classes is utilized, our test enriches $\mathcal{S}^{novel}$ following the pipeline of phase I and phase II of the proposed F4S to obtain a new support set $\mathcal{S}^{novel}_{new}$:

$$\mathcal{S}^{novel}_{new} \leftarrow \mathcal{S}^{novel} + \{(X_1, \hat{Y}_{X_1}), (X_2, \hat{Y}_{X_2}), ..., (X_k, \hat{Y}_{X_k})\} \tag{10}$$

Then, the query images will be segmented with the new support set $\mathcal{S}^{novel}_{new}$ to get better predictions.

5. Justification

5.1. Structural Causal Model

We construct a causal graph to formulate the causalities among the selected support sample, query prediction, and the noisy support set, which is shown in Fig. 5 (a). The causal graph consists of four nodes: X indicates the selected support sample; Y is the query label; D indicates the noisy support set, which includes the noisy intra-class and inter-class samples and acts as the *confounder* in the causal graph; M is the transformed representation of X in the low-dimensional manifold embedded in the latent high-dimension space via FSS model [40]. The directed path between two nodes indicates the causalities : cause $\rightarrow$ effect. Next, we detail the rationale of Fig. 5 (a).

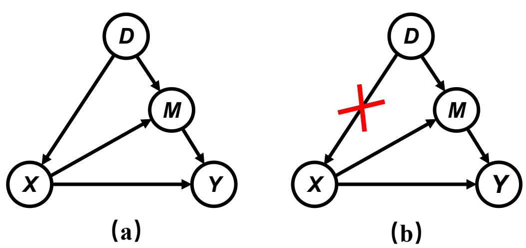

Figure 5: (a) The causal graph for FSS. The *confounder* D degrades FSS via $X \leftarrow D \rightarrow M \rightarrow Y$, i.e., noisy intra-class and inter-class samples in D are mistakenly selected as support samples X causing serious feature bias and bad query predictions of Y. (b) The revised causal graph of our F4S, where the proposed ranking algorithm in F4S can cut off the path towards X by *do(X)*, and thus ensures the selected support samples are noiseless.

$D \rightarrow X$. The support sample X is sampled from the noisy support set D.

$X \to Y$. The support sample X provides object cues to predict query label Y. However, this latent relevance between X and Y cannot obtained directly, and therefore a FSS model $f(\cdot)$ is needed here to learn a transformed representation M between X and Y.

$D \to M$. The transformed representation M is a subset of that of D due to that the FSS model $f(\cdot)$ is trained on D.

$X \to M \to Y$. The support sample X leads to the transformed representation M via FSS model, i.e., $M = f(X)$, and M contributes to the prediction of Y, i.e., $P(Y|X, M)$. X with less noise leads to better M, and finally benefits the prediction of Y.

Based on the causal graph, one can see that the *confounder* D degrades $P(Y|X)$ via the backdoor path $X \leftarrow D \to M \to Y$. Removing the backdoor path is the key challenge for improving F4S performance. Next, we show how to remove the confounding effect by causal intervention $P(Y|do(X))$.

5.2. Causal Intervention via Backdoor Adjustment

In this section, we propose to use the causal intervention $P(Y|do(X))$, which can remove the confounding effect by $do(\cdot)$ to get a better prediction of label Y. The key idea is to cut off the path $D \to X$ (Fig. 5(b)) via backdoor adjustment [38], i.e., identifying and eliminating noisy intra-class and inter-class samples when sampling X from D. Following [45, 38], we have:

$$\begin{aligned}
P(Y|do(X)) = \sum_{D=\{d_0,d_1\}} & P(Y|X, M = f(X, D))P(D) \\
=& P(Y|X, f(X, D = d_0))P(D = d_0) \\
&+ P(Y|X, f(X, D = d_1))P(D = d_1) \\
=& P(Y|X, f(X, D = d_0)) \cdot \alpha \\
&+ P(Y|X, f(X, D = d_1)) \cdot \beta
\end{aligned} \tag{11}$$

where the noisy support set D includes two types of noisy samples: d_0 indicates the noisy intra-class samples, and d_1 indicates the noisy inter-class samples. $P(D = d_0)$ and $P(D = d_1)$ indicate the ratio of d_0 and d_1 in D. For simplicity, they are set as two constants: α and β, respectively. Next, we estimate $P(Y|X, f(X, D = d_0))$ and $P(Y|X, f(X, D = d_1))$.

5.2.1. ***Estimation of*** $P(Y|X, f(X, D = d_0))$

Following [46], we implement the sampling process from the intervened distribution to get $P(Y = y|X = x, f(X = x, D = d_0))$, abbreviated as

$P(y|x, f(x, d_0))$. It represents the probability of predicting the label $Y = y$ under the condition of input $X = x$ with intra-class noise $D = d_0$. Intuitively, less intra-class noise d_0 leads to a higher probability P to predict the correct label $Y = y$, which can be reflected by a segmentation metric score. To this end, we can get:

$$P(y|x, f(x, d_0)) \propto m(y, \hat{y}) \tag{12}$$

where $\hat{y}$ is the prediction of label y, $m(\cdot, \cdot)$ indicates a segmentation metric score, e.g., mIoU.

However, the label $Y = y$ is unavailable since the noisy support set is not annotated, and thus $m(y, \hat{y})$ can not be calculated. Fortunately, the proposed intra-class confidence score R (Eq. 3) can estimate the credibility of prediction $\hat{y}$ in a blind way, i.e., without annotated label y. Therefore, we can further obtain:

$$P(y|x, f(x, d_0)) \propto m(y, \hat{y}) \propto R \tag{13}$$

In this way, the proposed intra-class confidence term R can estimate the target $P(Y|X, f(X, D = d_0))$ due to its correlation of metric score $m(\cdot, \cdot)$.

5.2.2. *Estimation of* $P(Y|X, f(X, D = d_1))$

Implementing the sampling process from the intervened distribution, we can get the term $P(y|x, f(x, d_1))$, which represents the probability of predicting the label $Y = y$ based on input $X = x$ with inter-class noise $D = d_1$. Intuitively, less inter-class noise d_1 leads to higher probability P to predict label $Y = y$, which can be reflected by the similarity between class prototype $\mathcal{P}$ and input noisy support sample x. Therefore, we have:

$$P(y|x, f(x, d_1)) \propto s(\mathcal{P}, f(x_s)) \tag{14}$$

where $\mathcal{P}$ is the class-specific prototype, $f(x_s)$ is the feature map of the input support sample x, $s(\cdot, \cdot)$ is a similarity metric, e.g., cosine similarity. Combining Eq. 14 with Eq. 9, we get:

$$P(y|x, f(x, d_1)) \propto T \tag{15}$$

In this way, the proposed inter-class confidence term T can estimate the target $P(Y|X, f(X, D = d_1))$ based on the feature similarities.

Finally, combining Eq. 13 with Eq. 15, we can rewrite Eq. 11:

$$P(Y|do(X)) \propto R \cdot \alpha + T \cdot \beta = E \tag{16}$$

Therefore, the proposed ranking mechanism can successfully remove the confounding effect in the noisy support set D following the causal intervention $P(Y|do(X))$.

6. Experiment

6.1. Setup

Datasets. We evaluate our method on PASCAL-5^i [1] and COCO-20^i [18] datasets and use the unlabeled 123,403 images in COCO2017 [47] for conducting experiments. Specifically, following the setup in [1], 20 categories in the PASCAL VOC 2012 dataset [48] are partitioned into 4 folds (i.e., fold-0, fold-1, fold-2, and fold-3) and each fold contains 5 categories. Following the setups in [18], 80 categories in the COCO dataset [47] are also divided into 4 folds and each fold contains 20 categories. The experiments are conducted in a cross-validation manner and the validation episode is set to 1000 for each fold.

Evaluation metrics. Following previous works [3, 4, 21, 49], we adopt mean intersection over union (mIoU) and foreground-background IoU (FB-IoU) as our evaluation metrics. The mIoU metric is computed by averaging IoU of all classes: $mIoU = \frac{1}{n}\sum_{i=1}^{n} IoU_i$. The FB-IoU metric is calculated by averaging IoU of foreground and background: $mIoU = \frac{1}{2}(IoU_F + IoU_B)$.

Implementation details. All of our experiments are conducted on two NVIDIA Titan XP GPUs and Intel Core i9-9900k CPU @ 3.60GHz$\times$ 16. Our code is constructed on PyTorch. We build our F4S framework based on the open-sourced code of methods in [8, 50]. In Sect. 4.2, multiple backbones are adopted as the two diverged networks $N_{\theta 1}$, $N_{\theta 2}$. The detailed settings of $N_{\theta 1}$, $N_{\theta 2}$ are shown in Table 1. The publicly released pretrained models in methods [8, 50] are used directly. For the PFENet (VGG16) on PASCAL-5^i and PFENet (ResNet101) on COCO-20^i, we train the models following the official settings in [8]. We set $m(\cdot,\cdot)$ to mIoU score in Sect. 4.2 and set $s(\cdot,\cdot)$ to cosine similarity in Sect. 4.3. The feature maps $\mathcal{F} \in \mathbb{R}^{C\times H\times W}$ in Sect. 4.3 are extracted from the last convolutional layer of the backbone. α and β in Eq. 1 are set to 0.3 and 0.7, respectively. In the training phase, pseudo labels with $E \geq 0.65$ are selected as new annotations of base classes. In the test phase, top 4 scored pseudo labels are introduced into the support set of novel classes. In phase III, the retraining setting strictly follows the base model [8, 50].

Table 1: The diverged networks in E_{imc}.

Method	$N_{\theta 1}$	$N_{\theta 2}$
HSNet [50]	ResNet50	ResNet101
	VGG16	ResNet101
PFENet [8]	VGG16	ResNet50
	VGG16	ResNet101

6.2. Quantitative Results

We evaluate the proposed F4S on PASCAL-5^i [1] and COCO-20^i datasets and compare the metric scores with recent FSS methods [2, 8, 50, 51, 53, 54]. Table 2 shows the mIoU and FB-IoU values of our method and the existing methods under 1-shot settings on PASCAL-5^i and COCO-20^i datasets, where "F4S (HSNet)" indicates that F4S is implemented on the HSNet [50]. Here, the F4S is set to 1-shot/5-shot with 4 noise support (as shown in Fig. 1 (c)) and our evaluation has two test ways: the conventional test in Fig. 3 (b) and our test in Fig. 3 (c), which are annotated as "†" and "‡" in Table 2, respectively.

Compared with the baseline (HSNet), we can observe that on the PASCAL-5^i dataset, "F4S (HSNet) †" achieves mIoU improvements of 1.6%, 0.8%, and 0.3% on three backbones under 1-shot, and achieves mIoU improvements of 0.7%, 0.6%, and 0.5% under 5-shot. Meanwhile, on the COCO-20^i dataset, "F4S (HSNet) †" also achieves further improvements of mIoU and FB-IoU on different backbones under 1-shot and 5-shot. These results demonstrate that the proposed F4S can benefit FSS models from the unlabeled support images in the retraining phase (Fig. 3 (a)) without noise disturbance. Besides, following our test (Fig. 3 (c)), "F4S (HSNet) ‡" achieves mIoU improvements of 8.2%, 6.8%, and 6.1% on three backbones on PASCAL-5^i, and mIoU improvements of 10.8%, and 10.2% on two backbones on COCO-20^i under 1-shot. Moreover, there are also remarkable performance improvements achieved by "F4S (HSNet) ‡" under 5-shot. These quantitative results verify that extending the support set with unlabeled support images via F4S can directly benefit the inference of FSS models in the test phase.

We also compare the proposed method with recent transductive and inductive methods. In Table 2, one can observe that the proposed method "F4S (HSNet) ‡" with different backbones obtains new state-of-the-art performances. On PASCAL-5^i and with ResNet101 backbone, our 1-shot and 5-shot results of "F4S (HSNet)‡" respectively achieve 3.7% and 0.9% of mIoU

Table 2: Performance of the proposed F4S on PASCAL-5^i and COCO-20^i datasets. "†" is the results of the conventional test. "‡" is the results of our test based on the F4S. "Oracle" is the 5-shot performance. "±0.1" is the standard deviation of repeating 5 times.

Dataset	Backbone	Method	Type	1-shot		5-shot	
				mIoU	FB-IoU	mIoU	FB-IoU
PASCAL-5^i	VGG16	PFENet [8]	inductive	58.0	72.0	59.0	72.3
		HSNet [50]	inductive	59.7	73.4	64.1	76.6
		HPA [51]	inductive	61.5	75.2	66.2	79.3
		DCP [24]	inductive	62.6	75.6	67.8	80.7
		BAM [22]	inductive	64.4	77.3	68.8	81.1
		BAM* [23]	inductive	65.3	77.5	**69.6**	**81.3**
		F4S (HSNet)†	inductive	61.3 (± 0.3)	74.4 (± 0.2)	64.8 (± 0.2)	76.9 (± 0.2)
		F4S (HSNet)‡	inductive	**67.9** (± 0.2)	**79.2** (± 0.1)	68.2 (± 0.3)	79.7 (± 0.3)
	ResNet50	RePRI [49]	transductive	59.1	-	66.8	-
		PFENet [8]	inductive	60.8	73.3	61.9	73.9
		HSNet [50]	inductive	64.0	76.7	69.5	80.6
		HPA [51]	inductive	64.8	76.4	68.9	81.1
		CDFS [52]	transductive	65.3	-	70.8	-
		DCP [24]	inductive	66.1	77.6	70.3	81.5
		BAM [22]	inductive	67.8	79.7	70.9	82.2
		BAM* [23]	inductive	68.3	80.3	71.8	**83.1**
		F4S (HSNet)†	inductive	64.8 (± 0.2)	77.2 (± 0.2)	70.1 (± 0.2)	81.0 (± 0.2)
		F4S (HSNet)‡	inductive	**70.8** (± 0.2)	**81.5** (± 0.1)	**72.0** (± 0.3)	82.3 (± 0.2)
	ResNet101	PFENet [8]	inductive	60.1	72.9	61.4	73.5
		DCAMA [53]	inductive	64.6	77.6	68.3	80.8
		HPA [51]	inductive	65.6	76.6	68.9	80.4
		HSNet [50]	inductive	66.2	77.6	70.4	80.6
		DCP [24]	inductive	67.3	78.5	71.5	82.7
		BAM [22]	inductive	68.6	80.2	72.5	**84.1**
		F4S (HSNet)†	inductive	66.5 (± 0.2)	78.2 (± 0.2)	70.9 (± 0.3)	81.1 (± 0.2)
		F4S (HSNet)‡	inductive	**72.3** (± 0.1)	**82.3** (± 0.1)	**73.4** (± 0.2)	82.6 (± 0.3)
COCO-20^i	ResNet50	RePRI [49]	transductive	34.0	-	42.1	-
		HSNet [50]	inductive	39.2	68.2	46.9	70.7
		CDFS [52]	transductive	42.0	-	49.8	-
		DCAMA [53]	inductive	43.3	69.5	48.3	71.7
		HPA [51]	inductive	43.4	68.2	50.0	71.2
		DCP [24]	inductive	45.5	-	50.9	-
		BAM [22]	inductive	46.2	-	51.2	-
		BAM* [23]	inductive	46.9	72.3	51.9	**74.7**
		F4S (HSNet)†	inductive	40.9 (± 0.3)	69.1 (± 0.2)	49.0 (± 0.4)	71.9 (± 0.5)
		F4S (HSNet)‡	inductive	**50.0** (± 0.4)	**72.6** (± 0.5)	**52.0** (± 0.3)	74.0 (± 0.3)
	ResNet101	PFENet [8]	inductive	38.5	63.0	42.7	65.8
		HSNet [50]	inductive	41.2	69.1	49.5	72.4
		DCAMA [53]	inductive	43.5	69.9	51.9	73.3
		HPA [51]	inductive	45.8	68.4	52.4	74.0
		BAM* [23]	inductive	48.5	69.9	52.7	74.1
		F4S (HSNet)†	inductive	42.8 (± 0.2)	69.8 (± 0.2)	51.2 (± 0.5)	73.3 (± 0.4)
		F4S (HSNet)‡	inductive	**51.4** (± 0.2)	**73.3** (± 0.3)	**54.1** (± 0.4)	**75.5** (± 0.4)

* indicates the improved version of the base method.

Table 3: Performance of the proposed F4S without the retraining phase on PASCAL-5^i and COCO-20^i datasets. "Oracle" is the 5-shot performance. "±0.1" is the standard deviation of repeating 5 times.

Dataset	Backbone	Method	Type	1-shot		5-shot	
				mIoU	FB-IoU	mIoU	FB-IoU
PASCAL-5^i	VGG16	PFENet [8]	inductive	58.0	72.0	59.0	72.3
		HSNet [50]	inductive	59.7	73.4	64.1	76.6
		HPA [51]	inductive	61.5	75.2	66.2	79.3
		DCP [24]	inductive	62.6	75.6	67.8	80.7
		BAM [22]	inductive	64.4	77.3	68.8	81.1
		BAM* [23]	inductive	65.3	77.5	**69.6**	**81.3**
		F4S (PFENet)‡	inductive	59.8 (± 0.2)	72.1 (± 0.2)	60.3 (± 0.3)	72.5 (± 0.3)
		F4S (HSNet)‡	inductive	**66.5** (± 0.2)	**78.4** (± 0.1)	67.1 (± 0.2)	78.9 (± 0.3)
	ResNet50	RePRI [49]	transductive	59.1	-	66.8	-
		PFENet [8]	inductive	60.8	73.3	61.9	73.9
		HSNet [50]	inductive	64.0	76.7	69.5	80.6
		HPA [51]	inductive	64.8	76.4	68.9	81.1
		CDFS [52]	transductive	65.3	-	70.8	-
		DCP [24]	inductive	66.1	77.6	70.3	81.5
		BAM [22]	inductive	67.8	79.7	70.9	82.2
		BAM* [23]	inductive	68.3	80.3	**71.8**	**83.1**
		F4S (PFENet)‡	inductive	62.4 (± 0.2)	73.3 (± 0.2)	62.9 (± 0.3)	73.5 (± 0.2)
		F4S (HSNet)‡	inductive	**70.6** (± 0.2)	**81.4** (± 0.1)	71.7 (± 0.3)	82.0 (± 0.3)
	ResNet101	PFENet [8]	inductive	60.1	72.9	61.4	73.5
		DCAMA [53]	inductive	64.6	77.6	68.3	80.8
		HPA [51]	inductive	65.6	76.6	68.9	80.4
		HSNet [50]	inductive	66.2	77.6	70.4	80.6
		DCP [24]	inductive	67.3	78.5	71.5	82.7
		BAM [22]	inductive	68.6	80.2	72.5	**84.1**
		F4S (HSNet)‡	inductive	**72.1** (± 0.1)	**82.1** (± 0.1)	**72.6** (± 0.3)	82.2 (± 0.3)
COCO-20^i	ResNet50	RePRI [49]	transductive	34.0	-	42.1	-
		HSNet [50]	inductive	39.2	68.2	46.9	70.7
		CDFS [52]	transductive	42.0	-	49.8	-
		DCAMA [53]	inductive	43.3	69.5	48.3	71.7
		HPA [51]	inductive	43.4	68.2	50.0	71.2
		DCP [24]	inductive	45.5	-	50.9	-
		BAM [22]	inductive	46.2	-	51.2	-
		BAM* [23]	inductive	46.9	**72.3**	**51.9**	**74.7**
		F4S (HSNet)‡	inductive	**49.7** (± 0.4)	72.2 (± 0.2)	51.0 (± 0.5)	72.9 (± 0.4)
	ResNet101	PFENet [8]	inductive	38.5	63.0	42.7	65.8
		HSNet [50]	inductive	41.2	69.1	49.5	72.4
		DCAMA [53]	inductive	43.5	69.9	51.9	73.3
		HPA [51]	inductive	45.8	68.4	52.4	74.0
		BAM* [23]	inductive	48.5	69.9	**52.7**	74.1
		F4S (PFENet)‡	inductive	41.5 (± 0.2)	63.8 (± 0.2)	43.3 (± 0.3)	66.4 (± 0.4)
		F4S (HSNet)‡	inductive	**51.1** (± 0.4)	**73.1** (± 0.5)	52.4 (± 0.4)	**74.5** (± 0.4)

* indicates the improved version of the base method.

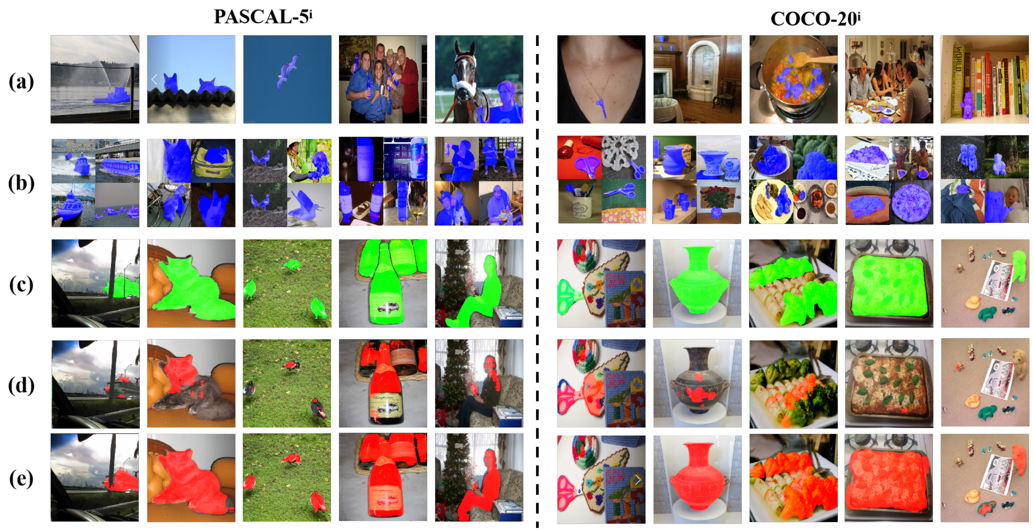

Figure 6: Qualitative results of the proposed F4S and its baseline. The left panel is from PASCAL-5^i, and the right panel is from COCO-20^i. From top to bottom: (a) 1-shot support images with ground truth, (b) 4 noise support images with pseudo labels via F4S, (c) query images with ground truth, (d) baseline predictions, (e) F4S predictions.

improvements over BAM [22]. On COCO-20^i and with ResNet101 backbone, "F4S (HSNet)‡" also outperforms recent methods with a sizable margin as well, achieving 2.9% and 1.4% of mIoU improvements over BAM* [23]. These results verify the superiority of the proposed method in the few-shot segmentation task.

Furthermore, we also evaluate F4S in the test phase directly without the retraining phase to save the training cost. Two popular FSS models, i.e., HSNet [50] and PFENet [8], are adopted to implement F4S. The quantitative results are shown in Table 3. One can observe that on PASCAL-5^i dataset and under the 1-shot setting, "F4S (PFENet)" achieves mIoU improvements of 1.8%, and 1.6% on VGG16 and ResNet50 backbones compared with PFENet performance (baseline), and "F4S (HSNet)" achieves mIoU improvements of 6.8%, 6.6%, and 5.9% on three different backbones compared with HSNet performance (baseline). On COCO-20^i dataset, "F4S (HSNet)" and "F4S (PFENet)" also obtain superior performance compared with the baseline. These quantitative results prove that the proposed F4S can benefit the inference of FSS models directly without extra training.

It is worth noting that in both Table 2 and Table 3, the performance of

Table 4: Performance comparison with recent semi-supervised few-shot segmentation methods on PASCAL-5^i and COCO-20^i datasets.

Dataset	Backbone	Method	1-shot		5-shot	
			mIoU	FB-IoU	mIoU	FB-IoU
PASCAL-5^i	ResNet50	CLRS [12]	56.4	-	67.7	-
		UaFSS [55]	67.0	79.2	68.9	80.2
		F4S (HSNet)†	64.8 (± 0.2)	77.2 (± 0.2)	70.1 (± 0.2)	81.0 (± 0.2)
		F4S (HSNet)‡	70.8 (± 0.2)	81.5 (± 0.1)	72.0 (± 0.3)	82.3 (± 0.2)
	ResNet101	CLRS [12]	64.3	-	68.2	-
		UaFSS [55]	68.5	79.4	69.5	79.4
		F4S (HSNet)†	66.5 (± 0.2)	78.2 (± 0.2)	70.9 (± 0.3)	81.1 (± 0.2)
		F4S (HSNet)‡	72.3 (± 0.1)	82.3 (± 0.1)	73.4 (± 0.2)	82.6 (± 0.3)
COCO-20^i	ResNet50	CLRS [12]	33.0	-	36.3	-
		UaFSS [55]	41.3	68.9	46.4	70.9
		F4S (HSNet)†	40.9 (± 0.3)	69.1 (± 0.2)	49.0 (± 0.4)	71.9 (± 0.5)
		F4S (HSNet)‡	50.0 (± 0.4)	72.6 (± 0.5)	52.0 (± 0.3)	74.0 (± 0.3)
	ResNet101	UaFSS [55]	43.6	69.9	46.8	70.7
		F4S (HSNet)†	42.8 (± 0.2)	69.8 (± 0.2)	51.2 (± 0.5)	73.3 (± 0.4)
		F4S (HSNet)‡	51.4 (± 0.2)	73.3 (± 0.3)	54.1 (± 0.4)	75.5 (± 0.4)

F4S (1-shot with 4 noise support) surprisingly surpasses the 5-shot performance of HSNet in some cases. This can be attributed to two aspects. First, the training of models is enhanced due to the additional support features from noisy and unlabeled support images introduced by F4S. Second, the annotated support samples in "Oracle" are randomly sampled from datasets and may include noisy intra-class samples, while the proposed F4S guarantees the exclusion of such noisy intra-class samples.

Finally, we also compare the proposed method with recent semi-supervised methods [12, 55] to show the superior performance in Table 4. One can see that on PASCAL-5^i dataset and with ResNet50 backbone, the proposed "F4S (HSNet)‡" achieves 3.8% of mIoU improvement in 1-shot setting and 3.1% of mIoU improvement in 5-shot setting over UaFSS[55]. Besides, with ResNet101 backbone, the proposed method also outperforms recent methods with a sizable margin as well, achieving 3.8% (1-shot) and 3.9% (5-shot) of mIoU improvements over UaFSS[55]. Besides, on COCO-20^i dataset and with ResNet50 and ResNet101 backbones, the 1-shot and 5-shot results of "F4S (HSNet)‡" are also superior to both UaFSS[55] and CLRS[12] with a remarkable margin.

6.3. Qualitative Results

Fig. 6 shows the qualitative results of "F4S (HSNet)" with ResNet101 backbone on PASCAL-5^i and COCO-20^i datasets. As can be noticed, (e) F4S predictions include more complete and accurate object regions compared with the (d) baseline, and are close to the (c) ground truth, which demonstrates that the proposed F4S achieves a comparable performance to 5-shot without increasing annotation cost.

6.4. Ablation study

We conduct a series of ablation studies to investigate the effectiveness of each component in the proposed F4S and the results are shown in Table 5. Without loss of generality, the ablation study experiments are performed on "F4S(HSNet)" with ResNet101 backbone on COCO-20^i dataset. In Table 5, one can observe that when only with the E_{sc}, E_{imc}, or E_{cyc}, the proposed method achieves mIoU improvement of 0.4%, 0.7% ,and 0.6% respectively, and their combination leads to 2.3% mIoU improvement. Then, when only using the inter-class confidence term T, the proposed method achieves mIoU improvements of 8.9%, and FB-IoU improvements of 2.6%. Next, with the existence of T, each component (E_{sc}, E_{imc}, and E_{cyc}) of the intra-class confidence term R contributes further mIoU improvements to different extents, which are shown in the 7^{th} to 9^{th} rows. Finally, the full combination of R and T achieves the best mIoU of 51.4% and FB-IoU of 73.3%. The ablation studies prove the effectiveness of both R and T in the F4S.

We notice that T contributes to larger mIoU improvement while R provides limited improvement. The reason is that the feature bias caused by inter-class noise is greater than intra-class noise, which explains the greater performance improvement of T. However, this does not mean that intra-class noise can be ignored. The results in the 2^{nd} to 5^{th} rows of Table 5 show that R is also essential for eliminating intra-class noise to improve FSS performance.

6.5. Analysis

6.5.1. Computational analysis

In Table 6, the 1^{st} row shows the computational complexity of the base model HSNet, which is regarded as the baseline. The 2^{nd} row shows the computational complexity of the proposed method in whole stages, including generating (Stage I) and selecting (Stage II) pseudo labels. The 3^{rd} to 5^{th} rows show the computational complexity of each stage respectively.

Table 5: Ablation study of F4S with different design choices. The results represent the mean metric scores of running 5 times. "±0.1" indicates the standard deviation of running 5 times.

R			T	Fold-0	Fold-1	Fold-2	Fold-3	mean	FB-IoU
E_{sc}	E_{imc}	E_{cyc}							
				37.2	44.1	42.4	41.3	41.2	69.1
✓				37.9	45.7	41.8	41.1	41.6 (±0.4)	69.3 (±0.3)
	✓			38.5	44.6	42.3	42.0	41.9 (±0.3)	69.8 (±0.4)
		✓		38.7	45.1	41.8	41.7	41.8 (±0.5)	69.6 (±0.6)
✓	✓	✓		39.7	47.0	44.4	42.8	43.5 (±0.7)	70.6 (±0.6)
			✓	47.1	53.4	50.3	49.7	50.1 (±0.4)	71.7 (±0.5)
✓			✓	46.7	56.2	50.8	48.7	50.6 (±0.8)	72.0 (±0.4)
	✓		✓	47.6	55.8	49.6	49.0	50.5 (±0.6)	71.9 (±0.3)
		✓	✓	47.6	55.6	51.3	49.6	51.0 (±0.4)	72.4 (±0.4)
✓	✓	✓	✓	46.6	56.7	51.5	50.7	51.4 (±0.2)	73.3 (±0.3)

Table 6: Computational complexity of F4S compared with the baseline.

Method	Stage			Learnable Params ↓	FPS ↑	FLOPS(G) ↓
	I	II	III			
HSNet (*baseline*)	-	-	-	2.6M	16.33	20.56
F4S (HSNet)	✓	✓	✓	2.6M	5.08	81.66
	✓			0	15.80	20.52
		✓		0	8.51	40.62
			✓	2.6M	16.45	20.52

Specifically, in stage I (3^{rd} row), the trained models of HSNet are officially provided to generate pseudo labels. Therefore, there are no learnable params in this stage, and the FPS and FLOPs are also close to the baseline. In stage II (4^{th} row), a diverged network $N_{\theta 2}$ is adopted here to compute E_{imc} in Eq. 6 and the base network N_θ is utilized to compute E_{cyc} in Eq. 7. Therefore, the FLOPS increases to 40.62G and the FPS decreases to 8.51. In stage III (5^{th} row), F4S (HSNet) is retrained with pseudo labels. Therefore, the learnable params is 2.6M, which is the same as the baseline. Besides, the FPS and FLOPs of F4S (HSNet) are 16.45 and 20.52G, respectively, which are also close to the baseline (16.33 and 20.56G).

Here we emphasize that although the proposed method has a high computational complexity in whole stages (2^{nd} row), the stage I and stage II only need to be performed once before the training and testing stages, and do not affect the computational complexity of the training and testing stages (5^{th} row). Therefore, in the actual testing process, the computational complexity

Table 7: Performance scores of different weight values. The results represent the mean metric scores of running 5 times. "±0.1" indicates the standard deviation of running 5 times.

α	β	Fold-0	Fold-1	Fold-2	Fold-3	mean	FB-IoU
0.5	0.5	72.3	74.7	68.3	70.4	71.4 (± 0.1)	81.5 (± 0.1)
0.4	0.6	72.5	75.0	69.6	70.1	71.8 (± 0.2)	81.9 (± 0.1)
0.2	0.8	72.2	74.5	69.5	71.9	72.0 (± 0.1)	82.0 (± 0.1)
0.3	0.7	72.3	75.4	71.1	70.6	72.3 (± 0.1)	82.3 (± 0.1)

Table 8: Precomputed α and β on PASCAL-5^i dataset.

	fold-1					fold-2				
	aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow
α	0.14	0.19	0.20	0.22	0.11	0.27	0.25	0.16	0.10	0.26
β	0.86	0.81	0.80	0.78	0.89	0.73	0.75	0.84	0.90	0.74
	fold-3					fold-4				
	diningtable	dog	horse	motorbike	person	pottedplant	sheep	sofa	train	tvmonitor
α	0.21	0.27	0.26	0.18	0.34	0.12	0.29	0.17	0.30	0.24
β	0.79	0.73	0.74	0.82	0.66	0.88	0.71	0.83	0.70	0.76

of the inference remains unchanged compared to the baseline.

6.5.2. Weights settings

Table. 7 shows the quantitative scores when α and β in Eq. 1 are set to different values. The experiments are conducted on "F4S(HSNet)" with ResNet101 backbone on PASCAL-5^i. One can observe that when $\alpha = 0.3$ and $\beta = 0.7$, the best quantitative scores (72.3% mIoU and 82.3% FB-IoU) are obtained. Besides, we also find that by using different α and β, the quantitative scores fluctuate within a narrow range (<1.0%), which demonstrates the stability of the proposed F4S to α and β.

Moreover, we conduct experiments of precomputed α and β to obtain the "oracle" performance. The α and β indicate the ratio of intra- and inter-class samples in the noisy unlabeled images. Therefore, we count the quantity of intra- and inter-class samples of each class. We conduct experiments on PASCAL-5^i dataset and the precomputed α and β of each class are shown in Table 8 and the "Oracle" results are shown in Table 9.

In Table 9, one can observe that with the precomputed α and β, the "Oracle" results of the proposed method achieve 73.4% (1-shot) and 73.9% (5-shot) of mIoU with ResNet101 backbone, which outperform "F4S (HSNet) ‡" with a sizable margin (1.1% and 1.1%). Besides, with VGG16 and ResNet50 backbones, the "Oracle" results also achieve remarkable mIoU improvements.

Table 9: "Oracle" performance by precomputed α and β on PASCAL-5^i dataset.

Backbone	Method	1-shot		5-shot	
		mIoU	FB-IoU	mIoU	FB-IoU
VGG16	F4S (HSNet) †	61.3 (± 0.3)	74.4 (± 0.3)	64.8 (± 0.2)	76.9 (± 0.2)
	F4S (HSNet) ‡	67.9 (± 0.2)	79.2 (± 0.1)	68.2 (± 0.3)	79.7 (± 0.3)
	Oracle	68.2	79.4	68.6	80.2
ResNet50	F4S (HSNet) †	64.8 (± 0.2)	77.2 (± 0.2)	70.1 (± 0.2)	81.0 (± 0.2)
	F4S (HSNet) ‡	70.8 (± 0.2)	81.5 (± 0.2)	72.0 (± 0.3)	82.2 (± 0.2)
	Oracle	71.9	82.4	72.5	83.0
ResNet101	F4S (HSNet) †	66.5 (± 0.2)	78.2 (± 0.2)	70.9 (± 0.3)	81.1 (± 0.2)
	F4S (HSNet) ‡	72.3 (± 0.2)	82.3 (± 0.2)	72.8 (± 0.2)	82.6 (± 0.3)
	Oracle	73.4	83.0	73.9	83.3

These results verify the effectiveness of precomputed α and β.

6.5.3. Statistical analysis of term R

To further investigate the terms E_{sc}, E_{imc}, E_{cyc} in the intra-class confidence term R, we sample the image X from the annotated PASCAL-5^i to calculate $m(Y_X, \hat{Y}_X)$, where the ground truth Y_X is available and $m(\cdot,\cdot)$ is set to mIoU score. Then, we calculate E_{sc}, E_{imc}, E_{cyc} following Sect. 4.2. In Fig. 7, we plot the scatter graphs of (a) E_{sc} and $m(Y_X, \hat{Y}_X)$, (b) E_{imc} and $m(Y_X, \hat{Y}_X)$, (c) E_{cyc} and $m(Y_X, \hat{Y}_X)$, (d) R and $m(Y_X, \hat{Y}_X)$ on the 4 folds of PASCAL-5^i. As can be noticed in Fig. 7 (a)-(c), there is a positive correlation between $m(Y_X, \hat{Y}_X)$ and E_{sc}, E_{imc}, E_{cyc}. In Fig. 7 (d), the score R combining the three components contributes to better scatter dots distribution: the dots mainly follow the line $y = x$, which presents a better positive correlation between R and $m(Y_X, \hat{Y}_X)$. Therefore, the results of the scatter graphs prove that the intra-class confidence term R can estimate the credibility of pseudo labels, i.e., $m(Y_X, \hat{Y}_X)$, and thus identify the noisy intra-class samples.

6.5.4. F4S performance change with different numbers of unlabelled examples

We have investigated the F4S performance change with different numbers of unlabelled examples. We choose "F4S (HSNet)" with ResNet101 backbone as the model to conduct the experiments. Here, Table 10 and Table 11 show the results on PASCAL-5^i and COCO-20^i datasets, respectively.

In Table 10 and Table 11, the "baseline" indicates the F4S performance under the 1-shot setting without any additional unlabelled examples in the test phase. The "+ N examples" indicates the F4S performance with additional unlabelled N examples, which are pseudo-labelled and selected by F4S.

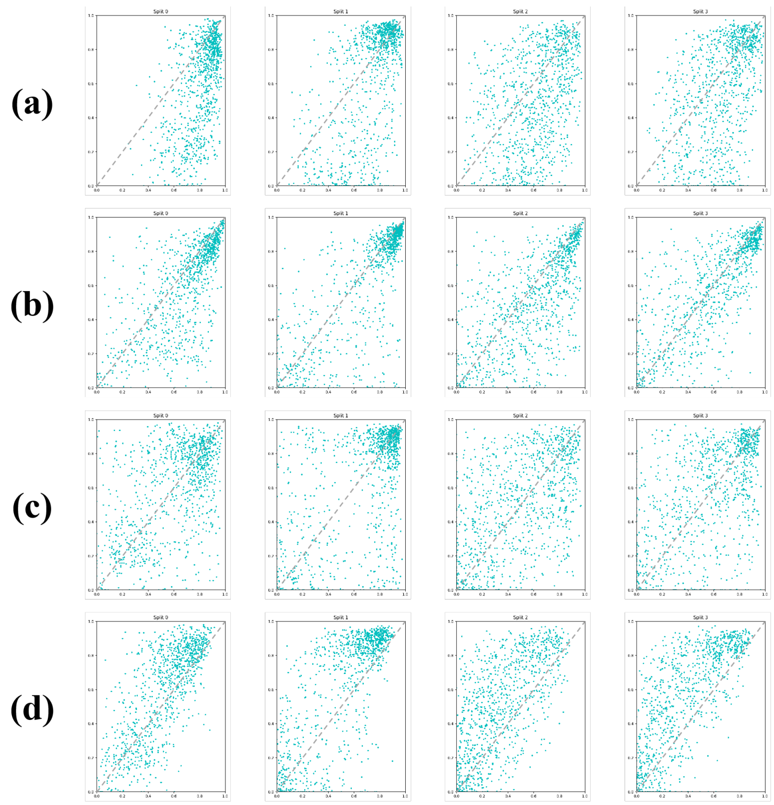

Figure 7: Scatter graphs of each term in score R. The y-axis indicates the mIoU score based on ground truth. The x-axis indicates the values of: (a) E_{sc}, (b) E_{imc}, (c) E_{cyc}, and (d) R. Each row shows the scatter graphs on the 4 folds of PASCAL-5^i.

In Table 10, the "baseline" performance is 66.5% mIoU score and 78.2% FB-IoU score over 4 folds on the PASCAL-5^i dataset. Then, with the increasing number of unlabelled examples, the performance scores of F4S also gradually improve. Finally, when with "+ 29 examples", the proposed F4S achieves 7.3% of mIoU improvements and 5.5% of FB-IoU improvements over the "baseline". In Table 11, when with "+ 29 examples" on the COCO-20^i dataset, the proposed F4S also outperforms "baseline" with a sizable margin as well, achieving 9.9% of mIoU improvements and 4.0% of FB-IoU improvements. Furthermore, we observed that with "+ 29 examples", the performance eventually plateaus in both PASCAL-5^i and COCO-20^i datasets. This outcome is attributed to the increased number of pseudo-labeled examples with lower scores E.

Table 10: F4S performance change with different numbers of unlabelled examples on PASCAL-5^i.

setting		Fold-0	Fold-1	Fold-2	Fold-3	mean	FB-IoU
baseline	1-shot	67.8	72.2	62.4	63.4	66.5 ($\pm$ 0.2)	78.2 ($\pm$ 0.2)
F4S	+ 4 examples	72.3	75.4	71.1	70.6	72.3 ($\pm$ 0.1)	82.3 ($\pm$ 0.1)
	+ 9 examples	73.0	76.0	72.2	71.6	73.2 ($\pm$ 0.1)	83.4 ($\pm$ 0.1)
	+ 19 examples	73.4	76.4	72.6	72.2	73.6 ($\pm$ 0.1)	83.5 ($\pm$ 0.2)
	+ 29 examples	73.5	76.5	72.8	72.6	73.8 ($\pm$ 0.1)	83.7 ($\pm$ 0.1)

Table 11: F4S performance change with different numbers of unlabelled examples on COCO-20^i.

setting		Fold-0	Fold-1	Fold-2	Fold-3	mean	FB-IoU
baseline	1-shot	38.4	47.8	43.2	41.8	42.8 ($\pm$ 0.2)	69.8 ($\pm$ 0.2)
F4S	+ 4 examples	46.6	56.7	51.5	50.7	51.4 ($\pm$ 0.2)	73.3 ($\pm$ 0.3)
	+ 9 examples	47.5	56.6	52.1	50.6	51.7 ($\pm$ 0.6)	73.6 ($\pm$ 0.5)
	+ 19 examples	47.2	57.9	52.7	50.5	52.1 ($\pm$ 0.6)	73.7 ($\pm$ 0.6)
	+ 29 examples	48.2	58.9	52.8	50.8	52.7 ($\pm$ 0.8)	73.8 ($\pm$ 0.7)

6.6. Discussion

In this section, we introduce the task settings of few-shot learning and semi-supervised learning, and summarize the similarities and differences between them.

Setting of Few-shot Learning. Few-shot learning (FSL) has a few available samples per class as the support set and aims to recognize the objects in the query set. In fact, FSL does not classify the data specifically, but makes a cluster to learn the similarity metric function [10]. Increasing the number of support images is a direct way to improve the performance of FSL models. However, it requires manual annotation and selection of high-quality intra-class data as new support images, which is a time- and labor-consuming process.

Setting of Semi-Supervised Learning. Semi-supervised learning (SSL) concerns with using labeled as well as unlabeled data to perform certain learning tasks. It permits harnessing the large amounts of unlabeled data available in many use cases in combination with typically smaller sets of labeled data [56]. Existing SSL methods based on deep neural networks can be categorized into: deep generative methods, consistency regularization methods, graph-based methods, pseudo-labeling methods, and hybrid methods [57]. Our proposed method falls within the category of pseudo-labeling methods.

Similarities. Both few-shot learning and semi-supervised learning face the challenge of data scarcity. In the FSL, there are typically very few samples available for training each category, while in the SSL, there is a small portion of labeled training data and the rest is unlabeled. Besides, both FSL and SSL place great demand on the model's generalization capability. The FSL and SSL models need to make accurate predictions on new data under data scarcity.

Differences. Few-shot learning and Semi-supervised learning differ in their primary objectives and approaches. FSL emphasizes how to effectively recognize novel classes with very few labeled samples. Therefore, existing FSL methods focus on the designing of network architectures, loss functions, and optimizers to improve FSL performance. However, SSL concerns with the utilization of unlabeled data to enhance supervised learning tasks. Taking pseudo-labeling methods as an illustration, this type of method concentrates on the generation of pseudo labels and the reduction of noise in order to enhance the diversity of classes within the dataset, consequently facilitating the supervised training of models.

7. Conclusion

We have presented a novel semi-supervised few-shot segmentation framework named F4S, where noisy and unlabeled support images, e.g., from other available datasets, are utilized to benefit both the training and test of few-shot segmentation networks via generating pseudo labels. Due to the feature-biased problem caused by noisy intra- and inter-class samples and resulting in FSS performance degradation, we propose a ranking algorithm in F4S to identify and eliminate the noisy samples via calculating and ranking confidence scores of noisy support images. Specifically, the ranking algorithm consists of an intra-class confidence score R to identify noisy intra-class samples based on their prediction confidence, and an inter-class confidence score T to identify noisy inter-class samples based on channel-wise feature similarity. Additionally, we have theoretically explained the effectiveness of the proposed method based on a Structural Causal Model (SCM) from the view of causal inference. We have conducted extensive experiments on PASCAL-5^i and COCO-20^i datasets to validate the proposed method. Compared with recent inductive and transductive FSS methods, the proposed method achieves superior performance under 1-shot and 5-shot settings. Besides, the ablation studies prove the effectiveness of each component in the score R and score T.

The proposed work still has some primary limitations: (1) the computational complexity in the stage II of the proposed method is costly. How to optimize the selection of pseudo labels to reduce the computational complexity is a crucial concern in the future. (2) The underlying characteristics of noisy samples need further investigation for designing the confidence score E and making the selection of pseudo labels more reliable. We hope our work may inspire the study of exploring the combination of semi-supervised learning with few-shot segmentation task.

8. Acknowledgement

This work was supported in part by the National Key R&D Program of China under Grant 2021ZD0112001, the National Natural Science Foundation of China under Grant 62271119, the Natural Science Foundation of Sichuan Province under Grant 2023NSFSC1972, the A*STAR AME Programmatic Funding A18A2b0046, the RobotHTPO Seed Fund under Project C211518008, the EDB Space Technology Development Grant under Project S22-19016-STDP, and the Natural Science Foundation of Jiangsu Province under Grant BZ2021013.

References

[1] A. Shaban, S. Bansal, Z. Liu, I. Essa, B. Boots, One-shot learning for semantic segmentation, British Machine Vision Conference (2017).

[2] K. Wang, J. H. Liew, Y. Zou, D. Zhou, J. Feng, Panet: Few-shot image semantic segmentation with prototype alignment, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9197–9206.

[3] Y. Liu, X. Zhang, S. Zhang, X. He, Part-aware prototype network for few-shot semantic segmentation, in: Computer Vision – ECCV 2020, Springer International Publishing, Cham, 2020, pp. 142–158.

[4] B. Yang, C. Liu, B. Li, J. Jiao, Q. Ye, Prototype mixture models for few-shot semantic segmentation, in: European Conference on Computer Vision, Springer, 2020, pp. 763–778.

[5] L. Yang, W. Zhuo, L. Qi, Y. Shi, Y. Gao, Mining latent classes for few-shot segmentation, in: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 8701–8710. doi:10.1109/ICCV48922.2021.00860.

[6] H. Sun, X. Lu, H. Wang, Y. Yin, X. Zhen, C. G. Snoek, L. Shao, Attentional prototype inference for few-shot segmentation, Pattern Recognition 142 (2023) 109726. doi:https://doi.org/10.1016/j.patcog.2023.109726.

[7] C. Zhang, G. Lin, F. Liu, R. Yao, C. Shen, Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5217–5226.

[8] Z. Tian, H. Zhao, M. Shu, Z. Yang, R. Li, J. Jia, Prior guided feature enrichment network for few-shot segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (2) (2022) 1050–1065. doi:10.1109/TPAMI.2020.3013717.

[9] H. Min, Y. Zhang, Y. Zhao, W. Jia, Y. Lei, C. Fan, Hybrid feature enhancement network for few-shot semantic segmentation, Pattern Recognition 137 (2023) 109291. doi:https://doi.org/10.1016/j.patcog.2022.109291.

[10] Y. Song, T. Wang, P. Cai, S. K. Mondal, J. P. Sahoo, A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities, ACM Computing Surveys (2023).

[11] K. Huang, J. Geng, W. Jiang, X. Deng, Z. Xu, Pseudo-loss confidence metric for semi-supervised few-shot learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 8671–8680.

[12] Y. Chen, C. Wei, D. Wang, C. Ji, B. Li, Semi-supervised contrastive learning for few-shot segmentation of remote sensing images, Remote Sensing 14 (17) (2022) 4254.

[13] Y. Tang, Z. Cao, Y. Yang, J. Liu, J. Yu, Semi-supervised few-shot object detection via adaptive pseudo labeling, IEEE Transactions on Circuits and Systems for Video Technology (2023).

[14] J. Li, R. Socher, S. C. Hoi, Dividemix: Learning with noisy labels as semi-supervised learning, in: International Conference on Learning Representations, 2019.

[15] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, M. Sugiyama, Co-teaching: Robust training of deep neural networks with extremely noisy labels, Neural Information Processing Systems(NeurIPS) (2018).

[16] J.-Y. Zhu, T. Park, P. Isola, A. A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2223–2232.

[17] T. Zhou, P. Krahenbuhl, M. Aubry, Q. Huang, A. A. Efros, Learning dense correspondence via 3d-guided cycle consistency, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 117–126.

[18] K. Nguyen, S. Todorovic, Feature weighting and boosting for few-shot segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 622–631.

[19] R. Zhang, H. Zhu, H. Zhang, C. Gong, J. T. Zhou, F. Meng, Semi-supervised few-shot segmentation with noisy support images, in: 2023 IEEE International Conference on Image Processing (ICIP), IEEE, 2023, pp. 1550–1554.

[20] N. Dong, E. Xing, Few-shot semantic segmentation with prototype learning, in: British Machine Vision Conference, 2018.

[21] G. Li, V. Jampani, L. Sevilla-Lara, D. Sun, J. Kim, J. Kim, Adaptive prototype learning and allocation for few-shot segmentation, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 8330–8339. doi:10.1109/CVPR46437.2021.00823.

[22] C. Lang, G. Cheng, B. Tu, J. Han, Learning what not to segment: A new perspective on few-shot segmentation, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 8057–8067.

[23] C. Lang, G. Cheng, B. Tu, C. Li, J. Han, Base and meta: A new perspective on few-shot segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence (2023).

[24] C. Lang, G. Cheng, B. Tu, J. Han, Few-shot segmentation via divide-and-conquer proxies, International Journal of Computer Vision (2023) 1–23.

[25] Z. Hu, Z. Yang, X. Hu, R. Nevatia, Simple: Similar pseudo label exploitation for semi-supervised classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 15099–15108.

[26] M. Yang, J. Ling, J. Chen, M. Feng, J. Yang, Discriminative semi-supervised learning via deep and dictionary representation for image classification, Pattern Recognition 140 (2023) 109521. doi:https://doi.org/10.1016/j.patcog.2023.109521.

[27] J. Li, G. Li, Y. Shi, Y. Yu, Cross-domain adaptive clustering for semi-supervised domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 2505–2514.

[28] K. Huang, J. Geng, W. Jiang, X. Deng, Z. Xu, Pseudo-loss confidence metric for semi-supervised few-shot learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 8671–8680.

[29] M. Xu, Z. Zhang, H. Hu, J. Wang, L. Wang, F. Wei, X. Bai, Z. Liu, End-to-end semi-supervised object detection with soft teacher, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 3060–3069.

[30] Y. Jin, J. Wang, D. Lin, Semi-supervised semantic segmentation via gentle teaching assistant, Advances in Neural Information Processing Systems 35 (2022) 2803–2816.

[31] Y. Wang, J. Zhang, M. Kan, S. Shan, Learning pseudo labels for semi-and-weakly supervised semantic segmentation, Pattern Recognition 132 (2022) 108925. doi:https://doi.org/10.1016/j.patcog.2022.108925.

[32] P. Mazumder, P. Singh, V. P. Namboodiri, Rnnp: A robust few-shot learning approach, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 2664–2673.

[33] J. Lu, S. Jin, J. Liang, C. Zhang, Robust few-shot learning for user-provided data, IEEE transactions on neural networks and learning systems 32 (4) (2020) 1433–1447.

[34] O. B. Baran, R. G. Cinbis, Semantics-driven attentive few-shot learning over clean and noisy samples, Neurocomputing 513 (2022) 59–69.

[35] K. J. Liang, S. B. Rangrej, V. Petrovic, T. Hassner, Few-shot learning with noisy labels, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9089–9098.

[36] Z. Chen, T. Ji, S. Zhang, F. Zhong, Noise suppression for improved few-shot learning, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 1900–1904.

[37] X. Luo, Z. Tian, T. Zhang, B. Yu, Y. Y. Tang, J. Jia, Pfenet++: Boosting few-shot semantic segmentation with the noise-filtered context-aware prior mask, IEEE Transactions on Pattern Analysis and Machine Intelligence (2023).

[38] J. Pearl, M. Glymour, N. P. Jewell, Causal inference in statistics: A primer. 2016, Google Ascholar there is no corresponding record for this reference (2016).

[39] D. B. Rubin, Essential concepts of causal inference: a remarkable history and an intriguing future, Biostatistics & Epidemiology 3 (1) (2019) 140–155.

[40] Z. Yue, H. Zhang, Q. Sun, X.-S. Hua, Interventional few-shot learning, Advances in neural information processing systems 33 (2020) 2734–2746.

[41] D. Zhang, H. Zhang, J. Tang, X.-S. Hua, Q. Sun, Causal intervention for weakly-supervised semantic segmentation, Advances in Neural Information Processing Systems 33 (2020) 655–666.

[42] R. Wang, M. Yi, Z. Chen, S. Zhu, Out-of-distribution generalization with causal invariant transformations, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 375–385.

[43] Y. Wang, X. Li, Z. Qi, J. Li, X. Li, X. Meng, L. Meng, Meta-causal feature learning for out-of-distribution generalization, in: European Conference on Computer Vision, Springer, 2022, pp. 530–545.

[44] T. Zhang, H.-R. Shan, M. A. Little, Causal graphsage: A robust graph method for classification based on causal sampling, Pattern Recognition 128 (2022) 108696. doi:https://doi.org/10.1016/j.patcog.2022.108696.

[45] L. G. Neuberg, Causality: Models, reasoning, and inference, by judea pearl, cambridge university press, 2000, Econometric Theory 19 (4) (2003) 675–685. doi:10.1017/S0266466603004109.

[46] B. Zhu, Y. Niu, X.-S. Hua, H. Zhang, Cross-domain empirical risk minimization for unbiased long-tailed classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 3589–3597.

[47] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, 2014, pp. 740–755.

[48] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, International Journal of Computer Vision 88 (2) (2010) 303–338.

[49] M. Boudiaf, H. Kervadec, Z. I. Masud, P. Piantanida, I. B. Ayed, J. Dolz, Few-shot segmentation without meta-learning: A good transductive inference is all you need?, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 13974–13983. doi:10.1109/CVPR46437.2021.01376.

[50] J. Min, D. Kang, M. Cho, Hypercorrelation squeeze for few-shot segmenation, in: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 6921–6932. doi:10.1109/ICCV48922.2021.00686.

[51] G. Cheng, C. Lang, J. Han, Holistic prototype activation for few-shot segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 45 (4) (2023) 4650–4666. doi:10.1109/TPAMI.2022.3193587.

[52] Y. Lu, X. Wu, Z. Wu, S. Wang, Cross-domain few-shot segmentation with transductive fine-tuning, arXiv preprint arXiv:2211.14745 (2022).

[53] X. Shi, D. Wei, Y. Zhang, D. Lu, M. Ning, J. Chen, K. Ma, Y. Zheng, Dense cross-query-and-support attention weighted mask aggregation for few-shot segmentation, in: Computer Vision – ECCV 2022, Springer Nature Switzerland, Cham, 2022, pp. 151–168.

[54] D. Kang, M. Cho, Integrative few-shot learning for classification and segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9979–9990.

[55] S. Kim, P. Chikontwe, S. An, S. H. Park, Uncertainty-aware semi-supervised few shot segmentation, Pattern Recognition 137 (2023) 109292.

[56] J. E. Van Engelen, H. H. Hoos, A survey on semi-supervised learning, Machine learning 109 (2) (2020) 373–440.

[57] X. Yang, Z. Song, I. King, Z. Xu, A survey on deep semi-supervised learning, IEEE Transactions on Knowledge and Data Engineering (2022).