Multi-Scale Brain Parcellator Documentation
Release 1.0.0-beta

Brain Communication Pathways Sinergia Consortium

Jan 11, 2019

Contents

This pipeline is developed by the Hagmann's group at the University Hospital of Lausanne (CHUV) for use within the SNF Sinergia Project 170873, as well as for open-source software distribution.
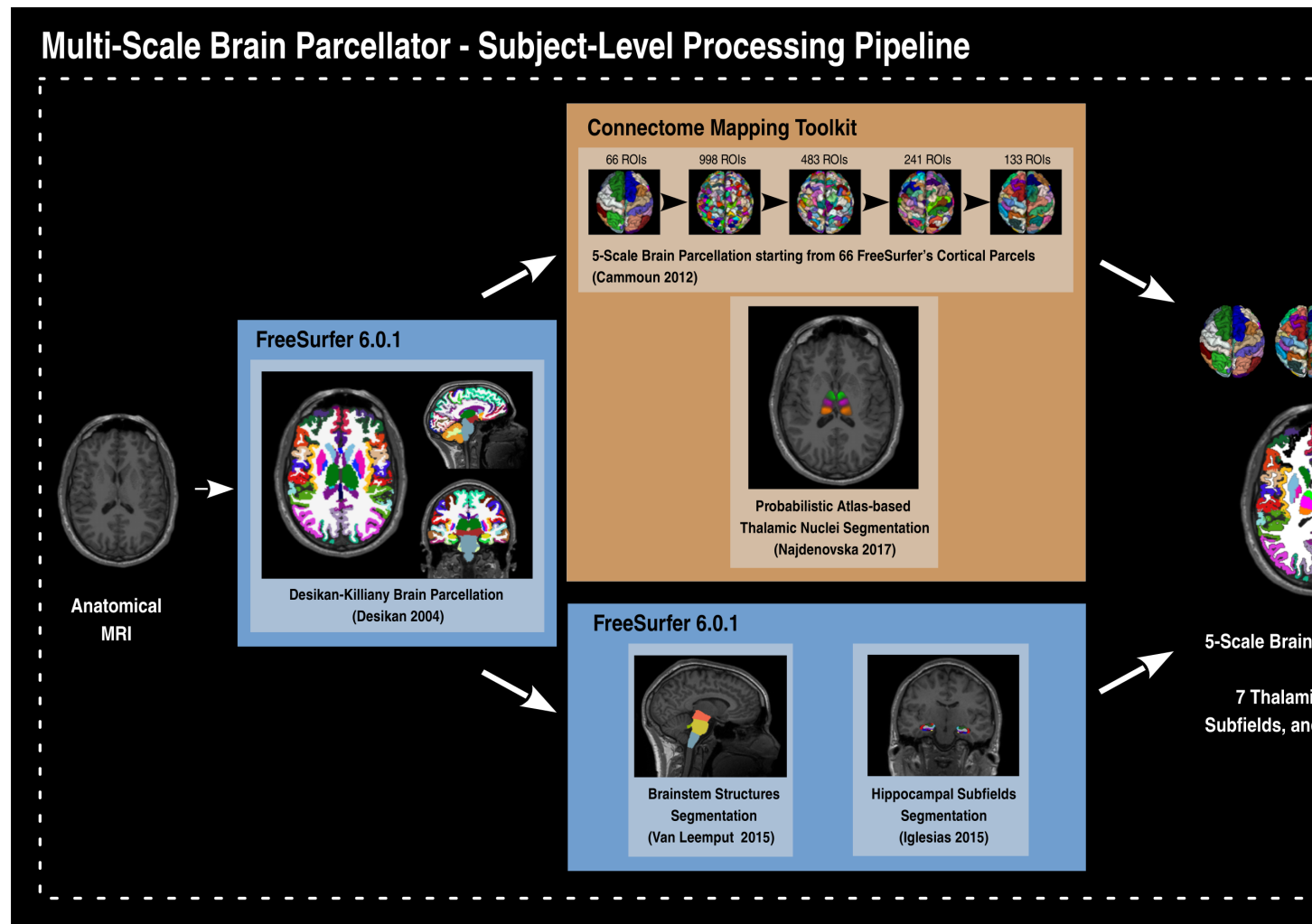
CHAPTER 1

About

`Multi-Scale Brain Parcellator`, part of the Connectome Mapping Toolkit (CMTK), is a BIDS App that implements a full anatomical MRI processing pipeline, from raw T1w data to structural brain parcellation at five different scales.

The `Multi-Scale Brain Parcellator` pipelines uses a combination of tools from well-known software packages, including **FSL_**, **FreeSurfer_**, **ANTs_** as well as in-house tools from CMTK.

This tool allows you to easily do the following:

- Take T1 from raw to brain parcellations at 5 different scales.
- Implement tools from different software packages.
- Automate and parallelize processing steps, which provides a significant speed-up from typical linear, manual processing.

Reproducibility and replicatibility is achieved through the distribution of a BIDSApp, a software container image which provide a frozen environment where versions of all external softwares and libraries are fixed.

Funding

CHAPTER 3

License information

This software is distributed under the open-source license Modified BSD. See license for more details.

All trademarks referenced herein are property of their respective holders.

Copyright (C) 2009-2019, Brain Communication Pathways Sinergia Consortium, Switzerland.

CHAPTER 4

Contents

4.1 Installation Instruction

> **Warning:** This software is for research purposes only and shall not be used for any clinical use. This software
> has not been reviewed or approved by the Food and Drug Administration or equivalent authority, and is for non-
> clinical, IRB-approved Research Use Only. In no event shall data or images generated through the use of the
> Software be used in the provision of patient care.

The Multi-Scale Brain Parcellator is a BIDS App that relies on Docker. Make sure that you have Docker installed or
follow the instructions found in *Installation of Docker*. Then, install the Multi-Scale Brain Parcellator by following
the instructions found in *Installation of the Multi-Scale Brain Parcellator*.

4.1.1 Installation of Docker

- Installed Docker Engine corresponding to your system
    - For Ubuntu 14.04/16.04/18.04, follow the instructions from the web page:

    ```
    $ firefox https://docs.docker.com/install/linux/docker-ce/ubuntu/
    ```

    - For Mac OSX (>=10.10.3), get the .dmg installer from the web page:

    ```
    $ firefox https://store.docker.com/editions/community/docker-ce-desktop-mac
    ```

    - For Windows (>=10), get the installer from the web page:

    ```
    $ firefox https://store.docker.com/editions/community/docker-ce-desktop-
    →windows
    ```

Note: Multi-Scale Brain Parcellator BIDSApp has been tested only on Ubuntu and MacOSX. For Windows users, it might be required to make few patches in the Dockerfile.

- For Ubuntu, Docker could be managed as a non-root user

    - Open a terminal

    - Create the docker group:

    ```
    $ sudo groupadd docker
    ```

    - Add the current user to the docker group:

    ```
    $ sudo usermod -G docker -a $USER
    ```

    - Reboot

    After reboot, test if docker is managed as non-root:

    ```
    $ docker run hello-world
    ```

4.1.2 Installation of the Multi-Scale Brain Parcellator

Installation and use of the Multi-Scale Brain Parcellator has been facilitated through the distribution of a Docker container image.

- Open a terminal

- Get the latest release of the BIDS App:

```
$ docker pull sebastientourbier/multiscalebrainparcellator:latest
```

- To display all docker images available:

```
$ docker images
```

You should see the docker image "multiscalebrainparcellator" with tag "latest" is now available.

Help/Questions

If you run into any problems or have any questions, you can post to the CMTK-users group. Code bugs can be reported by creating a "New Issue" on the source code repository.

4.2 Usage

4.2.1 Execution and the BIDS format

The `Multi-Scale Brain Parcellator` workflow takes as principal input the path of the dataset that is to be processed. The input dataset is required to be in valid BIDS (Brain Imaging Data Structure) format, and it must include at least one T1w or MPRAGE structural image. We highly recommend that you validate your dataset with the free, online BIDS Validator.

4.2.2 Commandline Arguments

The command to run `Multi-Scale Brain Parcellator` follow the BIDS-Apps definition with additional options specific to this pipeline.

Multi-scale Brain Parcellator BIDS App.

```
usage: multiscalebrainparcellator [-h]
                                  [--participant_label PARTICIPANT_LABEL [PARTICIPANT_
→LABEL ...]]
                                  [--number_of_cores NUMBER_OF_CORES]
                                  [--isotropic_resolution ISOTROPIC_RESOLUTION]
                                  [--thalamic_nuclei]
                                  [--hippocampal_subfields]
                                  [--brainstem_structures]
                                  [--skip_bids_validator] [-v]
                                  bids_dir output_dir {participant}
```

Positional Arguments

bids_dir	The directory with the input dataset formatted according to the BIDS standard.
output_dir	The directory where the output files should be stored. If you are running group level analysis this folder should be prepopulated with the results of theparticipant level analysis.
analysis_level	Possible choices: participant
	Level of the analysis that will be performed. Note that only participant level analysis is available.Multiple participant level analyses can be run independently (in parallel) using the same output_dir.

Named Arguments

--participant_label	The label(s) of the participant(s) that should be analyzed. The label corresponds to sub-<participant_label> from the BIDS spec (so it does not include "sub-"). If this parameter is not provided all subjects should be analyzed. Multiple participants can be specified with a space separated list.
--number_of_cores	The number of cores to be used forprocessing (Maximum number of available processing cores used by default)
--isotropic_resolution	The isotropic resolution in mmused to resample the original anatomical imagesand applied a tthe beginning of the processing pipeline.
--thalamic_nuclei	Whether or not to parcellate the thalamic nuclei
	Default: False
--hippocampal_subfields	Whether or not to parcellate the hippocampal subfields
	Default: False
--brainstem_structures	Whether or not to parcellate the brainstem structures
	Default: False
--skip_bids_validator	Whether or not to perform BIDS dataset validation
	Default: False

     -v, --version   Display the version of Multi-scale Brain Parcellator BIDS-App

4.2.3 Participant Level Analysis

To run the docker image in participant level mode (for one participant):

```
docker run -it --rm \
-v /home/localadmin/data/ds001:/bids_dataset \
-v /media/localadmin/data/ds001/derivatives:/bids_dataset/derivatives \
-v /usr/local/freesurfer/license.txt:/opt/freesurfer/license.txt \
sebastientourbier/multiscalebrainparcellator:latest \
/bids_dataset /bids_dataset/derivatives participant --participant_label 01 \
      --isotropic_resolution 1.0 \
      --thalamic_nuclei \
      --hippocampal_subfields \
      --brainstem_structures
```

Note: The local directory of the input BIDS dataset (here: `/home/localadmin/data/ds001`) and the output directory (here: `/media/localadmin/data/ds001/derivatives`) used to process have to be mapped to the folders `/bids_dataset` and `/bids_dataset/derivatives` respectively using the `-v` docker run option.

4.2.4 Debugging

Logs are outputted into `<output dir>/cmp/sub-<participant_label>/sub-<participant_label>_log-multiscalebrainparcellator.txt`.

4.2.5 Support and communication

The documentation of this project is found here: http://multiscalebrainparcellator.readthedocs.org/en/latest/.

All bugs, concerns and enhancement requests for this software can be submitted here: https://github.com/sebastientourbier/multiscalebrainparcellator/issues.

If you run into any problems or have any questions, you can post to the CMTK-users group.

4.2.6 Not running on a local machine? - Data transfer

If you intend to run `multiscalebrainparcellator` on a remote system, you will need to make your data available within that system first. Comprehensive solutions such as Datalad will handle data transfers with the appropriate settings and commands. Datalad also performs version control over your data.
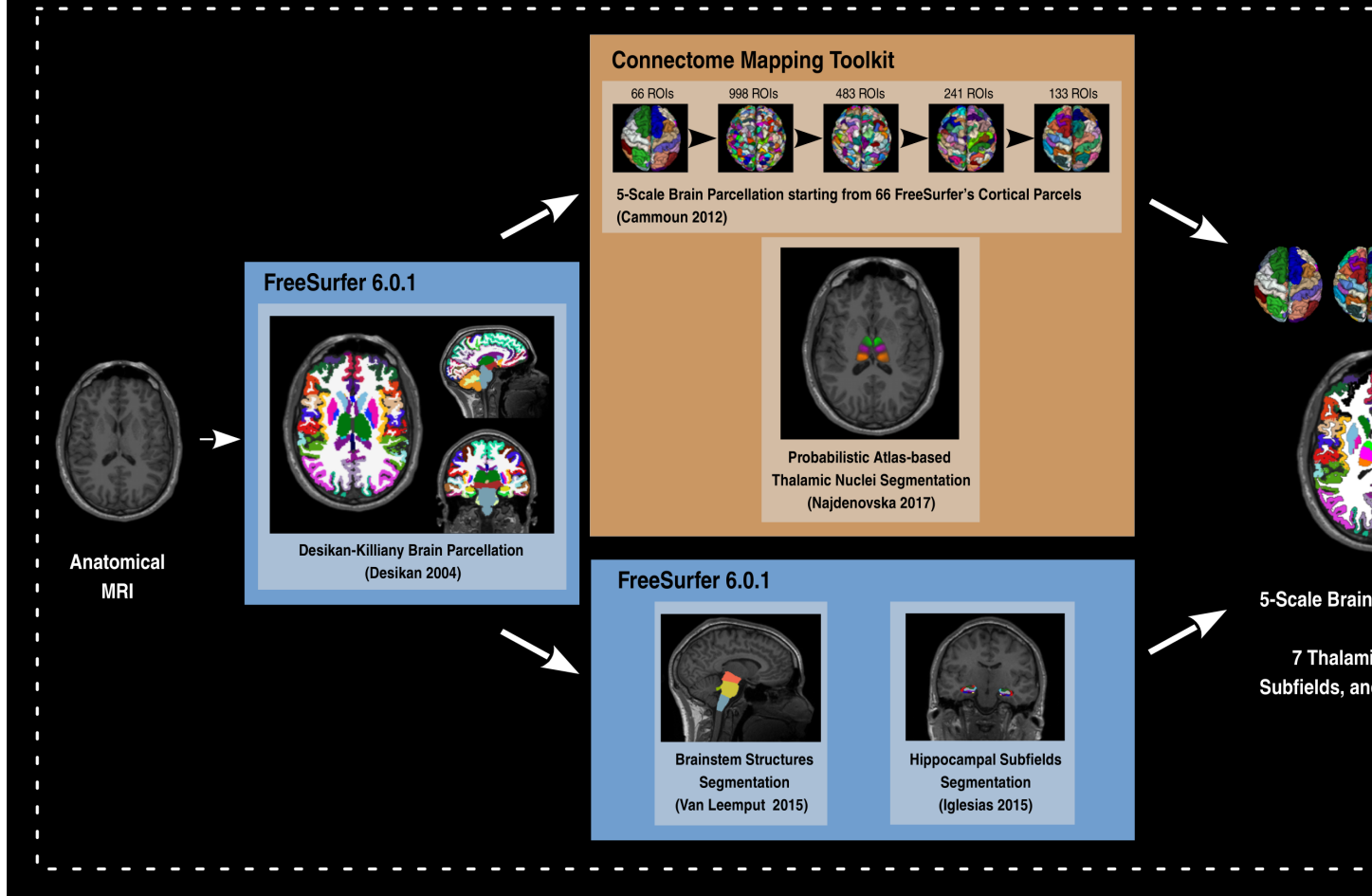
4.3 Processing Pipeline Details

4.3.1 Workflow

The `Multi-Scale Brain Parcellator` pipelines uses a combination of tools from the well-known software packages **FSL_**, **FreeSurfer_**, **ANTs_** as well as in-house tools from the Connectome Mapping ToolKit (CMTK).
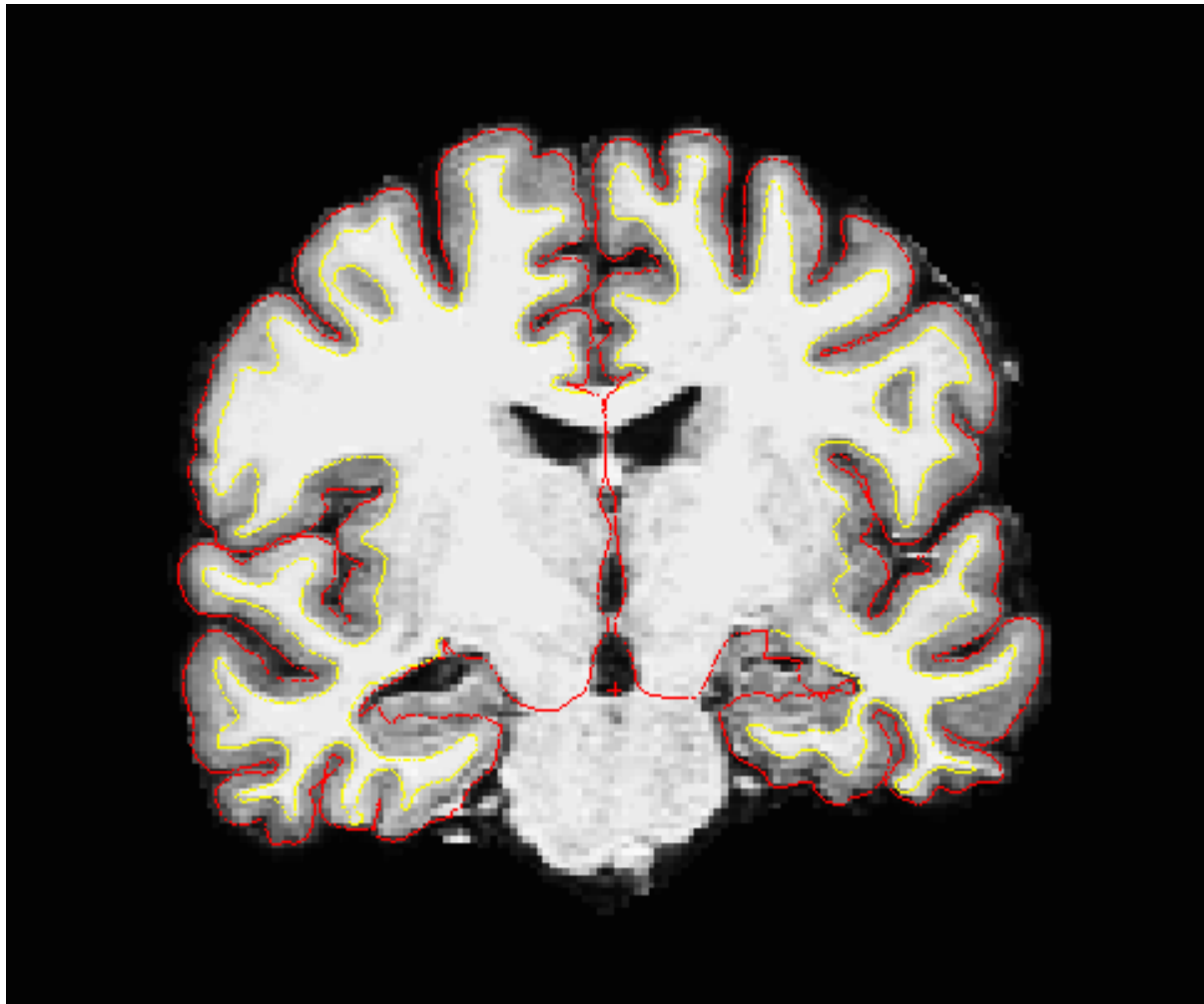
Multi-Scale Brain Parcellator - Subject-Level Processing Pipeline

Connectome Mapping Toolkit

66 ROIs   998 ROIs   483 ROIs   241 ROIs   133 ROIs

5-Scale Brain Parcellation starting from 66 FreeSurfer's Cortical Parcels
(Cammoun 2012)

FreeSurfer 6.0.1

Desikan-Killiany Brain Parcellation
(Desikan 2004)

Anatomical MRI

Probabilistic Atlas-based
Thalamic Nuclei Segmentation
(Najdenovska 2017)

FreeSurfer 6.0.1

Brainstem Structures
Segmentation
(Van Leemput 2015)

Hippocampal Subfields
Segmentation
(Iglesias 2015)

5-Scale Brain
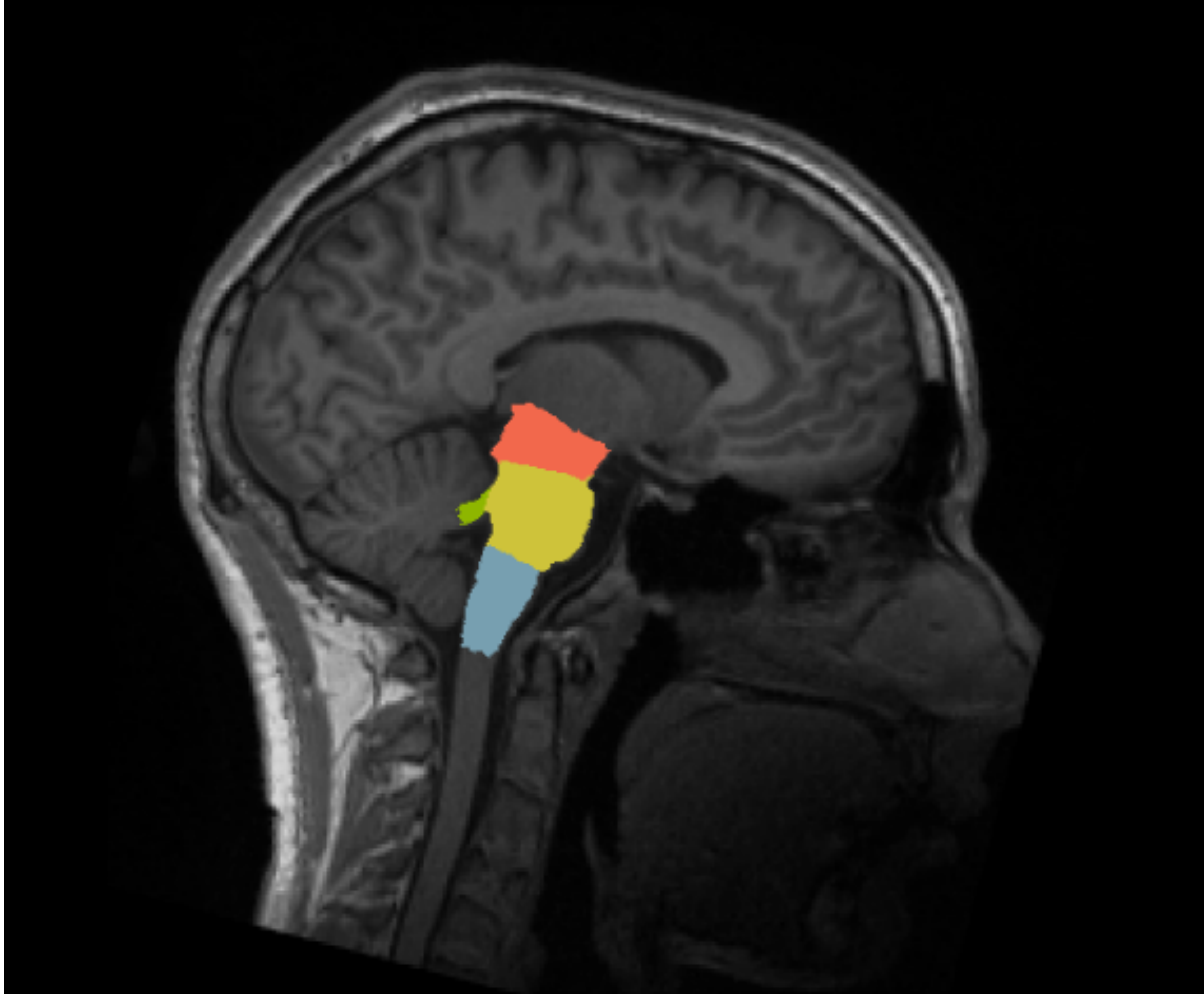
7 Thalami
Subfields, an

4.3.2 Segmentation

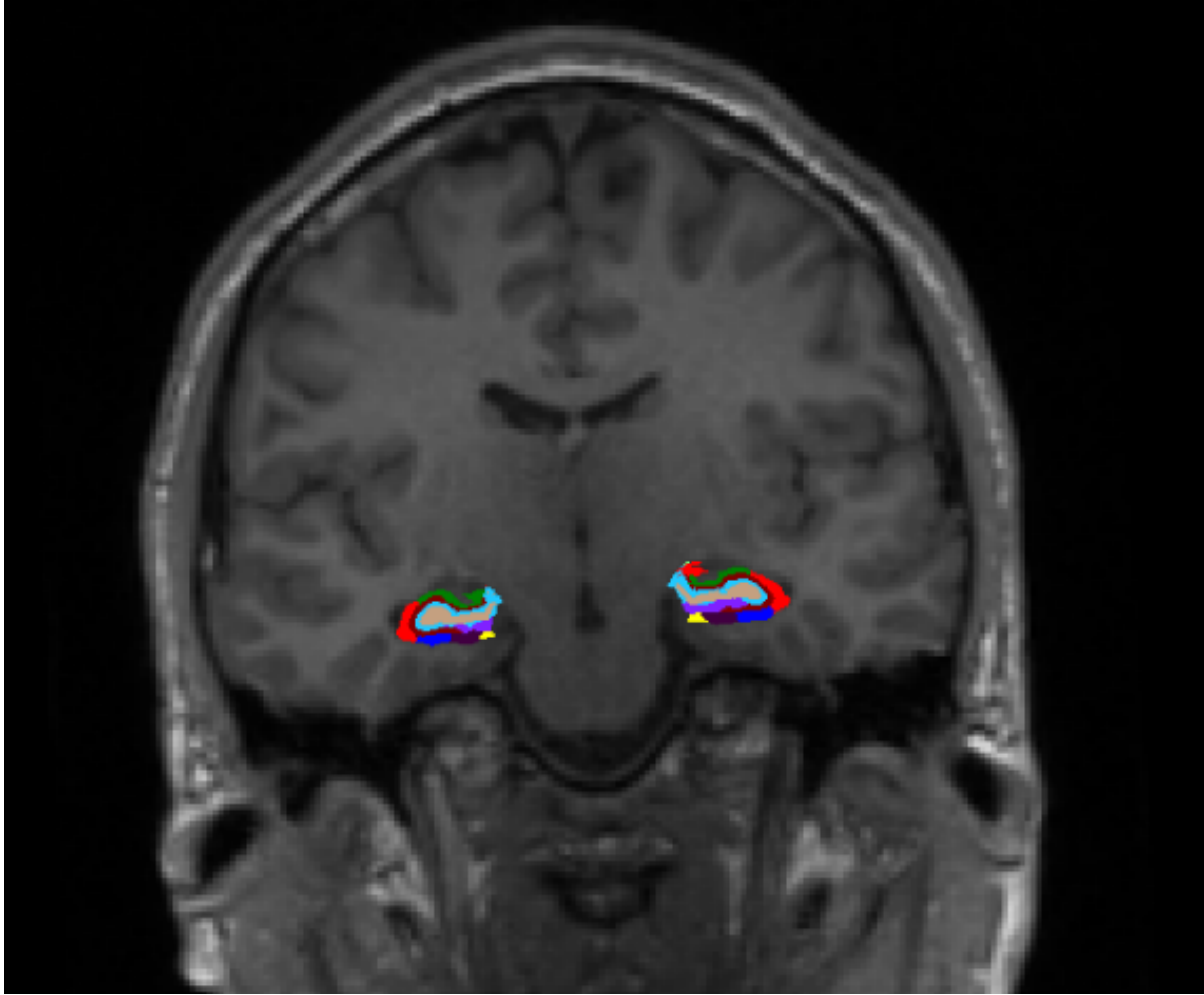Surfaces are extracted using Freesurfer.
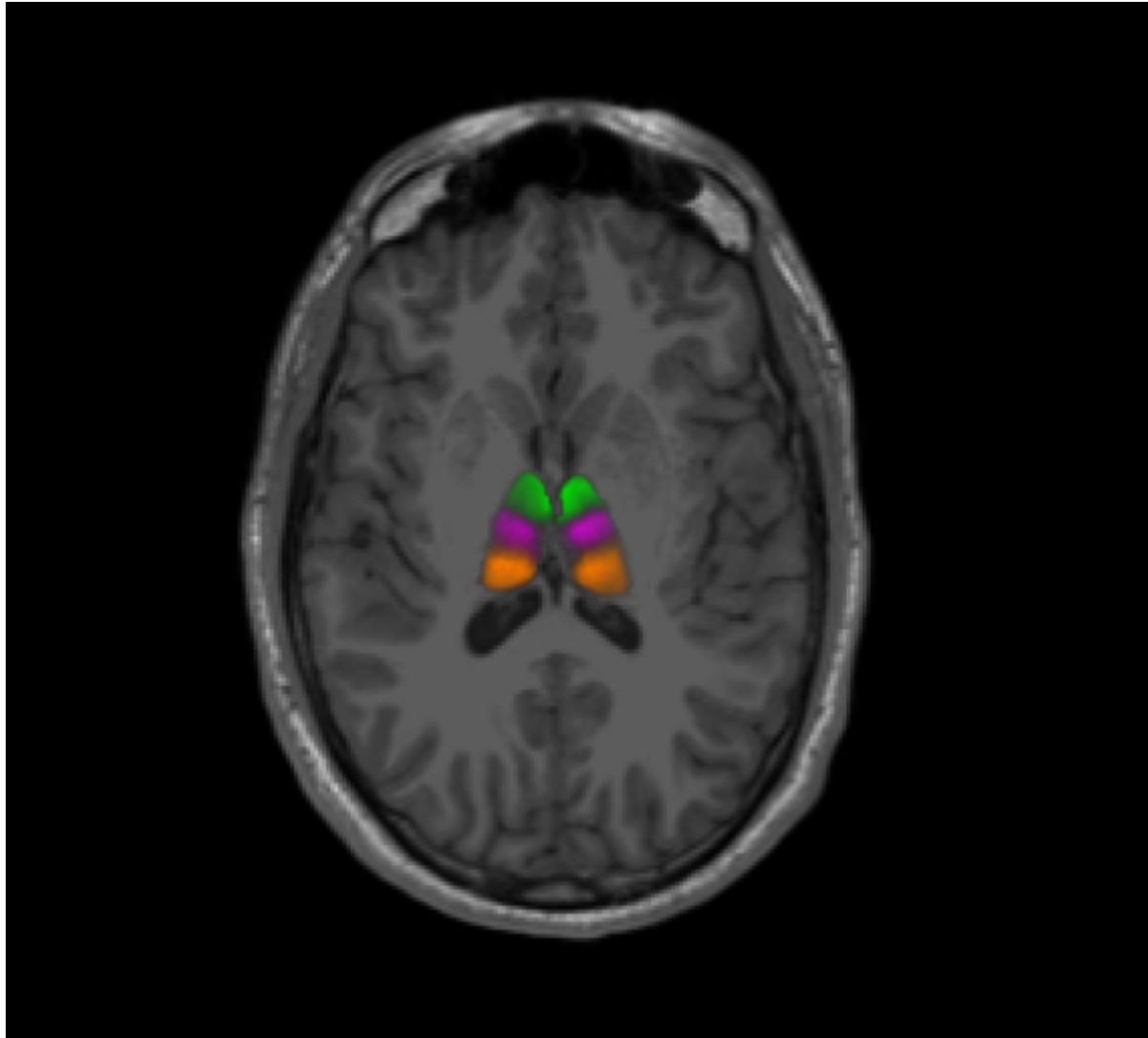
T1 is segmented using Freesurfer.

Brainstem sub-structures are segmented using Freesurfer.

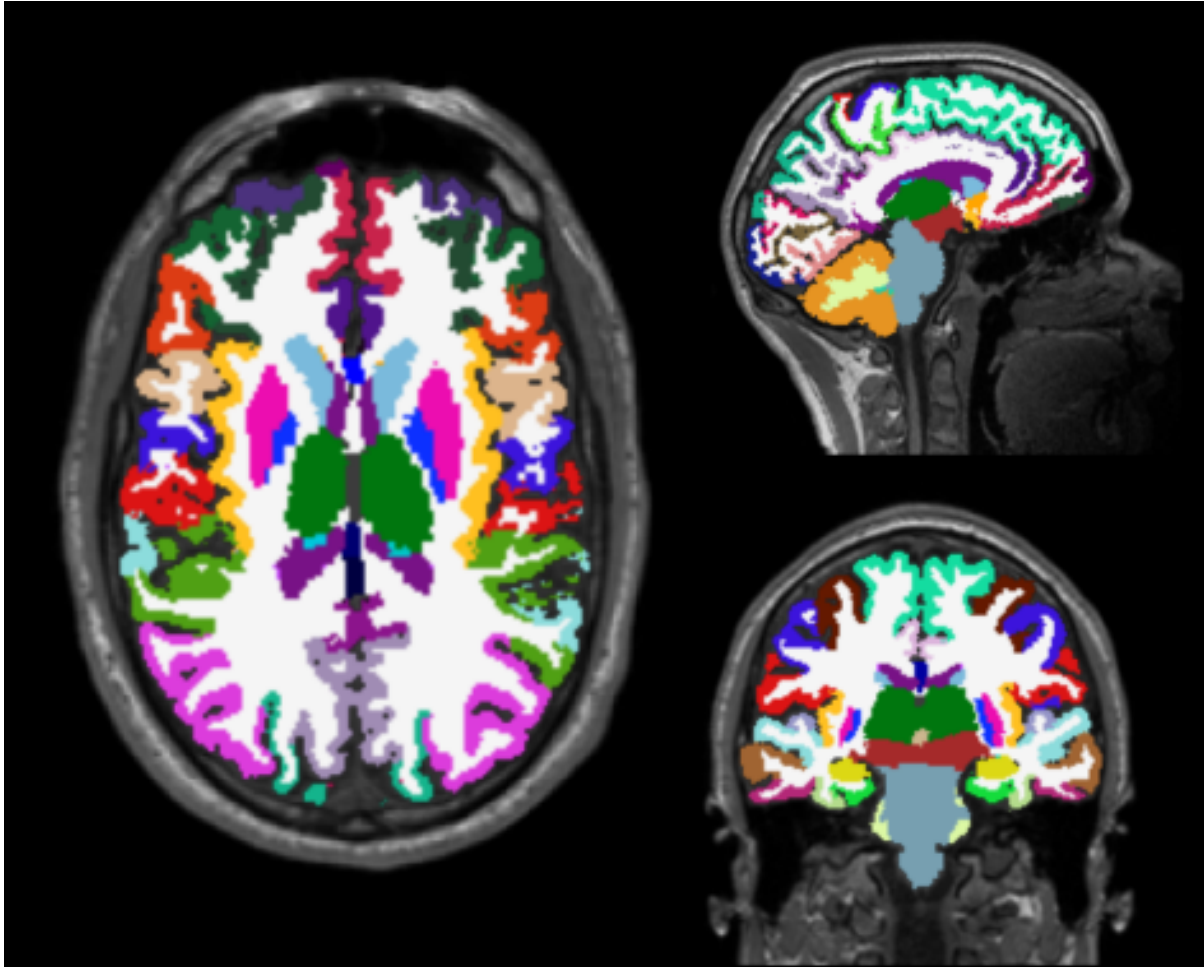Hippocampal subfields are segmented using Freesurfer.

Thalamic nuclei are segmented using an in-house tool of CMTK which implement the probabilistic atlas-based thalamic segmentation method with the help of ANTs (See Najdenovska 2018 in Citing Section).
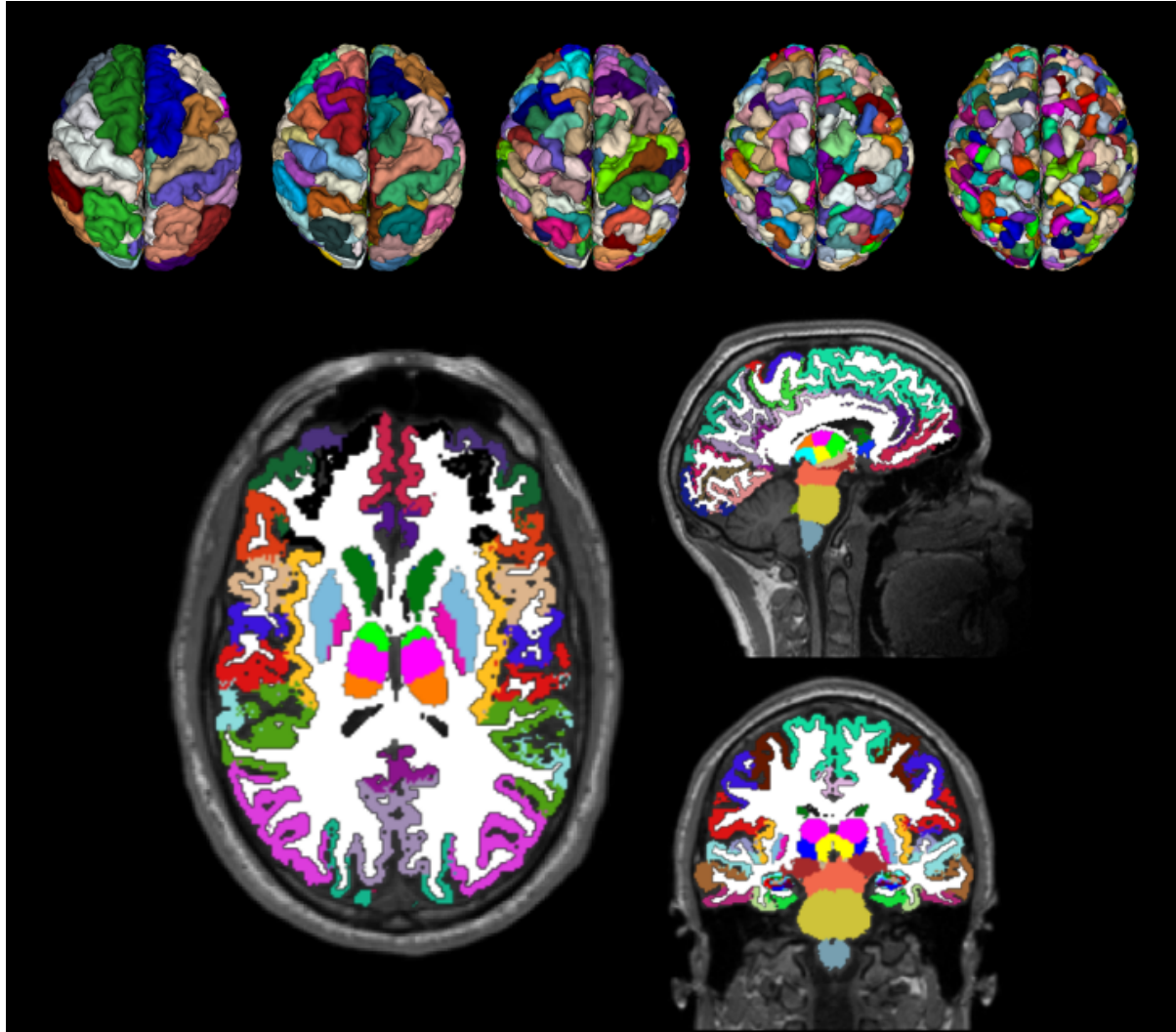
Parcellation

Desikan-Killiany brain parcellation is performed using Freesurfer.

5-Scale Brain parcellation is created according to Cammoun et al. 2012 (See `Citing`) at 5 different scales. All structures are then combined to create the final brain parcellation at each scale.

4.4 Outputs of Multi-Scale Brain Parcellator

Note: Multi-Scale Brain Parcellator outputs are currently being updated to conform to the BIDS Common Derivatives specification (see BIDS Common Derivatives Extension).

4.4.1 Multi-Scale Brain Parcellator Derivatives

Processed, or derivative, data are written to `<bids_dataset/derivatives>/cmp/ sub-<subject_label>/`. In this folder, a configuration file generated and used for processing each participant is saved as `sub-<subject_label>_anatomical_config.ini`. It summarizes pipeline workflow options and parameters used for processing.

Anatomical derivatives in the original `T1w` space are placed in each subject's `anat` subfolder including:

- `anat/sub-<subject_label>_desc-head_T1w.nii.gz`

- `anat/sub-<subject_label>_desc-brain_T1w.nii.gz`

- `anat/sub-<subject_label>_desc-brain_mask.nii.gz`

- `anat/sub-<subject_label>_label-WM_dseg.nii.gz`

- `anat/sub-<subject_label>_label-GM_dseg.nii.gz`

- `anat/sub-<subject_label>_label-CSF_dseg.nii.gz`

The five different brain parcellation are saved as:

- `anat/sub-<subject_label>_label-L2018_desc-<scale_label>_atlas.nii.gz`

where `<scale_label>`: `scale1`, `scale2`, `scale3`, `scale4`, `scale5` corresponds to the parcellation scale.

Additionally, the description of parcel labels and the updated FreeSurfer color lookup table are saved as:

- `anat/sub-<subject_label>_label-L2018_desc-<scale_label>_atlas.graphml`

- `anat/sub-<subject_label>_label-L2018_desc-<scale_label>_atlas_FreeSurferColorLUT.txt`

4.4.2 FreeSurfer Derivatives

A FreeSurfer subjects directory is created in `<bids_dataset/derivatives>/freesurfer`.

```
freesurfer/
    fsaverage/
        mri/
        surf/
        ...
    sub-<subject_label>/
        mri/
        surf/
        ...
    ...
```

The `fsaverage` subject distributed with the running version of FreeSurfer is copied into this directory.

4.5 Citing

Important:

- If your are using the Multi-Scale Brain Parcellator in your work, please acknowledge this software and its dependencies. To help you to do so, we recommend you to use, modify to your needs, and include in your work the following text:

  Results included in this manuscript come from the Multi-Scale Brain Parcellator version latest [1], a processing pipeline, written in Python which uses Nipype [2,3]. It is encapsulated in a BIDS app [4] based on Docker [5] and Singularity [6] container technologies. Resampling to isotropic resolution, Desikan-Killiany brain parcellation [7], brainstem parcellation [8], and hippocampal subfields segmentation [9] were performed using FreeSurfer 6.0.1. Final parcellations were created by performing cortical brain parcellation on at 5 different scales [10], probabilistic atlas-based segmentation of the thalamic nuclei [11],and combination of all segmented structures, using in-house CMTK tools and the antsRegistrationSyNQuick tool of ANTS v2.2.0 [12].

  1.Tourbier S, Aleman-Gomez Y, Griffa A, Hagmann P (2019, January 10) sebastientour-bier/multiscalebrainparcellator: Multi-Scale Brain Parcellator (Version v1.0.0-beta8). Zenodo. http://doi.org/10.5281/zenodo.2536778

2.Gorgolewski K, Burns CD, Madison C, Clark D, Halchenko YO, Waskom ML, Ghosh SS (2011). Nipype: a flexible, lightweight and extensible neuroimaging data processing framework in python. Front Neuroinform, vol. 5, no. 13. doi:10.3389/fninf.2011.00013.

3.Gorgolewski KJ, Esteban O, Ellis DG, Notter MP, Ziegler E, Johnson H, Hamalainen C, Yvernault B, Burns C, Manhães-Savio A, Jarecka D, Markiewicz CJ, Salo T, Clark D, Waskom M, Wong J, Modat M, Dewey BE, Clark MG, Dayan M, Loney F, Madison C, Gramfort A, Keshavan A, Berleant S, Pinsard B, Goncalves M, Clark D, Cipollini B, Varoquaux G, Wassermann D, Rokem A, Halchenko YO, Forbes J, Moloney B, Malone IB, Hanke M, Mordom D, Buchanan C, Pauli WM, Huntenburg JM, Horea C, Schwartz Y, Tungaraza R, Iqbal S, Kleesiek J, Sikka S, Frohlich C, Kent J, Perez-Guevara M, Watanabe A, Welch D, Cumba C, Ginsburg D, Eshaghi A, Kastman E, Bougacha S, Blair R, Acland B, Gillman A, Schaefer A, Nichols BN, Giavasis S, Erickson D, Correa C, Ghayoor A, Küttner R, Haselgrove C, Zhou D, Craddock RC, Haehn D, Lampe L, Millman J, Lai J, Renfro M, Liu S, Stadler J, Glatard T, Kahn AE, Kong X-Z, Triplett W, Park A, McDermottroe C, Hallquist M, Poldrack R, Perkins LN, Noel M, Gerhard S, Salvatore J, Mertz F, Broderick W, Inati S, Hinds O, Brett M, Durnez J, Tambini A, Rothmei S, Andberg SK, Cooper G, Marina A, Mattfeld A, Urchs S, Sharp P, Matsubara K, Geisler D, Cheung B, Floren A, Nickson T, Pannetier N, Weinstein A, Dubois M, Arias J, Tarbert C, Schlamp K, Jordan K, Liem F, Saase V, Harms R, Khanuja R, Podranski K, Flandin G, Papadopoulos Orfanos D, Schwabacher I, McNamee D, Falkiewicz M, Pellman J, Linkersdörfer J, Varada J, Pérez-García F, Davison A, Shachnev D, Ghosh S (2017). Nipype: a flexible, lightweight and extensible neuroimaging data processing framework in Python. doi:10.5281/zenodo.581704.

4.Gorgolewski KJ, Alfaro-Almagro F, Auer T, Bellec P, Capota M, Chakravarty MM, Churchill NW, Cohen AL, Craddock RC, Devenyi GA, Eklund A, Esteban O, Flandin G, Ghosh SS, Guntupalli JS, Jenkinson M, Keshavan A, Kiar G, Liem F, Raamana PR, Raffelt D, Steele CJ, Quirion P, Smith RE, Strother SC, Varoquaux G, Wang Y, Yarkoni T, Poldrack RA (2017). BIDS apps: Improving ease of use, accessibility, and reproducibility of neuroimaging data analysis methods. PLOS Computational Biology, vol.13, no. 3, e1005209. doi:10.1371/journal.pcbi.1005209.

5.Merkel D (2014). Docker: lightweight Linux containers for consistent development and deployment. Linux Journal, vol. 2014, no. 239. https://dl.acm.org/citation.cfm?id=2600239.2600241

6.Kurtzer GM, Sochat V, Bauer MW (2017). Singularity: Scientific containers for mobility of compute. PLoS ONE, vol. 12, no. 5, e0177459. doi: 10.1371/journal.pone.0177459

7.Desikan RS, Ségonne F, Fischl B, Quinn BT, Dickerson BC, Blacker D, Buckner RL, Dale AM, Maguire RP, Hyman BT, Albert MS, Killiany RJ. An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. NeuroImage, vol. 31, no. 3, pp. 968-980. doi:10.1016/j.neuroimage.2006.01.021.

8.Iglesias JE, Van Leemput K, Bhatt P, Casillas C, Dutt S, Schuff N, Truran-Sacrey D, Boxer A, Fischl B (2015). Bayesian segmentation of brainstem structures in MRI. Neuroimage, vol. 113, pp. 184-195. doi: 10.1016/j.neuroimage.2015.02.065.

9.Iglesias JE, Augustinack JC, Nguyen K, Player CM, Player A, Wright M, Roy N, Frosch MP, McKee AC, Wald LL, Fischl B, Van Leemput K (2015). A computational atlas of the hippocampal formation using ex vivo, ultra-high resolution MRI: Application to adaptive segmentation of in vivo MRI. Neuroimage, vol. 115, July, pp. 117-137. doi: 10.1016/j.neuroimage.2015.04.042.

10.Cammoun L, Gigandet X, Meskaldji D, Thiran JP, Sporns O, Do KQ, Maeder P, Meuli RA, Hagmann P (2012). Mapping the human connectome at multiple scales with diffusion spectrum MRI. Journal of neuroscience methods, vol. 203, no. 2, pp. 386-397. doi: 10.1016/j.jneumeth.2011.09.031.

11.Najdenovska E, Alemán-Gómez Y, Battistella G, Descoteaux M, Hagmann P, Jacquemont S, Maeder P, Thiran JP, Fornari E, Bach Cuadra M (2018). In-vivo probabilistic atlas of human thalamic nuclei based on diffusion- weighted magnetic resonance imaging. Scientific Data, vol. 5, no. 180270. doi: 10.1038/sdata.2018.270

12.Avants BB, Epstein CL, Grossman M, Gee JC (2008). Symmetric diffeomorphic image registration with

cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. Medical Image Analysis, vol. 12, no. 1, pp. 26–41. doi:10.1016/j.media.2007.06.004.