

Report from 2015 OHBM Hackathon (HI)

# Wrapping FreeSurfer 6 for use in High-performance Computing Environments

Project URL: https://github.com/hjmjohnson

Hans J. Johnson

## 1 Introduction

The purpose of this project is to wrap the recon-all workflow from FreeSurfer [1] 6.0 beta in a python processing framework for neuroimaging data called Nipvpe [2]. FreesSurfer, an image analysis suite, performs cortical reconstruction and volumetric segmentation. FreeSurfer's recon-all workflow provides researchers a simple interface for automatically processing of MRI data. Currently recon-all makes commandline program calls to other FreeSurfer processes from a tesh script. Nipype offers a framework in python for making command-line program calls that will allow for faster processing in a high-performance computing environment. This will allow high-performance computing environments to execute steps of the workflow that are independent of each other to be run in parallel. The wrapping of the workflow in Nipype will also allow for easier manipulation of the individual steps in the workflow.

# 2 Approach

FreeSurfer's recon-all tcsh workflow was used to process a set of MRI images. Recon-all then output a list of commands used to process the MRI data. The output commands were placed in a Nipype workflow. If a command did not have a pre-existing Nipype interface, then one was created. In order to ensure that the two workflows were analogous the Nipype and the tcsh workflow were run on the same set of MRI images and the outputs were compared. The comparisons were carried out with SimpleITK for image files and VTK for surface files.

Correspondence: hans-johnson@uiowa.edu

Carver College of Medicine, The University of Iowa, Iowa City, 200 Hawkins Drive, 52242, Iowa, USA

Full list of author information is available at the end of the article

# 3 Results

All output images and surfaces from FreeSurfer's recon-all were identical to those of the Nipype workflow. The stats files were also shown to be equivalent. Less significant files, such as log files, were either not thoroughly compared or were not output by the Nipype workflow.

### 4 Conclusions

The created Nipype workflow was shown to be analogous to pre-existing FreeSurfer tcsh workflow. With further development, Nipype could serve as a complete replacement to the tcsh workflow. This would allow for easier development of the workflow by researchers. Thorough testing on the processing times of the workflows in a high performance environment has not yet been evaluated, but it is hypothesized that the Nipype workflow will perform much faster. The collaborations at the 2015 OHBM BrainHack meeting were instrumental in accomplishing this task. Collaborations with the FreeSurfer, Nipype, and Human Connectome teams allowed members of this project to quickly identify problems and avoid unnecessary failures

#### Availability of Supporting Data

More information about this project can be found at: https://github.com/hjmjohnson. Further data and files supporting this project are hosted in the *GigaScience* repository REFXXX.

#### Competing interests

None

#### Author's contributions

HJJ performed the project and wrote the report

## Acknowledgements

The authors would like to thank the organizers and attendees of the 2015 OHBM Hackathon.

Page 2 of 2 Johnson

# References

- Fischl, B.: FreeSurfer. Neuroimage 62(2), 774–781 (2012)
  Gorgolewski, K., Burns, C.D., Madison, C., Clark, D., Halchenko, Y.O., Waskom, M.L., Ghosh, S.S.: Nipype: a flexible, lightweight and extensible neuroimaging data processing framework in python. Front Neuroinform **5**, 13 (2011)