Orthogonal Annotation Benefits Barely-supervised Medical Image Segmentation — Supplementary Material —

A. Algorithm Summary

The training procedure of our method is summarized in Algorithm 1.

Algorithm 1: Training Procedure with Sparse Orthogonal Annotation on Tiny Fraction of Volumes

For * steps: repeat at once by exchanging a and b.
Input: Training images $\{X_i | i \leq N\}$
Orthogonal annotations $\{Y_i : (Y_{ia}^{mi}, Y_{ib}^{ni}) | i \leq l\}$
Output: Segmentation models $\mathcal{M}_{seg \cdot a}, \mathcal{M}_{seg \cdot b}$

```
//pseudo label generation
```
for $i \in [1, l]$ **do**
  *pseudo label $\hat{Y}_{ia} \leftarrow \mathcal{M}_{reg}(X_i, Y_{ia}^{mi})$
  *pseudo label $\tilde{Y}_{ia} \leftarrow$ LabelMix$(\hat{Y}_{ia}, Y_i)$
end
```
//initialization
```
Cross-supervision weight $\lambda \leftarrow 0$
Decay rate $\alpha \leftarrow 0.95$
Generating weight map W_a and W_b with α
```
//model training
```
while *not converged* **do**
  foreach X_i *in minibatch* **do**
    *$P_{ia}, \bar{P}_{ia}, M_{un \cdot a} \leftarrow \mathcal{M}_{seg \cdot a}(X_i)$
    if $i \leq l$ **then**
      *$\mathcal{L}_{sup \cdot a} \leftarrow \mathcal{L}_{sup}(P_{ia}, \tilde{Y}_{ia}, W_a)$
    else
      *$\mathcal{L}_{cross \cdot a} \leftarrow \mathcal{L}_{cross}(P_{ia}, \bar{P}_{ib}, M_{un \cdot b})$
    end
  end
  *$\mathcal{L}_a \leftarrow (1 - \lambda)\mathcal{L}_{sup \cdot a} + \lambda \mathcal{L}_{cross \cdot a}$
  ```
  //dense to sparse
  ```
  Update decay rate α, weight map W_a, W_b
  Update cross-supervision weight λ
  Update $\mathcal{M}_{seg \cdot a}, \mathcal{M}_{seg \cdot b}$
end

B. Detailed Settings of Experiments in Sec. 3.2

This section provides detailed settings of three experiments in Sec. 3.2 in order.

Setting 1. The V-Net [5] model is trained with 100 fully annotated volumes. The features are extracted from the layer before classification layer. The parallel slices are from transverse plane, the orthogonal slices are from transverse plane and coronal plane. As the input volumes are cropped into $112 \times 112 \times 80$, the serial numbers for slices in transverse plane are randomly sampled from [1,80] and the serial numbers for slices in coronal plane are randomly sampled from [1,112], respectively. The results reported in Figure 3 of the main paper are the average HSIC [4] value of the slices selected from 100 volumes.

Setting 2. We randomly sample three slice serial number s_{t_1}, s_{t_2} and s_c. And the labeled slices to train models t_1, t_2 and c are the ${s_{t_1}}^{th}$ slice from transverse plane, ${s_{t_2}}^{th}$ slice from transverse plane and ${s_c}^{th}$ slice from coronal plane, respectively. The illustration in Figure 4 of the main paper is generated when $s_{t_1} = 43$, $s_{t_2} = 53$ and $s_c = 70$. And as long as the slices selected from transverse planes do not separate too far, the property illustrated in Figure 4 still holds.

Setting 3. The models in this part are trained with 8 volumes where only two slices are labeled in each volume. The serial numbers for slices in transverse plane are randomly sampled from [1,88], and the serial numbers for slices in coronal plane are randomly sampled from [1,132].

C. Monitoring of the Dice Coefficient on KiTS19 Dataset

To better illustrate 1) the effectiveness of our method in achieving stable and continuous improvement through the whole training process and 2) the difficulty of directly learning from sparse annotations, we save the intermediate models trained in Sec. 4.3 (MT [7], CPS [1], CTBCT [3], CoraNet [6] trained in **Sparse** setting and Ours) every 100 iterations and test their performance. The experiment setting has been introduced in Sec. 4.2. We use KiTS19 [2] dataset as an example and the result is shown in Figure 1.

As shown in Figure 1, our method achieves stable and continuous improvement, while other methods directly learning from sparse annotations suffer extremely unstable training and performance degradation in later stage. Also, their peak performances are inferior to ours.

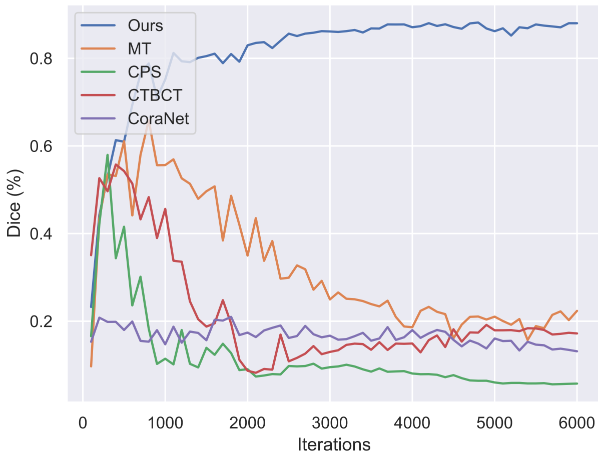

Figure 1. The performance comparison between our method and other methods (MT [7], CPS [1], CTBCT [3], CoraNet [6]) trained in Sparse setting on KiTS19 [2] dataset.

References

[1] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2613–2622, 2021. 1, 2

[2] Nicholas Heller, Niranjan Sathianathen, Arveen Kalapara, Edward Walczak, Keenan Moore, Heather Kaluzniak, Joel Rosenberg, Paul Blake, Zachary Rengel, Makinna Oestreich, et al. The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes. *arXiv preprint arXiv:1904.00445*, 2019. 1, 2

[3] Xiangde Luo, Minhao Hu, Tao Song, Guotai Wang, and Shaoting Zhang. Semi-supervised medical image segmentation via cross teaching between cnn and transformer. In *Medical Imaging with Deep Learning*, 2022. 1, 2

[4] Wan-Duo Kurt Ma, JP Lewis, and W Bastiaan Kleijn. The HSIC bottleneck: Deep learning without back-propagation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5085–5092, 2020. 1

[5] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *4th International Conference on 3D Vision*, pages 565–571. IEEE, 2016. 1

[6] Yinghuan Shi, Jian Zhang, Tong Ling, Jiwen Lu, Yefeng Zheng, Qian Yu, Lei Qi, and Yang Gao. Inconsistency-aware uncertainty estimation for semi-supervised medical image segmentation. *IEEE Transactions on Medical Imaging*, 41(3):608–620, 2022. 1, 2

[7] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in Neural Information Processing Systems*, 30, 2017. 1, 2