Variational prototype inference for few-shot semantic segmentation

Wang, H.; Yang, Y.; Cao, X.; Zhen, X.; Snoek, C.; Shao, L.

DOI
10.1109/WACV48630.2021.00057

Publication date
2021

Document Version
Final published version

Published in
2021 IEEE Winter Conference on Applications of Computer Vision

License
Article 25fa Dutch Copyright Act (https://www.openaccess.nl/en/in-the-netherlands/you-share-we-take-care)

Link to publication

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)
Variational Prototype Inference for Few-Shot Semantic Segmentation

Haochen Wang¹, Yandan Yang¹, Xianbin Cao¹,²,³†, Xiantong Zhen⁴,⁵, Cees Snoek⁴, Ling Shao⁵
¹School of Electronic and Information Engineering, Beihang University, Beijing, China
²Key Laboratory of Advanced Technology of Near Space Information System, Ministry of Industry and Information Technology of China
³Beijing Advanced Innovation Center for Big Data-Based Precision Medicine, China
⁴University of Amsterdam, Amsterdam, Netherlands
⁵Inception Institute of Artificial Intelligence, Abu Dhabi, UAE

{haochenwang, yangyandan, xcbao}@buaa.edu.cn, x.zhen@uva.nl, cgmsnoek@uva.nl, ling.shao@ieee.org

Abstract

In this paper, we propose variational prototype inference to address few-shot semantic segmentation in a probabilistic framework. A probabilistic latent variable model infers the distribution of the prototype that is treated as the latent variable. We formulate the optimization as a variational inference problem, which is established with an amortized inference network based on an auto-encoder architecture. The probabilistic modeling of the prototype enhances its generalization ability to handle the inherent uncertainty caused by limited data and the huge intra-class variations of objects. Moreover, it offers a principled way to incorporate the prototype extracted from support images into the prediction of the segmentation maps for query images. We conduct extensive experimental evaluations on three benchmark datasets. Ablation studies show the effectiveness of variational prototype inference for few-shot semantic segmentation by probabilistic modeling. On all three benchmarks, our proposal achieves high segmentation accuracy and surpasses previous methods by considerable margins.

1. Introduction

Semantic segmentation perceives the visual-world with pixel-level precision to help recognize and localize objects with rich details. Deep learning based models have achieved astonishing progress in semantic segmentation [2, 17]. However, they usually require a large amount of pixel-wise annotations for supervision which is expensive to obtain in practice. Moreover, the categories of objects to be segmented in the test stage must always be included in the training stage, which restricts its generality for practical use. Thus, few-shot semantic segmentation [25, 4] has recently emerged as a popular task to deal with the aforementioned issues in traditional semantic segmentation. The goal of few-shot segmentation is to segment the object of an unseen category in a query image with the support of only a few annotated images.

A critical challenge in few-shot semantic segmentation is the scarcity of annotated data for each object category to be segmented. Hence, faithfully extracting the class of the objects from the support images is key to guiding the segmentation of objects in the query image. Inspired by the prototype theory from cognitive science [24, 38] and prototypical networks for few-shot classification [28], the prototype-based framework has recently become popular for few-shot segmentation as well [19, 36, 4, 31, 26]. Generally, the prototype refers to some characteristic representation of a category, which is obtained by a deep neural network that takes support images and segment annotations as input. Subsequently, it guides the segmentation procedure of a query image by learning a certain metric [4]. The prototype-based methods have achieved good progress in few-shot semantic segmentation tasks. By mapping scarce support images to a deterministic class prototype, those methods learn transferable knowledge for segmenting arbitrary unseen classes. However, deterministic models suffer from two shortcomings: (1) Representing the prototype by a deterministic vector can be ambiguous and is vulnerable to noise because of the limited training data. (2) Capturing the information of objects by a single vector is inadequate, since objects in the same category usually exhibit great intra-class variations, as illustrated in Fig 1 (a). We address few-shot semantic segmentation by a new, probabilistic model that tackles these shortcomings.
Figure 1. Deterministic model (previous work) vs. probabilistic model (this work): The deterministic model embeds the support set into a single deterministic vector as the prototype, measuring the distance between the prototype vector and feature vectors of pixels on the query image. The deterministic prototype tends to be biased and lacks the ability to represent categorical concepts. The proposed probabilistic model infers the distribution of the prototype, which is treated as a latent variable, from the support set. The probabilistic prototype is more expressive of categorical concepts and endows the model with better generalization to unseen objects.

Our main contribution is to provide the first probabilistic framework for few-shot semantic segmentation. We model the class prototype as a distribution rather than a deterministic vector, which is able to better handle the uncertainty caused by limited support images and enhances the generalization for handling large intra-class variations of objects. Our second contribution is an optimization formulated as a variational inference problem, which we call variational prototype inference (VPI). The optimization objective is built upon a newly derived evidence lower bound, which well fits the few-shot segmentation problem and offers a principled way to incorporate the prototype into segmentation by conditional inference. To evaluate our proposal, we conduct extensive experiments on three benchmarks, i.e., Pascal-5i [25], MS-COCO [16] and FSS-1000 [32]. The ablative results demonstrate the benefit of the proposed probabilistic modeling for few-shot semantic segmentation. The comparison results show that our VPI outperforms the previous deterministic models on both 1-shot and 5-shot semantic segmentation tasks, showing its effectiveness for few-shot semantic segmentation.

2. Related Work

Many-Shot Semantic Segmentation  Semantic segmentation aims to segment given images within several predefined classes and is often regarded as a pixel-level classification task. State-of-the-art semantic segmentation methods based on deep convolutional neural networks [17, 37, 2, 23, 1, 15] have achieved astonishing success. The fully convolutional network [17] was the first model to introduce end-to-end convolutional neural networks into segmentation tasks, in which a fully convolutional architecture was designed. DeepLab [2] introduced the dilated convolution operation to enlarge the perception field while maintaining the resolution. However, to achieve good performance, fully convolutional networks must be heavily-parameterized and trained on a large number of images with pixel-level annotations, which are laborious to obtain. Moreover, the deep semantic segmentation models usually perform modest on new categories of objects that are unavailable in the training set, which restricts their use in practical applications.

Few-Shot Semantic Segmentation  In contrast to many-shot semantic segmentation, few-shot semantic segmentation aims to segment images from arbitrary classes by learning transferable knowledge with limited annotated support images. It has recently gained popularity in computer vision due to its promise in practical applications. Shaban et al. [25] introduced the first few-shot segmentation network based on a two-branch architecture, which uses a support branch to predict the parameters of the last layer of the query branch for segmentation. Recent works [4, 33, 36, 20, 26] follow the two-branch architecture for few-shot semantic segmentation. Dong and Xing. [4] introduced the idea of prototype learning from few-shot recognition for few-shot segmentation. They designed the PLNet in which the first branch learns a prototype vector that takes images and annotations as input and outputs the prototype; while the second branch takes both a new image and the prototype as input and outputs the segmentation mask. Since then, the prototype-based methods have been further developed in different ways [36, 20, 26, 31, 34]. Rakelly et al. [20] concatenated the pooled support features and the query image to generate the segmentation maps. Zhang et al. [36] introduced a masked average pooling operation to extract the representative prototype vector from support images and then estimated the cosine similarity between the extracted vector and the query feature map for predicting
the segmentation map. These works have demonstrated the effectiveness of prototype learning for few-shot semantic segmentation. However, a deterministic prototype vector is not sufficiently representative for capturing the categorical concept of the objects and therefore can cause bias and reduced generalization for handling huge variations of objects in the same categories.

**Variational Inference** Variational inference [11] approximates the probability densities of an unknown quantity through optimization given input data. The variational autoencoder (VAE) [13, 22] is a generative model that introduces variational inference into the learning of directed graphical models. Sohn et al. [29] developed the conditional variational auto-encoder (C-VAE) by extending VAE into the conditional generative model for supervised learning. Finn et al. [7] proposed a probabilistic meta-learning algorithm based on the above ELBO.

In order to infer the latent variable $z$, we maximize the conditional log-likelihood $\log p(y_q|x_q,S)$, which is expanded by incorporating the prior over $z$:

$$
\log p(y_q|x_q,S) = \log \int p(y_q|x_q,S)p(z|x_q,S)dz
$$

$$
= \log \int q(z|x_q,y_q)p(y_q|x_q,S) \frac{p(z|x_q,S)}{q(z|x_q,y_q)}dz,
$$

where we introduce a proposal distribution $q(z|x_q,y_q)$ to approximate the intractable true posterior. By applying the Jensen’s inequality to (1), we obtain

$$
\geq \int q(z|x_q,y_q) \log \frac{p(y_q|x_q,S)}{q(z|x_q,y_q)}dz + D_{KL}[q(z|x_q,y_q)||p(z|x_q,S)]
$$

$$
= -D_{KL}[q(z|x_q,y_q)||p(z|x_q,S)] + E_{q(z|x_q,y_q)}[\log p(y_q|x,z_S)] = \text{ELBO}.
$$

The $D_{KL}[]$ is the Kullback-Leibler (KL) divergence between the estimated posterior distribution $q(z|x_q,y_q)$ and the prior distribution $p(z|x_q,S)$. The second term of the ELBO is the expectation of a conditional generative distribution $p(y_q|x,z_S)$. We derive a variational objective based on the above ELBO.

**Variational Objective** Based on the ELBO, we construct a simplified variational objective, which allows efficient optimization and easy implementation. We replace the prior with $p(z|S)$ by conditioning it solely on the support set. We
We deploy the reparameterization trick proposed in [13] to the predicted segmentation map-wise cross-entropy loss to penalize the difference between for the second term in the empirical loss.

Further remove the condition on $S$ in the predictive posterior, which makes it computationally cheaper. We therefore attain the following objective:

$$
\mathcal{L} = - D_{KL}[q_\phi(z|x_i, y_q)||p_\theta(z|S)] + E_{q_\phi(z|x_i, y_q)}[\log p_\phi(y_q|z, x_q)]
$$

(3)

In the above optimization objective, minimizing the KL term narrows the gap between the posterior distribution $q_\phi(z|x_i, y_q)$ and the prior distribution $p_\theta(z|S)$. This encourages the inferred prototype from the query image to match that from the support images. Maximizing the expectation of log-likelihood $\log p_\phi(y_q|z, x_q)$ guarantees maximally precise prediction of the segmentation map. Based on (3), we attain the empirical objective for stochastic optimization as follows:

$$
\mathcal{L} = \sum_i \left( D_{KL}[q_\phi(z|x_i^k, y_q)||p_\theta(z|S^i)] + \frac{1}{L} \sum_{l=1}^L -\log p_\phi(y_q^i|x_q, z^{(l)}) \right),
$$

(4)

where $i$ indexes over the number of support-query pairs in the training data $D_{\text{train}}$, $z^{(l)} \sim p_\theta(z|S)$ and $L$ is the number of samples.

We take the multivariate Gaussian with a diagonal covariance structure for the distributions, and then the prior $p_\theta(z|S)$ and the posterior $q_\phi(z|x_i, y_q)$ can be parameterized by $N(z; \mu_{prior}, \sigma^2_{prior})$ and $N(z; \mu_{post}, \sigma^2_{post})$. As for the second term in the empirical loss $\mathcal{L}$, we adopt pixel-wise cross-entropy loss to penalize the difference between the predicted segmentation map $\hat{y}_q$ and the ground truth $y_q$.

We deploy the reparameterization trick proposed in [13] to solve the non-differentiable problem existing in the sampling process. Specifically, the class prototype $z$ is obtained by $z^{(l)} = \epsilon^{(l)} \odot \sigma_{prior} + \mu_{prior}$, where $\odot$ denotes the element-wise multiplication and $\epsilon^{(l)} \sim N(\epsilon; 0, 1)$. The number of samples $L$ is set to $1$ during training, as suggested in [13].

In the learning stage, as shown in Fig. 2 (a), we estimate the prior distribution $p_\theta(z|S)$ over the prototype $z$ and the posterior distribution $q_\phi(z|x_q, y_q)$ conditioned on the query image $x_q$ and ground truth $y_q$. To efficiently train the parameters with gradient descent, we rely on Monte Carlo sampling to draw $L$ samples $\{z^{(l)}\}_{l=1}^L$ from $p_\theta(z|S)$ and combine them with the query image to generate the segmentation map.

**Inference** The inference of segmentation maps is shared across learning and test stages. As shown in Fig. 2 (b), we utilize Monte Carlo sampling to draw $L$ potential prototypes $\{z^{(l)}\}_{l=1}^L$ from $p_\theta(z|S)$. The $\bar{y}_q$ is obtained by taking the average of $L$ segmentation maps based on the samples $z$.

$$
\bar{y}_q = \frac{1}{L} \sum_{l=1}^L p_\phi(y_q|x_q, z^{(l)}) , \quad z^{(l)} \sim p_\theta(z|S).
$$

(5)

For the $k$-shot setting, we generate a prior by each of the $k$ pairs of support images and masks: $\{N_i(z; \mu_i, \sigma^2_i)\}_{i=1}^k$, obtaining $k$ priors. We aggregate those $k$ priors with a variance-weighted average operation, which produces the overall aggregated distribution $N(z; \mu, \sigma^2)$:

$$
\mu = \sum_{i=1}^k \frac{1}{\sigma^2_i} \mu_i , \quad \sigma^2 = \frac{k}{\sum_{i=1}^k \frac{1}{\sigma^2_i}}.
$$

(6)

In contrast to the equal-weighted average operation, the variance-weighted average operation lets the distributions with small variance receive larger weights, resulting in the more representative distributions being enhanced, and less important being constrained.
space to approach the true posterior given a query image
a proposal posterior distribution
query image $x$
the support set $z$
net embeds the support set $n$ technique [13]. The networks that parameterize the three distri-

Prior Net The prior net deploys a CNN encoder to extract
marized in Algorithm 1.

Posterior Net Similar to the prior net, the posterior net
utilizes the same CNN encoder to extract the features of the
query image $x_q$, and then uses the ground-truth mask $y_q$
to acquire a global feature vector. Finally, a mean vector $\mu_{post}$ and
and a variance vector $\sigma^2_{post}$ are output from the posterior net
for the posterior distribution:

Segmentation Net The segmentation net takes the concate-
ation of the deep feature of the query image $x_q$ and
the prototype vector $z$ sampled from the prior (see also Fig. 2
(a)). Taking the feature representations of the query image
$x_q$ and the sampled $z$ as input, a CNN-based decoder produces
the output segmentation map:

Implementation Details We deploy the ResNet101 [9]
backbone pre-trained on ImageNet [3] as the encoder. The
decoder is composed of three convolutional blocks to gen-
erate feature maps, each of which is concatenated with the
respective encoded feature through the skip connec-
tions. The support and query images are randomly cropped
feature vector to a mean vector $\mu_{prior}$ and variance vector
$\sigma^2_{prior}$ in the latent space by two fully connected layers:

Inference:

Output: A query image $x_q$ and a support set $S$

Prior Net

Output: $p_\theta(z|S)$, $q_\phi(z|x_q, y_q)$, $p_\psi(y_q|x, z)$

Posterior Net

Output: $\tilde{y}_q$

3.2. Implementation with Amortized Networks

We implement the proposed VPI with neural networks of
the auto-encoder architecture using the amortization tech-
nique [13]. The networks that parameterize the three distri-

Datasets We conduct experiments on three commonly-
used benchmarks including the PASCAL-5$^i$, COCO-20$^i$
and FSS-1000 datasets.

PASCAL-5$^i$, we follow the setting in [25] dividing the
20 original classes in PASCAL VOC12 [5] and Extended
SDS [8] into four folds and conduct cross-validation among
those folds. Specifically, 15 object classes are used during
training, while the remaining 5 classes are for testing for
each fold. During evaluation, we sample the same test set
containing 1,000 support-query pairs for each category as in

COCO-20$^i$ [10], the 80 classes in MSCOCO are split into
four folds and we conduct four-fold cross-validation. Simi-
lar to the setting in PASCAL-5$^i$, 60 object categories are
used during training, while the remaining 20 categories are
used for testing. In each fold, we sample 1000 support-
query pairs from the selected 20 test classes, following [19].

FSS-1000 collected by [32], consists of 1000 object classes,
we choose the same 240 sub classes as in [32] for evaluation
and the remaining classes for training.

Implementation Details We deploy the ResNet101 [9]
backbone pre-trained on ImageNet [3] as the encoder. The
decoder is composed of three convolutional blocks to gen-
rate feature maps, each of which is concatenated with the
respective encoded feature through the skip connections.

In the decoder, we deploy a multi-layer skip-connections
structure [23] to incorporate more spatial information.

4. Experiments and Results

Authorized licensed use limited to: Universiteit van Amsterdam. Downloaded on May 31,2022 at 08:41:46 UTC from IEEE Xplore. Restrictions apply.
to 384 × 384 and augmented by random horizontal flipping and random rotation operations. The model is trained with the Adam optimizer [12] using a batch size of 16 on 4 NVIDIA GeForce TITAN X GPU for 40,000 iterations. The learning rate is fixed to 1e−6 for the backbone and 1e−5 for other layers, and the BN layers are frozen during training. The number of the samples $L$ is set to 6 during the test phase, which is analyzed in detail by our ablation study in Sec. 4.1.

We adopt two metrics for evaluation, Class-IoU [25] and Binary-IoU [20]. Class-IoU measures the Intersection-over-Union $\text{IoU} = \frac{TP}{TP + FP + FN}$, where TP, FP and FN are the number of pixels that are true positives, false positives and false negatives of the predicted segmentation masks for each foreground class $c$. Binary-IoU treats all object classes as foreground class and averages the IoU of foreground and background.

4.1. Ablation Study

**Benefit of Probabilistic Modeling** The main difference between our probabilistic model and previous deterministic models is that we estimate the distribution of the class prototype in the latent space instead of learning a deterministic prototype vector. To demonstrate the advantage of the proposed probabilistic modeling, we implement a deterministic counterpart. For fair comparison, we roughly keep the same network architecture and predict a deterministic class prototype vector by the $\mu$ branch and remove the KL divergence term during training. We implement both models with a VGG-16 [27], ResNet50 [9], and ResNet101 [9] backbone, which are commonly adopted in previous works [19, 33].

The results on PASCAL-5i are shown in Table 1. Our variational prototype inference achieves better performance than the deterministic models on both the 1-shot and 5-shot settings for the Class-IoU as well as the Binary-IoU metric. This is because the proposed probabilistic modeling of prototypes is more expressive of object classes and can better capture the categorical concept of objects, compared to the deterministic representation of prototypes. Therefore, the learned model is endowed with a stronger generalization ability to query images that usually exhibits huge variations. The results verify the advantage of probabilistic modeling for few-shot semantic segmentation. Note that, as expected the ResNet101 backbone outperforms the one with VGG16 as well as the one with ResNet50, and henceforth we adopt ResNet101 as our backbone network in our experiments.

**Effect of Monte Carlo Sampling** During the inference of segmentation maps, we utilize the Monte Carlo sampling to obtain multiple prototypes $z$ to produce multiple segmentation maps, which are aggregated into the final segmentation map. We study the effect of the Monte Carlo sampling on the segmentation results. As shown in Fig. 3, the segmentation map for each sampled prototype is not always...
Table 3. Comparison with state-of-the-art in terms of Class-IoU on PASCAL-5i.

<table>
<thead>
<tr>
<th></th>
<th>1-shot</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>5-shot</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>fold-0</td>
<td>fold-1</td>
<td>fold-2</td>
<td>fold-3</td>
<td>mean</td>
<td>fold-0</td>
<td>fold-1</td>
<td>fold-2</td>
<td>fold-3</td>
<td>mean</td>
</tr>
<tr>
<td>OSLSM [25]</td>
<td>33.6</td>
<td>55.3</td>
<td>40.9</td>
<td>33.5</td>
<td>40.8</td>
<td>35.9</td>
<td>58.1</td>
<td>42.7</td>
<td>39.1</td>
<td>43.9</td>
</tr>
<tr>
<td>Co-FCN [20]</td>
<td>36.7</td>
<td>50.6</td>
<td>44.9</td>
<td>32.4</td>
<td>41.1</td>
<td>40.3</td>
<td>55.3</td>
<td>49.9</td>
<td>40.1</td>
<td>46.4</td>
</tr>
<tr>
<td>AMP [26]</td>
<td>41.9</td>
<td>50.2</td>
<td>46.7</td>
<td>34.7</td>
<td>43.4</td>
<td>40.3</td>
<td>55.3</td>
<td>49.9</td>
<td>40.1</td>
<td>46.4</td>
</tr>
<tr>
<td>SG-One [36]</td>
<td>40.2</td>
<td>58.4</td>
<td>48.4</td>
<td>38.4</td>
<td>46.3</td>
<td>41.9</td>
<td>58.6</td>
<td>48.6</td>
<td>39.4</td>
<td>47.1</td>
</tr>
<tr>
<td>PANet [31]</td>
<td>42.3</td>
<td>58.0</td>
<td>51.1</td>
<td>41.2</td>
<td>48.1</td>
<td>51.8</td>
<td>64.6</td>
<td>59.8</td>
<td>46.5</td>
<td>55.7</td>
</tr>
<tr>
<td>CANet [34]</td>
<td>52.5</td>
<td>65.9</td>
<td>51.3</td>
<td>51.9</td>
<td>55.4</td>
<td>55.5</td>
<td>67.8</td>
<td>51.9</td>
<td>53.2</td>
<td>57.1</td>
</tr>
<tr>
<td>PGNet [33]</td>
<td>56.0</td>
<td>66.9</td>
<td>50.6</td>
<td>50.4</td>
<td>56.0</td>
<td>57.7</td>
<td>68.7</td>
<td>52.9</td>
<td>54.6</td>
<td>58.5</td>
</tr>
<tr>
<td>FWB [19]</td>
<td>51.3</td>
<td>64.5</td>
<td>56.7</td>
<td>52.2</td>
<td>56.2</td>
<td>54.8</td>
<td>67.4</td>
<td>62.2</td>
<td>55.3</td>
<td>59.9</td>
</tr>
<tr>
<td>VPI</td>
<td>53.4</td>
<td>65.6</td>
<td>57.3</td>
<td>52.9</td>
<td>57.3</td>
<td>55.8</td>
<td>67.5</td>
<td>62.6</td>
<td>55.7</td>
<td>60.4</td>
</tr>
</tbody>
</table>

Table 4. Comparison with state-of-the-art in terms of Binary-IoU on PASCAL-5i.

<table>
<thead>
<tr>
<th></th>
<th>1-shot</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>5-shot</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>fold-0</td>
<td>fold-1</td>
<td>fold-2</td>
<td>fold-3</td>
<td>mean</td>
<td>fold-0</td>
<td>fold-1</td>
<td>fold-2</td>
<td>fold-3</td>
<td>mean</td>
</tr>
<tr>
<td>Co-FCN [20]</td>
<td>60.1</td>
<td>60.2</td>
<td></td>
<td></td>
<td></td>
<td>61.2</td>
<td>62.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMP [26]</td>
<td>60.1</td>
<td>62.1</td>
<td></td>
<td></td>
<td></td>
<td>61.2</td>
<td>62.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PL+SEG [4]</td>
<td>61.2</td>
<td>62.3</td>
<td></td>
<td></td>
<td></td>
<td>63.9</td>
<td>65.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A-MCG [10]</td>
<td>66.2</td>
<td>69.6</td>
<td></td>
<td></td>
<td></td>
<td>66.5</td>
<td>70.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OSLSM [25]</td>
<td>61.3</td>
<td>61.5</td>
<td></td>
<td></td>
<td></td>
<td>66.5</td>
<td>70.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SG-One [36]</td>
<td>69.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>69.9</td>
<td>70.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CANet [34]</td>
<td>70.3</td>
<td>72.1</td>
<td></td>
<td></td>
<td></td>
<td>70.3</td>
<td>72.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PGNet [33]</td>
<td>70.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>70.3</td>
<td>72.1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4. Visualization of one-shot segmentation results on the PASCAL-5i dataset.

Comparison on Distribution Aggregation For k-shot learning, we adopt variance-weighted operation in Equation 6 for distribution aggregation. Here we compare it with another ordinary average operation:

$$\mu_o = \frac{\sum_{i=1}^{k} \mu_i}{k}, \quad \sigma^2_o = \frac{\sum_{i=1}^{k} \sigma^2_i}{k}. \quad (10)$$

Results in Table 2 shows the variance-weighted aggregation of $\mu$ and $\sigma$ outperforms the direct average operation.

4.2. Comparison with State-of-the-Art

PASCAL-5i In Table 3, we compare the performance of VPI with the state-of-the-art deterministic methods on PASCAL-5i in terms of the Class-IoU metric. In both 1-shot and 5-shot settings, our VPI outperforms other methods by considerable margins. We improve over the state-of-the-art set by Nguyen and Todorovic [19] by 1.1% and 0.5% for the 1-shot and 5-shot settings. The performance advantage of our VPI is larger on the 1-shot setting, which is more challenging compared to the 5-shot setting. Due to the probabilistic modeling of the prototype in VPI, it better captures the nature of objects even with only one support image. Table 4 shows the state-of-the-art comparison in terms of the Binary-IoU metric. Our VPI achieves the best scores in both the 1-shot and 5-shot settings with 70.3% and 72.1%.

Some qualitative results on PASCAL-5i are visualized in Fig. 4. The proposed VPI achieves accurate segmentation maps in various challenging scenarios, where the query images exhibit variation in appearance and object size from the associated support images. For instance, in the second
### Table 5. Comparison on COCO-20$^i$.

<table>
<thead>
<tr>
<th></th>
<th>Class-IoU</th>
<th>Binary-IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-shot</td>
<td>5-shot</td>
</tr>
<tr>
<td></td>
<td>1-shot</td>
<td>5-shot</td>
</tr>
<tr>
<td>A-MCG</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FWB</td>
<td>21.2</td>
<td>23.7</td>
</tr>
<tr>
<td>PANet</td>
<td>20.9</td>
<td>29.7</td>
</tr>
<tr>
<td>VPI</td>
<td>23.4</td>
<td>27.8</td>
</tr>
</tbody>
</table>

### Table 6. Comparison on FSS-1000.

<table>
<thead>
<tr>
<th></th>
<th>Positive-IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-shot</td>
</tr>
<tr>
<td>OSLSM</td>
<td>70.3</td>
</tr>
<tr>
<td>Co-FCN</td>
<td>71.9</td>
</tr>
<tr>
<td>FSS-1000</td>
<td>73.5</td>
</tr>
<tr>
<td>VPI</td>
<td>84.3</td>
</tr>
</tbody>
</table>

Figure 5. Class-wise comparison between VPI and Wei et al. [32] on FSS-1000.

---

column, the size and viewpoint of the plane in the query image is considerably different from the annotated plane in the support image; in the third column, the annotated bottle in the support image is much smaller than the one the query image. Moreover, the bottle in the support image is also partially occluded.

**COCO-20$^i$** Compared to PASCAL-$5^i$, the scenes in COCO-20$^i$ are more complicated with large intra-class diversity, which poses greater challenges for few-shot semantic segmentation. Therefore few-shot segmentation on COCO-20$^i$ has more ambiguity and it is difficult to acquire a precise class-specific deterministic prototype. As can be seen in Table 5, our method outperforms the state-of-the-art set by PANet [31] by 2.5% and 1.9% in terms of the Class-IoU and Binary-IOU metrics in the 1-shot setting.

**FSS-1000** We evaluate our method following the same settings as in Wei et al. [32]. The metric used for FSS-1000 is the intersection-over-union (IoU) of positive labels in a binary segmentation map, which we adopt for a fair comparison. The performance comparison with previous methods in terms of Positive-IoU is shown in Table 6. We improve over the state-of-the-art set by Wei et al. [32] by 10.8% and 7.6% in the 1-shot and 5-shot settings, showing the effectiveness of our proposal for few-shot semantic segmentation with a large number of categories. Fig. 5 shows the class-wise performance comparison between Wei et al. [32] and our proposed VPI. On most categories, VPI outperforms Wei et al. [32] by a good margin. Moreover, we observe that the performance of VPI does not change much across classes, indicating its robustness and generalization ability. The qualitative segmentation results on the FSS-1000 dataset are illustrated in supplementary materials, where VPI produces accurate segmentation close to the ground truth.

### 5. Conclusion

In this paper, we present a new, probabilistic model for few-shot semantic segmentation. We formulate the class prototype as a latent variable, the distribution over which is inferred from data. We develop variational prototype inference (VPI) to leverage the technique of variational inference for efficient optimization. By probabilistic modeling, we are able to estimate a more robust class prototype distribution that takes the inherent uncertainty in few-shot segmentation into account. Moreover, the probabilistic representation of prototypes better captures the categorical information of objects, which enhances the generalization ability of the model to new unseen categories of objects. We perform comprehensive experiments on three benchmark datasets. The thorough ablation studies demonstrate the benefit of our VPI by probabilistic modeling and the extensive comparison with state-of-the-art methods shows the performance advantage of VPI for few-shot semantic segmentation.

### Acknowledgements

This work was supported in part by the National Key Scientific Instrument and Equipment Development Project under Grant 61827901, the National Security Major Basic Research Program of China under Grant 15001303, the Guangxi Municipal Science and Technology Project under Grant 31062501, and National Natural Science Foundation of China under Grants 61976060 and 61871016.
References


[36] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Thomas Huang. 
Sg-one: Similarity guidance network for one-shot semantic 

[37] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang 
Wang, and Jiaya Jia. Pyramid scene parsing network. In 

[38] Xiantong Zhen, Yingjun Du, Huan Xiong, Qiang Qiu, Cees 
Snoek, and Ling Shao. Learning to learn variational semantic 