NP-SemiSeg: When Neural Processes meet Semi-Supervised Semantic Segmentation

Jianfeng Wang [1] **Daniela Massiceti** [2] **Xiaolin Hu** [3] **Vladimir Pavlovic** [4] **Thomas Lukasiewicz** [5 1]

Abstract

Semi-supervised semantic segmentation involves assigning pixel-wise labels to unlabeled images at training time. This is useful in a wide range of real-world applications where collecting pixel-wise labels is not feasible in time or cost. Current approaches to semi-supervised semantic segmentation work by predicting pseudo-labels for each pixel from a class-wise probability distribution output by a model. If the predicted probability distribution is incorrect, however, this leads to poor segmentation results, which can have knock-on consequences in safety critical systems, like medical images or self-driving cars. It is, therefore, important to understand what a model does not know, which is mainly achieved by uncertainty quantification. Recently, neural processes (NPs) have been explored in semi-supervised image classification, and they have been a computationally efficient and effective method for uncertainty quantification. In this work, we move one step forward by adapting NPs to semi-supervised semantic segmentation, resulting in a new model called NP-SemiSeg. We experimentally evaluated NP-SemiSeg on the public benchmarks PASCAL VOC 2012 and Cityscapes, with different training settings, and the results verify its effectiveness.

1. Introduction

Semi-supervised image segmentation has seen a rapid progress in recent years and involves assigning class labels to every pixel in an unlabeled image at training time. This has many real-world applications, from medical imaging to autonomous driving systems, where the cost and time to annotate large-scale training datasets with pixel-level labels is prohibitive.

Most recent works (Alonso et al., 2021; Chen et al., 2021a;b; French et al., 2020; Hu et al., 2021a; Ouali et al., 2020; Zhong et al., 2021; Wang et al., 2022b; Guan et al., 2022; Liu et al., 2022; Kwon & Kwak, 2022; Yang et al., 2022; Zhao et al., 2023) belong to the deterministic approach that aims at directly making a prediction for an input image. That is, it does not model the predictive distribution, and only gives a point estimate. In contrast, a method modeling a predictive distribution for the input is classified as probabilistic approach. Its key advantage is that one can estimate the uncertainty for an input by simply sampling from the posterior. The uncertainty provides information about whether the prediction is reliable, and thus how to estimate uncertainty should be considered under the setting of semi-supervised learning (SSL), as the performance of segmentation models is vulnerable to unlabeled data with incorrect pseudo-labels, and decision-makers need to know when they should not trust the models.

Unfortunately, the probabilistic approach is insufficiently investigated, as researchers barely explored its application to semi-supervised semantic segmentation for computer vision, and most related works focus on medical imaging (Sedai et al., 2019; Shi et al., 2021; Yu et al., 2019; Li et al., 2020; Wang et al., 2021; Wang & Lukasiewicz, 2022; Meyer et al., 2021; Xiang et al., 2022), in which Monte Carlo (MC) dropout has been the mainstream option for uncertainty quantification. MC dropout, however, has some limitations that hinder it from real scenarios. For instance, it can be time-consuming when it is combined with cumbersome segmentation models, as several feedforward passes are required for estimating uncertainty. In addition, architectural choices, such as where to insert dropout layers and the value of the dropout rate, are usually empirically set, which may result in a suboptimal performance. To tackle the limitations, a very recent work (Wang et al., 2022a) has studied neural processes (NPs) for SSL, in which a new model named NP-Match is proposed. Compared to MC-dropout-based SSL models, NP-Match is computationally significantly more efficient, as it only needs to perform one

[1]Department of Computer Science, University of Oxford, UK. [2]Microsoft Research, Cambridge, UK. [3]Department of Computer Science and Technology, Tsinghua University, Beijing, China. [4]Department of Computer Science, Rutgers University, New Jersey, USA. [5]Vienna University of Technology, Austria. Correspondence to: Jianfeng Wang <jianfeng.wang@cs.ox.ac.uk>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

feedforward pass to derive the prediction with an uncertainty estimate for a given input. Moreover, in NP-Match, NPs are directly built on top of convolutional neural networks (CNNs), and hence, unlike MC dropout, which has to be empirically set, NPs are more convenient to use.

Considering the success of NP-Match and insufficient exploration towards the probabilistic approach for semi-supervised semantic segmentation, in this work, we investigate the application of NPs on semi-supervised semantic segmentation, and propose a new model, called NP-SemiSeg. In particular, we primarily made two modifications when designing NP-SemiSeg. First, a global latent variable is predicted for each input image, rather than producing a global latent vector shared by different images.[1] This change is inspired by the fact that different images may have different prior label distributions. Hence, it is more reasonable to assume that every image has its own specific prior, and NP-SemiSeg should separately predict a global latent vector for every image, shared by all its pixels. Second, attention mechanisms are additionally introduced to both the deterministic path and the latent path. In the original NPs (Garnelo et al., 2018b), the information of context points or target points is summarized via a mean aggregator in both paths, and NP-Match also follows this practice. However, the mean aggregator introduces the issue that the decoder of NPs cannot capture relevant information for a given target prediction, as the mean aggregator gives the same weight to each point. Inspired by another model named attentive NPs (Kim et al., 2019), attention mechanisms are also integrated into NP-SemiSeg to solve this issue.

To validate the effectiveness of NP-SemiSeg, we conducted several experiments on two public benchmarks, namely, PASCAL VOC 2012 and Cityscapes, with diverse SSL settings, and the results show two merits of NP-SemiSeg. First, NP-SemiSeg is versatile and flexible, because it can be integrated into different segmentation frameworks, such as CPS (Chen et al., 2021b) or U^{2}PL (Wang et al., 2022b). Equipped with NP-SemiSeg, those frameworks are turned into probabilistic models, which are able to make predictions and quantify the uncertainty for input samples. Second, compared to the widely used MC-dropout-based segmentation models, the segmentation models with NP-SemiSeg are faster in terms of uncertainty quantification and are able to give higher-quality uncertainty estimates with less performance degradation, indicating that NP-SemiSeg can be a good alternative probabilistic method to MC dropout.

It should be noted that the principal objective of this research is not to introduce a new segmentation approach that surpasses all state-of-the-art methods. Rather, the aim is to present a novel probabilistic model for semi-supervised semantic segmentation, capable of delivering both a good performance and reliable uncertainty estimates. Summarizing, the main contributions of this paper are:

- We adjust NPs to semi-supervised semantic segmentation, and propose a new probabilistic model, named NP-SemiSeg, which is flexible and can be combined with different existing segmentation frameworks for making predictions and estimating uncertainty.
- We integrate an attention aggregator into NP-SemiSeg, which assigns higher weights to the information that is more relevant to target data, enhancing the performance of NP-SemiSeg.
- Compared to MC-dropout-based segmentation models, NP-SemiSeg not only performs better in terms of accuracy, but also runs faster regarding uncertainty estimation, showing its potential to be a new probabilistic model for semi-supervised semantic segmentation.

The rest of this paper is organized as follows. In Section 2, we briefly discuss related works. Section 3 elaborates our NP-SemiSeg, followed by our experimental details and results in Section 4. Finally, we give a conclusion and some future research directions in Section 5. The source code is available at: https://github.com/Jianf-Wang/NP-SemiSeg.

2. Related Works

In this section, we briefly review related works, including SSL for image classification, semi-supervised semantic segmentation, and the neural process (NP) family.

SSL for Image Classification. In the past few years, many methods have been proposed for semi-supervised image classification, which provide insights and research directions for semi-supervised semantic segmentation. The most prevalent method is FixMatch (Sohn et al., 2020). During training, it produces pseudo-labels for weakly-augmented unlabeled data based on a preset confidence threshold, and the pseudo-labels are used as the ground-truth for their strongly augmented version to train the whole framework. FixMatch (Sohn et al., 2020) thereafter inspired a series of promising methods (Li et al., 2021; Rizve et al., 2021; Zhang et al., 2021; Nassar et al., 2021; Pham et al., 2021; Hu et al., 2021b). For example, Li et al. (2021) incorporate contrastive learning through additionally designing the projection head that generates low-dimensional embeddings for samples. The low-dimensional embeddings with similar pseudo-labels are encouraged to be close, which improves the quality of pseudo-labels. Zhang et al. (2021) use dynamic confidence thresholds that are adjusted based on the model's learning status of each class, rather than the fixed preset confidence threshold. A more relevant method, named uncertainty-aware pseudo-label selection

[1] In NP-Match (Wang et al., 2022a), NPs generate a global latent vector shared by all images within a given batch, which follows the pipeline of the original NPs (Garnelo et al., 2018b).

(UPS) framework, was proposed by Rizve et al. (2021). This framework can be regarded as a probabilistic approach, as it applies MC dropout to obtain uncertainty estimates, based on which unreliable pseudo-labels are filtered out. Due to the weaknesses of MC dropout mentioned above, Wang et al. (2022a) try to explore a new alternative probabilistic model, i.e., NPs, for semi-supervised image classification, and propose a new method called NP-Match, which not only shows a promising accuracy on several public benchmarks, but also alleviates the problem of MC dropout. These results encourage us to further investigate the application of NPs on semi-supervised semantic segmentation.

Semi-supervised Semantic Segmentation. Most methods can be classified into two training paradigms, namely, consistency-training (French et al., 2020; Zhou et al., 2021; Ouali et al., 2020; Zhong et al., 2021; Liu et al., 2022; Ke et al., 2019) and self-training (Alonso et al., 2021; Chen et al., 2021a; Hu et al., 2021a; Wang et al., 2022b; Guan et al., 2022; Kwon & Kwak, 2022; Yang et al., 2022; Zou et al., 2021; Zhao et al., 2023).

The consistency-training methods aim to maintain the consistency among the segmentation results of different perturbations of the same unlabeled sample. For example, Ouali et al. (2020) propose a cross-consistency training (CCT) method, and it contains a main decoder and several auxiliary decoders, which share the same encoder. For the unlabeled examples, a consistency between the main decoder's outputs and the auxiliary outputs is maintained, over different kinds of perturbations leveraged to the inputs of the auxiliary decoders. Zhong et al. (2021) design a new framework, named PC^{2}Seg, which takes advantage of both the pixel-contrastive property and the consistency property during training, and their combination further enhances the performance. Considering the potential inaccurate training signal caused by perturbations, Liu et al. (2022) introduce an additional teacher model, a stricter confidence-weighted cross-entropy loss, and a new type of feature perturbation to improve consistency learning.

Self-training methods assign pixel-wise pseudo-labels to unlabeled data, and re-train the segmentation networks. For instance, PseudoSeg (Zou et al., 2021) utilizes the predictions of unlabeled data as the labels to re-train the whole framework. To obtain accurate pseudo-labels, a calibrated fusion module is incorporated, which fuses both the outputs of the decoder and the refined class activation map (CAM). The success of self-supervised learning motivates Alonso et al. (2021) to integrate the pixel-level contrastive learning scheme into their framework, which aims at enforcing the feature vector of a target pixel to be similar to the same-class features from the memory bank. Recently, Wang et al. (2022b) have discovered that some pixels may never be learned in the entire self-training process, due to their low confidence scores. Then, they propose a new framework, called U^{2}PL, which reconsiders those pixels as negative samples for training. Zhao et al. (2023) reconsider the data augmentation techniques used in the self-training process, and they design a new highly random intensity-based augmentation method and an adaptive cutmix-based augmentation method to enhance the performance.

All above methods do not involve any probabilistic model, and it is only valued in medical imaging (Sedai et al., 2019; Shi et al., 2021; Yu et al., 2019; Li et al., 2020; Wang et al., 2021; Wang & Lukasiewicz, 2022; Meyer et al., 2021; Xiang et al., 2022), where most methods rely on MC dropout for approximating BNNs and estimating uncertainty. In a nutshell, those methods usually leverage uncertainty maps given by MC dropout to refine pseudo-labels for unlabeled data, thereby boosting the capability of their models.

NP Family. The first member of the NP family comes from Garnelo et al. (2018a); it is called conditional NP (CNP). CNPs model the predictive distribution over context sets and target sets. However, CNPs only provide a point-wise uncertainty estimate. In most cases, it would be beneficial to exploit the correlation among different points during inference. Therefore, NPs are proposed to build the correlation points by introducing global latent variables as priors for those points (Garnelo et al., 2018b). Kim et al. (2019) have observed that NPs tend to underfit the context set, which is caused by the mean aggregator giving equal weights to all the context points. To remedy this issue, they propose a new model, called attentive NP, which uses an attention mechanism to attend to relevant context points with respect to target predictions. Concerning that the application areas of NPs are time series or spatial data, the translation equivalence should be an important property, i.e., if the data are translated in time or space, the predictions should be translated correspondingly. This property was ignored in previous models, until Gordon et al. (2020) designed a new model called convolutional CNPs. Besides, concerning that the global latent variables are not flexible for encoding inductive biases, Louizos et al. (2019) employ local latent variables along with a dependency structure among them instead, obtaining a new functional NP (FNP). Similarly, Lee et al. (2020) also point out the limited flexibility of a single latent variable to model functional uncertainty, and they use a classic frequentist technique, namely, bootstrapping, to model functional uncertainty, leading to a new NP variant, named Bootstrapping Neural Processes (BNPs). Bruinsma et al. (2021) propose a new NP variant called Gaussian NPs (GNPs), which not only involves translation equivariance with Gaussian processes (Rasmussen & Williams, 2006), but also provides universal approximation guarantees. Note that we only summarize some classical members of the NP family in this part, and for more variants and their applications, please refer to the survey paper (Jha et al., 2022).

3. Methodology

3.1. Background

Neural Processes (NPs) are a neural network-based formulation that learn to approximate a stochastic process through finite-dimensional marginal distributions (Garnelo et al., 2018b). Their working mechanism is closely related to a classical non-parametric model, Gaussian Processes (GPs). A GP makes the assumption that each point in the input space essentially maps to a normally distributed random variable. The GP model is fully specified by a mean function, which provides the expected value of these random variables, and a kernel function, which describes the dependencies among the variables. Thus, GPs are a powerful probabilistic model that can provide a measure of uncertainty along with predictions. However, GPs are computationally expensive for large datasets and require a careful choice and tuning of the kernel function, which hinders their practical applications. To address these issues, NPs have been proposed.

Before formally defining NPs, we first give the definition of a stochastic process. In general, a stochastic process can be defined as $\{F(x, \omega) : x \in \mathcal{X}\}$ over a probability space (Ω, Σ, Π) and an index set $\mathcal{X}$, where $F(\cdot\ ,\ \omega)$ is a sample function mapping $\mathcal{X}$ to another space $\mathcal{Y}$ for any point $\omega \in \Omega$. Therefore, for any finite sequence $x_{1:n}$, a marginal joint distribution function can be defined on the function values $F(x_1,\ \omega), F(x_2,\ \omega), \ldots, F(x_n,\ \omega)$, which satisfies two conditions given by the Kolmogorov Extension Theorem (Øksendal, 2003):

Exchangeability: *This condition indicates that the marginal joint distribution should remain unaffected by any permutation of the sequence.*

Consistency: *This condition requires that the marginal joint distribution should remain unaffected when a part of the sequence is marginalized out.*

With the two conditions, a stochastic process can be described by the marginal joint distribution function, namely:

$$p(y_{1:n}|x_{1:n}) = \int \pi(\omega) p(y_{1:n}|F(\cdot\ ,\ \omega), x_{1:n}) d\mu(\omega), \quad (1)$$

where π denotes density, namely, $d\Pi = \pi d\mu$. Here, the function $F(\cdot\ ,\ \omega)$ is determined by the kernels, which measure how all variables interact with each other.

To approximate stochastic processes, NPs parameterize the function $F(\cdot\ ,\ \omega)$ in the marginal joint distribution with neural networks and latent vectors. Specifically, let (Ω, Σ) be $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$, where $\mathcal{B}(\mathbb{R}^d)$ denotes the *Borel σ-algebra* of $\mathbb{R}^d$, and NPs use a latent vector $z \in \mathbb{R}^d$ sampled from a multivariate Gaussian distribution to govern the function $F(\cdot\ ,\ \omega)$. Then, $F(x_i,\ \omega)$ can be replaced by $\phi(x_i, z)$, where $\phi(\cdot)$ denotes a neural network, and Eq. (1) becomes:

$$p(y_{1:n}|x_{1:n}) = \int \pi(z) p(y_{1:n}|\phi(x_{1:n}, z), x_{1:n}) d\mu(z). \quad (2)$$

By doing this, NPs are capable of predicting and estimating uncertainty for each data point, circumventing the explicit access to kernel functions and comparisons of distances among distinct points. This capability renders them practical for application in real-world scenarios.

The training objective of NPs is to maximize $p(y_{1:n}|x_{1:n})$, which can be implemented by maximizing its evidence lower-bound (ELBO). The learning procedure reflects the NPs' property that they have the capability to make predictions for target data conditioned on context data (Garnelo et al., 2018b).

3.2. NP-SemiSeg

3.2.1. NPs for Semi-Supervised Semantic Segmentation

Semantic segmentation can be treated as a pixel-wise classification problem, and therefore, $p(y_{1:n}|\phi(x_{1:n}, z), x_{1:n})$ in Eq. (2) can be changed to the categorical distribution (denoted as $\mathcal{C}$). Specifically, a weight matrix ($\mathcal{W}$) and a softmax function (Φ) can be sequentially applied to the feature presentation of every pixel from the decoder $\phi(\cdot)$, outputting a probability vector that can parameterize $\mathcal{C}$. Furthermore, different images can have distinct prior label distributions, as some objects cannot appear in the same image. For example, if an image captures the main road of a city, fish will not appear, whose prior should be zero. But if the image records the creatures in the sea, the prior of fish is close to one. Because of this, rather than using a global latent variable for different images, we instead use a latent variable per image. This can be viewed as giving each image its own prior. Thus, we rewrite $p(y_{1:n}|\phi(x_{1:n}, z), x_{1:n})$ as follows:

$$p(y_{1:n}|\phi(x_{1:n}, z_{1:n}), x_{1:n}) = \mathcal{C}(\Phi(\mathcal{W}\phi(x_{1:n}, z_{1:n}))), \quad (3)$$

where the decoder $\phi(\cdot)$ can be learned through amortised variational inference. Specifically, as for a finite sequence with length n, we assume m context data ($x_{1:m}$) and r target data ($x_{m+1:\ m+r}$) in it, i.e., $m + r = n$. We also assume a variational distribution over latent variables, and the ELBO is given by (with proof in the appendix):

$$log\ p(y_{1:n}|x_{1:n}) \geq$$
$$\mathbb{E}_{q(z_{m+1:\ m+r}|x_{m+1:\ m+r}, y_{m+1:\ m+r})}\Big[\sum_{i=m+1}^{m+r} log\ p(y_i|z_i, x_i) -$$
$$log\ \frac{q(z_{m+1:\ m+r}|x_{m+1:\ m+r}, y_{m+1:\ m+r})}{q(z_{m+1:\ m+r}|x_{1:m}, y_{1:m})}\Big] + log\ p(y_{1:m}|x_{1:m}). \quad (4)$$

Then, one can maximize the ELBO to learn the NP model. During training, we follow the setting of NP-Match (Wang et al., 2022a) which treats only labeled data as context data and treats either labeled or unlabeled data as target data.

3.2.2. NP-SemiSeg Pipeline

We formulate NP-SemiSeg in a modular fashion, so that it can directly replace the classification layer of any segmentation pipeline without changing other modules in the pipeline, to output predictions with uncertainty estimates. As a result, NP-SemiSeg is flexible and can be used for different segmentation frameworks. To achieve this goal, the input of NP-SemiSeg should be feature maps,[2] which is consistent with the input of a classifier in other segmentation frameworks. To make explanations clearer, we only focus on NP-SemiSeg itself.

The overall pipeline of NP-SemiSeg is shown in Figure 1, where we represent the context and target data as generic feature maps, which could be obtained from any semantic segmentation pipeline such as U^{2}PL (Wang et al., 2022b) and AugSeg (Zhao et al., 2023). NP-SemiSeg has a training mode and an inference mode. The former aims to calculate loss functions with real labels or pseudo-labels during training, while the latter makes predictions for unlabeled data during training or test data during testing. In what follows, we describe these two modes:

Training mode. Given a batch of labeled data and a batch of unlabeled data, NP-SemiSeg is initially switched to inference mode, and it makes predictions for the unlabeled data. Those predictions are regarded as pseudo-labels for unlabeled data by taking the class with the highest probability. Then, NP-SemiSeg turns to training mode, and it duplicates the labeled samples and treats them as context data. Subsequently, the context data are passed through a deterministic path, which aims to obtain order-invariant context representations, and the target data are passed through a latent path, which aims to produce latent variables. The outputs from both paths are finally concatenated and then passed through a decoder before the loss is computed. Below, we provide details for the latent path and the deterministic path.

As for the latent path, target data are processed by a small ConvNet[3] at first for dimensionality reduction, whose outputs are transformed feature maps with a low channel dimension. The transformed feature maps are further split along the width (W) and the height (H), resulting in feature vectors. Based on the number of classes, a set of latent memory banks have been initialized, each of which is assigned to a category. Those feature vectors are passed to the latent memory banks according to their real or pseudo labels.[4] Then, a mean operator is used for each memory bank, and we can obtain a center for each class. Those centers and the target transformed feature maps are input to an attention aggregator, whose outputs are feature maps composed by target centers. Specifically, the feature vector of each location in such centers-based feature maps is the weighted summation of the centers, which intends to represent every location by the most relevant features from the memory. Thereafter, the global average pooling and MLPs are used to produce a mean vector and a variance vector for each target data point, followed by a reparameterization trick to get T latent vectors whose dimension is D_t. Finally, those latent vectors are copied for $W \times H$ times, thereby forming latent maps for each target data point with size $T \times D_t \times W \times H$.

As for the deterministic path, context data are processed in the same way as the target data, until we obtain the context centers for classes. Then, the context centers as well as the target transformed feature maps are fed to the attention aggregator, in order to get the feature maps composed by the context centers, which are further processed by global average pooling, leading to an order-invariant context representation with dimension D_c for each target data point. Finally, the order-invariant context representation is copied for $T \times W \times H$ times, thereby forming context maps for each target data point with size $T \times D_c \times W \times H$.

After the latent maps and the context maps are obtained for each target data point, they are concatenated with the original feature maps of the target data whose size is $T \times D \times W \times H$, and the concatenated feature maps will have the size $T \times (D + D_t + D_c) \times W \times H$, based on which a decoder $\phi(\cdot)$ makes pixel-wise predictions. The final prediction for each target data point can be obtained by averaging the T prediction maps, and the uncertainty map is computed as the entropy of the average prediction (Kendall & Gal, 2017). For saving space, only those centers are stored for inference after training, instead of saving those memory banks.

Inference mode. As for a set of test data, they are treated as target data and are first processed by the small ConvNet. Its outputs, the target centers, and the context centers are taken as inputs to the attention aggregator to acquire the feature maps composed by centers. Subsequently, the remaining steps are the same as in the training mode to generate concatenated feature maps where the decoder $\phi(\cdot)$ acts on to make predictions, along with their associated uncertainty estimates.

[2] In general, most segmentation frameworks are based on DeepLab (Chen et al., 2017), where a classifier acts on the final output feature maps from the decoder to predict for every location.

[3] The small ConvNet is mainly composed by 1×1 convolutions, and its outputs have the same spatial size as its inputs.

[4] Note that $q(z_*|x_{m+1:m+r}, y_{m+1:m+r})$ is conditioned on both data and labels, which is implemented by using them as inputs to MLPs in NP-Match, but in NP-SemiSeg, the condition on labels is implicitly implemented, i.e., how data are stored in memory banks is determined by the labels.

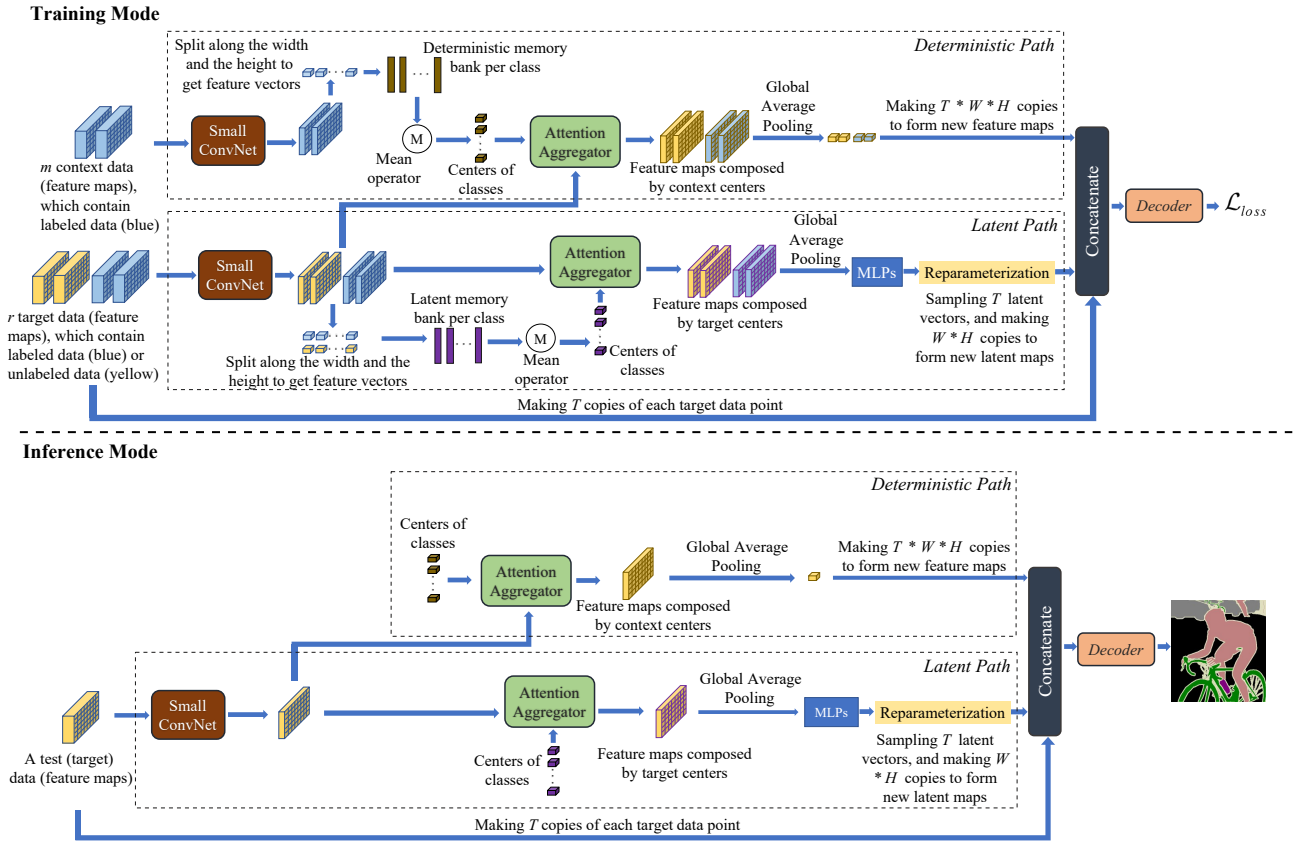


Figure 1. Overview of NP-SemiSeg. Both the small ConvNet and the attention aggregator are shared by the deterministic path and the latent path. T, W, and H represent the number of sampled latent vectors, and the width and height of the input feature maps, respectively.

3.2.3. Attention Aggregator

To predict a target data point, it is beneficial to gather relevant information from memory banks, as the centers close to the target provide similar representations. To achieve this, an attention aggregator is required, whose role is to produce centers-based feature maps based on the distance between query feature maps and a set of centers. We denote the input feature maps and the input centers as $\mathcal{M}$ and $\mathbf{C}$, respectively. The output $\mathcal{M}_{\mathbf{C}}$ is calculated as follows:

$$\mathcal{M}_{\mathbf{C}}[i,j] = \sum_{l} \frac{e^{-\Theta(\mathcal{M}[i,j],\mathbf{C}[l]))}}{\sum_{k} e^{-\Theta(\mathcal{M}[i,j],\mathbf{C}[k]))}} \mathbf{C}[l], \tag{5}$$

where i and j denote the index of feature maps along width and height. Both l and k denote the index of centers. Θ is defined as Euclidean distance over two vectors. In summary, the attention aggregator uses Θ to calculate the distance between the feature vector $\mathcal{M}[i,j]$ at the location (i,j) and every center, and all distances are further used to calculate weights through the softmax function for centers. Then, the output feature at location $[i,j]$, namely, $\mathcal{M}_{\mathbf{C}}[i,j]$, is the weighted combination of those centers. Similarly to ANPs (Kim et al., 2019), by using an attention aggregator, only the relevant information from the latent path and the deterministic path is involved for making predictions, thereby improving the model's performance.

3.2.4. Loss Functions

The loss function for NP-SemiSeg is derived from the ELBO (Eq. (4)). In particular, the first term can be achieved by pixel-wise cross entropy loss L_c for both labeled and unlabeled data, which is widely used in different segmentation frameworks. The second term is the KL divergence between $q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})$ and $q(z_{m+1:\,m+r}|x_{1:m}, y_{1:m})$. Due to the i.i.d assumption, those z_* are conditionally independent, and thus they can be calculated independently. We assume that the variational distribution follows a multivariate Gaussian with independent components, and for each target sample, the KL divergence term can be analytically written as:

$$\begin{aligned} L_{kl} =& 0.5 \times [\sum_{D_t} log\frac{\sigma_c^2}{\sigma_t^2} + \sum_{D_t}\frac{\sigma_t^2}{\sigma_c^2} - D_t + \\ & (m_c - m_t) diag(\sigma_c^{-2})(m_c - m_t)^T], \end{aligned} \tag{6}$$

where $diag(\cdot)$ receives a vector and converts it into a diagonal matrix. m_c and m_t denote the mean vector

Method	1/16 (92)	1/8 (183)	1/4 (366)	1/2 (732)
MT	48.37	58.44	65.49	68.92
PS-MT	63.32	67.78	74.68	76.54
U^{2}PL	62.13	68.11	73.22	75.60
AugSeg	64.22	72.17	76.17	77.40
MT w/ MC dropout	47.78	57.02	64.82	67.79
PS-MT w/ MC dropout	62.09	66.46	73.11	74.30
U^{2}PL w/ MC dropout	59.17	66.89	72.16	74.19
AugSeg w/ MC dropout	62.78	69.87	74.76	76.13
MT w/ NP-SemiSeg	49.02	58.91	65.27	69.34
PS-MT w/ NP-SemiSeg	63.76	68.17	74.93	76.33
U^{2}PL w/ NP-SemiSeg	59.45	68.73	74.16	75.77
AugSeg w/ NP-SemiSeg	65.78	72.38	75.77	77.40

Table 1. The mean IoU of different frameworks using ResNet-50 with either MC dropout or NP-SemiSeg on the *classic* PASCAL VOC 2012 validation set under different partition protocols.

Method	1/16 (662)	1/8 (1323)	1/4 (2646)	1/2 (5291)
MT	66.77	70.78	73.22	75.29
PS-MT	72.83	75.70	76.43	77.88
U^{2}PL	74.74	77.44	77.51	78.62
AugSeg	77.28	78.27	78.24	79.02
MT w/ MC dropout	65.46	69.29	72.39	74.67
PS-MT w/ MC dropout	71.28	74.03	74.97	75.97
U^{2}PL w/ MC dropout	73.79	76.23	76.56	76.41
AugSeg w/ MC dropout	76.42	76.87	77.02	77.56
MT w/ NP-SemiSeg	66.93	71.25	73.10	75.31
PS-MT w/ NP-SemiSeg	73.44	76.58	76.74	76.82
U^{2}PL w/ NP-SemiSeg	75.59	77.77	77.78	77.23
AugSeg w/ NP-SemiSeg	77.00	78.68	78.69	79.03

Table 2. The mean IoU of different frameworks using ResNet-50 with either MC dropout or NP-SemiSeg on the *blender* PASCAL VOC 2012 validation set under different partition protocols.

of $q(z_*|x_{1:m}, y_{1:m})$ and $q(z_*|y_{m+1:\,m+r})$, respectively. Similarly, σ_c^2 and σ_t^2 denote the variance vector of $q(z_*|x_{1:m}, y_{1:m})$ and $q(z_*|y_{m+1:\,m+r})$, respectively. The third term is a conditional distribution over the context data, but it is ignored in our loss function, as its maximization has been implicitly implemented by the attention aggregator, i.e., matching the transformed feature maps to the centers (classes) according to their distances. The overall loss function for NP-SemiSeg can be written as:

$$L_{loss} = L_c + \lambda_{kl} L_{kl}, \tag{7}$$

where λ_{kl} is the coefficient. When NP-SemiSeg is incorporated into different segmentation frameworks, L_{loss} can be naturally incorporated into their loss functions for end-to-end training.

4. Experiments

In this section, we present our experimental results. To save space, the implementation details are given in the appendix.

4.1. Datasets

We tested our models on two public segmentation benchmarks, namely, Cityscapes (Cordts et al., 2016) and PASCAL VOC 2012 (Everingham et al., 2010). Cityscapes is an urban scene understanding dataset containing 2, 975 training images with fine-annotated masks and 500 validation images. We followed previous works (Wang et al., 2022b; Zhao et al., 2023; Chen et al., 2021b) to use the sliding evaluation for fair comparisons. PASCAL VOC 2012 is a standard semantic segmentation dataset that has 20 semantic classes and 1 background class. There are 1,464 and 1,449 images in the training set and the validation set, respectively. Following Wang et al. (2022b); Zhao et al. (2023); Chen et al. (2021b), we used coarsely-labeled 9,118 images from the Segmentation Boundary dataset (SBD) (Hariharan et al., 2011) as additional training data, and we also evaluated our model on the *classic* set and the *blender* set. As in previous works (Wang et al., 2022b; Zhao et al., 2023; Chen et al., 2021b), the center-crops of images were used for evaluation.

Method	1/16 (186)	1/8 (372)	1/4 (744)	1/2 (1488)
MT	66.14	72.03	74.47	77.43
PS-MT	70.12	74.49	76.12	77.64
U^{2}PL	69.03	73.02	76.31	78.64
AugSeg	73.73	76.49	78.76	79.33
MT w/ MC dropout	65.25	71.09	72.48	74.96
PS-MT w/ MC dropout	68.83	73.11	75.25	75.47
U^{2}PL w/ MC dropout	67.89	72.13	75.11	75.85
AugSeg w/ MC dropout	72.28	75.84	77.69	78.04
MT w/ NP-SemiSeg	66.20	72.14	73.89	76.29
PS-MT w/ NP-SemiSeg	70.27	74.67	76.14	76.93
U^{2}PL w/ NP-SemiSeg	69.10	73.04	75.79	75.75
AugSeg w/ NP-SemiSeg	73.01	77.10	78.82	78.77

Table 3. The mean IoU of different frameworks using ResNet-50 with either MC dropout or NP-SemiSeg on the Cityscapes validation set under different partition protocols.

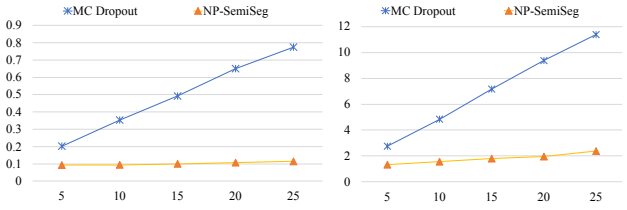
Dataset	Label Amount	MC Dropout	NP-SemiSeg
Cityscapes	1/16 (186)	82.89	84.05
	1/8 (372)	82.84	83.97
	1/4 (744)	82.78	84.55
	1/2 (1488)	82.92	84.61
VOC (*classic*)	1/16 (92)	85.79	86.87
	1/8 (183)	86.42	87.98
	1/4 (366)	87.05	88.74
	1/2 (732)	87.64	89.69
VOC (*blender*)	1/16 (662)	88.04	89.62
	1/8 (1323)	87.96	89.87
	1/4 (2646)	88.18	89.99
	1/2 (5291)	88.42	89.34

Table 4. The PAvPU of U^{2}PL (Wang et al., 2022b) using ResNet-50 with either MC dropout or NP-SemiSeg on different datasets.

4.2. Main Results

In the following, we report the main experimental results on the mean of Intersection over Union (mIoU), the Patch Accuracy vs. Patch Uncertainty (PAvPU) metric (Mukhoti & Gal, 2018), and the running time of NP-SemiSeg over the two benchmarks.

First, because of the flexibility of NP-SemiSeg, we integrated it into different segmentation frameworks to show



(a) Center-crop evaluation on PASCAL VOC 2012 (b) Sliding evaluation on Cityscapes

Figure 2. Time consumption of estimating uncertainty for U^{2}PL (Wang et al., 2022b) with MC dropout and NP-SemiSeg. The horizontal axis refers to the number of predictions used for the uncertainty quantification, and the vertical axis indicates the time consumption (sec).

Dataset	Label Amount	w/o Attention	w/ Attention
Cityscapes	1/16 (186)	67.86	69.10
	1/8 (372)	72.44	73.04
	1/4 (744)	75.33	75.79
	1/2 (1488)	75.45	75.75
VOC (*classic*)	1/16 (92)	58.52	59.45
	1/8 (183)	68.12	68.73
	1/4 (366)	73.72	74.16
	1/2 (732)	75.64	75.77
VOC (*blender*)	1/16 (662)	74.80	75.59
	1/8 (1323)	77.26	77.77
	1/4 (2646)	77.38	77.78
	1/2 (5291)	76.91	77.23

Table 5. Ablation studies of attention aggregation on different datasets. The results are all based on U^{2}PL (Wang et al., 2022b) using ResNet-50, and mean IoU is reported.

its performance. We chose four frameworks, namely, MT (Tarvainen & Valpola, 2017), PS-MT (Liu et al., 2022), U^{2}PL (Wang et al., 2022b), and AugSeg (Zhao et al., 2023). The first two frameworks are classified as the consistency-training method, while the rest belongs to the self-training method. Since MC dropout is the mainstream probabilistic approach in SSL, we also evaluated it by applying it to the four frameworks, and it is inserted after every activation layer in their decoders. From Tables 1, 2, and 3, we have two findings. First, on PASCAL VOC 2012, NP-SemiSeg can help to further improve the mIoU in most cases. In contrast, MC dropout leads to a poor performance, and it is outperformed by NP-SemiSeg with a healthy margin. Second, on Cityscapes, though NP-SemiSeg only achieves comparable results, it still performs clearly better than MC dropout. Thus, compared to MC dropout, NP-SemiSeg is a more favorable choice for semi-supervised semantic segmentation, as it does not cause a serious performance degradation. In the other experiments, we fixed a single framework, i.e., U^{2}PL (Wang et al., 2022b), to further explore NP-SemiSeg.

Second, we compare the PAvPU of NP-SemiSeg with that of MC dropout in Table 4 for the purpose of evaluating their uncertainty estimation. Under the same label amount setting

Dataset	Label Amount	w/o Attention	w/ Attention
Cityscapes	1/16 (186)	83.46	84.05
	1/8 (372)	83.61	83.97
	1/4 (744)	84.32	84.55
	1/2 (1488)	84.60	84.61
VOC (*classic*)	1/16 (92)	86.22	86.87
	1/8 (183)	87.54	87.98
	1/4 (366)	88.57	88.74
	1/2 (732)	89.53	89.69
VOC (*blender*)	1/16 (662)	89.46	89.62
	1/8 (1323)	89.53	89.87
	1/4 (2646)	89.57	89.99
	1/2 (5291)	89.35	89.34

Table 6. Ablation studies of attention aggregation on different datasets. The results are all based on U^{2}PL (Wang et al., 2022b) using ResNet-50, and PAvPU is reported.

for each dataset, NP-SemiSeg achieves a higher PAvPU metric than MC dropout, showing that the former can output more reliable uncertainty estimates. Therefore, it is more suitable than MC dropout for semi-supervised semantic segmentation in terms of uncertainty quantification.

Finally, we compare the running time of NP-SemiSeg and MC dropout for quantifying uncertainty, under two evaluation strategies, namely, the center-crop evaluation on PASCAL VOC 2012 and the sliding evaluation on Cityscapes. Note that the encoder of U^{2}PL in our experiments is a ResNet-50 (He et al., 2016) pretrained on the ImageNet dataset (Deng et al., 2009), and therefore MC dropout is only inserted into the decoder, and only the decoder performs T times of feedforward passes for saving time. From Figure 2, we have the following observations. First, when the number of predictions (T) increases, the time cost of MC dropout also rises accordingly, and the gap between NP-SemiSeg and MC dropout gradually becomes significant. Second, if the sliding evaluation is used, the time consumption of MC dropout is hardly acceptable, as MC dropout requires more numbers of feedforward passes than NP-SemiSeg for this strategy. For instance, to evaluate a large image, we need to move the sliding window for r strides in total, and in this case, MC dropout needs $T \times r$ feedforward passes, while NP-SemiSeg only needs r feedforward passes. These observations demonstrate that NP-SemiSeg is computationally more efficient than MC dropout for semi-supervised semantic segmentation.

4.3. Ablation Studies

We conducted ablation studies of the attention aggregator on two public benchmarks, which are shown in Tables 5 and 6. For the experiments without using the attention aggregator, we followed the previous work (Wang et al., 2022a) to use a mean aggregator for assembling the information instead.

The results show the importance of the attention aggregator.

In particular, when it is removed, we can observe that the mIoU decreases in Table 5. From the perspective of uncertainty quantification, we also see the gap regarding PAvPU between NP-SemiSeg with and without attention aggregator, even though the gap is marginal. These two findings support the significance of the attention aggregator, which involves relevant information from the memory banks to infer the latent maps and the context maps.

5. Conclusion and Outlook

In this work, we proposed a new probabilistic model, named NP-SemiSeg, which adjusts neural processes (NPs) to semi-supervised semantic segmentation. To better utilize the information from context data and target data, we integrated an attention aggregator into NP-SemiSeg for assigning higher weights to important information during aggregation, which is not considered in NP-Match. Our experimental results confirm the effectiveness of NP-SemiSeg in both accuracy and uncertainty estimation, thus highlighting its potential to supplant MC dropout as an innovative method for quantifying uncertainty in semi-supervised semantic segmentation.

For future research, it is valuable to explore NPs in other SSL tasks, such as object detection. In addition, it would also be interesting to see the application of NP-SemiSeg on semi-supervised medical image segmentation in the future.

6. Limitations

While NP-SemiSeg is superior to MC dropout with respect to uncertainty estimation, it is important to acknowledge its performance deterioration in some SSL settings, particularly with the Cityscapes dataset. This could potentially restrict its practical application. We hypothesize two potential causes for this degradation, both of which warrant further investigation.

Firstly, during the training phase, incorrect pixel-wise pseudo-labels may be assigned to unlabeled data. This could negatively affect NP-SemiSeg's ability to approximate the variational distribution to the true distribution over latent variables, leading to a subpar performance. A similar issue in NP-Match is partially resolved through an uncertainty-guided skew-geometric Jensen-Shannon (JS) divergence. However, it is challenging to directly apply this divergence to the task of segmentation.

Secondly, considering that the performance drop is pronounced in the Cityscapes dataset, it might be attributed to the sliding evaluation strategy, which contradicts NP-SemiSeg's use of global latent variables. NP-SemiSeg operates on the premise that a single latent variable is shared among all pixels in an image. This suggests that the global latent vector is dependent on the entire content (topic) of the target image. If a sliding evaluation strategy is employed, we do not obtain a global latent vector for the entire image, but rather a latent vector for the local region covered by the sliding window. This could negatively impact performance, given the importance of global information in generating a global latent vector for a target image.

7. Acknowledgements

This work was partially supported by the Alan Turing Institute under the EPSRC grant EP/N510129/1, by the AXA Research Fund, and by the EU TAILOR grant 952215.

References

Alonso, I., Sabater, A., Ferstl, D., Montesano, L., and Murillo, A. C. Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8219–8228, 2021.

Bruinsma, W. P., Requeima, J., Foong, A. Y., Gordon, J., and Turner, R. E. The Gaussian neural process. *Advances in Approximate Bayesian Inference*, 2021.

Chen, H., Jin, Y., Jin, G., Zhu, C., and Chen, E. Semi-supervised semantic segmentation by improving prediction confidence. *IEEE Transactions on Neural Networks and Learning Systems*, 2021a.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017.

Chen, X., Yuan, Y., Zeng, G., and Wang, J. Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2613–2622, 2021b.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The Cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3213–3223, 2016.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. The PASCAL visual object classes

(VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

French, G., Laine, S., Aila, T., Mackiewicz, M., and Finlayson, G. Semi-supervised semantic segmentation needs strong, varied perturbations. In *British Machine Vision Conference*, 2020.

Garnelo, M., Rosenbaum, D., Maddison, C., Ramalho, T., Saxton, D., Shanahan, M., Teh, Y. W., Rezende, D., and Eslami, S. A. Conditional neural processes. In *International Conference on Machine Learning*, pp. 1704–1713. PMLR, 2018a.

Garnelo, M., Schwarz, J., Rosenbaum, D., Viola, F., Rezende, D. J., Eslami, S., and Teh, Y. W. Neural processes. *arXiv:1807.01622*, 2018b.

Gordon, J., Bruinsma, W. P., Foong, A. Y., Requeima, J., Dubois, Y., and Turner, R. E. Convolutional conditional neural processes. *International Conference on Learning Representations*, 2020.

Guan, D., Huang, J., Xiao, A., and Lu, S. Unbiased subclass regularization for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9968–9978, 2022.

Hariharan, B., Arbeláez, P., Bourdev, L., Maji, S., and Malik, J. Semantic contours from inverse detectors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 991–998. IEEE, 2011.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Hu, H., Wei, F., Hu, H., Ye, Q., Cui, J., and Wang, L. Semi-supervised semantic segmentation via adaptive equalization learning. *Advances in Neural Information Processing Systems*, 34:22106–22118, 2021a.

Hu, Z., Yang, Z., Hu, X., and Nevatia, R. SimPLE: Similar pseudo label exploitation for semi-supervised classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15099–15108, 2021b.

Jha, S., Gong, D., Wang, X., Turner, R. E., and Yao, L. The neural process family: Survey, applications and perspectives. *arXiv:2209.00517*, 2022.

Ke, Z., Wang, D., Yan, Q., Ren, J., and Lau, R. W. Dual student: Breaking the limits of the teacher in semi-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6728–6736, 2019.

Kendall, A. and Gal, Y. What uncertainties do we need in Bayesian deep learning for computer vision? *Advances in Neural Information Processing Systems*, 2017.

Kim, H., Mnih, A., Schwarz, J., Garnelo, M., Eslami, A., Rosenbaum, D., Vinyals, O., and Teh, Y. W. Attentive neural processes. *International Conference on Learning Representations*, 2019.

Kwon, D. and Kwak, S. Semi-supervised semantic segmentation with error localization network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9957–9967, 2022.

Lee, J., Lee, Y., Kim, J., Yang, E., Hwang, S. J., and Teh, Y. W. Bootstrapping neural processes. *Advances in Neural Information Processing Systems*, pp. 6606–6615, 2020.

Li, J., Xiong, C., and Hoi, S. C. CoMatch: Semi-supervised learning with contrastive graph regularization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9475–9484, 2021.

Li, Y., Chen, J., Xie, X., Ma, K., and Zheng, Y. Self-loop uncertainty: A novel pseudo-label for semi-supervised medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 614–623. Springer, 2020.

Liu, Y., Tian, Y., Chen, Y., Liu, F., Belagiannis, V., and Carneiro, G. Perturbed and strict mean teachers for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4258–4267, 2022.

Louizos, C., Shi, X., Schutte, K., and Welling, M. The functional neural process. *Advances in Neural Information Processing Systems*, 2019.

Meyer, A., Ghosh, S., Schindele, D., Schostak, M., Stober, S., Hansen, C., and Rak, M. Uncertainty-aware temporal self-learning (UATS): Semi-supervised learning for segmentation of prostate zones and beyond. *Artificial Intelligence in Medicine*, 116:102073, 2021.

Mukhoti, J. and Gal, Y. Evaluating bayesian deep learning methods for semantic segmentation. *arXiv:1811.12709*, 2018.

Nassar, I., Herath, S., Abbasnejad, E., Buntine, W., and Haffari, G. All labels are not created equal: Enhancing semi-supervision via label grouping and co-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7241–7250, 2021.

Øksendal, B. Stochastic differential equations. In *Stochastic Differential Equations*, pp. 65–84. Springer, 2003.

Ouali, Y., Hudelot, C., and Tami, M. Semi-supervised semantic segmentation with cross-consistency training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12674–12684, 2020.

Pham, H., Dai, Z., Xie, Q., and Le, Q. V. Meta pseudo labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11557–11568, 2021.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian processes for machine learning*. MIT Press, 2006.

Rizve, M. N., Duarte, K., Rawat, Y. S., and Shah, M. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *International Conference on Learning Representations*, 2021.

Sedai, S., Antony, B., Rai, R., Jones, K., Ishikawa, H., Schuman, J., Gadi, W., and Garnavi, R. Uncertainty guided semi-supervised segmentation of retinal layers in OCT images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 282–290. Springer, 2019.

Shi, Y., Zhang, J., Ling, T., Lu, J., Zheng, Y., Yu, Q., Qi, L., and Gao, Y. Inconsistency-aware uncertainty estimation for semi-supervised medical image segmentation. *IEEE Transactions on Medical Imaging*, 2021.

Sohn, K., Berthelot, D., Li, C.-L., Zhang, Z., Carlini, N., Cubuk, E. D., Kurakin, A., Zhang, H., and Raffel, C. FixMatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems*, 2020.

Tarvainen, A. and Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in Neural Information Processing Systems*, 30, 2017.

Wang, J. and Lukasiewicz, T. Rethinking bayesian deep learning methods for semi-supervised volumetric medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 182–190, 2022.

Wang, J., Lukasiewicz, T., Massiceti, D., Hu, X., Pavlovic, V., and Neophytou, A. NP-Match: When neural processes meet semi-supervised learning. In *International Conference on Machine Learning*, pp. 22919–22934. PMLR, 2022a.

Wang, K., Zhan, B., Zu, C., Wu, X., Zhou, J., Zhou, L., and Wang, Y. Tripled-uncertainty guided mean teacher model for semi-supervised medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 450–460. Springer, 2021.

Wang, Y., Wang, H., Shen, Y., Fei, J., Li, W., Jin, G., Wu, L., Zhao, R., and Le, X. Semi-supervised semantic segmentation using unreliable pseudo-labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4248–4257, 2022b.

Xiang, J., Qiu, P., and Yang, Y. FUSSNet: Fusing two sources of uncertainty for semi-supervised medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 481–491. Springer, 2022.

Yang, L., Zhuo, W., Qi, L., Shi, Y., and Gao, Y. ST++: Make self-training work better for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4268–4277, 2022.

Yu, L., Wang, S., Li, X., Fu, C.-W., and Heng, P.-A. Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 605–613. Springer, 2019.

Zhang, B., Wang, Y., Hou, W., Wu, H., Wang, J., Okumura, M., and Shinozaki, T. FlexMatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34, 2021.

Zhao, Z., Yang, L., Long, S., Pi, J., Zhou, L., and Wang, J. Augmentation matters: A simple-yet-effective approach to semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Zhong, Y., Yuan, B., Wu, H., Yuan, Z., Peng, J., and Wang, Y.-X. Pixel contrastive-consistent semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7273–7282, 2021.

Zhou, Y., Xu, H., Zhang, W., Gao, B., and Heng, P.-A. C3-SemiSeg: Contrastive semi-supervised segmentation via cross-set learning and dynamic class-balancing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7036–7045, 2021.

Zou, Y., Zhang, Z., Zhang, H., Li, C.-L., Bian, X., Huang, J.-B., and Pfister, T. PseudoSeg: Designing pseudo labels for semantic segmentation. *International Conference on Learning Representations*, 2021.

Appendix

A. Implementation Details

NP-SemiSeg is a flexible module, and in our experiments, we evaluated it with four different segmentation frameworks, including MT (Tarvainen & Valpola, 2017), PS-MT (Liu et al., 2022), U^{2}PL (Wang et al., 2022b), and AugSeg (Zhao et al., 2023). When NP-SemiSeg is incorporated into them, we followed their original hyper-parameter settings for fair comparisons, and we only made the following changes due to limited computational resources. On the PASCAL VOC 2012 dataset, the training crop size is set to 480×480, and those frameworks with NP-SemiSeg are trained with 0.001 learning rate and 12 batch size. On the Cityscapes dataset, the training crop size is set to 580×580, and we used 0.005 learning rate and 8 batch size for training. When calculating PAvPU, we use a window size 64, and the uncertainty threshold is set to 0.4. The encoder is ResNet-50 (He et al., 2016) that is pre-trained on ImageNet (Deng et al., 2009).

The hyper-parameters of NP-SemiSeg include the length of each memory bank ($\mathcal{Q}$), the coefficient λ_{kl}, the number of latent maps T. We followed NP-Match to set $\mathcal{Q} = 2560$ for all memory banks. T was set to 5 at both the training phase and the testing phase. The coefficient λ_{kl} is set to 0.005. The configuration of the small ConvNet and the decoder are separately shown in Tables 7 and 8. The implementation of NP-SemiSeg is modified based on the public official source code of NP-Match (Wang et al., 2022a). All experiments are conducted on GeForce RTX 3090 GPUs.

Type	Configuration
2D Conv	# In-C: 512, # Out-C: 32, Kernel Size: 1×1, Stride: 1×1, Padding: 0
InstanceNorm	# In-C: 32, # Out-C: 32
ReLU	# In-C: 32, # Out-C: 32
2D Conv	# In-C: 32, # Out-C: 32, Kernel Size: 1×1, Stride: 1×1, Padding: 0
InstanceNorm	# In-C: 32, # Out-C: 32
ReLU	# In-C: 32, # Out-C: 32
2D Conv	# In-C: 32, # Out-C: 32, Kernel Size: 1×1, Stride: 1×1, Padding: 0

Table 7. Configuration of the small ConvNet. It is used for dimensional reduction, in order to save GPU memory. "In-C" and "Out-C" denote the channel dimension of the input feature maps and the output feature maps, respectively.

Type	Configuration
2D Conv	# In-C: 576, # Out-C: 256, Kernel Size: 3×3, Stride: 1×1, Padding: 1×1
InstanceNorm	# In-C: 256, # Out-C: 256
ReLU	# In-C: 256, # Out-C: 256
2D Conv	# In-C: 256, # Out-C: 256, Kernel Size: 3×3, Stride: 1×1, Padding: 1×1
InstanceNorm	# In-C: 256, # Out-C: 256
ReLU	# In-C: 256, # Out-C: 256
2D Conv	# In-C: 256, # Out-C: n_{class}, Kernel Size: 1×1, Stride: 1×1, Padding: 0

Table 8. Configuration of the decoder. "In-C" and "Out-C" denote the channel dimension of the input feature maps and the output feature maps, respectively. "n_{class}" represents the number of classes.

B. Hyper-parameter Exploration

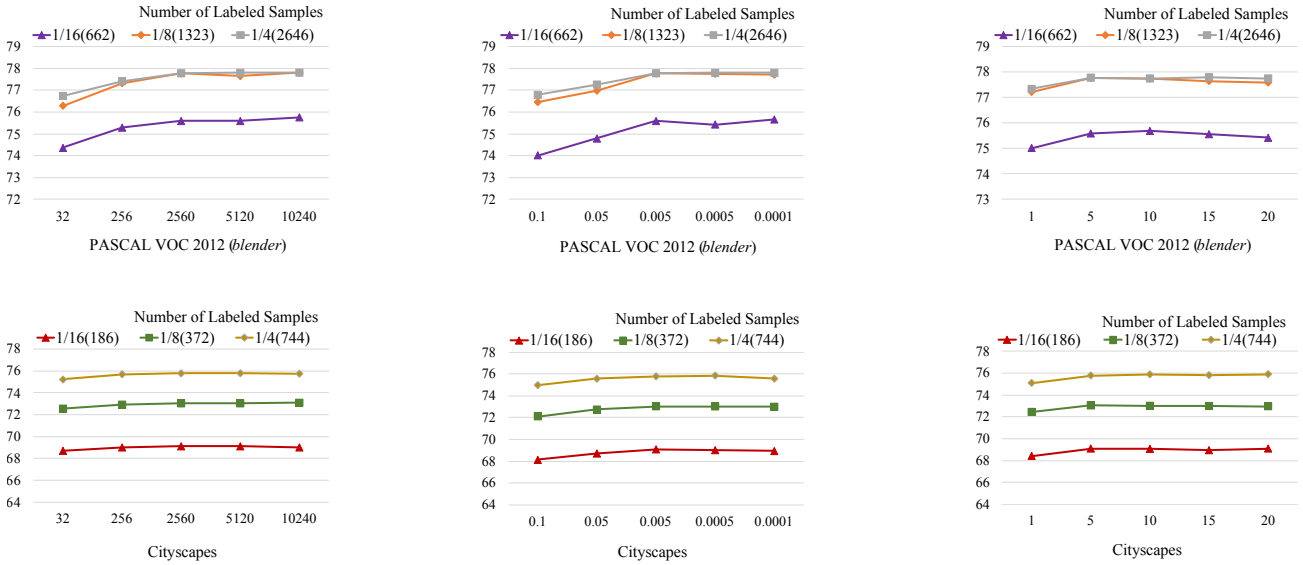


Figure 3. The mean IoU under different hyper-parameter settings for training.

Additional experiments are conducted on PASCAL VOC 2012 (*blender*) and Cityscapes with different amounts of labeled data for hyper-parameters exploration. Three hyper-parameters are investigated in total, including the length of each memory bank ($\mathcal{Q}$), the coefficient λ_{kl}, and the number of latent maps T. By Figure 3(a), $\mathcal{Q}$ should be set properly, as a small value leads to an inferior performance on both datasets. Once $\mathcal{Q}$ is large enough, further increasing the length will not affect performance. Figure 3(b) shows the results using different λ_{kl}. It can be seen that when λ_{kl} rises from 0.005 to 0.1, the performance of NP-SemiSeg gets worse. Conversely, decreasing λ_{kl} cannot impact the performance too much. Finally, Figure 3(c) shows the relationship between the number of latents and the performance. We can observe that the performance is insensitive to the setting of T, unless it is set to 1. Therefore, in our other experiments, it is a good practice to set $\lambda_{kl} = 0.005$, $\mathcal{Q} = 2560$, and $T = 5$.

C. Derivation of ELBO (Eq. (4))

Proof. As for the marginal joint distribution $p(y_{1:n}|x_{1:n})$ over n data points in which there are m context data points and r target data points (i.e., $m+r=n$), we assume a variational distribution over latent variables for the target data points, namely, $q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})$. According to the i.i.d assumption, those z_* are independent from each other, and we denote its integral domain as D_z. Then:

$$
\begin{aligned}
&log\, p(y_{1:n}|x_{1:n}) = log \int \cdots \int_{D_z} p(z_{m+1:\,m+r}, y_{1:n}|x_{1:n}) \\
&= log \int \cdots \int_{D_z} \frac{p(z_{m+1:\,m+r}, y_{1:n}|x_{1:n})}{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})} q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r}) \\
&\geq \sum_{i=m+1}^{m+r} \mathbb{E}_{q(z_i|x_{m+1:\,m+r}, y_{m+1:\,m+r})}[log\, \frac{p(z_i, y_{1:n}|x_{1:n})}{q(z_i|x_{m+1:\,m+r}, y_{m+1:\,m+r})}] \\
&= \mathbb{E}_{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})}[log\, \frac{p(y_{1:m}|x_{1:m}) p(z_{m+1:\,m+r}|x_{1:m}, y_{1:m}) \prod_{i=m+1}^{m+r} p(y_i|z_i, x_i)}{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})}] \\
&= \mathbb{E}_{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})}[\sum_{i=m+1}^{m+r} log\, p(y_i|z_i, x_i) + log\, \frac{p(z_{m+1:\,m+r}|x_{1:m}, y_{1:m})}{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})} + log\, p(y_{1:m}|x_{1:m})] \\
&= \mathbb{E}_{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})}[\sum_{i=m+1}^{m+r} log\, p(y_i|z_i, x_i) - log\, \frac{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})}{p(z_{m+1:\,m+r}|x_{1:m}, y_{1:m})}] + log\, p(y_{1:m}|x_{1:m}).
\end{aligned}
\tag{8}
$$

Similar to NPs (Garnelo et al., 2018b), $p(z_{m+1:\,m+r}|x_{1:m}, y_{1:m})$ is unknown, we replace it with $q(z_{m+1:\,m+r}|x_{1:m}, y_{1:m})$, and then we get:

$$
\begin{aligned}
&log\, p(y_{1:n}|x_{1:n}) \geq \\
&\mathbb{E}_{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})}\Big[\sum_{i=m+1}^{m+r} log\, p(y_i|z_i, x_i) - log\, \frac{q(z_{m+1:\,m+r}|x_{m+1:\,m+r}, y_{m+1:\,m+r})}{q(z_{m+1:\,m+r}|x_{1:m}, y_{1:m})}\Big] + log\, p(y_{1:m}|x_{1:m}).
\end{aligned}
\tag{9}
$$

□

D. Visualization Results

We visualize some prediction results and uncertainty maps given by NP-SemiSeg on both PASCAL VOC 2012 (*blender*) and Cityscapes. For the uncertainty maps, we calculate pixel-wise predictive entropy, and represent the uncertainty with gray images. Each uncertainty map uses pixel values, ranging from black to white, to denote the levels of uncertainty, starting from low to high.

According to the visualization results, NP-SemiSeg can provide a good quality of uncertainty estimates. In general, it can give a high uncertainty for the pixels that are wrongly predicted. Furthermore, the boundary of an object is more likely to be misclassfied, and therefore, NP-SemiSeg also gives high uncertainties to boundaries. Based on this information, one can make decisions or further improve the results in a real-world scenario.

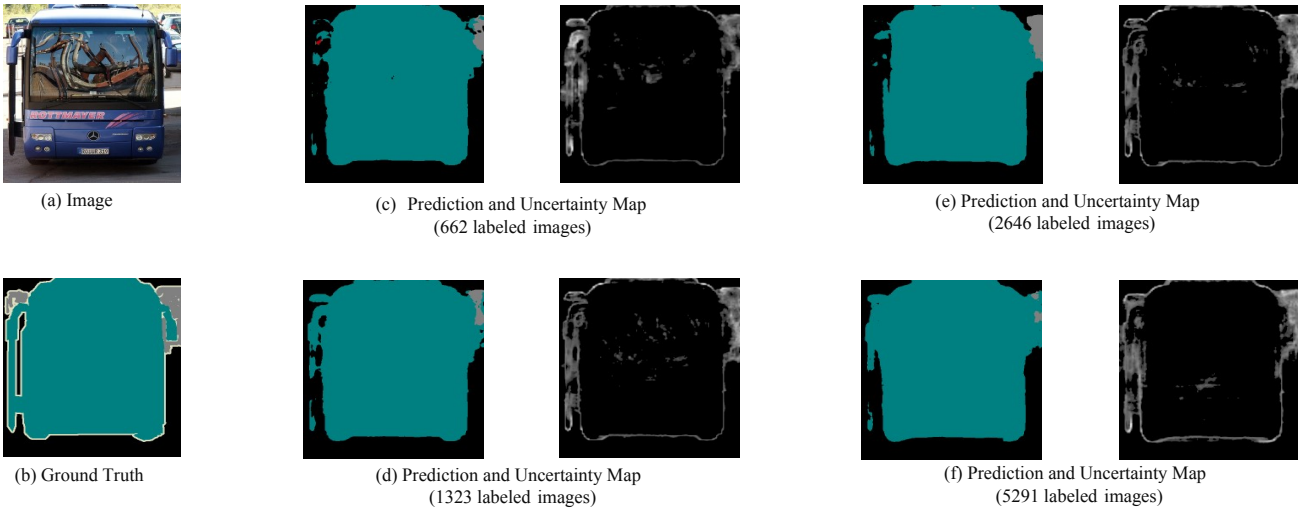


Figure 4. First set of visualization results on PASCAL VOC 2012 (*blender*) under different training protocols. The predictions and their corresponding uncertainty maps are shown.

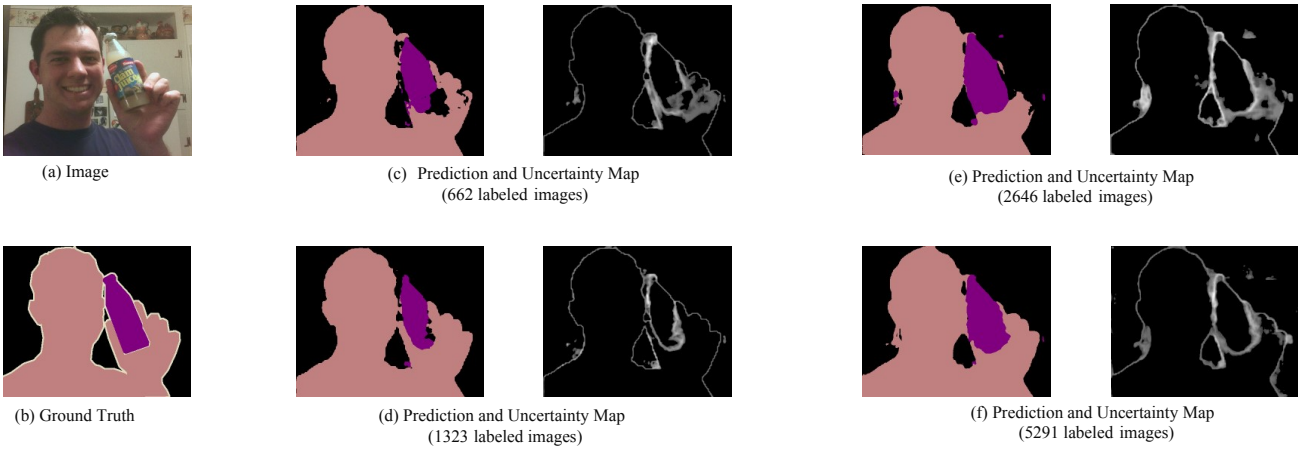


Figure 5. Second set of visualization results on PASCAL VOC 2012 (*blender*) under different training protocols. The predictions and their corresponding uncertainty maps are shown.

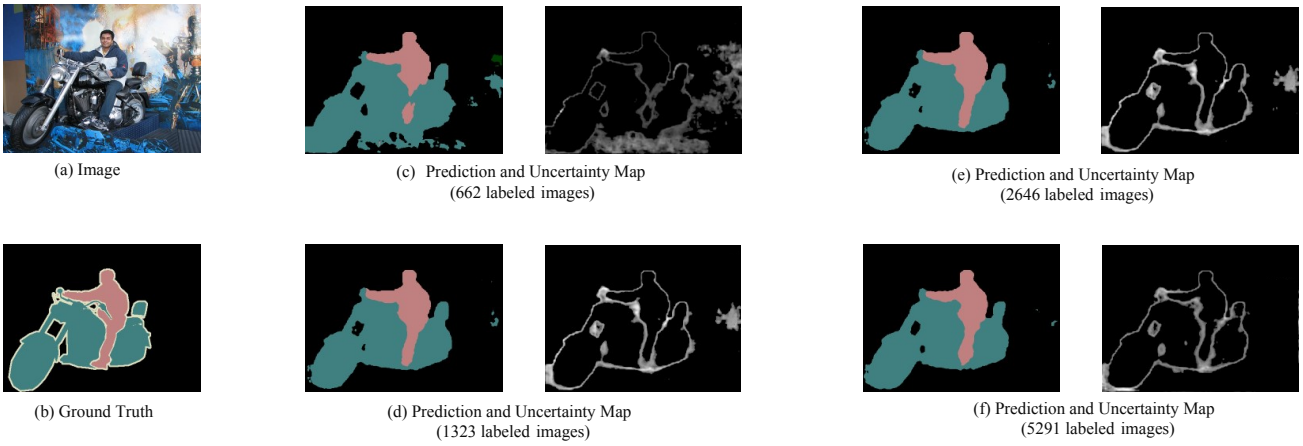
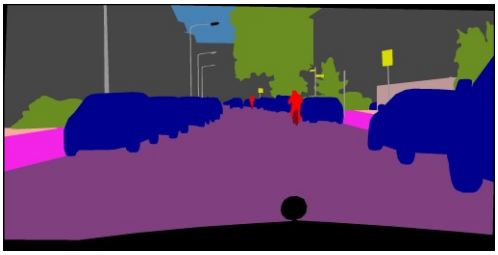
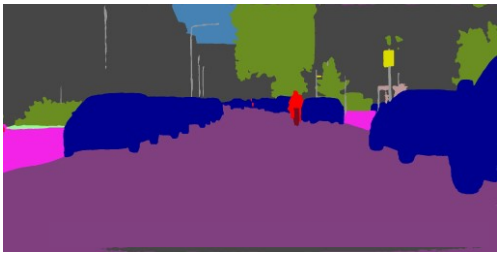


Figure 6. Third set of visualization results on PASCAL VOC 2012 (*blender*) under different training protocols. The predictions and their corresponding uncertainty maps are shown.

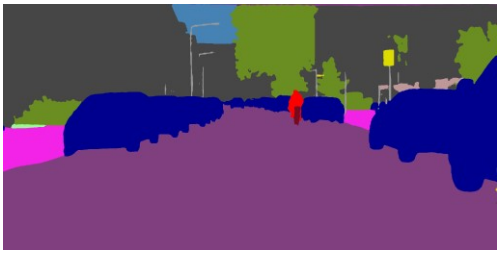
(a) Image



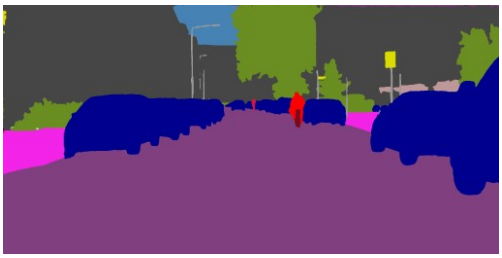
(b) Ground Truth



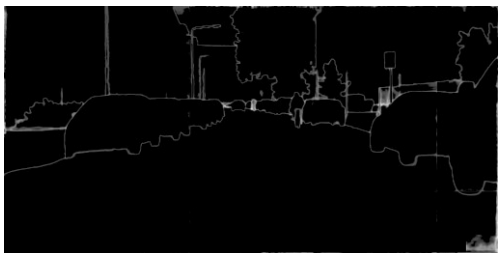
(c) Uncertainty Map and Prediction (186 labeled images)

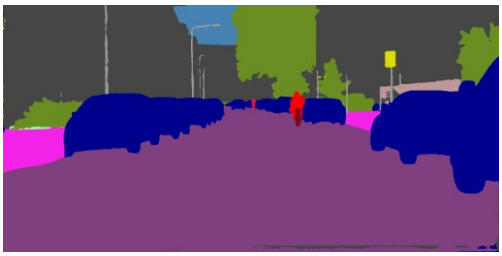


(d) Uncertainty Map and Prediction (372 labeled images)



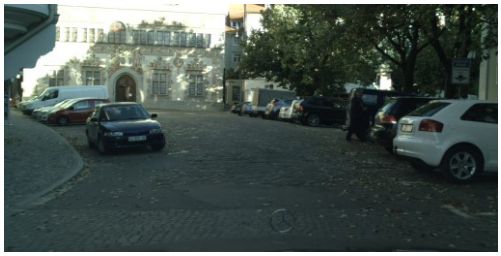
(e) Uncertainty Map and Prediction (744 labeled images)





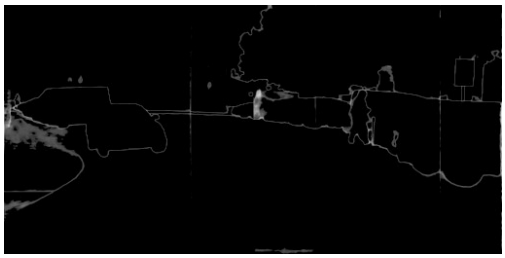
(f) Uncertainty Map and Prediction (1488 labeled images)

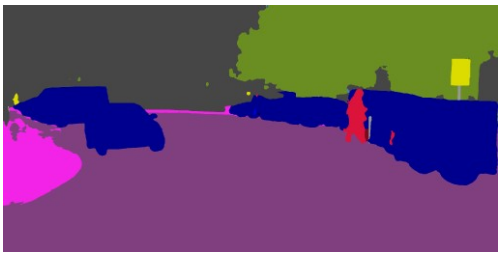
Figure 7. First set of visualization results on Cityscapes under different training protocols. The predictions and their corresponding uncertainty maps are shown.



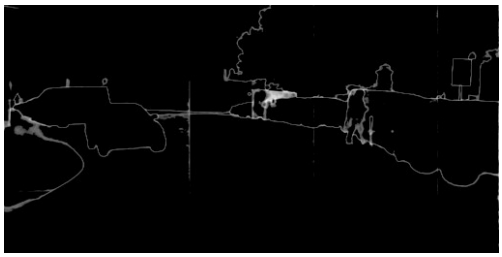
(a) Image

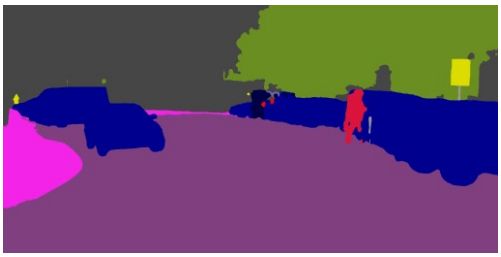
(b) Ground Truth



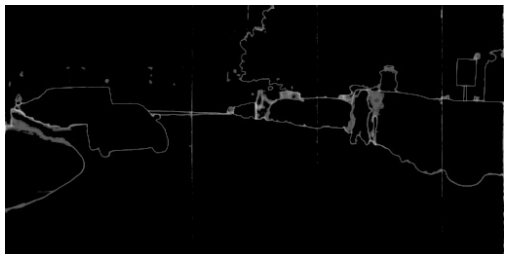


(c) Uncertainty Map and Prediction (186 labeled images)





(d) Uncertainty Map and Prediction (372 labeled images)

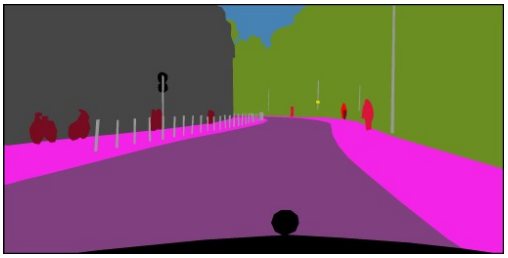


(e) Uncertainty Map and Prediction (744 labeled images)

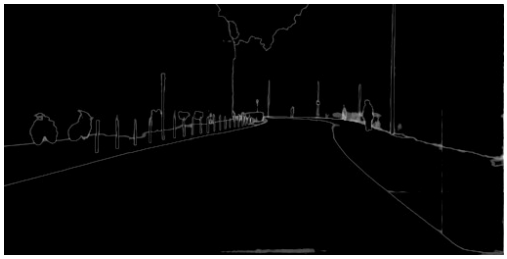
(f) Uncertainty Map and Prediction (1488 labeled images)

Figure 8. Second set of visualization results on Cityscapes under different training protocols. The predictions and their corresponding uncertainty maps are shown.

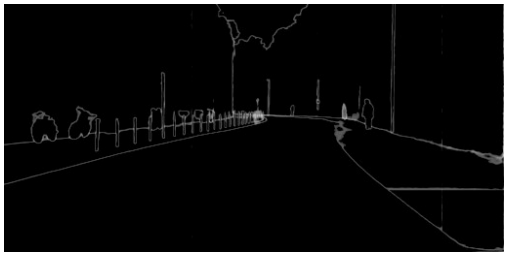
(a) Image



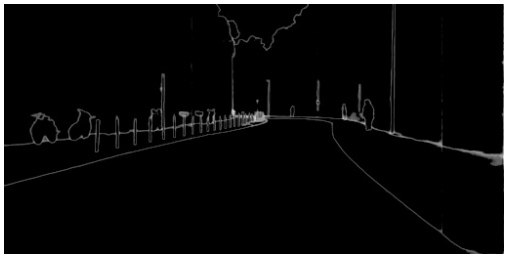
(b) Ground Truth

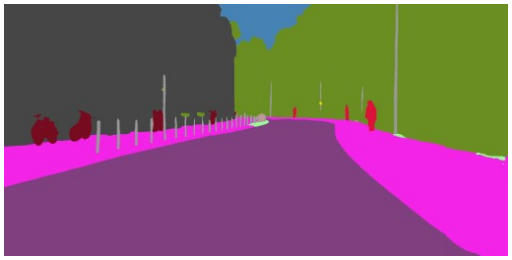


(c) Uncertainty Map and Prediction (186 labeled images)

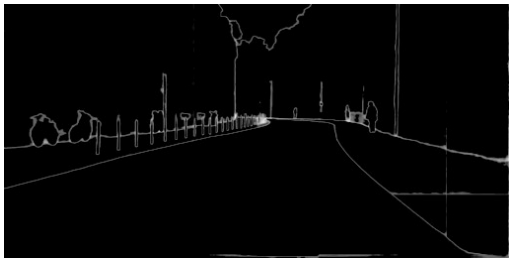


(d) Uncertainty Map and Prediction (372 labeled images)





(e) Uncertainty Map and Prediction (744 labeled images)



(f) Uncertainty Map and Prediction (1488 labeled images)

Figure 9. Third set of visualization results on Cityscapes under different training protocols. The predictions and their corresponding uncertainty maps are shown.