3D Patch based Convolutional Neural Network for WMH Segmentation

Miguel Luna[1], Sang Hyun Park[1]

[1] Department of Robotics Engineering,
Daegu Gyeongbuk Institute of Science and Technology, Republic of Korea,
shpark13135@dgist.ac.kr

Abstract. To address the problem in white matter hyperintensities (WMH) segmentation challenge at MICCAI 2017, we propose a 3D patch based convolutional neural network that can utilize features from multi-modality images, i.e., T1 and FLAIR images.

1 Introduction

Due to memory limitation on GPUs and the small amount of training data, it is often difficult to learn a convolutional neural network (CNN) which can handle the full size 3D images at once. A trivial solution is to use every 2D slices to learn a 2D CNN, but in this way, the accuracy may decrease since useful features in 3D space cannot be utilized. To address these issues, we sample a large amount of 3D patches from FLAIR and T1 images in the training set and then learn the 3D CNN which can predict the WMH label in the 3D patch. For inference, we sample 3D patches with a regular interval from the input set of FLAIR and T1 images and then predict the label using the 3D CNN. The final result is obtained by merging the predictions in every 3D patches.

2 Method

The proposed network consists of 3 stages, *i.e.*, encoder, transition, and decoder. The encoder takes the 3D input patches extracted from the set of FLAIR and T1 images. The patch size was heuristically defined as $8 \times 24 \times 24$ voxels on the z, y and x coordinates by considering the spacing information. The patches from two images are concatenated (*i.e.*, the block size is $8 \times 24 \times 24 \times 2$) and then pass through a series of three convolutional layers with 64 filters in cascade (Level 1). The output feature maps are reduced to the half of original size (*i.e.*, $4 \times 12 \times 12$) by average pooling and then pass through convolutional layers with 128 filters in cascade (Level 2). In the same manner, the feature maps are reduced more to the size of $4 \times 6 \times 6$ and pass through the convolutional layers with 256 filters (Level 3).

The decoding stage starts from the feature maps generated at Level 3. Like the U-Net [2], we apply a deconvolution layer with a kernel $1 \times 2 \times 2$ and stride

$1 \times 2 \times 2$ to generate the features maps which are compatible with the feature maps from the encoder at Level 2. The features maps from Levels 2 and 3 are concatenated followed by two convolutional layers and then upsampled by a deconvolution layer. Above process is repeated to get the features maps at Level 1. For those feature maps, the last convolution operation is applied to reduce the number of feature maps to one, followed by a sigmoid activation function. These scores are approximated to the nearest integer, 0 or 1, to create a segmentation map with a value of 1 for the WMH and 0 for others.

Unlike the U-Net, we add transition layers [3] between the encoder and the decoder. Specifically, the feature maps generated in the last layer of encoder at Level 1 pass through a convolutional layer with 16 filters and then are connected to the decoder at Level 1. Similarly, the feature maps on the encoder at Level 2 pass through a convolutional layer with 32 filters and then are connected to the decoder at Level 2.

All convolutional layers have $3 \times 3 \times 3$ kernels and ReLU activation and a batch normalization layer [1] are used after each convolutional layer.

3 Implementation Details

To address the intensity variations between subjects, we normalize the intensities in each patch so that the mean of intensities equals to zero and the standard deviation equals to one for each modality.

To reduce the training time and accurately train a model from randomly initialized weights, we perform an initial training using the patches randomly selected near the WMH regions. After 10000 thousand mini batches with the size 128 were run, we allow the model to learn from the patches extracted at any position in the image.

For inference, we sample the 3D patches with a stride of $4 \times 12 \times 12$ from the input set of FLAIR and T1 images. Since the stride is half of the 3D patch size, the label on every voxel is predicted 8 times. The final prediction is obtained by averaging all those 8 prediction scores. Among 60 training images, we used 48 images for training and the remaining 12 images for validation to find the optimal setting.

References

1. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37. (2015) 448–456
2. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Proceedings of MICCAI. (2015)
3. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. arXiv:1802.02611 (2018)