A UNet Pipeline for Segmentation of New MS Lesions

Cory Efird, Dylan Miller, and Dana Cobzas

MacEwan University, Edmonton AB, Canada

Abstract. A pipeline for the second multiple sclerosis segmentation challenge (MSSEG-2) hosted by MICCAI is proposed. Two FLAIR images taken at different time-points are used as a multi-channel input to a 3D CNN to detect new lesions. Patch sampling strategies are adopted to keep the input volume shape manageable in terms of memory requirements. To further improve results, multiple models and patch orientations are ensembled. Performance is evaluated against nn-UNet.

Keywords: Multiple Sclerosis · Segmentation · Deep Learning

1 Method

The proposed method uses a 3D convolutional neural network (CNN) to detect new multiple sclerosis (MS) lesions in FLAIR images taken at two different time-points. We have chosen a relatively simple approach where both time-points are used as a multi-channel input to the CNN. This is viable due to the accurate co-registration that was performed between the two time-points. As a result, the network is able to produce spatio-temporal features early in the first few layers. The network is trained using patches that are randomly sampled from brain volumes. At inference time, predictions are generated for evenly spaced overlapping patches that cover the entire volume. Afterward, the patches are combined by interpolating the overlapping regions.

A survey of MS lesion segmentation with convolutional neural networks (CNNs) was recently published [1], which guided the development of our method. Furthermore, the success of the nn-UNet pipeline [3] influenced the pipeline design, and served as a baseline method for comparisons.

1.1 Data Processing

Pre-Processing First, the standard processing package that is provided by the contest organizers is applied. This includes brain extraction, bias field correction, and masking of non-brain image regions. The processing is extended with the following steps: All images are re-sampled to a target spacing that is close to $1\times1\times1\text{mm}^3$, with a tolerance of 0.11mm for each axis. The tolerance allows for integer multiples and divisions of the original spacing to be preferred (e.g. a spacing of 0.45mm is resampled to 0.9mm). The volumes are cropped to the

bounding box of the brain mask. At this point, augmentations are applied to training data. Finally, the intensity of the FLAIR images is adjusted by clipping values below the 0.05^{th} and above the 99.5^{th} percentiles, and then transforming the remaining values onto the range $[-1, 1]$.

Augmentation A number of spatial and intensity augmentations implemented in the TorchIO package [2] are applied to the training data before intensity normalization. First, a random volume orientation is chosen. The volume axes may be permuted or flipped so that 48 orientations are possible. If a spatial augmentation is applied (p=0.75), it is either an elastic deformation (p=0.2), or an affine transformation (p=0.8). Intensity augmentations include procedurally generated bias fields (p=0.5), modifying gamma by raising image values to a random power (p=0.8), a random blur kernel (p=0.2), and random high-frequency noise (p=0.35).

1.2 Implementation

Patch Extraction During training, patches with a size of 96×96×96 are sampled from random locations in the brain volume. A weighted volume is used to set the relative probability that a voxel will be chosen as the center of a patch. Voxels that are background, brain, and lesion have probabilities of 0, 1, and 100 respectively.

Model Our network architecture is a standard 3D UNet with 6 layers and residual blocks (Figure 1). The number of feature maps are linearly increased by 40 every 2 downsampling operations, as opposed to the common doubling approach. This saved many parameters without harming segmentation performance. Adding an anti-aliasing step when downsampling was shown to improve shift-invariance for classification tasks [5]. Anti-aliased convolutions with a stride of 2 are used for upsampling and downsampling to test this operation in a segmentation setting.

Loss Function A recent review of segmentation loss functions reported that a combination of dice and distribution-based loss functions is most reliable [4]. We adopt a commonly used hybrid of logistic and dice losses, which is formulated as follows:

$$\mathcal{L} = \frac{1}{C}\sum_{c=1}^{C}\left[\underbrace{-\frac{1}{N}\sum_{n=1}^{N} g_{n,c}\log p_{n,c}}_{\text{logistic term}} + \underbrace{1 - \frac{2\sum_{n=1}^{N} p_{n,c}g_{n,c}}{\sum_{n=1}^{N} p_{n,c}^2 + \sum_{n=1}^{N} g_{n,c}^2}}_{\text{dice term}}\right]$$

Training Five models are trained for cross validation. For each model 4 folds make up 32 participants in the training set, and the remaining fold has 8 participants for validation. The SGD optimizer is used with a batch size of 4 and a

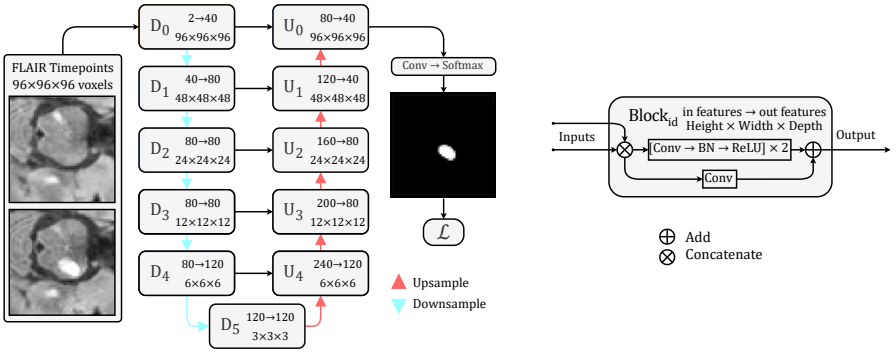

Fig. 1. Proposed model architecture. The convolutional block has a typical [Conv → BN → ReLU] × 2 local topology. A residual path has a single convolution to match the number of output channels. Feature maps are channel-wise concatenated when blocks have multiple input connections.

learning rate of 0.001. Training was halted if there was no improvement in mean dice score across the validation subjects for 2000 iterations, which resulted in an average of 7000 iterations per model. To train all models it took 330 hours on Tesla V100 GPUs with 32GB of RAM. The code was written using the PyTorch framework.

Inference To generate final predictions, a sliding window extracts patches with a 50% overlap across the entire volume. Due to the overlap, there are 8 patches which contribute to a voxel. Additionally, 8 patch flips are passed into the model and ensembled, and then all 5 models from the cross validation are ensembled at test time. In total there are $5 \cdot 8 \cdot 8 = 320$ forward passes through the model that contribute to every voxel.

2 Results

The commonly used Dice Similarity Coefficient (DSC) is reported, however, it has shortcomings when evaluating lesion segmentation. For subjects with no lesions, DSC is undefined. To handle this we let DSC=1 if the model correctly predicts no lesions, and DSC=0 if the model predicts even a single lesion voxel. A more robust test [6] was re-implemented to obtain precision, recall, and F1 measures for lesion detection. The performance on the test set has not been released at the time of writing this paper. Instead, single-model cross validation results are reported in Table 1, and 3D surface renderings of lesion segmentations are displayed in figure 2. In summary, our pipeline has a high detection recall, but lower precision when compared to nn-UNet.

Pipeline	Detection F1	Detection Precision	Detection Recall	Dice
nn-UNet	**0.77**	**0.70**	0.55	0.50
Ours	0.71	0.58	**0.88**	**0.54**

Table 1. Mean detection F1, precision, recall, and dice scores for all 40 participants. Lesion segmentations were produced by the model where the participant was in the validation set.

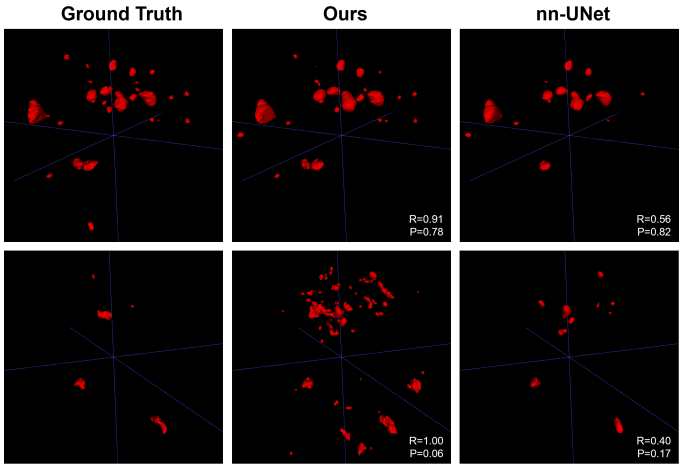

Fig. 2. 3D surface rendering of segmentation results for subjects 095, 057, 094 and 026 (from top to bottom row). Detection precision (P), and detection recall (R) is displayed for predictions from nn-UNet and our pipeline.

References

1. Zhang H., Oguz I. Multiple Sclerosis Lesion Segmentation - A Survey of Supervised CNN-Based Methods. In: Crimi A., Bakas S. (eds) Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. BrainLes 2020. Lecture Notes in Computer Science, vol 12658, 2021.
2. F. Pérez-García, R. Sparks, and S. Ourselin. TorchIO: a Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning. Computer Methods and Programs in Biomedicine (June 2021), p. 106236. ISSN: 0169-2607
3. Isensee, F., Jaeger, P.F., Kohl, S.A.A. et al. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nat Methods 18, 203–211 (2021). https://doi.org/10.1038/s41592-020-01008-z
4. J. Ma. Loss odyssey in medical image segmentation. Medical Image Analysis, vol. 71 2021.
5. Zhang, R. Making Convolutional Networks Shift-Invariant Again. ICML, 2019.
6. Commowick, O., Istace, A., Kain, M. et al. Objective Evaluation of Multiple Sclerosis Lesion Segmentation using a Data Management and Processing Infrastructure. Sci Rep 8, 13650 (2018). https://doi.org/10.1038/s41598-018-31911-7