Contents

An introduction to the custom preprocessing pipeline built for surfaced-based analysis

1. Motivation – Why surface?

2. Overall scheme – Covering multiple softwares (e.g., FSL, AFNI, ANTs, Freesurfer)

3. Pipeline details – Comparison with existing preprocessing pipelines

4. Practical applications – Not limited to surface analysis

Overall scheme of this pipeline

A) Make directories for preprocessed data (r1)

B) Convert DICOM images to NIFTI format (using dicm2nii.m, which was adapted from https://github.com/xiangruili/dicm2nii) (r2-r4)

C) Basic environmental setup for preprocessing (s1)

D) Preprocessing structural data

For surface-based preprocessing:

1. Use recon-all (Freesurfer) to correct bias-field, extract brain tissue, reconstruct structural images to cortical surface, and do anatomical segmentation (s2)

2. Use ciftify_recon_all (CIFTIFY, Dickie et al., 2019) to normalize structural images to MNI space, and resample native surface images onto Conte69 164k and 32k CIFTI surface (Van Essen et al., 2012) using MNI normalization parameters (s3)

For Volume-based preprocessing:

1. Use antsBrainExtraction (ANTs) to correct bias-field and extract brain tissue (s4)

2. Use antsRegistrationSyN (ANTs) to normalize structural images to MNI space (s4)

3. Use FAST (FSL) to do anatomical segmentation (s4)

https://github.com/cocoanlab/surface_preprocessing

2. Overall scheme

E) Preprocessing functional data

1. Use 3dTshift (AFNI) to do slice-timing correction (s5, if needed)
2. Use 3dvolreg (AFNI) to do motion-correction (s6)
3. Use topup/applytopup (FSL) to do distortion-correction (s7)
4. Use flirt (FSL) with BBR cost function to co-register functional images to structural images (s8)
5. For surface-based preprocessing, additionally use bbregster (Freesurfer) to refine co-registration (s8)
6. Use ICA-AROMA (Pruim et al., 2015) to remove motion-related signals (s9)
7. Use 3dTproject (AFNI) to remove nuisance signals (s10)
8. Use applywarp (FSL, for surface-based preprocessing) or antsApplyTransforms (ANTs, for volume-based preprocessing) to normalize functional images to MNI space
9. For volume-based preprocessing, use susan (FSL) to spatially smooth functional images
10. For surface-based preprocessing, use ciftify_subject_fmri (CIFTIFY, Dickie et al., 2019) to transform functional images to Conte69 32k CIFTI surface, and spatially smooth (based on both surface-based and volume-based smoothing) functional images

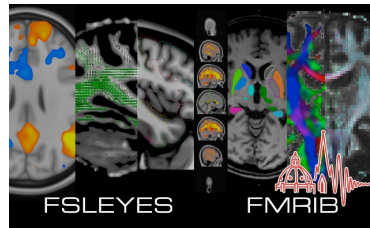Note: 8 can precede 6 and 7. depending on your purpose.

https://github.com/cocoanlab/surface_preprocessing
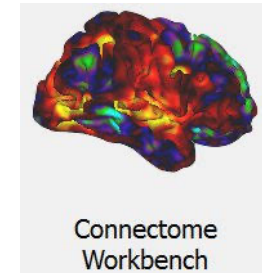
Super useful!!!

Provide nice functions
for some steps of preprocessing
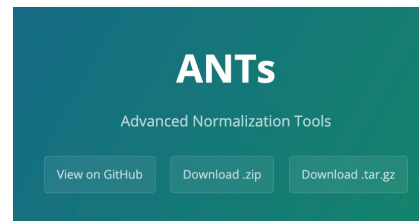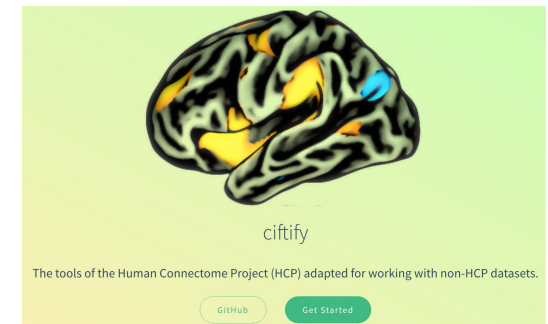

Specialized tool for analysis
based on CIFTI space

Best tool for automatic cortical segmentation
and surface-based analysis

Very good at MNI normalization

A savior for legacy MRI data – CIFTI-fy!

https://docs.google.com/document/d/1wjh2fWaBKGc2XKo2alKkbHGLzOIGgxyBImCeL7VLI0o/edit

Surface Preprocessing Pipeline
Software Installation Manual

April 2019, COCOAN Lab

Introduction

This manual is written to guide those who wish to use the 'surface preprocessing' pipeline of Cocoan Lab. To use this new preprocessing pipeline, many softwares such as AFNI, ANTs, freesurfer, Connectome Workbench, ICA-AROMA, and CIFTIFY are needed to be installed in your workstation. Installing these softwares is not easy and takes many steps and efforts. Therefore, this manual aims to help the installation of the softwares, plus further information about running the code and minor issues to save your time and energy from troubleshooting!

https://github.com/cocoanlab/surface_preprocessing

Computational Cognitive Affective Neuroscience Lab (Cocoan Lab) https://cocoanlab.github.io

You can also use either 1) fMRIprep only or 2) fMRIprep + ciftify!

nature methods ARTICLES
https://doi.org/10.1038/s41592-018-0235-4

fMRIPrep: a robust preprocessing pipeline for functional MRI

Oscar Esteban [1]*, Christopher J. Markiewicz [1], Ross W. Blair[1], Craig A. Moodie [1], A. Ilkay Isik [2], Asier Erramuzpe [3], James D. Kent[4], Mathias Goncalves[5], Elizabeth DuPre [6], Madeleine Snyder[7], Hiroyuki Oya[8], Satrajit S. Ghosh [5,9], Jessey Wright[1], Joke Durnez [1], Russell A. Poldrack[1,10] and Krzysztof J. Gorgolewski [1,10]*

Contents lists available at ScienceDirect

NeuroImage

journal homepage: www.elsevier.com/locate/neuroimage

ELSEVIER

Ciftify: A framework for surface-based analysis of legacy MR acquisitions

Erin W. Dickie [a,*], Alan Anticevic [b,c,d,e], Dawn E. Smith [a], Timothy S. Coalson [f], Mathuvanthi Manogaran [a], Navona Calarco [a], Joseph D. Viviano [a], Matthew F. Glasser [f,g], David C. Van Essen [f], Aristotle N. Voineskos [a,h]

[a] Kimel Family Translational Imaging-Genetics Laboratory, Campbell Family Mental Health Research Institute, Centre of Addiction and Mental Health, Toronto Canada
[b] Department of Psychiatry, Yale University School of Medicine, New Haven, CT, USA
[c] Division of Neurocognition, Neurocomputation, & Neurogenetics (N3), Yale University School of Medicine, New Haven, CT, USA
[d] Interdepartmental Neuroscience Program, Yale University, New Haven, CT, USA
[e] Department of Psychology, Yale University, New Haven, CT, USA
[f] Departments of Radiology and Neuroscience, Washington University School of Medicine, St Louis, USA
[g] St. Luke's Hospital, Chesterfield, MO, USA
[h] Department of Psychiatry, University of Toronto, Toronto, Canada

HCP Grayordinates

If CIFTI output is enabled, the motion-corrected functional timeseries (in T1w space) is first sampled to the high resolution 164k vertex (per hemisphere) `fsaverage`. Following that, the resampled timeseries is sampled to *HCP Pipelines*'s `fsLR` mesh (with the left and right hemisphere aligned) using Connectome Workbench's `--metric-resample` to generate a surface timeseries for each hemisphere. These surfaces are then combined with corresponding volumetric timeseries to create a CIFTI2 file.

https://fmriprep.org/en/stable/index.html

fmriprep_ciftify_BIDS-app

Runs a combination of fmriprep and ciftify pipelines from BIDS specification

Usage

```
run.py <bids_dir> <output_dir> <analysis_level> [options]
```

https://edickie.github.io/ciftify/#/02_bids-app