ACU-NET: AN EFFICIENT CONVOLUTIONAL NETWORK FOR BIOMEDICAL IMAGE SEGMENTATION

A PREPRINT

Tianyu Ding
Department of Biostatistics
University of Pittsburgh
Pittsburgh, PA 15260
tid16@pitt.edu

July 6, 2020

1 Data

60 MRI scans(T1 and FLAIR) from WMH 2017 Challenge are used in the experiments. We have used 5-fold cross-validation to obtain optimal hyperparameter settings such as learning rate and batch size. The cross-validation is subject level, not slice level. Then, the model is trained on the full 60 scans.

2 Data preprocessing

ROBEX [1] is used for brain extraction. Since our model is a 2D model, we expand a 3D scan on axial view. For example, if a scan is with dimension $256 \times 256 \times 60$, we processed it as 60 slices with dimension 256×256. In addition, we have applied following image processing steps sequentially: (1) cropping to remove background slices, (2) padding all slices to square image, (3) resizing all slices to 256×256 dimension and (4) intensity normalization.

After above steps, we generated dataloader in PyTorch [2] with online data augmentation i.e. the augmented image data is generated in each epoch. The data augmentation techniques we used including: cropping, rotation and flipping.

3 Method

ACU-Net (Asymmetric Compact U-Net) is based on U-Net [3]. The goal of ACU-Net is to build a compact and efficient CNN for medical image segmentation. Compared with U-Net, ACU-Net is over 40x smaller on number of parameters and 140x on FLOPs. While tested on a same machine and same dataset, ACU-Net is 5 times faster than U-Net on an training epoch where the data loading time is bottleneck for ACU-Net on GPU. In addition, ACU-Net is 10 times faster than U-Net on CPU makes ACU-Net is available for CPU machines. ACU-Net can achieve competitive performance compared with U-Net.

The architecture of ACU-Net is described in Fig 1. It is different from U-Net in three main aspects: (1) ACU-Net is asymmetric, decoder parts with green color have fewer operations compared with their corresponding encoder parts with blue color, (2) ACU-Net is compact, it uses a light encoder and decoder design with much fewer channels compared with U-Net while still keep channel concatenation at the beginning of each decoder block and (3) ACU-Net is composed with depthwise convolutional layers. There are three techniques used in ACU-Net to build compact and efficient convolutional blocks described in Fig 2: (1) Depthwise separable convolution [4], (2) Inverted residual with linear bottleneck [5] and (3) Squeeze-and-Excitation(SE) [6].

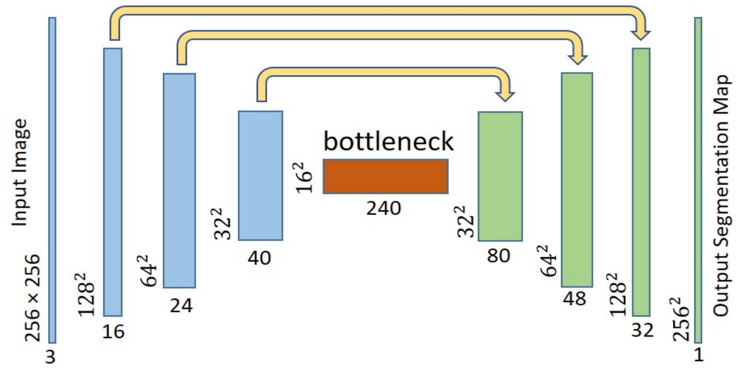

Figure 1: ACU-Net architecture

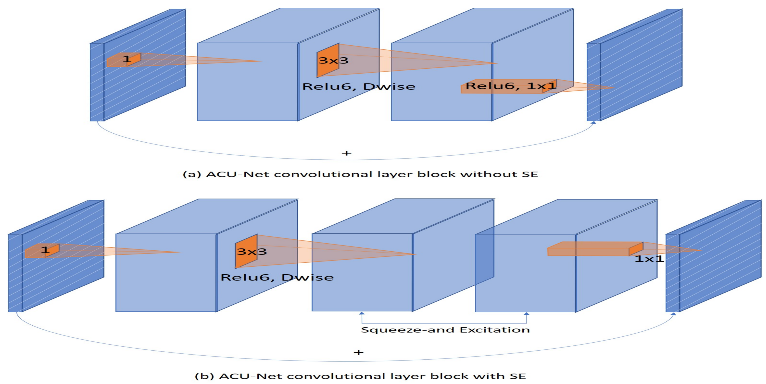

Figure 2: ACU-Net convolutional layer blocks
ACU-Net convolutional layer block without Squeeze-and-Excitation in (a) and with Squeeze-and-Excitation in (b).

References

[1] Juan Eugenio Iglesias, Cheng-Yi Liu, Paul M Thompson, and Zhuowen Tu. Robust brain extraction across datasets and comparison with publicly available methods. *IEEE transactions on medical imaging*, 30(9):1617–1634, 2011.

[2] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[4] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[5] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[6] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.