P^2SAM: Probabilistically Prompted SAMs Are Efficient Segmentator for Ambiguous Medical Images (Supplementary Material)

Anonymous Authors

The following contents are provided in the supplements:

- More experimental details (Sec. 4.2 in main paper).
- Details of ablation study (Sec. 4.6 in main paper).
- Details of network architecture of P^2SAM.
- More visualization about our experiments. (Sec. 4.4 and Sec. 4.5 in the main paper).

1 DETAILED EXPERIMENTAL SETUP

Throughout both stages of P^2SAM training, the ℓ_{seg} loss function was employed, which amalgamates binary cross entropy loss (bce loss), focal loss, and dice loss. During the LoRA fine-tuning of the image encoder Enc_I, the low rank was set at 4. Additionally, the Adam optimizer was utilized to fine-tune both the mask decoder Dec_M and prompt encoder Enc_P. The AdamW optimizer was also deployed for the diversity-aware assembling module, with both learning rates established at 10^{-3}. The experiments' initial and subsequent stages were executed on a computer equipped with an NVIDIA 4090 GPU. Notably, the first stage of training spanned approximately 1 hour, whereas the second stage was condensed to around 10 minutes, during which 100 training epochs were conducted on the entire LIDC dataset.

2 MORE DETAILS ABOUT ABLATION STUDY

In our study, we confined the ablation experiments to the LIDC dataset and limited the evaluation tasks to a subset of 500 samples from the test set. We established a baseline with the SAM model that underwent dataset-specific fine-tuning. This fine-tuning process was completed during the initial stage of training, encompassing only 50 epochs of adjustment. As observed from Table A1, while the vanilla adapted SAM exhibits performance analogous to other models with respect to D_{max} metrics, it demonstrates significant disparities in terms of GED and HM-IoU. This suggests that despite the ablility of fine-tuned SAM to sustain a certain degree of segmentation precision, the diversity of the samples it generates is markedly deficient.

To rectify this shortcoming, we incorporated the probabilistic prompt module into the fine-tuned SAM model. Experimental outcomes illustrate that by sampling the probability distribution of input prompts, we attain segmentation outcomes that display increased diversity (i.e., enhanced performance on GED and HM-IoU metrics), while simultaneously boosting the segmentation benchmark (D_{max}).

Furthermore, we integrated the diversity assembly module into the fine-tuned SAM model. From the experimental results, it is evident that though there is no substantial enhancement in GED in comparison to the baseline model, the surge in the D_{max} index is quite considerable. This substantiates the efficacy of the module in amalgamating multiple segmentation outputs, and the resultant samples can fit the labels more accurately, thereby enhancing the overall performance of models.

Table A1: Ablation study of the key strategies of the proposed P^{2}SAM on LIDC-IDRI dataset.

Method	GED(↓)	D_{max}(↑)	HM-IoU (↑)
Vanilla Adapted SAM	0.381	0.705	0.359
SAM + Probabilistic Prompt	<u>0.340</u>	0.803	<u>0.454</u>
SAM + Diversity Assembling	0.376	<u>0.853</u>	0.402
P^{2}SAM (Full Model)	**0.208**	**0.919**	**0.627**

3 NETWORK ARCHITECTURE

3.1 Prompted SAM

In the experiment, we adopted the Vit-b version of the SAM model and accommodated Enc_I by reducing the size of the output feature map F_I by $\frac{1}{8}$ compared with the original. This change is expected to reduce the required memory usage during the training process and accelerate the inference speed of the model. In addition, we adjusted the SAM model to multi output mode with 8 outputs, and set the pixel mean and pixel std parameters to 0 and 1, respectively.

3.2 Prompt Generation Network (PGN)

The network mainly consists of two parts. (1) Encoder: This part contains 4 convolutional blocks, each with 3 convolutional layers inside. These 4 convolutional blocks have channel numbers of 32, 64, 128 and 192, respectively, to gradually extract and deepen features. (2) Axis Gaussian Generation Network: This network consists of a 1x1 convolutional layer with 256 channels and an axial Gaussian distribution generator. This design first increases the dimensionality of the feature map output by the Encoder through a 1x1 convolutional layer to obtain 256 dimensional μ and σ, and then these two parameters are fed into a Gaussian generator to generate the distribution of $\tilde{T_P}$.

3.3 Diversity-aware Assembling Module

In our experimental design, we set the number of mask weights $\mathcal{W}$ to 8 and initialize each weight to $\frac{1}{8}$. This setting aims to correspond to 8 outputs of SAM model. In the first stage of the experiment, these weight $\mathcal{W}$ will be trained to meet the model requirements. In the second stage, we will freeze these weights. This is to enable the prompt generation network to generate more diverse and representative segmentation results, thereby effectively guiding the modeling of $\tilde{T_P}$.

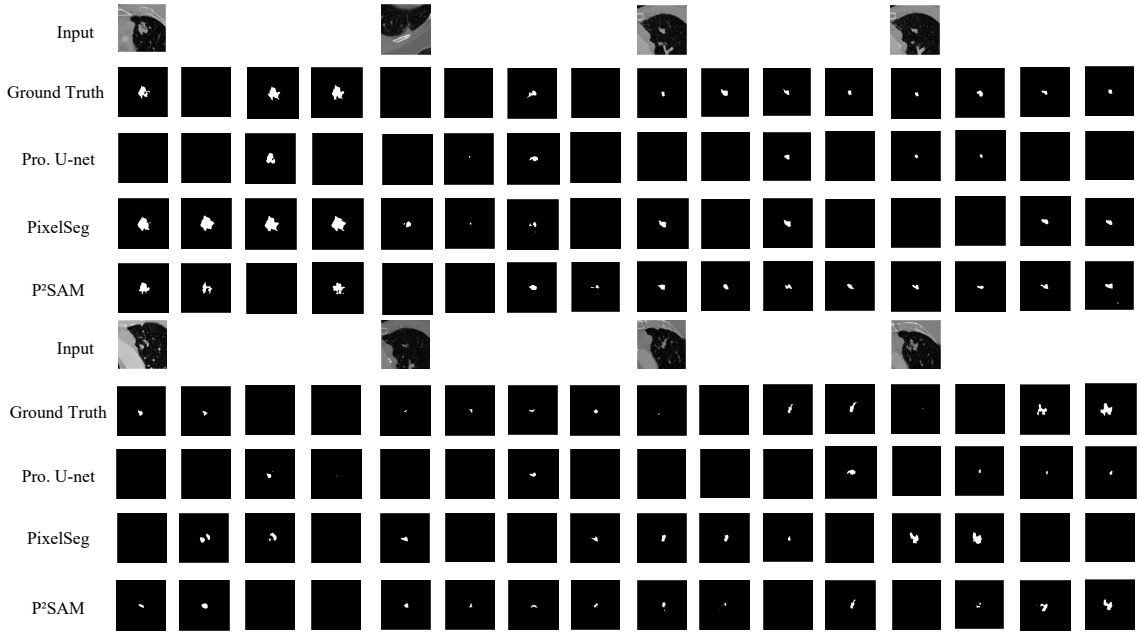

Figure A1: More visualization on the LIDC dataset, displaying only the first 4 samples.

4 MORE VISUALIZATION ABOUT EXPERIMENTS

As demonstrated in Fig. A1 and Fig. A2, these illustrations provide an extensive visualization of our research outcomes. These figures meticulously depict various aspects of our data, aiding readers in gaining a profound understanding of our research findings.

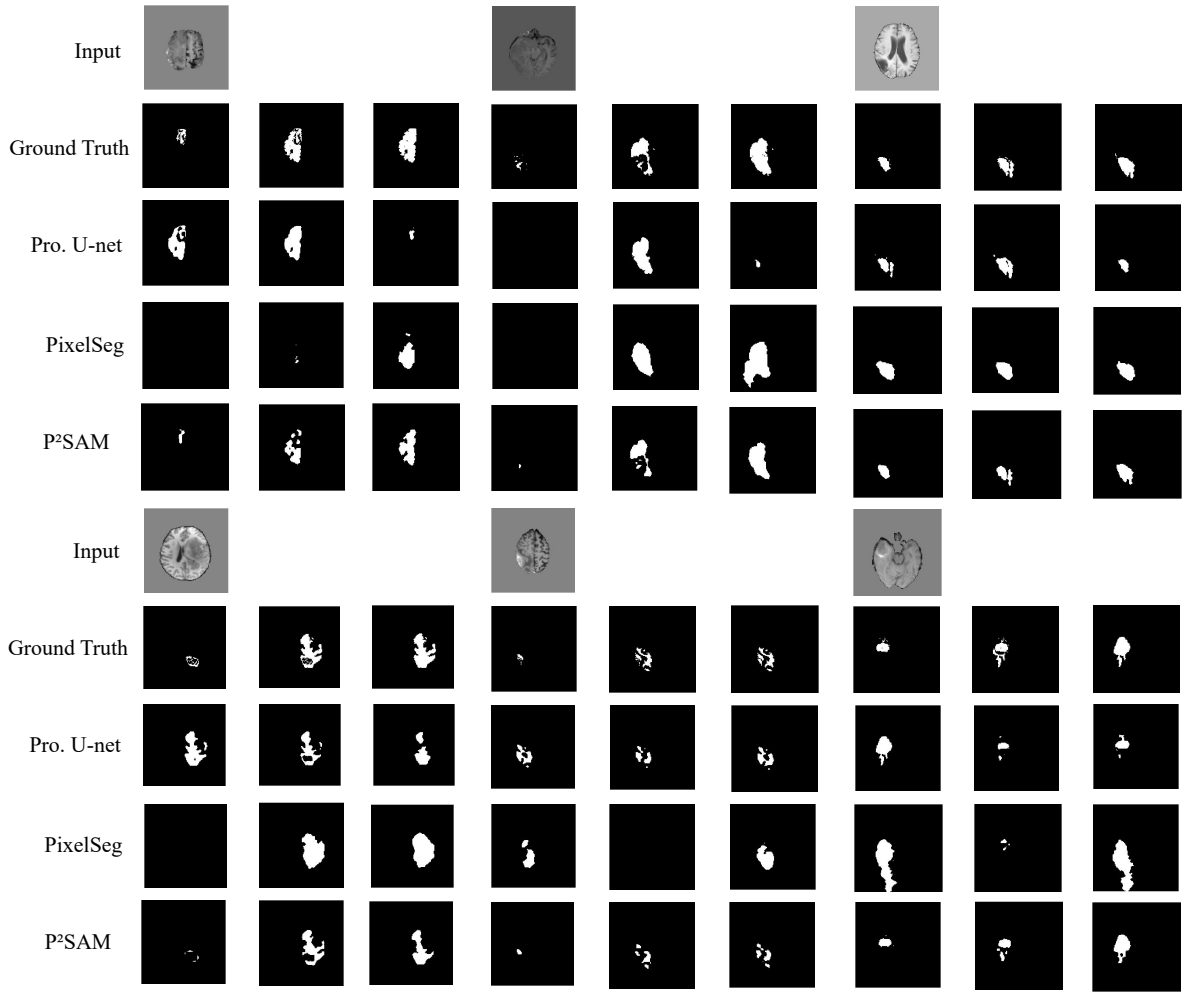

Figure A2: More visualization on the BraTS2017 dataset, displaying only the first 4 samples.