White Matter Hyperintensities Segmentation Using Fully Convolutional Network and Transfer Learning

Yongchao Xu [1,2,3], Thierry Géraud [1], Élodie Puybareau [1], Isabelle Bloch [2], Joseph Chazalon [1,4]

[1] EPITA Research and Development Laboratory, France

[2] Télécom ParisTech, Université Paris-Saclay, France

[3] Huazhong University of Science and Technology, China

[4] Université de La Rochelle, France

July 2017

Foreword: A longer description than the following one is given in the recent paper [4].

We applied a fully convolutional network and transfer learning to segment the white matter intensities. The method is fully automatic, and uses both T1 and FLAIR sequences. An overview of the proposed method is given in Fig. 1. The details of the whole pipeline is given hereafter.

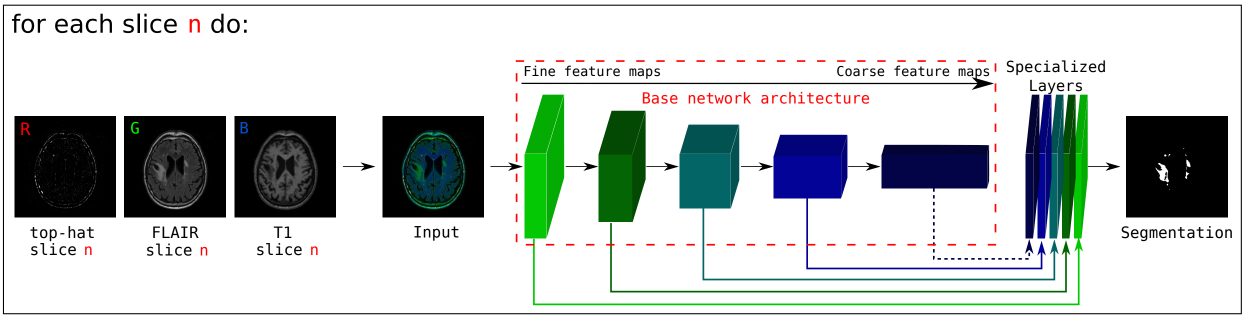

Figure 1: Architecture of the proposed network. We fine tune it and combine linearly fine to coarse feature maps of the pre-trained VGG network [3]. Note that each color image (`Input`) is built from the slice n of the T1 and FLAIR sequences.

Pre-processing. Our segmentation method uses both T1 and FLAIR sequences of brain. We perform a requantization of voxel values on 8bit. The FLAIR slices are filtered using a morphological operator (an area opening), so that small lesions are filtered out, and we compute the residue (difference between the original FLAIR image and the filtered one); in this final image, small lesions are particulary visible and large ones do not appear.

Deep FCN for white matter intensities segmentation. Efficient natural image segmentation can be achieved thanks to deep fully convolutional network (FCN) and transfer learning [1]. In this paper, we propose to rely on this same method to segment 3D brain MR images, although those images are very different from natural images. We rely on the 16 layers VGG network [3] pre-trained on millions of natural images in ImageNet for image classification. For our application, we discard the fully connected layers at the end of VGG network, and keep the 5 stages of convolutional parts called "*base network*". This base network is mainly composed of convolutional layers: $z_i = w_i \times x + b_i$, Rectified Linear Unit (ReLU) layers for non linear activation function: $f(z_i) = \max(0, z_i)$, and max pooling layers between two successive stages, where x is the input of each convolutional layer, w_i is the convolution parameter, and b_i is the bias term. The four max pooling layers divide the base network into five stages of fine to coarse feature maps. Inspired by the work in [1,2], we add specialized convolutional layers (with a 3×3 kernel size) with K (e.g. $K = 16$) feature maps after the convolutional layers at the end of each stage. We resize all the specialized layers to the original image size, and concatenate them together. A last convolutional layer with kernel size 1×1 is appended at the end that combines linearly the fine to coarse feature maps in the concatenated specialized layers, to produce the final segmentation result. The proposed network architecture is depicted in Fig. 1.

The architecture described above is very similar with the one used in [2] for retinal image analysis, where the retinal images are already 2D color images. For our application, the question amounts to how to prepare appropriate inputs given that a brain MR image is a 3D volume. To get RGB input images, we propose to stack successive 2D slices. Precisely, to form an input artificial color image for the pre-trained network to segment the n^{th} slice, we use the slice n of FLAIR and T1 and a filtered FLAIR as respectively the green, blue and red channels. This process is depicted in Fig. 1 (left). Each 2D color image thus forms a representation of a part (a slice of FLAIR and T1) of the MR volume. Using such a 2D representation avoids the expensive computational and memory requirements of fully 3D FCN.

For the training phase, we use the multinomial logistic loss function for a one-of-many classification task, passing real-valued predictions through a softmax to get a probability distribution over classes. During training, we use the classical data augmentation strategy by scaling and rotating, and also subtract 127 (to center values at 0) for each channel in the training images. We fine tune the network for the first 50k iterations using a learning rate of $lr = 10^{-8}$, and the last 100k with a smaller learning rate ($lr = 10^{-9}$). We rely on stochastic gradient descent with momentum to minimize the loss function with $momentum = 0.99$ for the first 50k iterations and 0.999 for the next 100k, and $weight\_decay = 0.0005$. The loss function is averaged over 20 images.

At test time, after having pre-processed the 3D volume (re-quantization), we prepare the set of 2D color images. Then we subtract 127 for each channel, and pass every image through the network.

Post-processing. After inference, the 3D volume is reconstructed from the 2D slices.

We run the test phase on a GPU card: `NVIDIA GeForce GTX 1080 Ti`, having 11Go.

References

[1] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In Proc. of CVPR, pages 3431–3440, 2015.

[2] Kevis-Kokitsi Maninis et al. Deep retinal image understanding. In Proc. of MICCAI, pages 140–148, 2016.

[3] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556, 2014.

[4] Yongchao Xu, Thierry Géraud, and Isabelle Bloch. From neonatal to adult brain MR image segmentation in a few seconds using 3D-like fully convolutional network and transfer learning. In Proceedings of the 23rd IEEE International Conference on Image Processing (ICIP), Beijing, China, September 2017. To appear.