A Fast and Memory-Efficient Brain MRI Segmentation Framework for Clinical Applications

Nejad, Ashkan; Masoudnia, Saeed; Nazem-Zadeh, Mohammad-Reza

Published in:
2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)

Publication date:
2022

A Fast and Memory-Efficient Brain MRI Segmentation Framework for Clinical Applications

Ashkan Nejad[1], Saeed Masoudnia[2] and Mohammad-Reza Nazem-Zadeh[2]

***Abstract*— Current segmentation tools of brain MRI provide quantitative structural information for diagnosing neurological disorders. However, their clinical application is generally limited due to high memory usage and time consumption. Although 3D CNN-based segmentation methods have recently achieved the state-of-the-art and come up with timely available results, they heavily require high memory GPUs. In this paper, we customize a memory-efficient (GPU) brain structure segmentation framework, named FLBS, based on nnU-nets which enables our framework to adapt its architecture based on memory constraints dynamically. To further reduce the need for memory, we also reduce multi-label brain segmentation to the fusion of sequential single-label segmentations. In the first step, single label patches are extracted from the T1w and segmentation maps by locating the approximate area of each structure on the MNI305 template, including the safety margin. These considerations not only decrease the hardware usage but also maintains comparable computational time. Moreover, the target brain structures are customizable based on the specific clinical applications. We evaluate the performance in terms of Dice coefficient, runtime, and GPU requirement on OASIS-3 and CoRR-BNU1 datasets. The validation results show our comparable accuracies with state-of-the-arts and confirm the generalizability on unseen datasets while significantly reducing GPU requirements and maintaining runtime duration. Our framework is also executable on a budget GPU with a minimum requirement of 4G RAM.**

***Clinical relevance*— We develop a memory-efficient deep Brain MRI segmentation tool that significantly reduces the hardware requirement of MRI segmentation while maintaining comparable accuracy and time. These advantages make FLBS suitable for widespread use in clinical applications, especially for clinics with a limited budget. We plan to release the framework as a part of a free clinical brain imaging analysis tool. The code for this framework is publicly available*.**

I. Introduction

The precise segmentation of human brain MRI is an important intermediate step for the quantitative analysis process of neurological studies. It also provides valuable structural information for the diagnosis, diseases monitoring, and treatment control for most neurological disorders. There are two available approaches for MRI segmentation; traditional neuroimaging software and recently developed deep network-based methods. While the first approach suffers high time and computational load, the second has recently been proposed to address these limitations.

The deep supervised networks replace time-consuming sequential pipelines with a feedforward network. In this approach, Fully Convolutional Networks (FCNs) are used to learn segmentation in an end-to-end manner from image. Their parallel implementations on graphical processing units (GPU) reduces computation time from hours to few minutes.

U-Net architecture is one of the most successful FCN models in MRI segmentation [1]. Different U-net architectures are employed to segment MRI images, including 2D and 3D U-Nets. The 2D U-Nets process MRI images slice by slice and thus suffer from missing the entirely contextual and neighborhood information. Although 3D U-nets were suggested to address this limitation, another issue was raised. The 3D U-nets encountered GPU memory constraints when dealing with tens of millions of computations corresponding to 3D feature spaces of high-resolution brain MRI volumes.

Several techniques were proposed to resolve the GPU memory constraints, e.g., SLANT [2] and AssemblyNet [3]. Despite using smaller networks and dividing input volume, these methods still occupy a minimum of 8-12 GB of memory in the inference phase. As a result, this requirement makes it difficult to use such methods in widespread clinical use with limited computational resources where this limitation is severe in the low-income communities. From other viewpoint, in most clinical applications, segmentation of only a limited number of structures is needed based on the user requirements where the whole-brain segmentation methods are not customizable in this manner. For this reason, recent methods [4], [5] focus on specific essential brain regions rather than whole-brain segmentation.

This paper aims to tackle the computational challenge of running a state-of-the-art brain segmentation tool on a budget GPU. To this end, we present and customized a low-cost framework with a core of an already state-of-the-art 3D U-net, nnU-net[6], for MRI brain segmentation called FLBS. Reducing the hardware requirement for running the tool will expand the application of this technology in medical clinics.

II. Proposed Method

Our proposed framework consists of three parts: 1) *Preprocessing* prepares MR images for training and testing by transforming them into a standard space to facilitate further processes; 2) *Segmentation* trains a series of nnU-Nets [6] on the designated extracted patches from the brain MR images; 3)*Post-processing* fuses the labels by solving the label conflict in the overlapping areas and transforms the

[1]Ashkan Nejad is with the Machine Learning Lab, Data Science Center in Health (DASH), University Medical Center Groningen, University of Groningen, The Netherlands a.nejad@rug.nl

[2]Saeed Masoudnia and Mohammad-Reza Nazem-Zadeh are with Medical Physics and Biomedical Engineering Department and the research center for molecular and cellular imaging (RCMCI), Tehran University of Medical Sciences (TUMS), Tehran, Iran s.masoudnia@gmail.com and mnazemzadeh@tums.ac.ir

* The code is publicly available on github.com/arnejad/FLBS

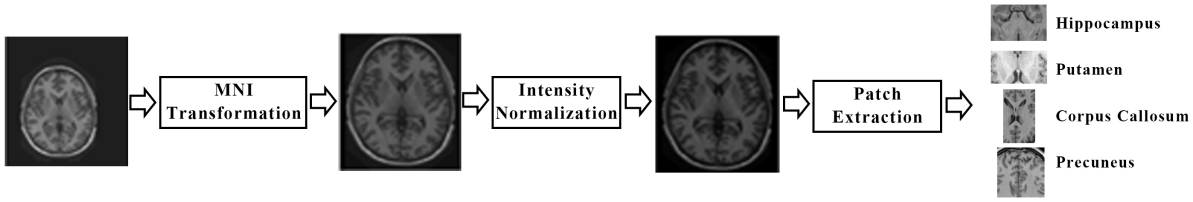

Fig. 1: The overall workflow of preprocess including affine transformation to MNI space, normalizaing the intensity values, and structure-based patch extraction, respectively. In the final step, each box extracts a neighborhood with a safe margin around pre-defined structures where their approximated locations are known in MNI305 space. We obtain the ranges of the filling area by each structure for the training data and add 10% to the ranges in each direction for a safe margin.

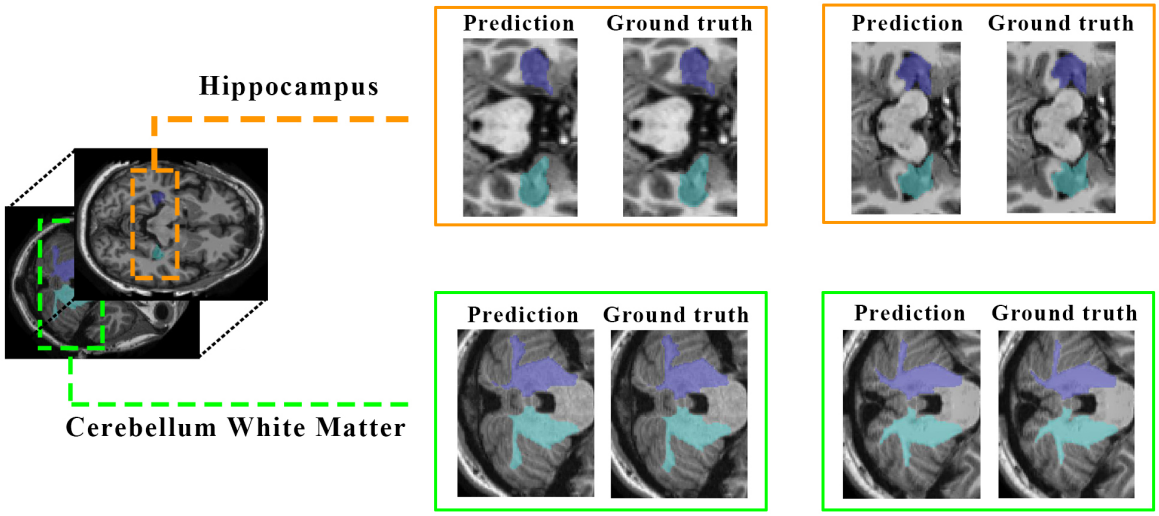

Fig. 2: Segmentation outputs of FLBS for Hippocampus and Cerebellum White Matter from two MRI samples. The ground truth segmentation refers to FreeSurfer (the most commonly used segmentation tool) segmentation output.The results show that FLBS provides segmentation with a smoother border.

result back to the space of input image before the pre-processing. The following subsections explain each stage.

A. *Pre-processing*

The image's intensity scale and size and the head's location and scale are prone to significant variations due to different imaging instrument calibrations and patient placement. The first pre-processing step transforms the brain image into a template space. We use the NiftyReg [7] for affine transformation of images into the MNI305 template [8]. The transformation also facilitates the later extraction of structure-surrounding patches. In order to solve the issue of diverse image intensity scales, we then perform the N4-bias correction [9] algorithm. This method scales all the intensity values into a normal domain.

The final step of pre-processing extracts a set of 3D patches (boxes) with predefined sizes and locations from the image. In MNI305 space, the approximate locations of structures are known. Each box extracts a neighborhood with a safe margin around a structure. We extract a patch for each structure on T1-weighted images and the corresponding segmentation map. The area to cover by each patch is obtained by computing the corresponding structure's minimum and maximum voxel on the training data. Then, the ranges are stretched by 10% to include a safe margin. Figure 1 illustrates the overall workflow of pre-processing. The following stage segments the extracted patches.

B. *Segmentation*

We employ the nnU-Net architecture [6] in MRI segmentation for two main reasons: 1. this architecture achieves state-of-the-art results in biomedical image segmentation. 2. This network benefits from a strategy known as the *network topology dynamic adaptation* [19]. The architecture adapts automatically based on specific constraints, e.g., size of the input images and memory limitation. These flexibility advantages are utilized in FLBS.

We consider one separate network for each brain structure. The extracted patch around the structure in fed into the

TABLE I: Summary of used collections from MRI datasets

Dataset	# Subjects	MRI Sample size	Purpose
OASIS-3	200	$256 \times 256 \times 256$	train & test
CoRR-BNU1	20	$144 \times 256 \times 256$	test-only

TABLE II: The comparison of resulted structure-specific average Dice coefficient for FLBS with state-of-the-art methods on OASIS dataset. The results for other methods were adapted from [2], which were compared under similar conditions. Only [10] differs in terms of using combination of several datasets in addition to OASIS.

Structure	U-Net [11]	Patch CNN [12]	NLSS [13]	JLF [14]	SLANT [2]	3D hemisphere CNN [10]	FLBS
Amygdala - Left	68.77%	25.44%	67.04%	62.03%	70.74%	85.3%	**88.46**%
Amygdala - Right	68.24%	12.05%	64.9%	60.9%	66.86%	84.9%	**88.52**%
Cerebellum White Matter - Left	78.27%	66.65%	79.82%	75.83%	82.22%	93.3%	**94.43**%
Cerebellum White Matter - Right	79.48%	60.6%	80.47%	76.07%	82.7%	93.5%	**94.67**%
Hippocampus - Left	79.16%	49.84%	79.96%	76.89%	82.62%	89.1%	**92.82**%
Hippocampus - Right	79.36%	54.35%	81.53%	77.62%	83.42%	89.9%	**93.29**%
Pallidum - Left	74.49%	48.18%	77.75%	71.06%	81.07%	83.5%	**89.03**%
Pallidum - Right	76.73%	24.17%	79.17%	73.23%	80.9%	85.2%	**90.89**%
Putamen - Left	86.52%	64.24%	87.78%	82.92%	89.31%	89.9%	**92**%
Putamen - Right	86.28%	61.37%	88.3%	81.08%	89.18%	90.1%	**91.51**%
Thalamus - Left	86.48%	73.84%	87.85%	85.59%	87.42%	**93.2**%	90.03%
Thalamus - Right	87.3%	73.16%	88.53%	85.93%	88.52%	**94**%	90.24%

TABLE III: The comparison results of FLBS with state-of-the-art methods for transfer learning on CoRR-BNU1 dataset. The segmentation results are reported in terms of average Dice coefficients for each structure. The results of other methods were adapted from [15], which were compared based on similar experimental settings.

Structure	SAU-Net [15]	U-Net [1]	U-Net++ [16]	FastSurfer [17]	QuickNAT [18]	FLBS
Hippocampus - Left	88.06%	88.73%	88.14%	88.25%	85.81%	**91.83**%
Hippocampus - Right	87.58%	88.41%	87.24%	87.13%	84.44%	**88.97**%
Amygdala - Left	83.89%	81.19%	81.78%	82.41%	75.89%	**85.88**%
Amygdala - Right	84.38%	83.62%	80.55%	82.04%	71.18%	**85.38**%
Putamen - Left	89.32%	**89.75**%	88.35%	89.35%	87.48%	85.12%
Putamen - Right	90.04%	88.67%	90.3%	**90.38**%	87.75%	87.92%
Pallidum - Left	83.95%	**85.17**%	76.06%	82.04%	75.81%	83.8%
Pallidum - Right	87.68%	**88.81**%	86.18%	87.15%	80.78%	85.42%
Thalamus - Left	**92.78**%	92.02%	91.96%	91.68%	87.65%	87.96 %
Thalamus - Right	**92.78**%	92.72%	92.75%	91.68%	87.61%	87.84%

corresponding nnU-Net. Training on the extracted patches allows the networks to focus on its sub-task and learn the fine details of segmentation rather than both localization and segmentation. In the training stage, a combination of Dice and cross-entropy loss is computed at the final layer as the loss to back-propagate into all layers for weight adjustment. Hence, the loss function is defined as

$$\mathcal{L} = \mathcal{L}_{Dice} + \mathcal{L}_{CE}. \quad (1)$$

Having K classes and I pixels, Dice loss is obtained from

$$\mathcal{L}_{Dice} = -2|K| \sum_{k\in K} \sum_{i\in I} u_i^k v_i^k \sum_{i\in I} u_i^k + \sum_{i\in I} v_i^k, \quad (2)$$

where u is the softmax output of the network and v is the one hot encoding of the ground truth map.

The cross-entropy loss is calculated using

$$\mathcal{L}_{CE} = -\sum_{k\in K} \sum_{i\in I} v_i^k \log u_i^k + ((1 - v_i^k) \log(1 - u_i^k)) \quad (3)$$

In the prediction stage, boxes obtained from all networks are merged into a single 3D matrix. The final fusion and transformation are conducted in the post-processing stage. Since segmentation of the structures is independent in this step, this task is parallelizable to reduce the execution time significantly.

The segmentation core uses nnU-Net library which implemented in PyTorch and is publicly available. The segmentation networks were trained using $\mathcal{L}$ loss function and Adam optimizer for 1000 iterations. Other optimization parameters are similar to the default in the used nnU-Net library.

C. Post-processing

The post-processing procedure fuses the predicted labels for all structures into one 3D matrix. The conflicting regions are divided between contributing structures using a morphological procedure. At first, the borders of the overlapping area are identified. The interior is then filled layer by layer by convolving a 5-by-5-by-5 window on the overlapping region. This process is done for each structure that contributes to the creation of the common area in turn. The morphological process is finished when no unlabeled voxel remains. After reattaching all patches and correcting the overlapping areas, it is transformed to the initial space using NiftyReg algorithm.

III. Experimental Results

We evaluated the framework on two benchmark MRI datasets: OASIS-3 [20] and CoRR-BNU1 [21] datasets. The information of these two datasets is summarized in table I. The accuracies are reported based on the Dice coefficient (eq. 2). The first experiment was conducted on the OASIS-3 dataset, containing T1w MR images with the segmentation maps, where the first 126 samples and the successive 20 samples comprise the training and testing set, respectively. The obtained structure-specific accuracies on the OASIS-3 dataset are reported separately and compared with state-of-the-art methods on this dataset in table II. As shown, FLBS outperforms all of the other methods on the OASIS-3 dataset. In the second experiment, we also tested the trained framework on OASIS-3 for segmentation of MRI images

TABLE IV: Comparison of mean average time for thee structure segmentation on CoRR-BNU1 dataset. The inference time consumption is computed by averaging the mean time for each structure. The methods were executed on a GTX 1080Ti GPU. The minimum requirement of memory sizes during inference phase are also reported. The runtime of freesurfer was adapted from [15] based on running on CPU

Method	runtime	min. required memory size
SAU-Net	3.6 s	11 GB
QuickNAT	3.08 s	8 GB
FreeSurfer	3.2 m	-
SLANT	9.28 s	11 GB
FastSurfer	2.67 s	11 GB
FLBS	**2.72** s	**4** GB

in the CoRR-BNU1 [21] dataset. One of the limitations in current methods is the lack of good generalizability of the trained models on other unseen datasets. Table III presents the class-specific Dice coefficients in the case of transfer learning on another unseen dataset. This table shows that our trained framework on OASIS-3 is transferable with comparable accuracies on CoRR-BNU1. Moreover, figure 2 illustrates the segmentation outputs of FLBS for Hippocampus and Cerebellum White Matter.

The runtime comparison between the state-of-the-art methods and ours based on GPU hardware is provided in table IV. The inference runtime is calculated by averaging the mean computation time for each structure. However, the minimum requirements of memory size do not necessarily show the exact ones, since we did not monitor exact memory usages in these cases. We tested the models on GTX 1050Ti, 1080, and 1080Ti, respectively and reported the memory size corresponding to the weakest GPU which can run the model. Table IV shows that despite the sequential computation of FLBS, our time consumption is comparable to state-of-the-art methods. However, the memory requirement of our framework is significantly less than the other methods, while ouvrs is runnable on a budget GPU, e.g., GeForce GTX 1050Ti, compared methods needs to high-performance and expensive GPUs. By accurately monitoring the training and inference phase, we found that FLBS occupies a maximum of only 6 GB of RAM while training and 2.7 GB while testing. Its running time is also about 9 seconds on GTX1050Ti which shows feasibility of our solution in terms of time consumption.

IV. Discussion

Structural brain segmentation is one of the essential steps in diagnosing neurological disorders. Current methods for automating this task need high GPU memories or significant time on the CPU. These challenges limit clinical use, especially in low-income and highly populated regions (our local challenges). We addressed these challenges and developed a memory-efficient deep brain MRI segmentation tool that is runnable on a budget GPU for a few seconds but achieves high accuracy comparable with state-of-the-arts.The advantages were achieved since we reduced the simultaneous localization and fine segmentation in multi-structure brain segmentation into only the structure segmentation. It was implemented by limiting the input MRI volume to the estimated surrounding of the structure and reducing multi-label into the fusion of single-label segmentations. Better accuracies on OASIS-3 and good generalizations in the case of transfer learning on CoRR-BNU1 may confirm that the presented framework is a reliable choice for clinical use. However, ensuring its generalizations on different datasets and different imaging devices without retraining requires further evaluations. We plan to release the framework as a part of a free clinical brain imaging analysis tool.

References

[1] Olaf Ronneberger *et al.*, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015, pp. 234–241.

[2] Yuankai Huo *et al.*, "3d whole brain segmentation using spatially localized atlas network tiles," *NeuroImage*, vol. 194, 2019.

[3] Pierrick Coupé *et al.*, "Assemblynet: A large ensemble of cnns for 3d whole brain mri segmentation," *NeuroImage*, vol. 219, 2020.

[4] Han Shuo *et al.*, "Automatic cerebellum anatomical parcellation using u-net with locally constrained optimization," *NeuroImage*, vol. 218, pp. 116819, 2020.

[5] Zhengshi Yang *et al.*, "Cast: A multi-scale convolutional neural network based automated hippocampal subfield segmentation toolbox," *NeuroImage*, vol. 218, pp. 116947, 2020.

[6] Fabian Isensee *et al.*, "nnu-net: a self-configuring method for deep learning-based biomedical image segmentation," *Nature methods*, vol. 18, no. 2, pp. 203–211, 2021.

[7] Sébastien Ourselin *et al.*, "Reconstructing a 3d structure from serial histological sections," *Image and vision computing*, vol. 19, 2001.

[8] Alan C Evans *et al.*, "3d statistical neuroanatomical models from 305 mri volumes," in *1993 IEEE conference record nuclear science symposium and medical imaging conference*, 1993, pp. 1813–1817.

[9] Nicholas J Tustison *et al.*, "N4itk: improved n3 bias correction," *IEEE transactions on medical imaging*, vol. 29, no. 6, pp. 1310–1320, 2010.

[10] Evangeline *et al.* Yee, "3d hemisphere-based convolutional neural network for whole-brain mri segmentation," *Computerized Medical Imaging and Graphics*, vol. 95, pp. 102000, 2022.

[11] Özgün Çiçek *et al.*, "3d u-net: learning dense volumetric segmentation from sparse annotation," in *MICCAI*, 2016, pp. 424–432.

[12] Raghav Mehta *et al.*, "Brainsegnet: a convolutional neural network architecture for automated segmentation of human brain structures," *Journal of Medical Imaging*, vol. 4, no. 2, pp. 024003, 2017.

[13] Andrew J Asman *et al.*, "Hierarchical performance estimation in the statistical label fusion framework," *Medical image analysis*, vol. 18, no. 7, pp. 1070–1081, 2014.

[14] Hongzhi Wang *et al.*, "Multi-atlas segmentation with joint label fusion and corrective learning—an open source implementation," *Frontiers in neuroinformatics*, vol. 7, pp. 27, 2013.

[15] Minho Lee *et al.*, "Split-attention u-net: A fully convolutional network for robust multi-label segmentation from brain mri," *Brain Sciences*, vol. 10, no. 12, pp. 974, 2020.

[16] Zongwei Zhou *et al.*, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep learning in medical image analysis*, pp. 3–11. Springer, 2018.

[17] Leonie Henschel *et al.*, "Fastsurfer-a fast and accurate deep learning based neuroimaging pipeline," *NeuroImage*, vol. 219, 2020.

[18] Abhijit Guha Roy *et al.*, "Quicknat: A fully convolutional network for quick and accurate segmentation of neuroanatomy," *NeuroImage*, vol. 186, pp. 713–727, 2019.

[19] Fabian Isensee *et al.*, "nnu-net: Self-adapting framework for u-net-based medical image segmentation," *arXiv preprint arXiv:1809.10486*, 2018.

[20] Pamela J LaMontagne *et al.*, "Oasis-3: longitudinal neuroimaging, clinical, and cognitive dataset for normal aging and alzheimer disease," *MedRxiv*, 2019.

[21] Xi-Nian Zuo *et al.*, "An open science resource for establishing reliability and reproducibility in functional connectomics," *Scientific data*, vol. 1, no. 1, pp. 1–13, 2014.