Improved nnUNet for Pulmonary Airway Segmentation from 3D Chest CT

Yingao Liu[1,2], Hongrong Wei[2] and Na Wang[2]

[1] University of Science and Technology of China, China
[2] Sensetime, China
sa20048051@mail.ustc.edu.cn

Abstract. Accurate segmentation of pulmonary airway (PA) is important for chronic obstructive pulmonary disease diagnosis and bronchoscopic-assisted surgery navigation. Due to the fine-grained pulmonary airway structure, yet still challenging in extracting small peripheral bronchi and suffer the risk of airway leakage for the SOTA framework nnUNet. In this paper, we propose an improved nnUNet framework, which incorporates more data augmentation, new voxel sampling strategy and our proposed loss function, to improve the segmentation performance of remote PA branches and focus on hard samples mining. The validation results on ATM2022 dataset demonstrate that the improved nnUNet can achieve more refine segmentation for the remote branches and is promising for clinically used.

Keywords: Pulmonary Airway Segmentation, nnUNet, Hard Samples Mining.

1 nnUNet baseline

In this section, we first describe our baseline frameworks nnUNet[1].

Network backbone. Since 3D convolution network had been demonstrated to be more powerful for the pulmonary airway segmentation from 3d chest CT. Therefore, in this paper, we use 3D version of nnUNet as the network backbone. It closely follows the original U-Net. The network depth is determined by performing max-pooling until the feature map is reduced to 4 voxels. In our work, the network consists of seven stages. Batch normalization is applied immediately after each convolutional layer except for the last one to better converge the network. The channel number of the first block is set to 32. Every time a feature map passes through the max pooling layer in a dimension with a stride of 2, the size of the feature map in that dimension is halved. In the first four stages, the channel number is doubled after the feature map passes each layer, while in the last three stages the channel number is fixed to 320, which is the default maximum number of channels in nnUNet. All the feature maps in the encoder part are concatenated with their counterparts in the decoder part by skip connections. The encoder and decoder parts are symmetric.

Implementation Details. A combination of cross-entropy loss and Dice loss is applied as the loss function, which is defined as:

$$L_{total} = L_{CE} + L_{Dice}$$

With $L_{CE}(y,\hat{y}) = -\frac{1}{N}\sum_i y\log(\hat{y}) + (1-y)\log(1-\hat{y})$, $L_{Dice}(y,\hat{y}) = -\frac{2y\hat{y}}{y+\hat{y}}$

where $\hat{y}$ refers to the predicted value and y to the ground truth label. Augmentation techniques, including random rotations, random scaling, and mirroring, were applied to train nnU-Net. The Adam optimizer with an initial learning rate of 3×10^{-4} is used to train the network. The epoch is set to 1000 and the batch size to 2. An epoch is defined as the iteration over 250 mini-batches. The training was conducted on a Linux machine with an NVIDIA GTX 1080Ti GPU and took 100 h.

2 Improved implementation details

In this section, we describe our training strategy in detail.

Data Augmentation. Elastic transformation and brightness transformation are also applied for data augmentation. We replace the percentage clipping with fixed CT window. The window is [-1200,600], and the maximum HU value is randomly selected from 400 to 600 for data augmentation in the training stage.

Sampling Strategy. The main trachea can be easily and accurately predicted for most deep learning models. However, the small peripheral bronchi are easily missed. As the main trachea accounts for a large percentage of the whole airway, random selection of the center point is inappropriate in accurate airway segmentation. To sample the small peripheral bronchi more, we divide the airway tree into two parts, main branch and other branches. And the central points are located more on the small branches. The ratio of the main branch and other branches is 3:2.

Loss Function. It is worth noting that small peripheral bronchi are critical in surgical planning, but are easily missed by the deep learning network due to low contrast, small area, etc. To alleviate this problem, a combination of top-k loss and dice loss is applied as the loss function, which is defined as:

$$L_{total} = L_{top-k} + L_{dice}$$

With $L_{top-k}(y,\hat{y}) = -\frac{1}{K}\sum_{i\in K} y\log(\hat{y}) + (1-y)\log(1-\hat{y})$, $L_{dice}(y,\hat{y}) = -\frac{2y\hat{y}}{y+\hat{y}}$

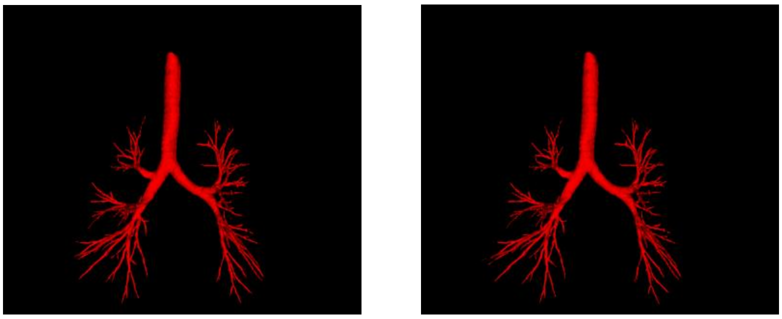
where k is set to 10%. Top-k loss is first proposed to do hard sample mining. To balance the learning for simple samples and hard samples, the dice loss is combined with top-k loss. In our experiments, top-k loss easily makes the network training unstable. Thus, the proposed loss function is only used to finetune the network with 100 epochs.

3 Experiments

In this section, we show the quantitative and qualitative result of the proposed improved nnUNet, and compared it with the original nnUNet in Table 1 and Fig 1 respectively. The experiment was carried out using 240 subjects for training and 60 subject for test. Results shows that the improved nnUNet can improves the segmentation of the PA branch segmentation.

Table 1. Comparison of segmentation performance on ATM2022 challenge dataset.

Methods	BD	TD	Dice	Precision
original	**82.69**	**90.86**	93.61	94.17
DA	79.82	88.37	93.91	95.04
DA+SS	79.92	88.45	93.93	95.02
DA+SS+TD loss	81.11	89.26	**94.15**	**95.12**



(a) nnUNet (b) improved nnUNet

Fig. 1. Segmentation results of nnUNet and improved nnUNet.

4 Conclusions

In this paper, we proposed an improved nnUNet framework to enhance the segmentation of the PA branch. It incorporates new data augmentation, voxel sampling strategy and our proposed loss function. The results demonstrated that all the proposed training strategies are effective for more accurate PA segmentation.

References

1. Isensee, F., Jaeger, P.F., Kohl, S.A.A. et al. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nature Methods 18(2), 203–211 (2021).