Attention Guided 3D U-Net for KiTS19

Zhusi Zhong[1], Zhenxi Zhang[1], Zhicheng Jiao[2]

[1] School of Electronic Engineering, Xidian University, Xi'an 710071, China
[2] Department of Radiology and BRIC, University of North Carolina at Chapel Hill,
Chapel Hill, NC 27599, USA

Abstract. We use a two-stage 3d U-Net model to predict the multi channels segmentations from coarse to fine. The second stage is guided by the predictions from the first stage.

1 Method

We proposed a two stages method to segment CT image from coarse to fine. The two stages are trained with different learning scope and are assigned with different learning missions.

1.1 Stage 1 – Coarse stage

Data preprocess. Firstly, we downscale the training data to a normal shape, in order to make sure the model can take a whole image at once. All the images and segmentations are downscale to 128*128*32 (height*width*depth). The segmentation files are transformed to 3-channels arrays, in which the channels-wise pixel values represent kidneys, tumors and the background (without kidneys and tumors) in order.

Training. We train the standard 3D U-Net follow with a softmax layer. While training, we apply some data augmentation to the training data, including normalize, random contrast, random flip and random rotate. We input all the 210 cases training data and train the model to regress the multi-channel segmentations. We apply with pytorch, and the learning rate is 0.1 which divide 0.1 in 300000 epochs and 500000 epochs. We use the Binary Cross Entropy Loss as loss function.

Predicting. The 90 cases testing images are preprocessed the same with the training images then input to the trained model. The channel-wise predictions are scaled back the original shape. We take the first 2 channels of the predictions, represent as the segmentation of kidneys and tumors, then package as the **.nii.gz** files.

1.2 Stage 2 – Fine stage

Attention guide. We extract bound boxes of the kidneys by obtain the minimum and maximum pixel location in the segmentations. Each box represents as a boundary of one kidney. While inferring, we extract the boxes from the coarse segmentations predicted from the first stage.

Data preprocess. We crop the training data in the bound boxes, but set the minimum size of the cropping box, which is 128*128*32 (height*width*depth). And we record the crop locations, in order to recover the crop segmentation to original size. The cropped segmentation patches are transformed to 3-channels arrays, in which the channels-wise pixel values represent kidneys, tumors and the background (without kidneys and tumors) in order.

Training. We train the standard 3D U-Net follow with a softmax layer. While training, we apply some data augmentation to the training data, including normalize, random contrast, random flip and random rotate. We apply 7-fold cross-validation on all the cropped kidney regions training data and train the model to regress the multi-channel segmentations. We apply with pytorch, and the learning rate is 0.1 which divide 0.1 in 300000 epochs and 500000 epochs. We use an average of Binary Cross Entropy Loss and focal loss [2] as loss function.

Predicting. The 90 cases testing images are cropped by the boxes extract from the first stage segmentations then input to the trained model. The patch predictions are placed in the recorded box location to original shape. 7-fold trained models generate 7 sets predictions, we average the predictions, and the first 2 channels represent as the segmentation of kidneys and tumors.

1.3 Predictions fine tuning

We crop the testing images in 4 different scopes, then we get 4 sets of averaged predictions. The 4 sets are weighted to 0.4, 0.3, 0.2 and 0.1 in the order of the size of scopes, then are summarized. We apply the region grow method to reduce the false positive predictions. The region grow generates the masks on kidneys, and the seed points are obtained from the mean position of the summarized prediction masked by the coarse masks from the stage 1. Finally the summarized predictions multiply by the masks generated from the region grow, then package as the **.nii.gz** files

References

1. Özgün Çiçek, et al.: 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In: MICCAI, (2016).
2. Lin, T.-Y., et al.: Focal loss for dense object detection. In: ICCV, pp. 2980-2988. (2017)