INSTANCE 2022: Challenge Report

Md Mahfuzur Rahman Siddiquee, Dong Yang, Yufan He, Daguang Xu, and Andriy Myronenko

NVIDIA, Santa Clara, CA
{mdmahfuzurr,dongy,yufanh,daguangx,amyronenko}@nvidia.com

1 Method

1.1 The Network

We implemented our approach with MONAI[1] [2]. We use the 2D version of encoder-decoder backbone based on [5] with an asymmetrically larger encoder to extract image features and a smaller decoder to reconstruct the segmentation mask [6–8].

We use 2D segmentation network operation on axial slices, with a crop size of 384x384 without re-sampling. We considered 3 difference version of 2D input: a) using 3 neighboring slices as a 3 channel 2D input b) using same slice at 3 different CT window-levels as 3 channel 2D input c) combine both approaches as 9 channel 2D input. The final ensembling is based on 20 trained checkpoints, which includes a combination of all such approaches.

Encoder part The encoder part uses 2D ResNet [3] blocks. We have used 5 stages of down-sampling; each stage has 2, 4, 4, 4, and 4 convolutional blocks, respectively. We have used batch normalization and ReLU. Each block's output is followed by an additive identity skip connection. We follow a common CNN approach to downsize image dimensions by 2 progressively and simultaneously increase feature size by 2. For downsizing, we use strided convolutions. All convolutions are 3x3 with an initial number of filters equal to 32. The encoder is trained with a 384x384 input region.

Decoder part The decoder structure is similar to the encoder one, but with a single block per each spatial level. Each decoder level begins with upsizing with transposed convolution: reducing the number of features by a factor of 2 and doubling the spatial dimension, followed by the addition of encoder output of the equivalent spatial level. The end of the decoder has the same spatial size as the original image, and the number of features equal to the initial input feature size.

[1] https://github.com/Project-MONAI/MONAI

2 Training Method

2.1 Dataset

We have used the INSTANCE dataset [1] only for training the model. We have randomly split the entire dataset into 5-folds and trained a model for each.

2.2 Loss

We have used DiceCE loss for training.

2.3 Optimization

We use the AdamW optimizer with an initial learning rate of $2e^{-4}$ and decrease it to zero at the end of the final epoch using the Cosine annealing scheduler. We have used a batch size of 16 per gpu and trained it on 8 GPUs machine. All the models were trained for 300 epochs. We have also used deep supervision during training.

2.4 Regularization

We use L2 norm regularization on the convolutional kernel parameters with a weight of $1e^{-5}$.

2.5 Data preprocessing and augmentation

We have applied random rotation and random zoom on each axis with a probability of 0.4; random contrast adjustment and random Gaussian noise with a probability of 0.2. We have also applied random coarse shuffle [4] and random flips on each axis with a probability of 0.5.

3 Results on Cross-Validation

Our cross-validation results on the 5-folds can be found in Tab. 1.

Fold 1	**Fold 2**	**Fold 3**	**Fold 4**	**Fold 5**	**Average**
0.7303	0.8513	0.8205	0.7619	0.8155	0.7959

Table 1. Average DICE among classes using 5-fold cross-validation.

4 The Team

Team Name: NVAUTO
Team Members:
Md Mahfuzur Rahman Siddiquee, Dong Yang, Yufan He, Daguang Xu, Andriy Myronenko

References

1. Instance22, https://instance.grand-challenge.org/Dataset/
2. Project-monai/monai, https://doi.org/10.5281/zenodo.5083813
3. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: European conference on computer vision. pp. 630–645. Springer (2016)
4. Kang, G., Dong, X., Zheng, L., Yang, Y.: Patchshuffle regularization. arXiv preprint arXiv:1707.07103 (2017)
5. Myronenko, A.: 3D MRI brain tumor segmentation using autoencoder regularization. In: International MICCAI Brainlesion Workshop. pp. 311–320. Springer (2018)
6. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
7. Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J.: Unet++: A nested u-net architecture for medical image segmentation. In: Deep learning in medical image analysis and multimodal learning for clinical decision support, pp. 3–11. Springer (2018)
8. Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J.: Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. IEEE transactions on medical imaging **39**(6), 1856–1867 (2019)