

# Table of Contents

## I Mouse Brain Images

<b>I.0</b>	<b>Overview</b>	<b>9</b>
<b>I.1</b>	<b>Mouse MRI</b>	<b>11</b>
I.1.1	<b>Getting Started</b>	<b>11</b>
I.1.2	<b>Image Processing Prior to Atlasing</b>	<b>12</b>
A.	Reorienting image files	12
B.	Resampling image files	12
C.	Skull stripping	12
B.	Bias field correction	14
<b>I.2</b>	<b>Probabilistic Atlasing</b>	<b>15</b>
I.2.1	<b>Using Pipeline for Data Registration</b>	<b>16</b>
A.	Alignlinear	16
B.	Align_warp	17
C.	Registering multiple files	18
<b>I.3</b>	<b>Segmenting Brain Areas</b>	<b>19</b>
<b>I.4</b>	<b>Data Analyses</b>	<b>21</b>
A.	Volume analysis	21
B.	Degree of warping	21



# 2 Mouse Stained Slices

## 2.1 Converting Image Files to an Appropriate Format **23**

- 2.1.1 Convert to Analyze Image Format **23**
  - A. JPG, PNG, tiff, etc. 23
  - B. Resample images at a lower resolution 24
  - C. Orient images appropriately 24

### 2.1.2 Masking the Images **25**

### 2.1.3 Making the Images Uniform **25**

## 2.2 Registering the 2D Images \_\_\_\_\_ **27**

## 2.3 Warp to Fit Standardized Atlas \_\_\_\_\_ **29**



# 3 Human MRI

- 3.1 Overview of Data Processing and Analysis 32**
- 3.2 General Preprocessing Steps for Cortical and Subcortical Analyses \_\_\_\_\_ 37**
  - 3.2.1 Anonymize Subject **37**
  - 3.2.2 Registering Images to Another Atlas or Image (Registration) **37**
    - A Automatic 39
    - B Manual 42
- 3.3 Steps for Subcortical Processing and Analysis \_\_\_\_\_ 44**
  - 3.3.1 Subcortical Structure Delineation **45**
  - 3.3.2 Subcortical Analysis **48**
    - A Area 48
    - B Volume 49
    - C Creating Maps of the Structures and Analyses of these structures. 50
- 3.4 General Preprocessing Steps for Cortical Analyses \_\_\_\_\_ 51**
  - 3.4.1 Creating a Brain Mask **52**
    - A General guide for generating a mask using BrainSuite2 52
    - B General guide for generating a mask using Display 52
    - C Creating a mask that removes the skull 53
    - D. Creating masks that divide hemispheres 57
    - E. Additional Notes on Delineating Structures using masks 58
  - 3.4.2 Applying a Mask **59**
    - A. Using masks in BrainSuite2 59
    - B. Using mincmask 59
- 3.5 Processing Steps For Cortical Surface Extraction And Sulcal Analysis \_\_\_\_\_ 61**
  - 3.5.1 Cortical Surface Extraction: **62**



	A. Using the Pipeline to create a surface model	62
	B. Using BrainSuite2	66
<b>3.5.2</b>	<b>Drawing Sulcal Lines</b>	<b>67</b>
	A. Overview	67
	B. General guidelines for using BrainSuite2	69
	C. General guidelines for using Display	70
<b>3.5.3</b>	<b>Warping, Flatmapping and Reinflation</b>	<b>72</b>
	A. Lateral	73
<b>3.6</b>	<b>Separation of Gray Matter, White Matter, and CSF (Segmentation)</b>	<b>75</b>
	A. Automatic with PVC	76
	B. Manual	77
<b>3.7</b>	<b>Cortical Analyses</b>	<b>83</b>
<b>3.7.1</b>	<b>Volume of Gray Matter, White Matter, and CSF</b>	<b>83</b>
	A. Calculating Volumes from Segmented Images	83
<b>3.7.2</b>	<b>Cortical Surface Analyses</b>	<b>85</b>
	A. Sulcal Analysis (gyral pattern variability)	86
	B. Complexity of Cortical Surface	88
	C. Local brain size	89
	D. Gray matter density	90
	E. Maps of cortical thickness	91
<b>3.7.3</b>	<b>Analyzing 4D UCFs</b>	<b>93</b>
	A. Calculating Group Average Cortical Thickness/Density Files	93
	B. Calculating Group Variance of Cortical Thickness/Density Files	93
	C. Calculating Percentage Difference Between Two Group Average UCFs	93
	D. Covariates and P-maps	94
	E. Permutation Tests	95



# LONI HANDBOOK



# LONI HANDBOOK

## 0 Table of Contents



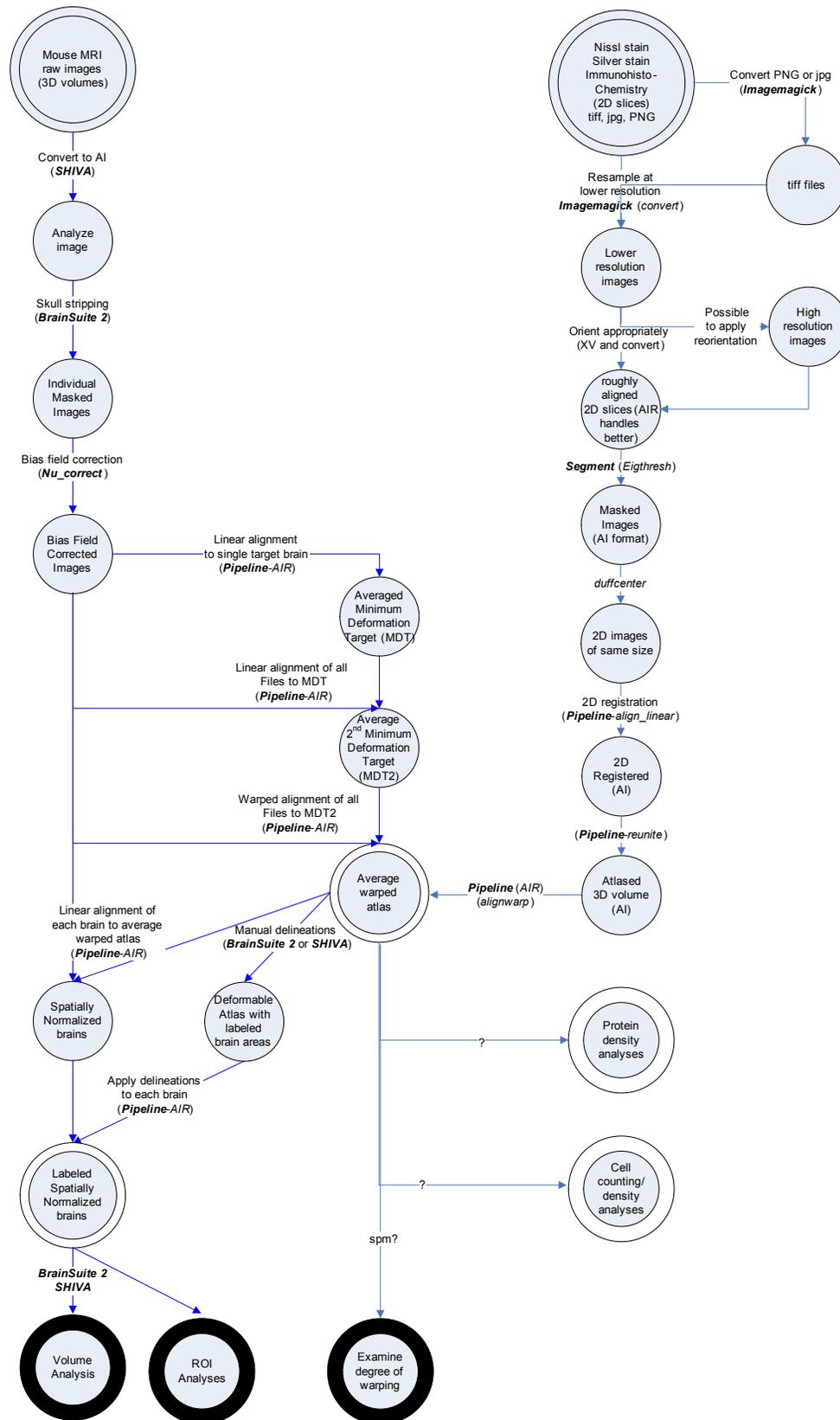
# I Mouse Brain Images

## I.0 Overview

Different modalities of images from the mouse brain can be processed and registered to a similar space (atlasing the image data), which allows an investigator to examine, using multiple approaches to a singular problem. The processing steps discussed in this tutorial have the goal of putting mouse MRI images into a single atlas space. The figure below shows the steps needed to convert MRI image into data that can be atlased and analyzed for volumetric information.



# LONI HANDBOOK





## I.1 Mouse MRI

### I.1.1 Getting Started

Before processing MRI data, you need to make sure your data are in place and the proper tools are installed on your system (this list is for the PC).

Install the following software tools:

#### **BrainSuite 2 (for Windows only)**

- Download this software at [http://www.loni.ucla.edu/Software/Software\\_Detail.jsp?software\\_id=19](http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=19)
- Download and install this software into your preferred directory
- This will create an application named BrainSuite 2 on your PC

#### **Java**

- Both Pipeline and SHIVA require JAVA version 1.4.2 or greater
- Download this software from <http://java.com/en/> and install on your computer

#### **Pipeline Version 3**

- Download this software at <http://www.loni.ucla.edu/twiki/bin/view/MouseBIRN/MouseBIRNTutorials>
- The LONI Pipeline is distributed as a single Java jar program, pipeline.jar.
- Create the directory, C:\mouseBIRN\Pipeline and place the jar file in that directory
- To run the program, double click the pipeline.jar file.
- If you wish to have the program on your desktop, create a shortcut that points to \mouseBIRN\Pipeline\pipeline.jar file
- If running the program in Client/Server mode, you are allowed access to LONI modules, pipelets, and pipelines on the LONI server. However, for this mode you will need a LONI Pipeline account, to get this account apply at [http://www.loni.ucla.edu/NCRR/Application/Collaborator\\_Application.jsp](http://www.loni.ucla.edu/NCRR/Application/Collaborator_Application.jsp)
- For advanced installing options see [http://www.loni.ucla.edu/Software/Installing\\_Detail.jsp?software\\_id=2](http://www.loni.ucla.edu/Software/Installing_Detail.jsp?software_id=2)

#### **SHIVA**

- Download this software at [http://www.loni.ucla.edu/Software/Software\\_Detail.jsp?software\\_id=12](http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=12)
- Create the directory, \mouseBIRN\SHIVA and place the jar file in that directory
- To run the program, double click the SHIVA.jar file.
- If you wish to have the program on your desktop, create a shortcut that points to the \mouseBIRN\SHIVA\shiva.jar file

For hands on practice of this whole processing step, download the data set from the SRB at /BIRN/mouse/Sharing/Tutorials/MRI/AHM session to your computer in \mouseBIRN\data



## I.1.2 Image Processing Prior to Atlasing

The image of the brain should be as clean as possible. Removing the skull from the image by masking and adjusting for spatial variations in the image using a bias field correction both facilitate the process of atlasing the mouse brain.

### A. Reorienting image files

For the registration process to work its best, the scans need to be aligned in the same orientation within 45 degrees of each other. Thus, it may be necessary for you to reorient your data files to the same orientation.

- Open your files in SHIVA
- Open the Reorient Tool from the tool menu
- If necessary flip the axis or axes necessary to reorient your file, make sure all the images are within 45 degrees of each other
- Save them in this configuration

### B. Resampling image files

If data files exceed 500 MB, the files will need to be downsampled to a size of  $512 \times 256 \times 256$  in order for skull stripping to succeed using BrainSuite 2.

- This may be done manually using a command line program called manual reslice
- This may also be done using the program ImageJ

If you wish to use the higher image resolution for further data analysis, you will need to do the following:

- Run most of your analyses using the lower resolution image data
- Upsample both the image data as well as their masks
- Apply the high resolution masks to their respective images

### C Skull stripping

Masking is the process of selecting a portion/region of the raw MRI scan by creating a binary file that covers the region of interest. Members of this laboratory have found that registration is much more effective with just the brain in MRI images taken of the mouse. This section describes how to use the masking method to strip the skull and leave only an image of the brain.

*Image type:* Analyze Image format

*Required preprocessing steps:* In order to use BrainSuite 2, image data cannot be larger than  $512 \times 256 \times 256$ . If necessary, resample your image files (see section II.A.2.b)

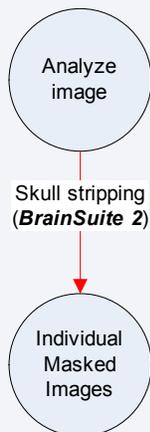
*Suggested preprocessing steps:* None

*General tips before beginning:* A good approximation of a stripped skull can be created semi-automatically using BSE (run within BrainSuite2). You will then want to check and clean up the image volume manually using BrainSuite2.

Keep your masks consistent across brains by defining the edges of your brain before you begin.



*Processing steps:*



1. You must have BrainSuite2 installed on your PC
2. Open the Analyze image data volume within BrainSuite
3. On the BrainSuite2 toolbar, select the BSE option
4. Click Do All. When it is finished, close the BSE window
5. If you wish, you can change some of the values in the BSE window to see if your mask improves
6. This will generate a mask file that will mask out most of the non-neural tissue. You will want to save this mask separate from your original file.
7. Click File, Save, Mask, and name your mask file (original\_name.mask.img)
8. Further editing will require manual manipulation of your mask file, so make sure both the original image and the mask file are open in BrainSuite2
9. Open the Mask tool and click Apply edits to mask. The Mask tool must be kept open while editing the mask, and you may wish to change the brush size frequently, so often it's handy to keep that open as well.
10. Once the skull has been automatically stripped, you have the option of dilating and eroding (either in a diamond or cube format) this mask using the buttons in the Mask tool.
11. Since you will need to go through every slice of the brain, you may choose to edit all the images in one plane and then check them in another.
12. For additional information on manually editing this mask, cleaning a mask.doc on <http://www.loni.ucla.edu/twiki/bin/view/MouseBIRN/Mouse-BIRNTools> for a demonstration of this process.
13. Save your file often as you are editing.

*Other tips or troubleshooting:* None

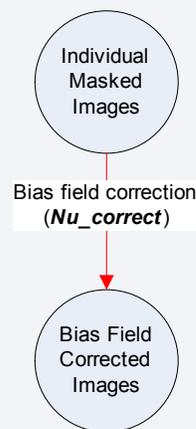
*Specific information:* None



## B. Bias field correction

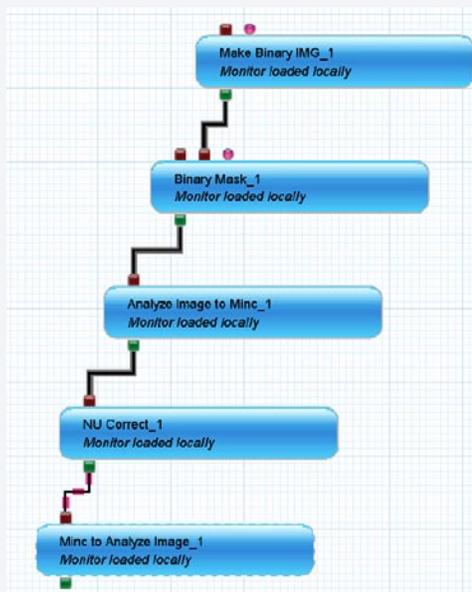
Some MRI images contain spatial variation in the intensity that is due to an artifact. Running a bias field (or RF) correction can compensate for the spatial variation in most MRI files.

The Nu\_correct module in the LONI Pipeline will run a bias field correction on a MINC File format, thus the image must be converted to MINC file format for the correction and then converted back to AI format.



- Processing steps:*
1. BrainSuite 2 creates 8 bit masks that must first be converted to binary masks
  2. When this binary mask is added to the original image file, in the module Binary Mask a single Analyze image volume is created that is the sum of these images
  3. This masked Analyze image volume is sent to the Analyze Image to MINC module to convert it to Minc format
  4. Bias field correction of the image occurs in the module Nu Correct
  5. Finally, this image is converted back into th Analyze image format for further processing

Key Variables: *(Suggestions in parenthesis)*



Make Binary IMG:

Input: 8 bit mask (*input\_file.mask.img*)  
Output: binary image mask (*binary.mask.img*)

Binary Mask:

Input 1: Analyze image volume (*input\_file.img*)  
Input: binary image mask (*binary.mask.img*)  
Output: masked image volume (*masked\_volume.img*)

Analyze Image to MINC:

Input: masked image volume (*masked\_volume.img*)  
Output: MINC image volume (*masked\_volume.mnc*)

NU Correct:

Input: MINC image volume (*masked\_volume.mnc*)  
Output: RF\_corrected\_MINC\_image (*output\_file.mnc*)

MINC to Analyze Image:

Input: MINC image volume (*output\_file.mnc*)  
Output: Analyze image volume (*output\_file.img*)

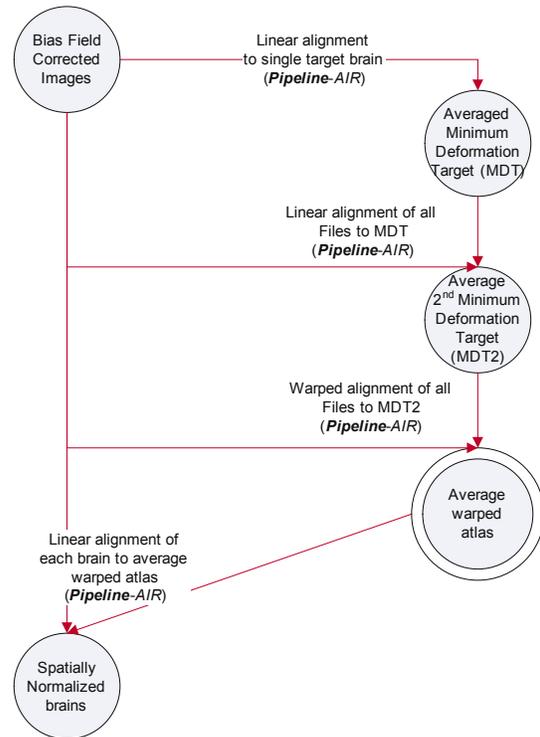


## I.2 Probabilistic Atlasing

There is no single representative brain, nor a simple method of generating an “average” anatomy to represent 3D anatomic variations, let alone variations across strains, genetically manipulated animals, and disease states. Probabilistic atlasing is a research strategy for generating anatomical templates, expert diagnostic systems, and knowledge-based imaging tools that retain quantitative information on intersubject variations in brain architecture. Intensity-based approaches concentrate on generating “average” representations of anatomy by intensity averaging multiple MRI scans.

A fundamental problem for brain-mapping studies that integrate data from many different subjects, is that there are significant anatomic variations in the size, shape, and position of neuroanatomical structures. Registration is the image-processing tool used in brain-mapping research to reduce inter-individual anatomic variance by matching homologous spatial features of a “source brain” to those of a “target brain.”

The figure to the right shows the process used by people in this laboratory for creating average warped atlases and spatially normalized brains from the average atlas.



## 1.2.1 Using Pipeline for Data Registration

Registration is the process of aligning an image to another image. You will be using the LONI Pipeline to call the AIR modules in the AIR package to register mouse MRI images. You may register two images, or you can register a group of images to a single image, or create an average image. This registration process may involve a linear alignment, which scales the size and orientation of the images and places them in the same coordinate frame or a nonlinear alignment, which actually warps the images into a common space.

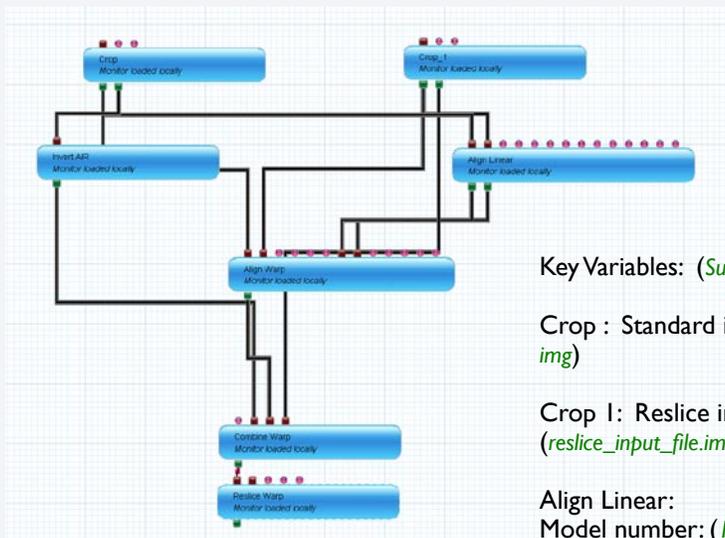
The AIR package includes two programs for automated registration of Analyze image files, **Alignlinear** and **Align\_warp**.

In many instances, the procedure that provides the closest fit for the registration involves first running a linear alignment followed by a nonlinear alignment. It should be noted, that while you can reverse the results of a linear alignment, you cannot do the same for a nonlinear alignment.

### A Alignlinear

- This module includes 2D and 3D variations of all linear spatial transformation models. It generates a .air file that contains the linear transformation parameters that when applied, can be used to resample one of the images to match the other.
- The AIR module `reslice_AIR` must be used after `alignlinear` to apply the transformation parameters. It resamples the reslice file to match the standard file for a linear alignment.
- An example of a Linear alignment of two files using the Pipeline is below

- Processing steps:*
1. Crop your standard and reslice files
  2. Send them to the `alignlinear` module to generate a .air transformation file. The cost function argument should be assigned 3 if you are trying to align similar images, but 1 if they are a different modality.
  3. After the files are aligned, resample the reslice file to the space of the original standard file.
  4. Since the standard file has been cropped, you need to invert the crop transformation to put your standard file back into its original space. This can be accomplished by running the `invert_AIR` module on the standard file, which will create an .air transformation file that holds these parameters.
  5. Run the `combine_AIR` module with the aligned transformation and the “uncropped” standard transformation to generate an .air transformation file that allows for placing the aligned reslice image into the standard space.
  6. Use this .air file with the module `reslice_AIR` on the reslice image file to apply these transformations to the image file and create a new realigned file.
  7. The SHIVA module at the end will allow you to visualize these files, and you can check their alignment, or you can manually open the file within SHIVA or another visualization program to check the registration.



Key Variables: (*Suggestions in parenthesis*)

Crop : Standard input volume (*standard\_input\_file.img*)

Crop 1: Reslice input volume (*reslice\_input\_file.img, or list of files*)

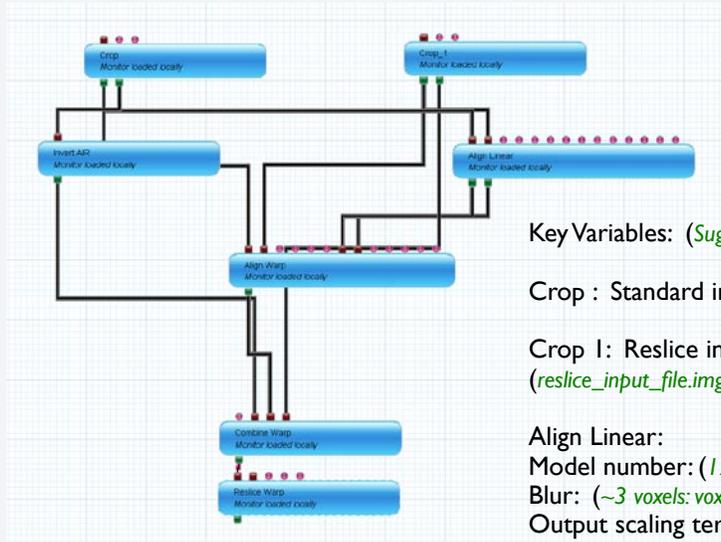
Align Linear:  
Model number: (*12*)  
Blur: (*~3 voxels so for voxel size of 0.3 mm = 1mm*)  
Cost function: (*1 or 3*)

Reslice Air:  
Resliced output volume (*resliced\_output\_file.img*)

## B Align\_warp

- This module includes 2D and 3D variants of nonlinear polynomial spatial transformation models. It generates a .warp file that contains nonlinear transformation parameters that when applied, can be used to resample one of the images to match the other.
- Reslice\_warp must be used after align\_warp to apply the transformation parameters a nonlinear alignment.
- This process is slightly different from the linear alignment of two files, but basically it follows the same idea.
- An example of a nonlinear alignment using the LONI Pipeline is below.

- Processing steps:*
1. Send the cropped files are sent to alignlinear
  2. The parameters from this module are sent to the alignwarp module (.air transformation file and the scaling termination file, which specify the transformation initialization and scaling initialization for the alignwarp module). The alignwarp module uses these parameters as the starting point before beginning a nonlinear alignment of the two volumes.
  3. Again, the idea is to register the reslice volume to the standard volume, so invert\_air reverses the crop and puts the standard volume back into standard space.
  4. Combining the output of alignwarp with the original generates a .warp transformation file that allows for placing the aligned reslice image into the standard space.
  5. Using this .warp file with the module reslicewarp on the reslice image file will now apply these transformations to the image file and creates a new realigned file.
  6. The resulting image files can be sent to SHIVA to visualize the registration.



Key Variables: (Suggestions in parenthesis)

Crop : Standard input volume (*standard\_input\_file.img*)

Crop 1: Reslice input volume (*reslice\_input\_file.img, or list of files*)

Align Linear:

Model number: (12)

Blur: (~3 voxels: voxel size of 0.3 mm = 1 mm)

Output scaling termination file:

Align\_Warp:

Model number: (2 5)

Blur: (~3 voxels: voxel size of 0.3 mm = 1 mm)

Output scaling termination file:

Reslice Warp:

Resliced output volume

(*resliced\_output\_file.img*)

Divide by intensity scale factor:

Model #: 3D windowed sine (5 | | | | |)

## C Registering multiple files

For the registration modules in this tutorial, you need to specify which image will be your standard (your atlas or the image to which you want to align the other file) and which will be the reslice image (the file which is resampled to match the standard). It is also possible to align multiple images to your standard with a single pipeline.

In general, you can use a list of files instead of single file names to run the same process on each file in the list. For aligning multiple file, pass the list file instead of a single reslice input volume to Crop 1 in the Pipelines shown above. To begin, make a text file (name it something that makes sense to you) that holds a list of your .img files with the directory. An example for the PC is shown to the right, however the same concept holds for Linux, UNIX or Mac operating systems. For the PC environment, prefix this directory list with the drive letter, i.e. C:\mBIRN\data\MRI\_001.img.

Contents of "input\_files.list":

C:\mBIRN\data\MRI\_001.img

C:\mBIRN\data\MRI\_002.img

C:\mBIRN\data\MRI\_003.img

C:\mBIRN\data\MRI\_004.img



## I.3 Segmenting Brain Areas

You may wish to segment certain brain areas in order to perform further analyses. This is done by painting labels over your MRI images and saving these labels as a separate, but related file.



*Image type:* Analyze Image format

*Required preprocessing steps:* None

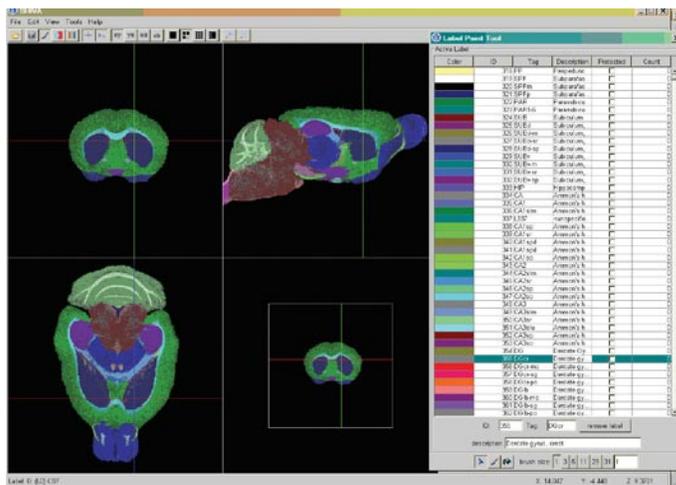
*Suggested preprocessing steps:* Skull Stripping  
Bias field correction  
Registration to an Atlas space

*General tips before beginning:* None

*Processing steps:* In order to segment the brain into separate brain areas, you will need to manually draw the outlines on the data volume. This can be accomplished using BrainSuite2.

1. Open your data volume in SHIVA or BrainSuite2.
2. Open the Label Painter and select the “Edit Labels” box on the label painter.
3. See the BrainSuite2 tutorial for information on how to draw these labels.

Once the areas have been drawn, it is straightforward to count the number of voxels in each label. Knowing the size of the pixels, you can easily compute the volume of this area from this information.



*Other tips or troubleshooting:* None

*Specific information:* None



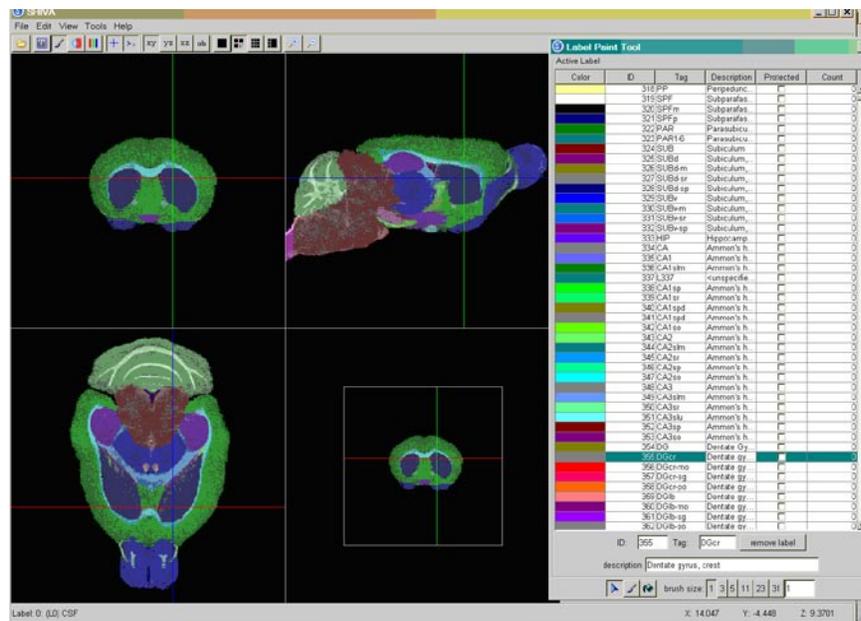


## I.4 Data Analyses

### A. Volume analysis

After you have segmented your data using SHIVA or BrainSuite2, it will automatically tell you how many voxels are in each delimited area. These values can be recorded and compared to the volumes in another atlas. In order to make valid comparisons, the brains should be registered to each other, which in essence should “normalize” the atlases.

1. Open your label volume in SHIVA or BrainSuite2.
2. Open the Label[Painter] tool
3. The number of voxels in each area are listed in the Label Painter tool
4. Multiply each voxel by the spatial resolution of the voxels within your image file



### B. Degree of warping

After you have aligned your data to “fit” a standard atlas, you can examine your transformation files to quantify exactly how much each file needed to be warped to fit the standard.



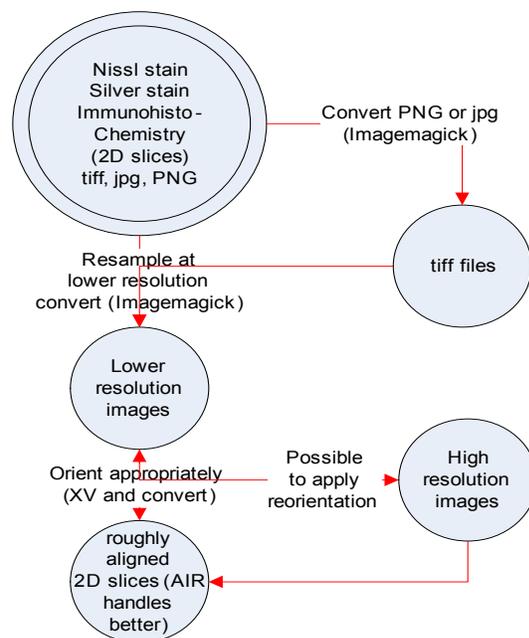
## 2 Mouse Stained Slices

### 2.1 Converting Image Files to an Appropriate Format

A major difference to remember about these files compared to MRI image files, is that they are collected as individual 2D slices rather than as a volume. These individual slices can be aligned to each other to create a 3D volume.

After data files are collected, they must be in the proper format so they can be manipulated with the AIR package. This is the Analyze image file format.

Depending on what your data looks like after data collection, you may need to process it using some of several of the steps listed below.



#### 2.1.1 Convert to Analyze Image Format

##### A. JPG, PNG, tiff, etc.

###### Using SHIVA

SHIVA has file converters that allow you to open an individual slice and save it as Analyze image format.

###### Using ImageMagick

Use the ImageMagick command line utility `convert`, to convert a JPG image to an Analyze image file. One may also use the ImageMagick pipeline as made available through the LONI pipeline library.

```
convert (options) input_file.jpg gray output_file.img
```

If you plan to resize the image, convert can do this simultaneously (see below). Note that when using raw files with this utility (ImageMagick calls them raw gray samples), you may be required to specify the size.

### Command Line

There is a command line utility that can convert tiff images to Analyze image format

```
tiff2duff -i input_file.tiff -o output_file.img [options]
```

A corresponding conversion tool converts analyze image format, tiff, and duff files to raw grey files.

```
duff2grey -i input_file -o output_file [options]
```

## B. Resample images at a lower resolution

Almost all images will be at much higher resolution than is necessary for this process. In fact, if the files are too large, it will immensely slow down the processing of your data. It is recommended that your images be 256 x 256 in the X and Y dimensions.

This can be done using almost any commercially available image processing software, but a simple method employs the ImageMagick utility convert.

```
convert -size 256x256 input_file.img output_file.img
```

To convert file type and size at the same time

```
convert -size 256x256 input_file.tiff output_file.img
```

## C. Orient images appropriately

If the slices are close to the same orientation, the registration of these images can give a better approximation of a 3D volume. Thus, it is suggested that you go through each slice and visually align them as close as possible to the proper orientation. View them using your preferred visualization program (i.e. SHIVA, BrainSuite2, XV, Photoshop), and several of these programs also rotate the image.

### Using SHIVA

SHIVA contains the GEM import tool which allows you to rotate and flip your image.

1. Open SHIVA and open the GEM import tool.
2. Browse for your import file and open it.
3. Select the appropriate slice (axial, sagittal, coronal)
4. Reorient the image file to the orientation shown in the GEM importer
5. Save the image file

### Using ImageMagick

You can rotate them using the ImageMagick command line utility convert with the rotate option. It is likely the file will need the size option, so it knows the size of the raw files.

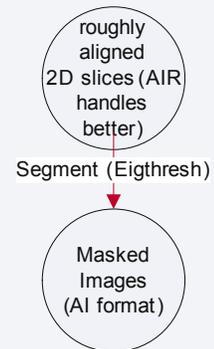
```
convert -rotate degrees_rotate_cw input_file.img output_file.img
```



## 2.1.2 Masking the Images

Once the images have been converted, resampled, and rotated, the background needs to be masked. The background will almost always be clear (or white), but for further processing, it should be black. To create this mask automatically, use the command line program `eigthresh`. From Inire, use the command

```
/nethome/users/shattuck/armiger/bin/eigthresh input_file.img output_file.img
```



## 2.1.3 Making the Images Uniform

After converting, rotating, and resampling your images, they may no longer be the same size and centered. This is important for the registration process, so to do this, use the command line `duffcenter`. On Inire, use the following command

```
/nethome/users/shattuck/armiger/bin/duffcenter input_file.img output_file.img
```





## 2.2 Registering the 2D Images

*Image type:* Analyze image format, see the section on converting files.

*Required preprocessing steps:* Backgrounds masked from white to black, and all files same size

*Suggested preprocessing steps:* Reorientation of files to an approximate align, resampled to lower resolution, bias field corrected if necessary (see the mouse MRI section)

*General tips before beginning:* This process uses the Pipeline environment and many modules from the AIR package to linearly align a set of 2D images to each other. Once these are aligned and united, they can be combined into a single 3D image file.

This process is somewhat like a cross between aligning two files and multiple files as discussed above in the MRI section. Again, you will have a group of data files, but instead of being from several subjects, they will be several slices from a single subject. Each file will refer to the next serial section of brain, so the goal is to align two neighboring slices, then align one to the next serial slice and so on. For this process you will need two list files, offset by one slice. This order allows the serial slices to be aligned in order.

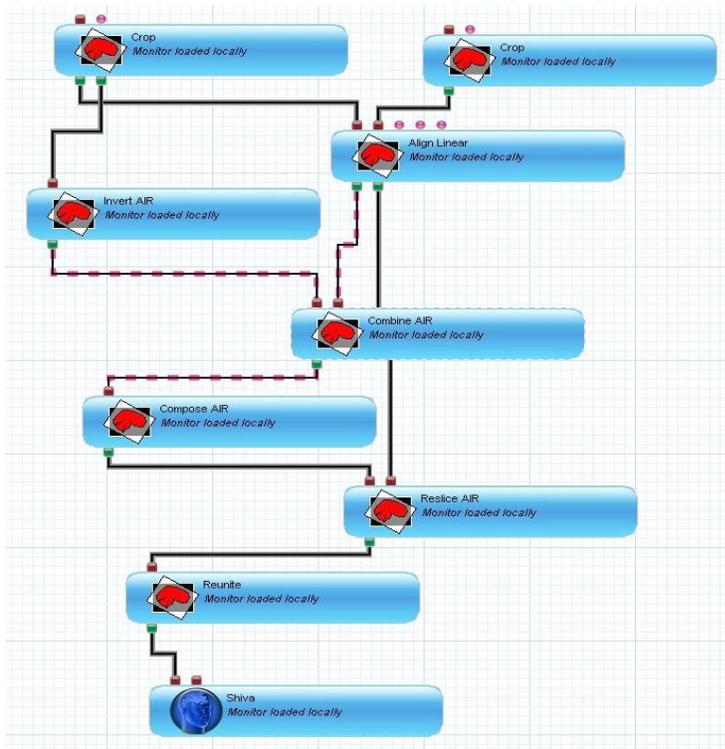
- Processing steps:*
1. To begin with, make two text files (name them something that makes sense to you)
    - a. the first will hold a list of your .img files (you will probably want to include the directory the files are in)
    - b. the second one will have the same list of files offset by one.
    - c. An example is to the right, notice how the first file is repeated in the offset list, and the list ends one short of the input list.

<b>Contents of "input_files.list":</b>	<b>Contents of "offset_files.list":</b>
C:\mBIRN\data\MRI_001.img	C:\mBIRN\data\MRI_001.img
C:\mBIRN\data\MRI_002.img	C:\mBIRN\data\MRI_001.img
C:\mBIRN\data\MRI_003.img	C:\mBIRN\data\MRI_002.img
C:\mBIRN\data\MRI_004.img	C:\mBIRN\data\MRI_003.img

2. Your "standard" input will be the first list file, and the "reslice" input will be the offset list file.
3. Input to the left Crop module your first "standard" list of files and to the other Crop module your "reslice" list of files.
4. These are then sent to the alignlinear module to generate an .air transformation file.
5. Using the compose\_AIR module generates a volume from the individual 2D slices
6. After the files are aligned, you want to put the reslice file in the space as the original standard file.
  - a. However, the standard file has been cropped, so you need to invert the crop transformation to put your standard file back into its original space.
  - b. This can be accomplished by running the invertAIR module on the standard file, which will create an .air transformation file that holds these parameters.
7. Running the combine\_AIR module with the aligned transformation and the "uncropped" standard transformation generates an .air transformation file that allows for placing the aligned reslice image into the standard space.



8. Using this .air file with the module `reslice_AIR` on the reslice image file will now apply these transformations to the image file and create a new realigned file.
  - a. Note that `reslice_AIR` is sent an intensity scaling parameter file from the `alignlinear` module, which can be combined with a spatial transformation initialization file to restart the algorithm at the same location in parameter space where it left off.
  - b. `Reslice` can use this information to create a final image that is intensity corrected as well as spatially corrected.
  - c. In addition, the scaling parameter can be used as an intensity normalization factor for subsequent statistical analysis of the registered data.
9. The `reunite` module combines the resliced images into a 3D volume file. And running the `SHIVA` module at the end allows you to visualize this 3D volume file.



Key Variables: *(Suggestions in parenthesis)*

Crop 1: List of input volumes (*input\_file.list*)

Crop 2: List of offset volumes (*offset\_file.list*)

Align Linear:

Model number: *(23)*

Blur: *(0.1)*

Output scaling termination file: ✓

Reslice Air:

Divide by intensity scale factor: ✓

Reunite:

Assembled output volume  
(*assembled\_output\_file.img*)



## 2.3 Warp to Fit Standardized Atlas

Once you have aligned your 2D slices into a “3D volume,” you can align this volume to an existing 3D volume or atlas. For how to do this, see the section for aligning two files in the mouse MRI section.





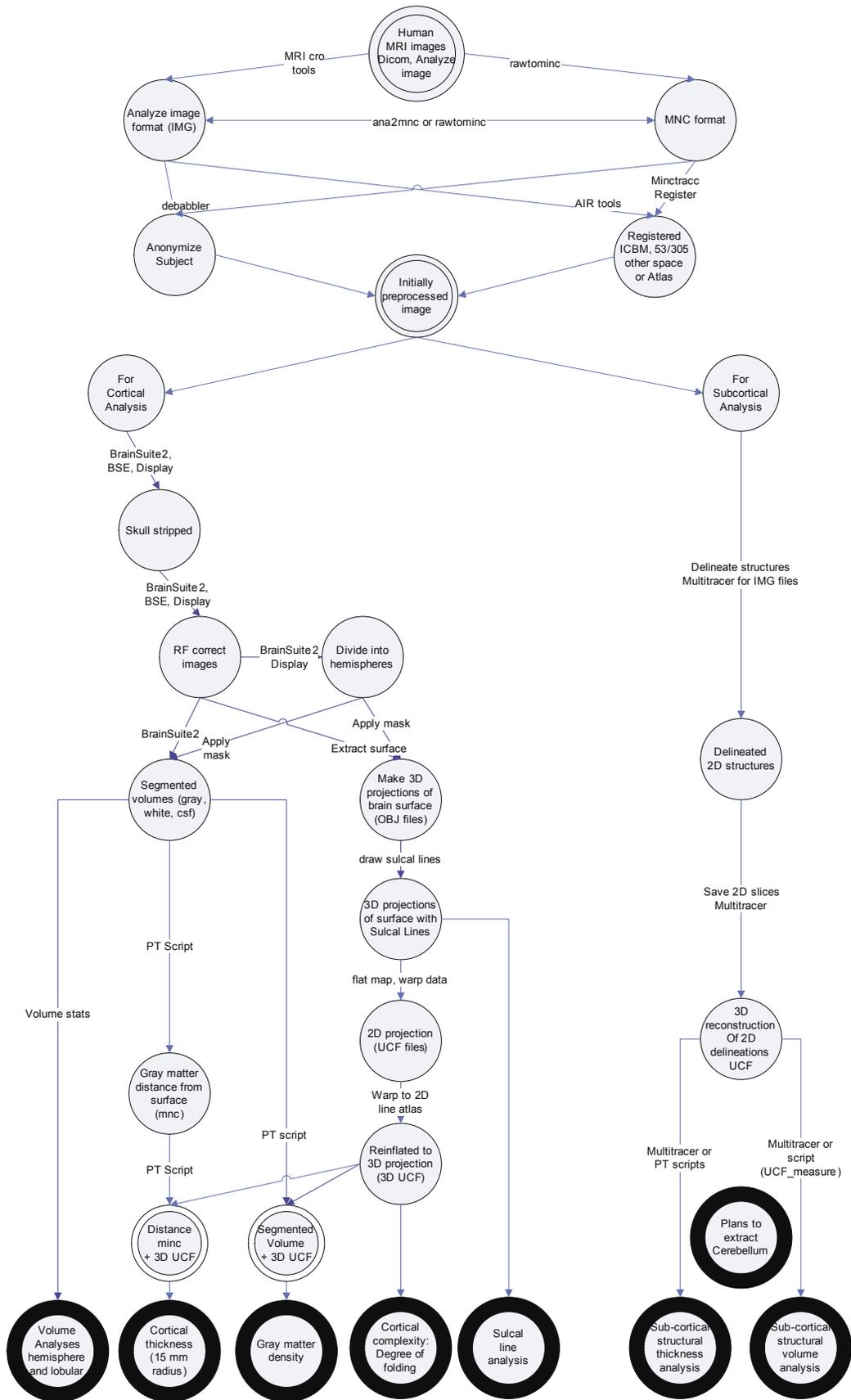
# 3 Human MRI

## 3.1 Overview of Data Processing and Analysis

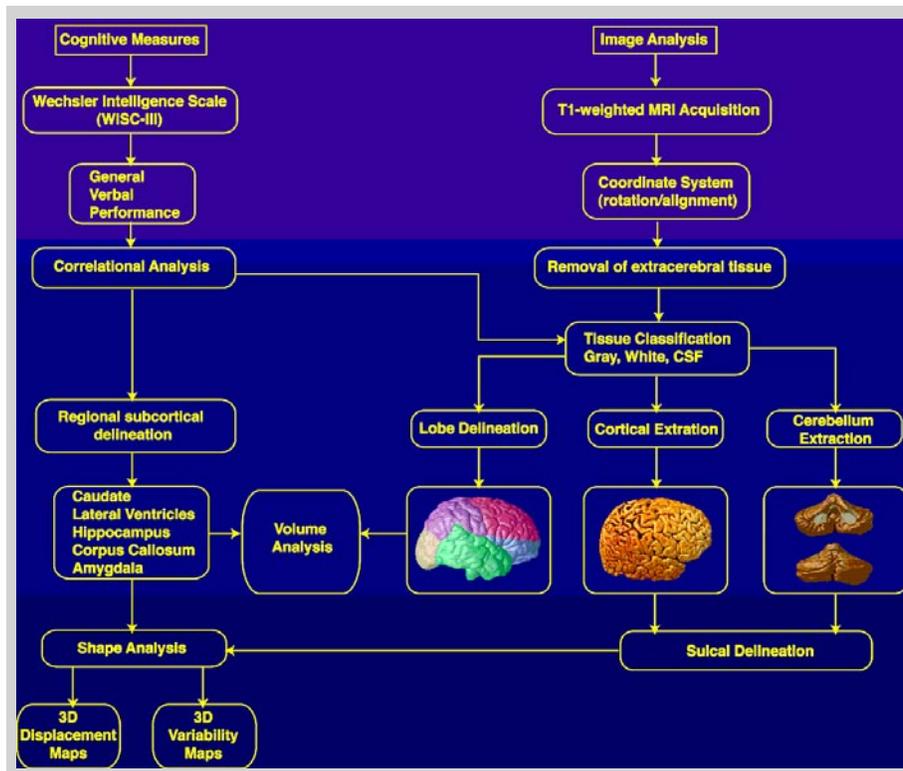
The best approach to using this manual is to first decide on the type of analysis that is of interest to you. Once determined, simply find and follow the pre- and processing steps suggested. Note however, that most of the analyses require many of the same basic pre-processing steps.

The figure below illustrates the flow of data processing for most of the analyses done at LONI, where the circles represent the “state” of the data, and the arrows represent the processes necessary to move from one state to the next, where the dark outlined circles represent an “end analysis” state for the data. It is worth noting that this is a simplified version of the steps required, yet covers all of the essentials.

Currently, analysis is carried out on either Analyze image format or Nifti format (with LONI offering software, both the LONI Debabeler, and the LONI pipeline, which allow conversion between most commonly used formats). However, it is worth noting that this laboratory is moving towards using the Analyze image format in all analysis, and developing tools to replace ones available to nifti files.



The figure below gives an example of how an experiment may drive the processing and analysis of MRI images and how the image data may be related to the behavioral information that has been collected. Here the results of cognitive tests and general verbal tests are correlated to anatomical measures of volume, size, and shape in both subcortical and cortical structures.

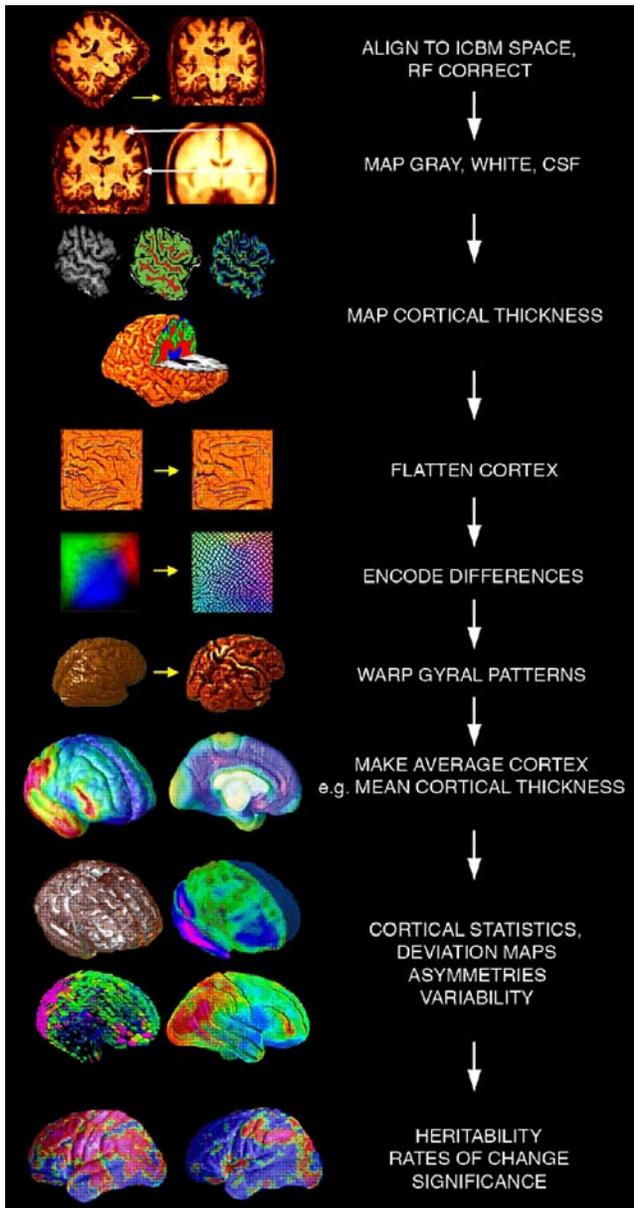


For instance if you are studying the effect of the expression of the ApoE 4 versus ApoE 3 genes and their effect on gray matter density and their potential linkage to Alzheimer's, here are a few points to consider:

**Experiment:** Subjects are chosen that fit one of these two groups (ApoE 3 or ApoE 4), scored on behavioral cognitive tests for Alzheimer's symptoms, and given a T1 weighted MRI scan.

**Goal:** To find if expressing ApoE 4 vs. ApoE 3 genes relates to the gray matter density and the severity of Alzheimer's cognitive deficits.

While the behavioral tests can be scored and instantly quantified, a great deal more would be required to have the MRI scans into a state where one can examine and quantify gray matter density of the cortex and its correlation to the disease. Once this has been accomplished, statistical tests can be undertaken to examine gray matter differences across the groups of normals versus the diseased. This figure illustrates in more detail the steps involved.



The figure to the left shows many of these steps.

Preprocessing steps:

- Anonymize subject
- Register to ICBM space
- Skull stripping
- RF correction
- Tissue Segmentation (grey matter, white matter, and CSF)
- Split into hemispheres (if interested in studying the medial as well as lateral surface)
- 3D cortical extractions
- Drawing of the sulcal lines
- Flat mapping
- Warping to template (average of all subjects)
- Reinflate to 3D cortical UCF mesh

Analysis steps:

- Determine gray matter density from the MRI of each subject
- Create an average of gray matter density of each group
- P maps (significance map, using any covariants of interest. i.e. age, gene, gender, behavioral cognitive scores)
- Permutations (testing whether P maps are truly significant)

In this analysis, the fundamental question is how expression of ApoE 4 vs. ApoE 3 relates to differences in gray matter density. Looking for these effects by examining significance maps (comparing the differences of GMD density between ApoE4 and ApoE3 groups) is one of the final analyses. Additionally, although analysis is described for cortical gray matter density above, the steps are the same as those for studying cortical thickness, with a few minor changes in the latter portion of the analysis.

As with any analysis, there exist several approaches to undertaking. Some may be completed wholly automatically or semi-automatically by programs or scripts that have been written for that specific task, but some instances call for much more manual processing. The processing steps can be completed with one of the software packages built for MRI data analysis (i.e. BrainSuite2, Display, MRlcro), or with scripts built for the UNIX environment, or a tool built to automate much of the analysis, the LONI Pipeline.



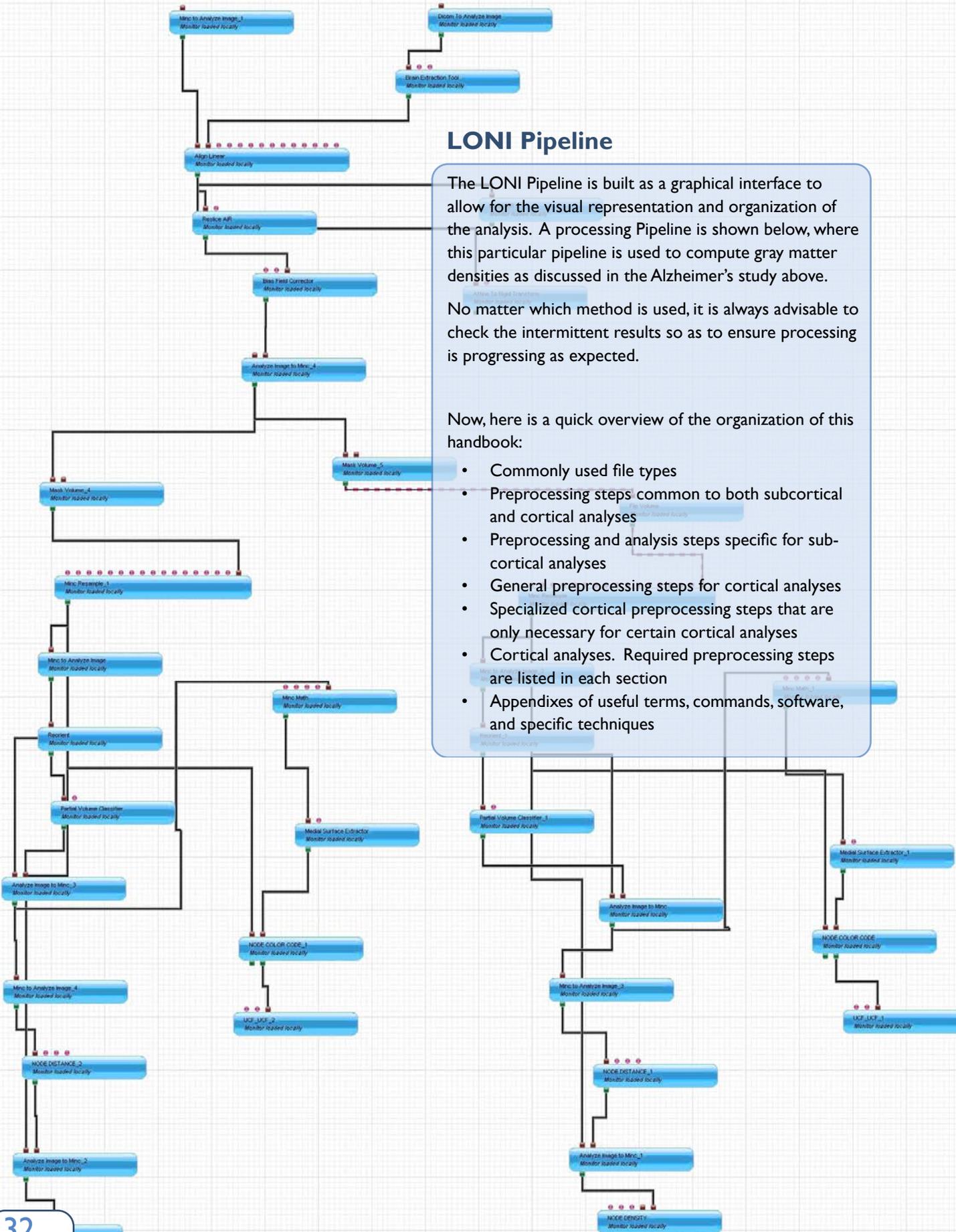
## LONI Pipeline

The LONI Pipeline is built as a graphical interface to allow for the visual representation and organization of the analysis. A processing Pipeline is shown below, where this particular pipeline is used to compute gray matter densities as discussed in the Alzheimer's study above.

No matter which method is used, it is always advisable to check the intermittent results so as to ensure processing is progressing as expected.

Now, here is a quick overview of the organization of this handbook:

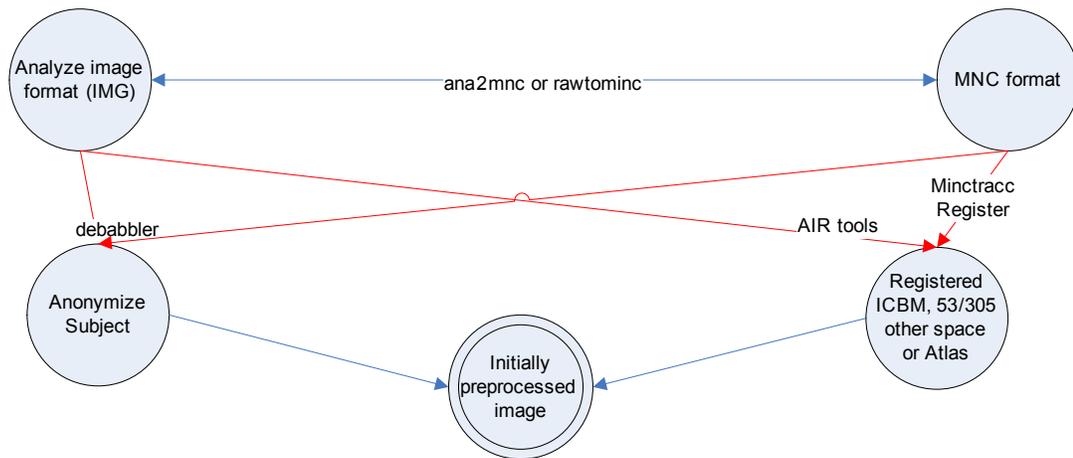
- Commonly used file types
- Preprocessing steps common to both subcortical and cortical analyses
- Preprocessing and analysis steps specific for subcortical analyses
- General preprocessing steps for cortical analyses
- Specialized cortical preprocessing steps that are only necessary for certain cortical analyses
- Cortical analyses. Required preprocessing steps are listed in each section
- Appendixes of useful terms, commands, software, and specific techniques





## 3.2 General Preprocessing Steps for Cortical and Subcortical Analyses

The following preprocessing steps are necessary for all or nearly all of the analyses that you may wish to perform on your data. The figure shows an overview of most of these steps. For some of them, the order in which the steps are performed may actually affect the results. For suggestions about this order, see the individual processing steps in the section below.



### 3.2.1 Anonymize Subject

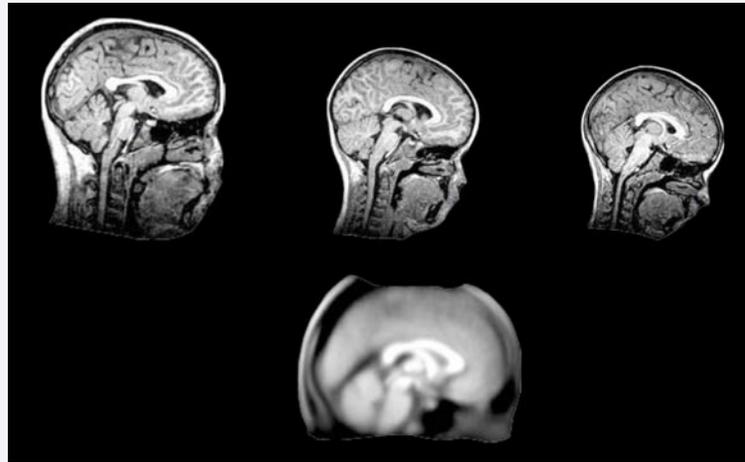
The LONI tool Debabbler has a debablet that can be used to anonymize or deidentify the subject. Download the applet at [http://services.loni.ucla.edu/debablet/LONI\\_Debablets.html](http://services.loni.ucla.edu/debablet/LONI_Debablets.html). You can select the input files and input a directory and patient ID for the deidentified files, then click run. Your anonymized file will be placed in the specified directory with the new patient ID.

### 3.2.2 Registering Images to Another Atlas or Image (Registration)

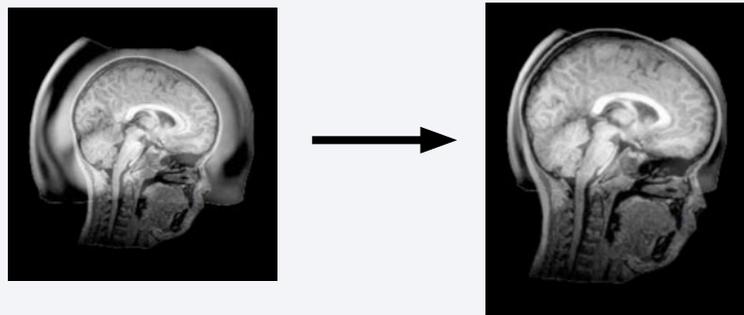
Registration is the process of positioning a subject's brain into a defined standard/average/atlas space. An atlas represents some brain (or average of brains) that has been set as a standard. Spatial normalization allows us to easily make comparisons across subjects, as it ensures constant position and space. Adapted from Eric Kan's movie, this process is shown below.

Each individual image is spatially registered to the average brain, both in size and in orientation. After this process, all scans are in the standardized space.

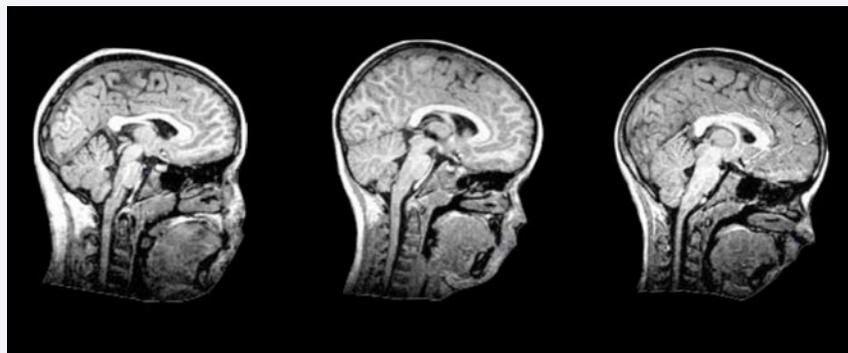
Whether or not registration is performed before skull stripping seems to be a matter of preference. When using Pipeline tools, having the brain skull stripped before this step seems to give better results (see section D for notes on skull-stripping), but when using minctracc, the preference seems to be registering the images before stripping the skull.



*Figure of 3 individual images relative to a standard average*



*An individual image is rotated and scaled to fit the average*



*The result is all 3 subjects are the same size and in the same orientation as the average*



## A Automatic

Most commonly, files are registered using the LONI Pipeline environment, which provides access to multiple suites, many of which include registration tools, including AIR and FSL, as well as MNI's minctracc.

### Using analyze image file format and AIR

*Image Type* Analyze image format

*Required preprocessing steps*

*Suggested preprocessing steps* **Skull stripping**

For AIR to work optimally, the scans need to be in the same coordinate frame and orientation as the atlas. The images should be no more than 45 degrees off from the desired atlas.

You may also use the programs SHIVA or BrainSuite2 to examine your Analyze image files and flip your data files into the proper orientation if necessary using the tool REORIENT, which is a part of AIR.

If you would prefer to run a script on several files, it might be easier to use the AIR module reorient to quickly change all the files into the proper orientation.

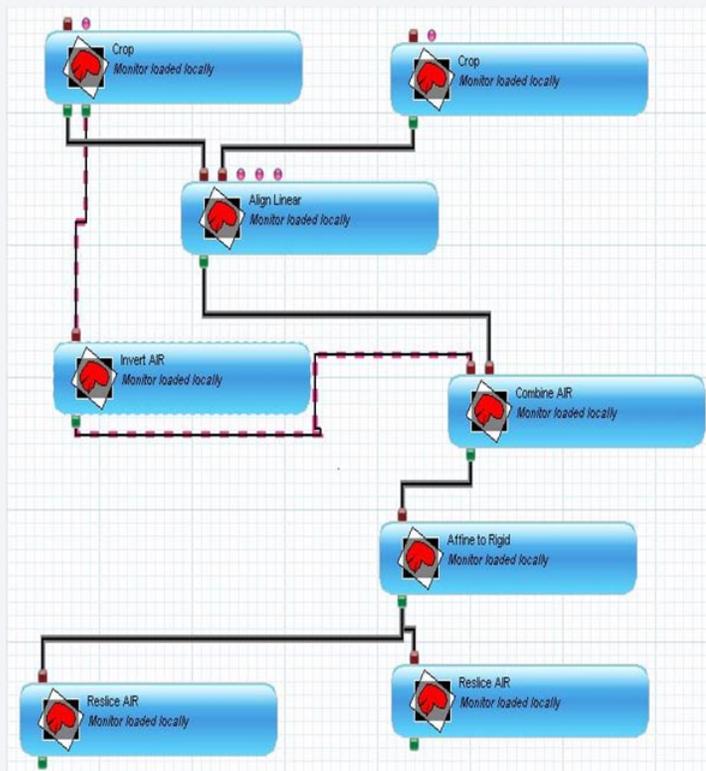
*reorient AI\_ input\_filename.img AI\_output\_filename.img (options)*

*General tips before beginning* For more information about registration of Analyze image files using AIR tools see the mouse section on registering images to another image.

- Processing steps*
1. You will create a .air file, a transformation matrix, which is needed to register the skull stripped file to an atlas. Include the path of the files. Also, if you want to run AIR using a different number of parameters (use 12 to start), change the values after the -m flag.
  2. As input, you can process either a single image file, or a list of image files.
  3. Crop your standard and reslice files
  4. 4. Send them to the alignlinear module to generate a .air transformation file. Each individual image is linearly registered to the standard file using this module.
  5. After the files are aligned, you want to resample the reslice file to the space of the original standard file.
    - a. However, the standard file has been cropped, so you need to invert the crop transformation to put your standard file back into its original space.
    - b. This can be accomplished by running the invertAIR module on the standard file, which will create an .air transformation file that holds these parameters.

- Running the `combine_AIR` module with the aligned transformation and the “uncropped” standard transformation generates an `.air` transformation file that allows for placing the aligned reslice image into the standard space.
- Use this `.air` file with the module `reslice_AIR` on the reslice image file and applying these transformations to the image file creates a new realigned file.

Check the registration of the file to the Atlas using a software visualization program (Register for minc files or SHIVA for either analyze image or minc)



Key Variables: *(Suggestions in italics)*

**Crop 1 (left):**

Input: *standard\_input\_file.img*  
Output: *crop\_standard\_input\_file.img* and *crop\_standard\_transform.air*

**Crop 2 (right):**

Input: *reslice\_input\_file.img* (or list)  
Output: *crop\_reslice\_input\_file.img* and *crop\_reslice\_transform.air*

**Align Linear:**

Input: *crop\_standard\_input\_file.img* and *crop\_reslice\_input\_file.img*  
Output transform: *alignlinear\_transform.air*  
Model number: (12)  
Blur: (?)

**Invert Air:**

Input: *crop\_standard\_transform.air*  
Output: *invert\_crop\_standard\_transform.air*

**Combine Air:**

Input: *invert\_crop\_standard\_transform.air*, *alignlinear\_transform.air*, and *crop\_reslice\_transform.air*  
Output: *combined\_transform.air*

**Affine to Rigid:**

Input: *combined\_transform.air*  
Output: *total\_rigid\_transform.air*

**Reslice Air:**

Resliced output volume  
(*resliced\_output\_file.img*)

**Using the Minc file format and Minc tools***Image type:* Minc*Required preprocessing steps:* None*Suggested preprocessing steps:* None*General tips before beginning:* Have the brain oriented in the same direction as the target brain*Processing steps:* Generate an xfm (transformation matrix file) using minctracc  
`minctracc [options] <input_raw.mnc> <average_target.mnc> <to_target_space.xfm>`

Transform the native space mnc to the defined standard space using the above xfm.

`mincresample <input_raw.mnc> <output_target_space.mnc> -like <average_target.mnc> -transformation <to_target_space.xfm>`

While much faster, automatic registration does not always accurately register the MR data to the target space. To check if the brain has been properly registered, use the program “register” to compare the standard atlas with the newly registered brain (output\_target\_space.mnc):

`register <average_target.mnc> <output_target_space.mnc>`

To determine if the test brain has in fact been aligned accurately to its target space, inspect the third window/column of the “register” program. This window superimposes the average atlas and the brain you are registering. Scroll through the three views (using the left/right arrow keys) to ensure that the atlas and the newly registered brain more or less line up with one another. That is, make sure that the lobes and ventricles come in and out of view together. If there is any asymmetry, the brain has not been well aligned to its target space, and you may either retry automatic registration by editing some of the options in minctracc, or you may opt to register the brain manually.

*Other tips or troubleshooting:* There are many options that can be used with minctracc. Two options that are always good to use are `-lsq9` (or `-lsq#` depending on desired parameterization) and `-mi` (for mutual information):`minctracc -mi -lsq9 <input_raw.mnc> <average_target.mnc> <to_target_space.xfm>`In the event that the brain does not register well using these options, other good flags to try are `-est_center`, `-est_translations`, and `-est_rotations`. Note that when using these additional flags, you should still always apply the `-lsq#` and `-mi` options



## B Manual

*Image type:* Minc

*Required preprocessing steps:* None

*Suggested preprocessing steps:* None

*General tips before beginning:* Have the brain oriented in the same direction as the target brain

*Processing steps:* Load the raw MRI and average target brain together in the program “register”

`register average_brain.mnc raw_mri.mnc`

If you want to use previously defined tag points

`register average_brain.mnc -tag tag_file.tag raw_mri.mnc`

Set the average brain to gray, and adjust the intensity of the raw MRI to a comfortable intensity. The image can be enlarged by holding shift + the middle mouse button, and it can be moved by holding shift + the left mouse button.

Pick the tag points on the raw scan corresponding to the points on the average brain. The suggested tag points are in the table below.

Tag#	Anatomical Name	Slice View	Tips/Landmarks
1	Left Cerebellum	Sagittal	Apex of triangle formed by the transverse sinus (coronal view)
2	Right Cerebellum	Sagittal	Apex of triangle formed by the transverse sinus (coronal view)
3	Anterior-most Point of Corpus Callosum	Sagittal	Most anterior point where corpus callosum curves
4	Posterior-most Point of Corpus Callosum	Sagittal	Most posterior point where corpus callosum curves
5	Left Eye Socket	Axial	Point where the diameter of the eye socket bones are the largest (coronal view)
6	Right Eye Socket	Axial	Point where the diameter of the eye socket bones are the largest (coronal view)
7	Fourth Ventricle	Sagittal	Most middle and posterior point (axial view)
8	Left Temporal Lobe	Coronal	Use most anterior point of left temporal lobe
9	Right Temporal Lobe	Coronal	Use most anterior point of right temporal lobe
10	Mamillary Bodies	Sagittal	Point where mamillary bodies and pons are most clearly defined

For more specific information about these tag points, see the Appendix on 10 suggested tag points

Be sure to always pick the points with the “record tags” icon located on the left of the screen. Never use the right mouse button, otherwise the tag points will not stay in order. NOTE: The very center pixel of the cross hair is the actual location of the tag point.

After selecting the tag points, go into the third window, and bring the blend bar all the way to the left. Now scroll through the brain and check: if the temporal lobes come in together, the ventricles come in and leave together, and if the parietal lobes come in together. If there is an asymmetry, check the other lobes and ventricles, and decide if the asymmetry is due to bad tag points, or



actual asymmetries in the brain. Also check to see if the midsagittal line is straight up and down.

Enter the name of the tag file in the top white box, and then press “save tags.”

```
raw_filename_reslice.tag
```

Enter the name of the transform file in the second white box, and then press the blank yellow box to save the file.

```
raw_filename_reslice.xfm
```

The reorientation transform file is now used to create a reoriented file.

```
mincresample raw_filename.mnc raw_filename_reslice_raw.mnc raw_filename_reslice.xfm
```

```
mincresample -short input_file_raw.mnc reslice_raw.mnc -like template.mnc -transformation reslice.xfm
```

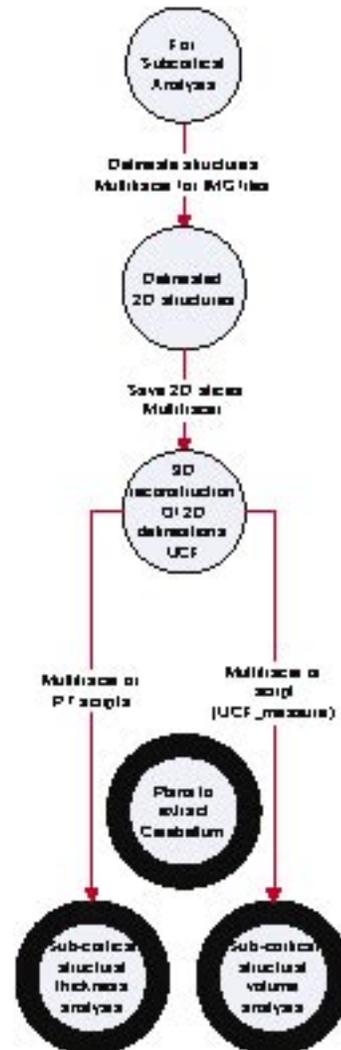
*Other tips or troubleshooting:* None

*Specific information:* None



## 3.3 Steps for Subcortical Processing and Analysis

Generally speaking, the only preprocessing steps necessary for the analysis of subcortical structures other than structure delineation, is anaonymizing the subject, registering the brain to an atlas or other standard, and possibly a bias field correction. These are all done in the manner discussed in their respective sections.





### 3.3.1 Subcortical Structure Delineation

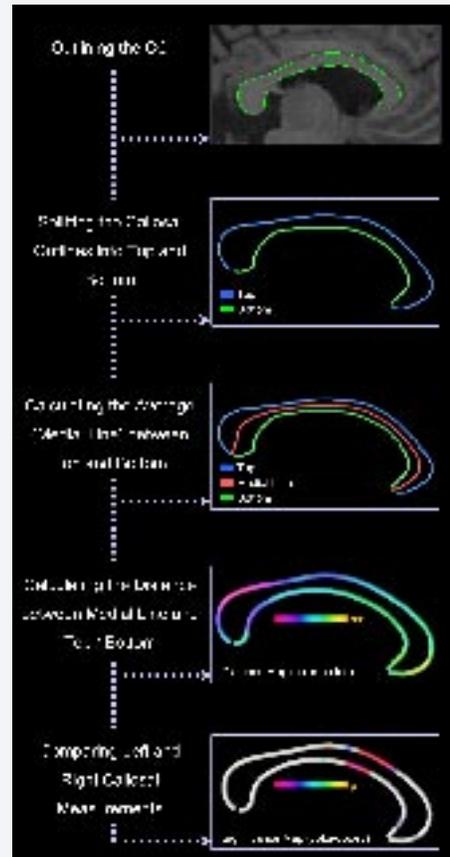
This method can be used to first trace the shape of an individual structure of interest. Once these delineations are done, it is possible to perform an area, length, and/or volume analyses. When these analyses are used in careful population mappings, they can be a powerful method for examining differences that arise in different populations, or to examine changes over time. An example of this method is shown to the right for the Corpus Callosum.

Depending on the aims of the analysis, you may want to examine a structure in a single “slice” through the brain, or repeat and compile this process in many “slices” to give a 3D structural thickness or volume analysis. We will begin with the process of how to delineate a structure in a single slice, and after that discuss how to use this process to create a volume.

A detailed manual explaining how to use the MultiTracer software for this whole process can be downloaded at <http://www.loni.ucla.edu/download/MultiTracer/>, where the program can also be downloaded.

For more information about individual structures, see <http://www.loni.ucla.edu/NCRR/Downloads/Protocols/Segmentation.html>.

General steps involve first outlining the area of interest using the software program MultiTracer. Once this delineation is complete, a series of scripts may be used (the actual one used will vary depending on the structure you are examining) to perform the intended analyses (see section below). We will use the Corpus Callosum (CC) as an example.



*Image type:* Multi-tracer: Minc or Analyze image format.  
Tracer: Analyze image format.

*Required preprocessing steps:* None

*Suggested preprocessing steps:* Bias field correction  
Registration to an average atlas space

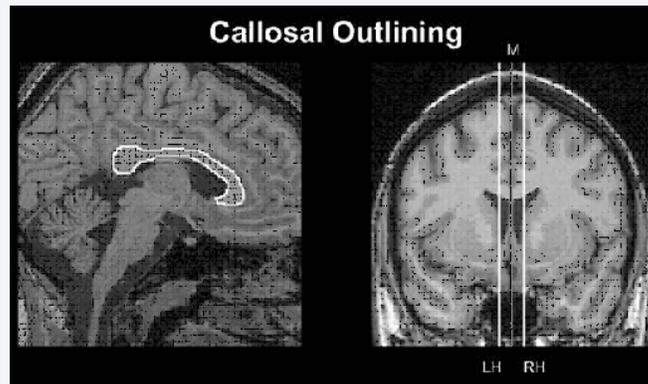
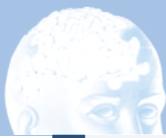
*General tips before beginning:* If you are interested in asymmetry measurements, make sure you understand the orientation of the subject’s head before you start tracing (i.e. which is right and left).

*Processing steps:* Defining a new structure

- Open the file within MultiTracer
- From the “Contours” menu, select “Add Structure”
- Enter the name of the structure. The name you choose will also determine the name under which MultiTracer will save the structure and file name. It will not accept a name if a file with that name already exists. This name can be preceded by a Unix style directory path if you like (e.g., “/data/patients/patient01/left upper hippocampus”).
- Press the “OK” button
- If the structure has been successfully defined, the name that you entered will appear in the control panel in the main window, and the text of the “Trace” button on the control panel will turn red indicating that MultiTracer is now in tracing mode.
- You can repeat the above steps to define additional structures if you like (Structures are only defined for one orientation. For example, if you defined a structure with the images displayed transversely, that structure will not be accessible if you switch to a coronal or sagittal display. If you want to trace the same structure in more than one orientation, you will need to define unique names for the structure in each orientation).

Tracing a defined structure

- Go to a plane where the structure is visible. For our example the CC, this is a midsagittal brain section. Midsagittal brain sections are defined by identifying the interhemispheric fissure in the coronal and sagittal planes and confirmed by the presence of the falx cerebri. However, if you wanted to do an asymmetry analyses, you would choose two planes 6 mm from the midsagittal sections (parasagittal) in the LH and RH (indicated by the white lines in the coronal view in Figure).
- Depending on whether you want traces to be open or closed (i.e., whether the trace should connect back to its starting point), from the “Contours” menu, select the “Style” item and then pick “Open” or “Closed” from the submenu.
- Move the mouse to the anatomically defined starting point of the structure. Press and then release the mouse button to start tracing at this point.
- Move the mouse in the direction defined in your anatomic protocol along the anatomic structure. Don’t worry about small errors; you can fix them later. If you make a grotesque error, you can “shrug off” the bad trace immediately by moving the mouse outside of the main window before you end the trace.
- When the trace is complete, end it by clicking the mouse button again.
- If you want to improve the trace, you can modify it (see the manual for detailed methods for modifying the trace).
- You can proceed to another plane, or change to another structure to trace on the current plane.



#### Saving your delineation

- Even if you have already exported your traces to UCF format, it is often useful to also save the traces. All of your traces will be saved into a single file that can be reloaded later. Unlike the exported UCF's, saved traces can be modified and reviewed using MultiTracer.
- To save your traces
  - From the “File” menu, select “Save Contour Data”
  - In the “Save” dialogue, select the directory where you want the data saved and enter a file name. Press the “Save” button on the dialog

#### Exporting your delineation as UCF'S

- The UCF file format is a format that is used by a variety of surface analysis tools at LONI. MultiTracer does not allow exported UCF's to be reloaded for display or editing, so it is always a good idea to save the traces in addition to exporting them.
- The names of the exported UCF files are determined by the names that were used to define the structures before tracing them.
- To export UCF's
  - From the “File” menu, select “Export Contours as UCF's”
  - The UCF's will be exported to directories that are defined with respect to the directory from which MultiTracer was launched unless you have reset the working directory.
- UCF files can be remapped into other files using the Automated Image Registration (AIR) package.



## 3.3.2 Subcortical Analysis

This section discusses some of the potential analyses you may want to perform on subcortical structures. These include 2D or 3D structural shape analyses.

After the structures of interest have been delineated using MultiTracer (see section on preprocessing subcortical structures), these analyses may be used to examine the shape of the structure of interest. When used in careful population mappings, it can be a powerful method for examining differences that arise in certain populations, or to examine changes over time.

Depending on the aims of the analysis, you may want to examine a structure in a single “slice” through the brain, or repeat and compile this process in many “slices” to give a 3D structure on which to conduct a volume analysis. Note that a thickness measure or analysis may be obtained for both single and multiple slice tracings. Thus, the basic analysis steps are the same for both types of traces. We will begin with the analyses in a single slice and after that discuss the analyses involving many slice tracings.

A very good detailed manual explaining how to use the software (MultiTracer) for this whole process can be downloaded at <http://www.loni.ucla.edu/download/MultiTracer/>, where the software can also be downloaded.

### A Area

On any given slice, the areas and lengths of structures can be calculated.

#### Using MultiTracer to calculate two dimensional areas and lengths

Areas and Lengths can be calculated by doing the following:

- Choose “Analyze 2D Contours on This Plane” from the “Analyze” menu in the main window.
- In the displayed table, the volumes of each valid structure are shown:
  - Area: This is the area of the structure on this plane.
  - Open Length: This is the length of the contour on this plane without including the distance from the final point back to the initial point.
  - Closed Length: This is the length of the contour on this plane including the distance from the final point back to the initial point.
  - X Min: This is the lowest x-coordinate found for the structure on this plane as measured in calibrated real world units with the origin located at the center of the first voxel in the original image

#### Using UCFMeasure

To measure the area (although `ucfmeasure` allows for various gauges) of your 2D delineated structure,

```
ucfmeasure -area 2D_traced_structure.ucf
```

The Corpus Callosum analysis has specialized scripts, which when applied, break the structure into several pieces before analysis. See the Appendix for this information.



## B Volume

General steps include first outlining the area of interest using a software program (most likely MultiTracer). For volume analyses, this step should be repeated through each 2D “slice” that holds the structure of interest.

### Using MultiTracer to calculate volumes

Once structures have been traced, their volumes can be calculated. For the volume of a structure to be defined, it must be traced on more than one plane and the planes on which it is traced must all be contiguous. Volumes can be calculated by doing the following:

- Choose “Analyze Contours” from the “Analyze” menu in the main window.
- In the displayed table, the volumes of each valid structure are shown:
  - Classic Vol: This is the volume calculated by summing the areas for each plane and multiplying by the slice thickness
  - Trap Vol: This is the volume calculated by assuming that the structure extends from the center of the first plane on which it was drawn to the center of the last plane on which it was drawn with areas varying linearly when moving from the center of one plane to the center of the next
  - Frust Vol: This is the volume calculated by assuming that the structure extends from the center of the first plane on which it was drawn to the center of the last plane on which it was drawn with the square root of areas varying linearly when moving from the center of one plane to the center of the next
  - X Min: This is the lowest x-coordinate found for the structure as measured in calibrated real world units with the origin located at the center of the first voxel in the original image
  - Y Min: This is the lowest y-coordinate found for the structure on this plane as measured in calibrated real world units with the origin located at the center of the first voxel in the original image
  - Z Min: This is the lowest z-coordinate found for the structure on this plane as measured in calibrated real world units with the origin located at the center of the first voxel in the original image
  - X Max: This is the highest x-coordinate found for the structure on this plane as measured in calibrated real world units with the origin located at the center of the first voxel in the original image
  - Y Max: This is the highest y-coordinate found for the structure on this plane as measured in calibrated real world units with the origin located at the center of the first voxel in the original image
  - Z Max: This is the highest z-coordinate found for the structure on this plane as measured in calibrated real world units with the origin located at the center of the first voxel in the original image
- Structures that have not been traced on this plane will be listed with zeros for all values.

You can choose “Copy” from the “Edit” menu of the table’s window to copy all of the data in tab delimited format that can be pasted into a spreadsheet.

### Using UCFMeasure

Measure the volume of your 3D delineated structure,

```
ucfmeasure -volume 3D_traced_structure.ucf
```



## C Creating Maps of the Structures and Analyses of these structures.

Although in the next revision of the manual we will expand on this section, for now, we'd like to make available as guide an example, specifically, Corpus Callosum (CC) measurements:

Callosal outlines from midline (or left and right hemisphere) are automatically divided into top and bottom segments. The randomly digitized points making up each callosal surface are then redigitized to render them spatially uniform using surface-based mesh modeling methods. Subsequently, the 2D average (the medial CC line) is calculated from spatially homologous surface points representing the upper (top) and lower (bottom) callosal surface boundaries in each hemisphere. Finally, the distances between each of 100 equidistant surface points making up the medial CC line and 100 equidistant surface points making up the callosal surface boundaries (top and bottom) are calculated. Regions exhibiting significant differences (between groups of interest) are coded in color and mapped onto the average callosal surface model.



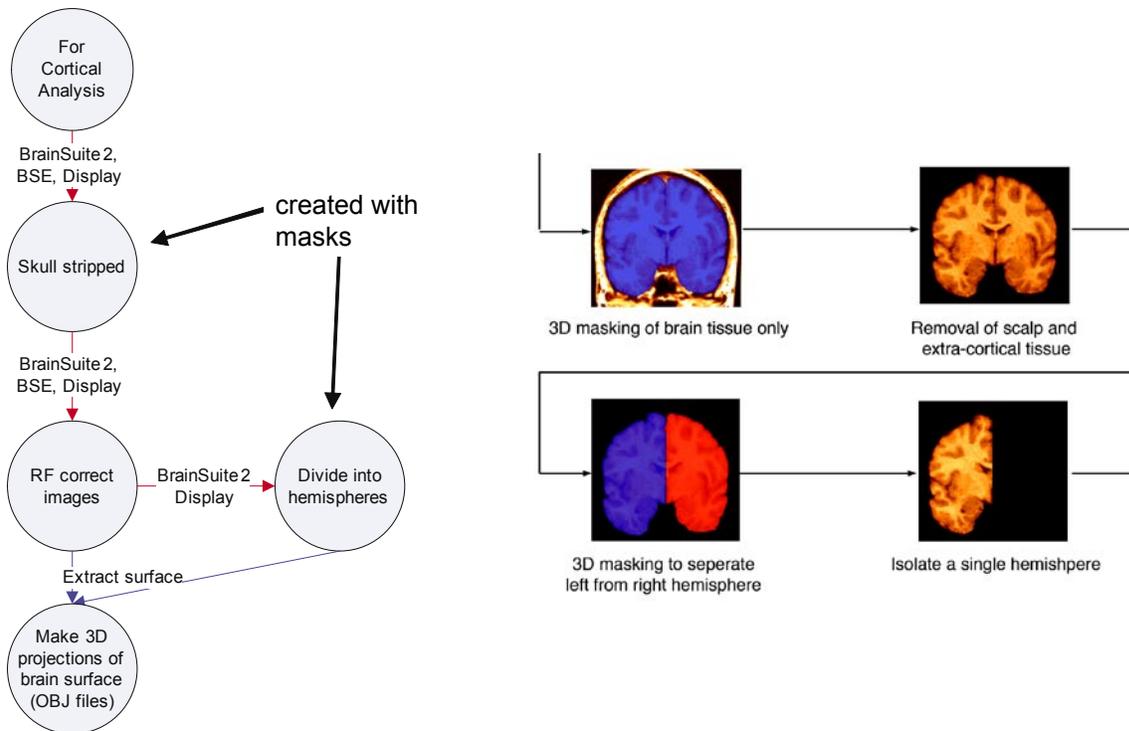
## 3.4 General Preprocessing Steps for Cortical Analyses

Masking is vital to ensuring proper demarcation of brain tissue and/or regions of interest and would be a good investment of time to learn, as it is useful for removing the skull and extra-cortical tissue, as well as for dividing the brain into hemispheres and extracting ROIs.

Division of the brain into hemispheres allows for analysis of the medial surface, which presently represents the most advanced state of cortical analysis at LONI. One may choose to undertake analysis without the study of the medial surface, hence bypassing the hemispheric masking described below. It is worth noting that the hemispheres could be united in the end, providing entire brain analysis, if so desired.

Additionally, as tissue classified files are required for nearly all analyses, you must use a skull-stripped, whole brain file (i.e. not a hemisphere) as an input into such tissue classifying software, in order to ensure consistency of the classification.

The following diagrams depict the general steps:





## 3.4.1 Creating a Brain Mask

Masking is the process of creating a binary file covering the region(s) of the MRI file which are of interest. Generally, a mask is used to remove the skull, to divide the brain into hemispheres, or to isolate/represent an ROI.

Below, we describe two programs that can be used for creating and/or editing a mask. While it is possible to use these programs to create a mask from scratch, it is advisable that when removing the skull, or dividing the brain into hemispheres, you use a program that automatically creates a rough mask (see sections c and d below), and then manually edit using one of the two programs described here.

### A General guide for generating a mask using BrainSuite2

This program utilizes **Analyze image** file format.

One may also use BrainSuite2 to manually generate and/or edit existing masks. The steps for editing a mask in BrainSuite2 are as follows:

1. Open your file using BrainSuite2 (follow i & ii below if mask exists and skip to 7, else skip ii and move to 2)
  - i. Click on File, then Open Volume, find the location and double click on the file (the original volume)
  - ii. Click on File, then Open, then Label Volume. Change the "File of Types" to "Analyze Images" find your file (the mask) and double click.
2. From either the toolbar, or the Tools menu, open the Mask Tool.
3. Select Apply edits to Mask.
4. Select "Auto update" and adjust the "Threshold MRI" value.
5. A new mask will be produced automatically.
6. You may now edit the mask and save it under a new name.
7. To add voxels to the mask, hold down the Control key and left mouse button
8. To erase parts of the mask, hold down the Control key and right mouse button
9. If you make a mistake, you can do undo in the Edit toolbar.
10. You may want to save your mask frequently so you can easily go back to a previous version.

### B General guide for generating a mask using Display

This program opens and saves files in Minc format.

One may use Display to manually generate and/or edit existing masks. The steps for editing a mask in Display are as follows:

1. Open Display, with the original brain minc file and the corresponding mask (should you have one):  
*Display mri.mnc (if no pre-existing mask)*  
*Display mri.mnc -label mri\_mask.mnc (if pre-existing exists)*
2. Part of the Display tool is a "keyboard" (called a pop-up menu) that contains all program options.
3. You may choose to make selections from this pop-up menu directly with your mouse, or by typing the corresponding letters (for each option) on your actual keyboard.
4. A few tips for editing the mask in Display are listed below:
  - Be sure that "Segmenting" (F) has been selected from main pop up menu
  - To add voxels (line): right mouse button
  - To add voxels (fill an area): E

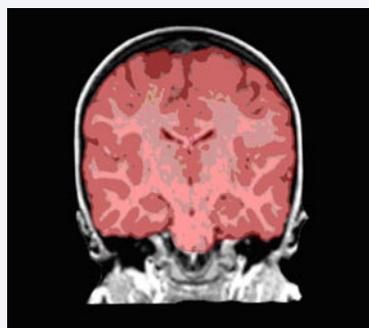


- To remove voxels (line): shift + right mouse button
  - To remove voxels (fill an area): T
  - To undo: 7
  - When drawing a mask, it is possible to temporarily make the labels for a given slice “invisible.” This can be a useful tool when a clearer view of the original volume is needed.
    - From the “Segmenting” page, go to “Show Lables:” (B)
    - This will make the labels disappear.
    - Hitting “Show Lables:” again will make the labels reappear
5. Once you are ready to save your mask:
- Spacebar to get back to main pop up menu
  - Select “File” (T) from main pop up menu
  - Be sure that “Crop Save Lbls” (I) are OFF
  - Select “Save Labels.mnc” (W) and enter a filename in terminal window
  - Note: file will be saved in the directory in which the Display program was opened (unless another path is otherwise designated while saving)

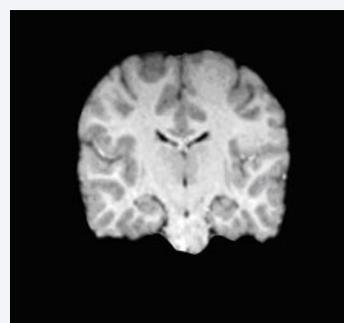
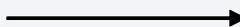
## C Creating a mask that removes the skull

Several reasons exist for removing the skull, including accurate 3D surface representation of the cortex, and improved spatial alignment when using programs such as AIR, where skull stripped data results in superior registration. A mask is required to create a skull-stripped image. A “native” mask may be created, using programs such as BSE and BET, but frequently, such masks need touching up to ensure all brain regions are included and that other non-brain tissue is excluded.

Below we illustrate a mask, and the final product produced as a result of applying the mask to the MR data.



Original 2D slice with mask overlay



Non-brain “stripped” from image

### Automatic Skull Stripping using BSE

BSE (Brain Surface Extractor), a tool built specifically for extracting the brain from MR data, provides a good approximation of the brain, and is used to create an initial skull stripped file or mask to be touched up and used for skull stripping. More information about BSE can be found at <http://neuroimage.usc.edu/BSE/>

**Running BSE from your PC using Brainsuite2**

BrainSuite2 is useful for skull stripping because BSE will be run automatically within the program and then you can manually edit the file.

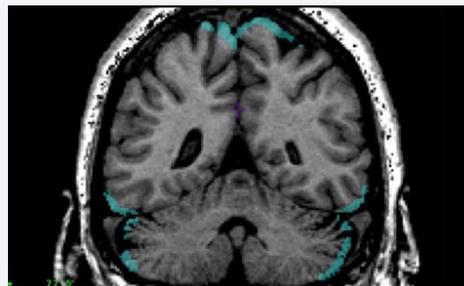
*Image type:* Analyze image (others, to some extent. See documentation on BrainSuite2)

*Required preprocessing steps:* None

*Suggested preprocessing steps:* None

*General tips before beginning:* None

- Processing steps:*
1. Open your non-skull-stripped file using BrainSuite2 (Click on File, then Open Volume, find the location and double click on the file)
  2. Click on BrainSuite, then on Skull Stripping (BSE) to open the BSE window.
  3. You may exercise the “Do All” option to run the whole skull stripping routine, or step through it by clicking the Next button.
  4. If a poor mask is created, set/change some defaults in the BSE window, and retry.
    - Altering the anisotropic filter and the edge detector kernel value, will result in better skull stripped data. For more information about the effects of changing these values, consult the BrainSuite2 Manual
  5. Save your edited file as a mask file. Click on File, then Save to Mask, and enter a file name (filename.mask.img)
  6. Use the visualization program of your choice (Display for Minc format, BrainSuite2 for Analyze image format) to check and manually edit the skull-stripped results. Generally, the dura and superior sagittal sinus cause the most problems. Thus, you will want to pay particular attention to these areas when editing your mask





## Running BSE in the Pipeline

Brain Surface Extractor (BSE) module exists in the LONI Pipeline BrainSuite2 package. As with all pipeline modules, the directory and file name must be specified within the module. To learn more, see the help section on Pipeline.

*Image type:* Analyze image format

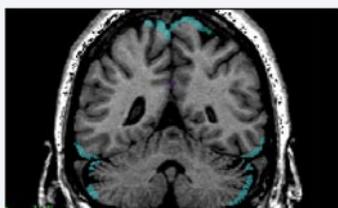
*Required preprocessing steps:* None

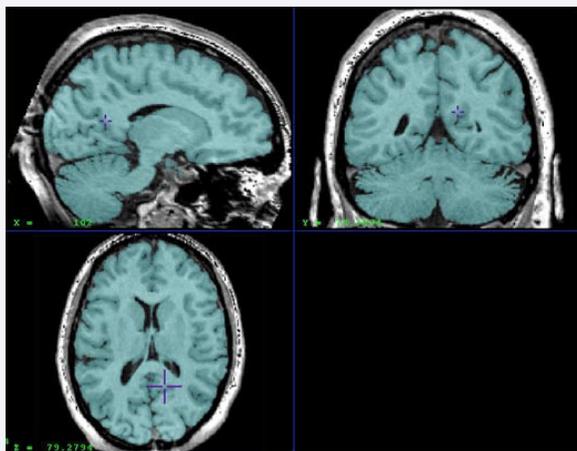
*Suggested preprocessing steps:* None

*General tips before beginning:* As noted in the previous section, BSE possesses several options (anisotropic filter and edge detector kernel values) that may need changing based on your data to effectively increase the precision of the output mask file. Thus, it is recommended that you try several cases on your PC first using the Brainsuite2 GUI, to find ideal settings. These values can then be entered into the node in the Pipeline environment.

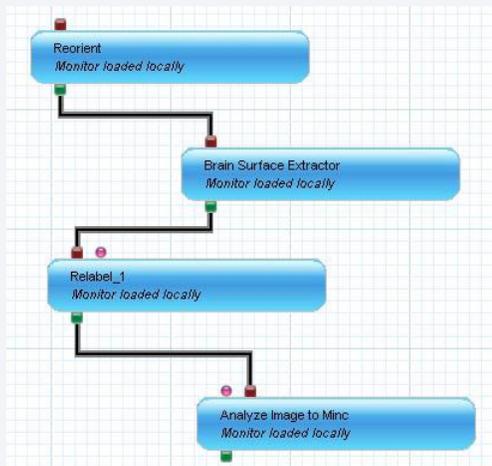
The pipeline below includes a conversion of the output mask to Minc format. This is useful for individuals who plan to further edit their mask using the program Display. However, if you are using BrainSuite2 for further imaging processing, you can do without the Analyze to Minc module and use the image and mask in Analyze image format.

- Processing steps:*
1. As input, you can use a single image file, or a list of image files.
  2. These images are put into the same orientation or a standard orientation to make later analyses easier, using the reorient module. This module also ensures that certain meta-data fields are set in the analyze file's header, which BSE requires.
  3. After reorienting the file, the reoriented image is sent to the BSE module for binary mask generation used for skull stripping, which results in a native mask, meaning, it may need to be touched up to ensure quality of extraction.
  4. To make the mask image easier to use, "relabel" module converts the value of the mask to 1 from its convention of 255.
  5. The Analyze Image to Minc converter converts the mask to the Minc file format. Be certain to use the `-like` option in this node to ensure the mask is in the same coordinate space as the input, else, the starting points of the minc file will be shifted to 0, 0, 0.
  6. Use the visualization program of your choice (Display for Minc format, BrainSuite2 for Analyze image format) to check and manually edit the skull-stripped results. Generally, the dura and superior sagittal sinus cause the most problems. Thus, you will want to pay particular attention to these areas when editing your mask.





Shown in blue, are the regions of the brain typically included when ensuring proper skull-stripping.



Key Variables: (Suggestions in parenthesis)

Reorient:

Input volume (*input\_file.img*) or list of input volumes (*input\_file.list*)

Reorient command: desired reorientation  
output volume (*reoriented\_image.img*)

Brain Surface Extractor:

Anisotropic filter: (*3 5.0*)

Edge Detector: (*0.75*)

Relabel:

Input volume *stripped\_mask\_image.img*

Analyze Image to Minc 2:

Model input *reoriented\_minc\_image.mnc*

Input volume *stripped\_mask\_image.img*

Output volume *stripped\_mask\_image.mnc*



## D. Creating masks that divide hemispheres

Dividing the brain into hemispheres is needed for analyses in which you are examining the medial surface of the brain.

### Automatic Hemisphere Division using BrainSuite2

BrainSuite2 will label the brain into two separate hemispheres and the cerebellum.

### Running the Cerebrum tool from your PC in Brainsuite2

*Input image type:* Analyze image

*Required preprocessing steps:* Registration to the reference atlas

Skull stripping

Bias field correction

*Suggested preprocessing steps:* None

*General tips before beginning:* To use this option in the BrainSuite2 GUI, the spatial orientation of the template must match the input case. For more information, reference the BrainSuite2 manual.

- Processing steps:*
1. Open your skull-stripped image file in BrainSuite2.
  2. Click on BrainSuite, then on the Cerebrum tool and click all.
  3. For more detailed information or more flexibility in this process, see the BrainSuite2 tutorial.
  4. Use the visualization program of your choice (Display for Minc format, BrainSuite2 for Analyze image format) to check and manually edit the hemisphere masks. For additional notes on editing hemisphere masks, see section ii below.

### Manually Editing a Hemisphere Mask

As with skull stripping, it is often necessary to edit masks that were created by programs automatically. Editing can be done in either BrainSuite2 (for analyze image format) or Display (for minc format). Please see the general guidelines at the beginning of this section for more specific instructions on both of these programs.

Dura can cause the greatest problems in hemisphere divisions. Some areas in which to exercise extra care are noted below:

1. In the most posterior regions, note the dura mater in between the two hemispheres.
2. The next non-brain tissue to avoid is the transverse and sigmoid sinus separating the cerebrum from the cerebellum.
3. After these, take note of the meninges and superior sagittal sinus in between the two hemispheres. These will continue throughout the entire brain.
4. When the brain stem appears, only cover the part that is above the most posterior portion, bottom, of the cerebellum.
5. Soon after, the brain stem will disappear with the pons leaving last. Everything in between the temporal lobes, from the pons forward, is non-brain, and so it is removed.

6. Sometimes the border between the temporal lobes and the non-brain is very difficult to distinguish.
7. The optic nerves emerges along side of the temporal lobes, and then passes up into the gap between the temporal lobes and the insula. Use the raw image, in particular, to make sure these aren't included in the brain mask.
8. The final region to watch out for is in the most anterior portion of the brain. Like the most posterior region, the dura will be in between the two hemispheres.

## E. Additional Notes on Delineating Structures using masks

### Cortical structures:

- If a 3D model of the cortex has been created, in Brainsuite2 or Display, you may project the label on the 3D model, thus allowing distinction of the ROI being labeled. Please refer to the Brainsuite2 reference section for more details.

### Subcortical structures:

- In Display, it helps to keep the color in Hot Metal to make the subcortical structures look more apparent.

### Processing steps with the existence of a 3D object model of the cortical surface:

The advantage of having a 3D object model is particularly helpful for labeling gyri, as the label can be projected onto the 3D model to help find boundaries.

1. To open the MRI, object files, and label, type:  
`Display 5342_7_atlas .mnc -label 5342_right.mnc 5342_7_obj`
2. Go to File menu [T], and turn the crop off [I]
3. To color code the 3D object model, go to Color menu [D], then Color code object [A]
4. Go the Segmentation menu [F], go to Set Threshold [Y] before you edit, and set the range to 1 10000.



## 3.4.2 Applying a Mask

Now that you have created the mask, it is important to understand how to use it. As noted earlier, a mask is used to define any ROI and eliminate everything else. Depending on the file format, either Brainsuite2 (for analyze image files) or Mincmask (for minc files) can be used to extract the portion of the brain represented by your mask. AIR also offers tools for extraction of ROIs using a binary mask.

### A. Using masks in BrainSuite2

Please refer to the Brainsuite2 tutorial for applying masks.

### B. Using mincmask

When working with minc files, you can apply a mask using the command or pipeline module, “minc-mask.” This command will use the original volume and mask as an input. The output file will contain only the areas of the volume represented by the mask:

```
mincmask input_volume.mnc input_mask.mnc output_volume.mnc
```



## 3.4.2 Brain Delineation Using Display

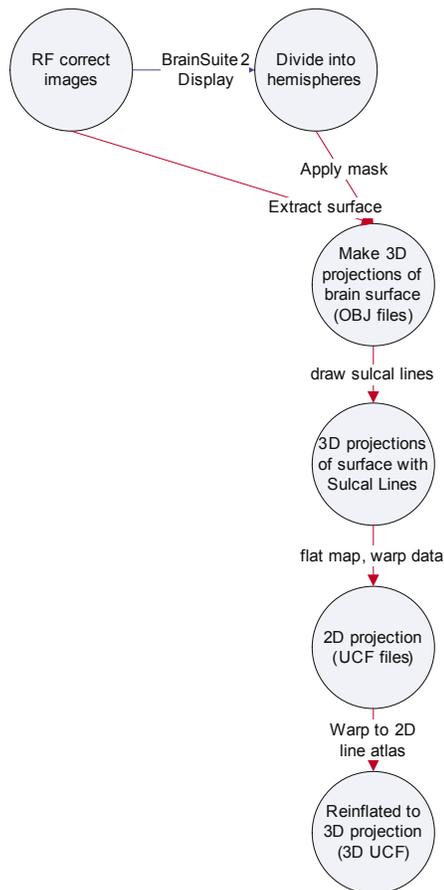
### 3.4 General Preprocessing Steps for Cortical Analyses

#### 3 Human MRI



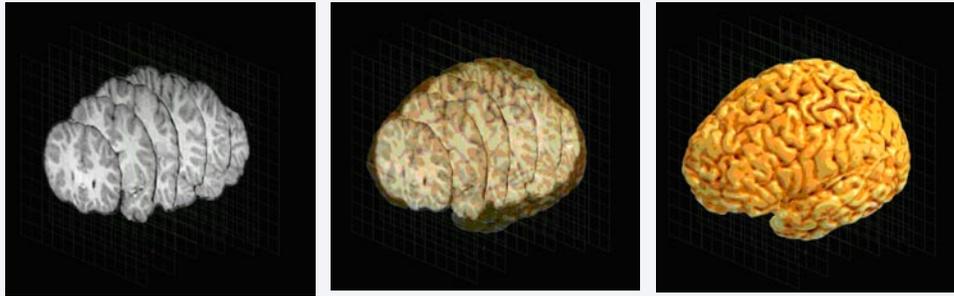
## 3.5 Processing Steps For Cortical Surface Extraction And Sulcal Analysis

Upon completing the processing steps described to this point, you can conduct some basic volume analysis of your data (see section F-G). Moreover, you can proceed with the following steps described here that will enable you to generate 3D cortical surface models of the brain. These can be used to analyze sulcal anatomy or to map cortical gray matter (see section G). The steps covered in this section include extracting the cortical surface, drawing sulcal lines, and cortical pattern matching (flatmapping, warping, and re-inflation).



## 3.5.1 Cortical Surface Extraction:

Here, we describe how to construct a 3D representation of the cortical surface. Once the surface has been created, it is possible to identify a series of anatomical landmarks (specifically sulcal lines) across subjects (see sections 2-3 for more details on drawing sulcal lines and cortical pattern matching).



Two primary methods are used for extracting a cortical surface. One method employs tools provided by BrainSuite2, using the analyze image file type. However, the surfaces generated by BrainSuite2 are not properly parameterized, nor compatible with the flat-mapping and warping steps outlined below. Thus, should you wish to use complete the gray matter analyses described in section G, you must use the pipeline described below, which deals with files in minc format.

### A. Using the Pipeline to create a surface model

The pipeline described in this section, uses a skull-stripped MRI image (in minc format) as its input. A threshold intensity is selected that best distinguishes the gray matter/CSF boundary. A template object (in the same space and orientation, but greatly simplified in shape and features) is gradually deformed to the MRI image, until it reaches a voxel at or very near the given threshold, ultimately revealing the gyral patterning of the cortical surface.

The brain can be examined in its entirety (called a lateral extraction) or by hemisphere (called a medial extraction). The processes for creating a 3D surface are similar, but not identical, as they use different model templates. As such, we describe the pipelines used for both lateral and medial extractions in the sections that follow.

As noted above, prior to surface extraction, an appropriate threshold intensity is required. The threshold is determined by the intensity represented at the gray matter/CSF interface.

#### i. Choosing a threshold for cortical extractions (both lateral and medial)

1. Open the MRI in Display.  
*Display input\_file.mnc*
2. The Display program utilizes a “keyboard menu” (called a pop-up menu) that contains all program options. From this keyboard, select the following:
  - Select “S” to open the Slice View menu
  - Then, “G” to Recompute Histogram\*

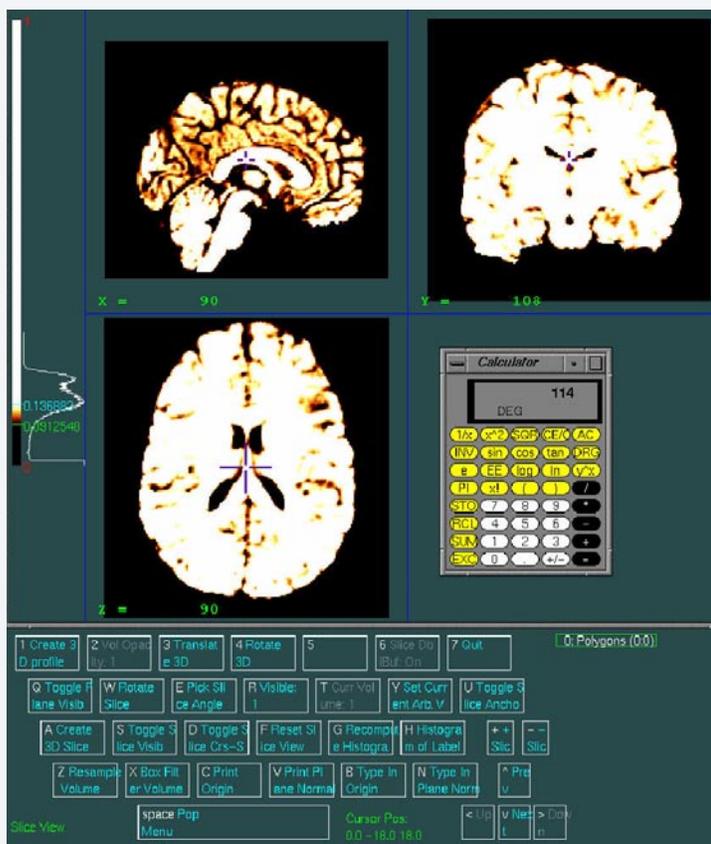
\*To get a histogram for the entire image, keep your cursor in the menu box (if the cursor lies in one of the x, y, or z views, you will only get the histogram for that particular slice).
3. This will produce a histogram displayed next to the contrast bar (shown at right)
4. Looking at the histogram, place the bottom marker on the contrast bar at the point at which



the slope begins to rise to the first peak (end of fluid signal intensities) and the other marker at the highest point of the first peak (gray matter signal intensities).

- Calculate the value directly in between these two points. This will be the approximate threshold for the subject.

Note: One can further validate this value by scrolling the cursor over the brain image where the gray matter and CSF meet, and observing the VI value located in the bottom left corner of the display screen. The intensities represented by VI, should closely match that calculated from the histogram.



## ii. Lateral (whole brain) extraction

*Input image type:* Minc format

*Required preprocessing steps:* Skull stripping

RF correction

Registration to a common space or an xfm file describing transformation to atlas space, or the space where the template lies.

*Suggested preprocessing steps:* For your skull stripped brain, you may input either a minc file which includes the CSF, or one which has had the CSF removed. Typically, mincs with CSF removed result in better object models.

*Processing steps:*

- In order to create what is referred to as a lateral object model, or a 3D object model of the entire brain, the “Cortical Surface Extractor” node of the pipeline is used, which requires three inputs.
  - The first is the skull stripped, RF corrected minc file
  - The second, is the surface threshold for the same minc, which is based on the histogram of the minc and provides a numerical value describing the gray matter/CSF boundary of the minc. This value is found by using the module “Threshold Cortex” which takes the same minc file as an input.
  - The third required input is an xfm file describing the transformation of your minc file from native space (the space in which the scan was originally acquired) to an atlas space. This xfm file is created by inputting your minc file into the “MRI to Talairach” pipeline node.

Please note that this xfm file is not required if your input minc is already in the same space as the template, which in this instance is the ICBM space.

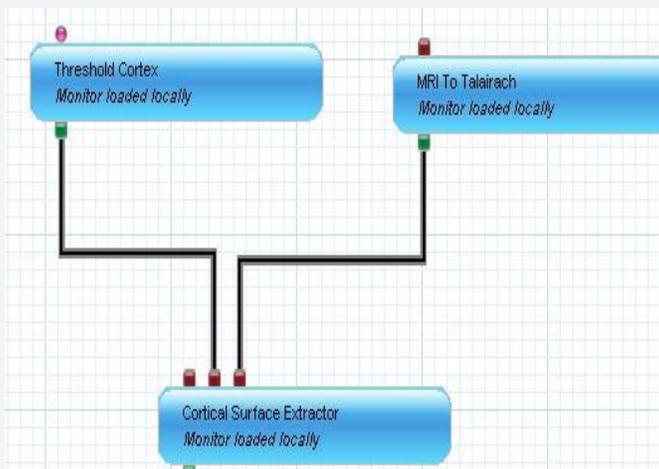
2. These three inputs are now used in “Cortical Surface Extractor” which will yield a 3D rendering of the inputted minc saved as an MNI obj file.
3. This process takes approximately 20 minutes.

NOTE: Often, a single threshold will not yield the best results and multiple threshold will need to be used. For this, it is recommended that one runs a series of duplicates of this pipeline simultaneously to ensure a multitude of objs to choose from. The alteration needing to be made to this pipeline would be the elimination of the “Threshold Cortex” node with an input list text file (with all thresholds of interest) into the “Cortical Surface Extractor” node (see the Pipeline appendix for list explanations). Recall from above, that you can find an “ideal” threshold using the Display program, and then select a range about that threshold for your input list.

4. Once the process has completed, open the output (cortical surface object file) in Display to confirm that it has been extracted at an appropriate threshold.

*Display output\_surface.obj*

5. The object file should clearly reveal the sulcal fissures of the brain. If a threshold is too high, the object will look too atrophied. If a threshold is too low, the object will look too smooth, hence the need to try multiple thresholds



Key Variables: *(Suggestions in parenthesis)*

**Threshold Cortex:**

Input: Whole Brain Volume (*minc file*)  
Output: Surface Threshold (*txt file*)

**MRI to Talairach:**

Input: T1 Volume (*minc file*)  
Input: Starting Transformation (*xfm file*)  
Output: Transform to Talairach (*xfm file*)

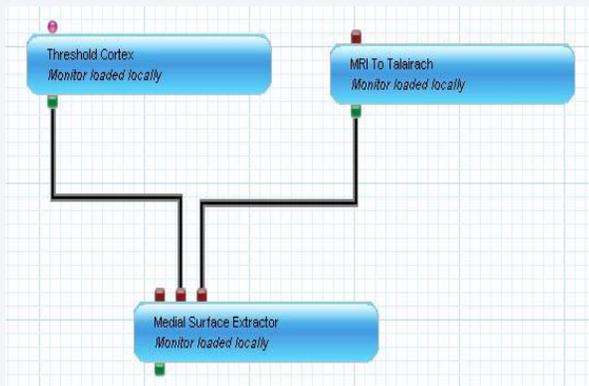
**Cortical Surface Extractor:**

Input: T1 Volume (*minc file*)  
Input: Surface Threshold (*value*)  
Input: Transform To Talairach (*xfm file*)  
Output: Surface Object (*obj file*)



### iii. Medial extraction or extraction of a hemisphere

The only distinction between the lateral and medial surface extraction is the node used in the pipeline, which in essence, references either the lateral or medial model to be deformed. This node is contained within the pipeline library and is called “Medial Surface Extractor.”



Key Variables: *(Suggestions in parenthesis)*

**Threshold Cortex:**

**Input:** Minc volume (*input\_file.mnc*) or list of input volumes (*input\_file.list*)

**Output:** Numerical Cortical Value for inputted minc

**MRI To Talairach:**

**Input:** Minc volume (*input\_file.mnc*) or list of input volumes (*input\_file.list*)

**Input:** Initial transformation file if one exists. Not Required.

**Output:** Numerical value of the threshold for the Inputted minc.

**Medial Surface Extractor:**

**Input:** Minc volume (*input\_file.mnc*)

**Input:** Surface Threshold

**Input:** XFM transformation matrix

**Output:** 3D object Model.



## B. Using BrainSuite2

Please note that to date, the surfaces generated using Brainsuite2 may be used to analyze the sulcal anatomy of the brain, however, cannot be used to study cortical gray matter (density and thickness) as described in section G.

**Input image type:** Analyze image (others, to some extent. See documentation on BrainSuite2)

**Required preprocessing steps:** Skull stripping  
RF correction  
Segmentation (or tissue classification)

**Suggested preprocessing steps:** None

**General tips before beginning:** None

**Processing steps:** These steps must follow one another where the cortical surface extraction involves the following:

1. Generating a cortex model
2. Running a topology correction
3. Generating a cortical surface (one for each hemisphere)

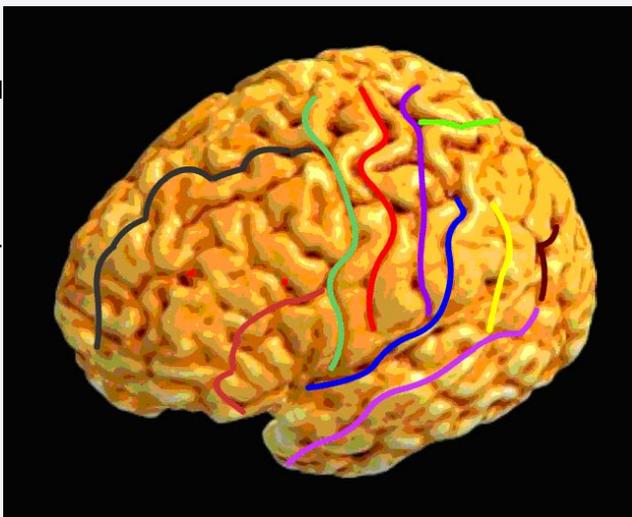
See the BrainSuite2 manual sections about Cortex Model, Topology correction, Surface Generation, Working with surfaces, and Skull and Scalp Modeling (if it fits your needs) for more information on how to perform these steps.



## 3.5.2 Drawing Sulcal Lines

Once you have generated a cortical surface, the next step involves delineating the sulcal fissures of the surface. This will allow you to complete the flat-mapping and warping steps that follow.

There are two programs that can be used to complete the tracing step; BrainSuite2 and Display.



### A. Overview

Before beginning with either program, it is important to familiarize yourself with the cerebral surface anatomy, and to be aware of some of the “rules” that we employ to handle ambiguous cases, which we clarify in the Drawing Tips, and Lines sections below.

#### Drawing Tips

- ALWAYS draw the sulcus in same direction, that is, be consistent in the directions in your starting and ending points for each sulci (especially helpful for Frontal Sulci).
- If presented with 2 equally good opportunities to jump when drawing any of the sulci, always jump sooner than later
- Generally, choose the shortest direct route between start and termination points for each sulcus. Do not jump gyri unless the sulcus is clearly discontinuous, and follow all twists and turns of each sulcus when it is an unambiguous path. If you must jump a gyrus, always choose the shortest distance and do not select points on the gyrus. Specifically, when you come to the last point before you must jump, just choose that one point on the other side of the gyrus (shortest distance possible) where you will draw the continuation of the sulcus.
- Starting points for most sulci tend to be in deepest, darkest contrast sulcal area whereas an ending point can have less contrast if the sulcus is continuous.
- If the sulcus is not present (most notably 22 and 2c), do not draw it.
- Note, change line weighting back to 0 before drawing control lines, as often, they do not fall within any sulcal patterns found on the cortex.
- Imagine a plane bisecting the inter-hemispheric fissure. Generally, keep the control lines parallel to this plane without veering too far laterally to accommodate gyrification.
- Draw lines roughly at the location which best distinguishes the separation between the lateral and medial surface of the hemisphere.



## The Lines

The lines that we traced are summarized in the table below. For a more specific protocol, go to:

[http://www.loni.ucla.edu/~esowell/elevel/new\\_sulcvar.html](http://www.loni.ucla.edu/~esowell/elevel/new_sulcvar.html) (for the lateral surface) and <http://www.loni.ucla.edu/~esowell/elevel/MedialLinesProtocol.htm> (for the medial surface)

Line#	Line Name	Direction
1	Central Sulcus	Superior to Inferior
2a	Superior Temporal Sulcus, Main Body	Anterior to Posterior
2b	Superior Temporal Sulcus, Ascending Branch	Inferior to Superior
2c	Superior Temporal Sulcus, Posterior Branch	Anterior to Posterior
3	Postcentral Sulcus	Superior to Inferior
4	Inferior Frontal Sulcus	Anterior to Posterior
5	Superior Frontal Sulcus	Anterior to Posterior
6	Olfactory Sulcus	Anterior to Posterior
7	Occipital-Temporal Sulcus	Anterior to Posterior
8	Collateral Sulcus	Anterior to Posterior
9	Inferior Temporal Sulcus	Anterior to Posterior
15	Intraparietal Sulcus	Anterior to Posterior
16	Sylvian Fissure	Anterior to Posterior
19	Precentral Sulcus	Superior to Inferior
20	Secondary Intermediate Sulcus	Superior to Inferior
21	Transverse Occipital Sulcus	Medial to Lateral
22	Primary Intermediate Sulcus	Superior to Inferior
10	Olfactory Control Line	Anterior to Posterior
11	Superior Frontal-Precentral Control Line	Anterior to Posterior
12	Central-Postcentral Control Line	Anterior to Posterior
13	Postcentral-Transverse Occipital Control Line	Anterior to Posterior
14	Occipital Control Line	Anterior to Posterior
17	Olfactory-Superior Frontal Control Line	Anterior to Posterior
18	Precentral-Central Control Line	Anterior to Posterior
30	Callosal Sulcus	Anterior to Posterior
31	Inferior Callosal Outline Segment	Anterior to Posterior
32a	Cingulate Sulcus, Anterior Segment	Anterior to Posterior
32b	Cingulate Sulcus, Posterior Segment	Anterior to Posterior
32c	Cingulate Sulcus, Double Parallel	Anterior to Posterior
33	Paracentral Sulcus	Superior to Inferior
34	Superior Rostral Sulcus	Posterior to Anterior
35	Inferior Rostral Sulcus	Posterior to Anterior
36	Parieto-Occipital Sulcus	Superior to Inferior
37a	Calcarine Sulcus, Anterior Segment	Posterior to Anterior
37 b	Calcarine Sulcus, Posterior Segment	Posterior to Anterior
38	Subparietal Sulcus	Anterior to Posterior
50	Superior Frontal-Precentral Control Line	Anterior to Posterior
51	Precentral-Posterior Cingulate Control Line	Anterior to Posterior
53	Posterior Cingulate-Parieto-Occipital Control Line	Anterior to Posterior
54	Parieto-Occipital-Calcarine Control Line	Anterior to Posterior
55	Occipital Control Line	Anterior to Posterior



## B. General guidelines for using BrainSuite2

*Input image type:* Cortical surface (.obj or .dfs surface)

*Required preprocessing steps:* Skull stripping

RF correction

Surface extraction

*Suggested preprocessing steps:* Alignment to a common atlas

Hemisphere masking and extraction (for medial sulcal analysis)

*Processing steps:*

1. If you are working with the object files created using the pipeline described above in the above section, it is a good recommendation to first color code your object, although this could be done in Brainsuite2 as well (please refer to the Brainsuite2 tutorial). Using Display:
  - Load the surface (.obj) as well as the corresponding mri (.mnc) in the program Display  
*Display mri.mnc surface.obj*
  - Adjust the color bar on the mri such that GM, WM and CSF are all clearly visible, and the contrast between them is unambiguous
  - From the main pop-up menu, go to “Colour Coding” (D)
  - Select “Colour Code Obj” (A)
  - To save this file, hit spacebar (to get back to main pop-up menu) and go to “File” (T)
  - Select “Save File” (D)
  - Enter a name for the file in the terminal window where the Display program was opened.
2. Load the .obj or .dfs file on which you will be tracing the lines in BrainSuite2
  - i. Click on File, then Open > Surface...
  - ii. Find the location of the surface you wish to open, and double click on the file
3. The image will be loaded in its own window (GL Window). If you right click within this window, you can select either “Curve Editor” or “Curve Protocol Tool.”
  - While either tool can be used to delineate the sulci, the curve protocol tool has the added advantage of displaying and guiding you through the most current version of the LONI cerebral sulcal protocol. As such, it is recommended that you use the curve protocol tool when tracing.
4. To begin a trace, hold the Shift key and left mouse button. Point the cursor to where you would like to begin. Once you are satisfied with the location of origin, release the Shift key.
5. The program will automatically choose the path of “least resistance” between your first mouse click, and subsequent clicks. Thus, in many cases, you can simply left click again at the end of the sulcus, and a complete trace will appear. However, in cases where the path of the sulcus is less direct (for example, a gyrus is jumped), you will need to click several times along the path of the sulcus.
  - Note that any time you need to cross a gyrus, you will want to de-select the “Stickiness” function located in the Curve Protocol Tool. Check the box next to “Stickiness” again, once you have crossed the gyrus and are resuming your trace within the sulcal bed.

6. In the event that you are unhappy with the path selected, use the “Clear Segment” button to “go back” or “undo.”
  - Note that using the “Clear All” button will clear the entire curve, while “Clear Segment” only clears the last section of the curve (clears to the last anchor point).
7. If you are using the “Curve Protocol Tool” to trace the sulci, you will want to follow the order defined by the tool, and each line will be assigned the proper number. In the event you are using the “Curve Editor Tool,” you will want to manually define a name and/or number for each line, in the “Curve” box.
8. In either case, the curves will be saved in one file (a .dfc file). To save the curves, hit the “Save Curves” button, and type a name for the file.

## C. General guidelines for using Display

Tracing the sulcal anatomy can be done using the program Display.

*Input image type:* Cortical Surface (.obj)

*Required preprocessing steps:* Alignment to a common atlas

Skull stripping

RF correction

Surface extraction using the McDonal/MNI pipeline module/executable

*Suggested preprocessing steps:* Hemisphere division

*General tips before beginning:* Be sure that you have a clean extraction that is well aligned to a common space. If the brain has not been well aligned, you will likely get unwanted “bulges” or protrusions on your surface.

When tracing in Display, it can be helpful to load and use the mri to which the surface corresponds. While you will rely primarily on the object view, the reslice view can be helpful in determining which path to take. All starting and stopping points must be chosen in orthogonal views. Rotate the object file so that you can see down the length of the entire sulcus (you must be able to clearly see the beginning and ending markers)

- Processing steps:*
1. Before beginning, it is recommended (and indeed necessary for later steps) that you color code your object:
    - Load the surface (.obj) as well as the corresponding mri (.mnc) in the program Display
      - Display mri.mnc surface.obj*
    - Adjust the color bar on the mri such that GM, WM and CSF are all clearly visible, and the contrast between them is unambiguous
    - From the main pop-up menu, go to “Colour Coding” (D)
    - Select “Colour Code Obj” (A)
    - To save this file, hit spacebar (to get back to main pop-up menu) and go to “File” (T)
    - Select “Save File” (D)

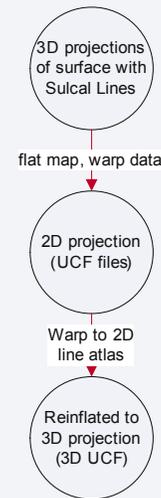
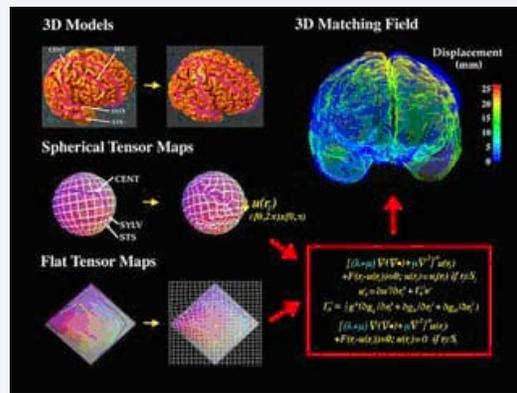


- Enter a name for the file in the terminal window where the Display program was opened.
2. Load the color-coded .obj file on which you will be tracing the lines in Display
  - Display surface\_colored.obj*
  - As noted above, it may also be helpful to load and reference the original mri while tracing:
    - Display mri.mnc surface\_colored.obj*
3. From the “keyboard” or main pop-up menu in Display, go to “Surface Curves” (V)
4. Before tracing, you will need to adjust the curve weight according to the type of line you are tracing.
  - All sulcal lines, except #31 (the inferior callosal segment) are to be drawn with a weight of -50. The inferior callosal segment ought to be traced with a weight of 100.
  - All control lines are to be drawn with a weight of 0.
  - Go to “Curve Weight” (X)
  - Enter the appropriate weight in the terminal window where the Display program was opened
  - Note that you only need to change the curve weight each time you change curve types.
5. To trace a line:
  - Go to “Start Surf Curve” (F) from the Surface Curves menu
  - Click with left mouse inside of “groove”/sulcus where you want the line to be drawn. More specifically, click at the point where you want your sulcal line to start, and then continue clicking every so often inside the sulcal groove. You DO NOT want to hold down the left mouse button and drag it through the sulcus.
  - If you are unhappy with the line (or a portion of the line) that you have traced, you will need to retrace the line. There IS NOT an “undo” function for line-drawing in Display. To start your curve over, go to “Reset Curves” (A), and then “Start Surf Curve” (F) again.
  - If you are pleased with your line, go to “End Surf Curve” (D) and then “Permanent Curve” (Z). You will now want to save this line (see next step below).
6. To save a line:
  - Highlight the curve (represented by #: Lines) by left-mouse clicking on the right-hand side of the pop-up screen
  - Hit the spacebar to go to the main pop-up menu
  - Go to “File” (T) from the main pop-up menu
  - Select “Save File” (D)
  - Enter a name for the file in the terminal window where the Display program was opened. It is recommended that you indicate brain id, hemisphere, and line number in the file name (e.g. Lbrain\_16.obj)
7. Now you are ready to being a new curve:
  - Hit the spacebar to go to the main pop-up menu
  - Go to “Start Surf Curve” (F) and repeat steps 5-6 above

### 3.5.3 Warping, Flatmapping and Reinflation

After delineating the sulci for each subject, the lines may be used to match the cortical anatomy across subjects. This process transforms an image 3D object file into a 2D flatmap of the cortex which is stored as a 2D uvl file. Once in this form, the averaging of all sulci are used as anchors to warp the flatmaps of individual cases into an average space. After this process is finished, the warped flatmaps are reinflated to 3D UCF files. These then act as the framework for the rest of the cortical analyses. The 4th dimension of these UCF files will contain the relevant information for each of the positions of the cortex specified by the 3D UCFs, e.g., cortical gray matter thickness, density, or a multitude of other relevant information, such as correlation between gray matter and csf densities for each point of the cortex.

For some additional information on troubleshooting during this process, see the Appendix.



Input image type: Obj

Required preprocessing steps: Registration to standard space

Skull stripping

RF correction

Cortical sulcal lines drawn

For medial flatmapping, divide into hemispheres

Suggested preprocessing steps: Tissue classification

General tips before beginning: Verify that all lines are drawn in the same direction, and named consistently. Use the line naming convention as specified on the Dr. Sowell's web site as some of the Pipeline modules look for numbers used in this name convention (please refer to the numerical convention as specified on ip address [http://www.loni.ucla.edu/~esowell/elevel/new\\_sulcvar.html](http://www.loni.ucla.edu/~esowell/elevel/new_sulcvar.html) and <http://www.loni.ucla.edu/~esowell/elevel/MedialLinesProtocol.htm> ).

Lateral and medial flatmapping require generally the same steps; however, lateral flatmapping is done for the whole brain, which excludes the medial portion of the brain. In order to analyze those areas, medial flatmapping must be undertaken, which involves flatmapping each hemisphere separately, and is generally, more time consuming. The underlying programs that run these processes are slightly different for the whole brain and hemisphere analysis, so



even though the general idea is the same, the actual implementation differs slightly. Additionally, if medial object models are being made for sulcal analysis, be aware that all files, that is, the 3D surface and all sulcal lines, must be transferred into left hemispheric space, which may be accomplished by a simple transformation of the files using either an .air transformation or an xfm.

## A. Lateral

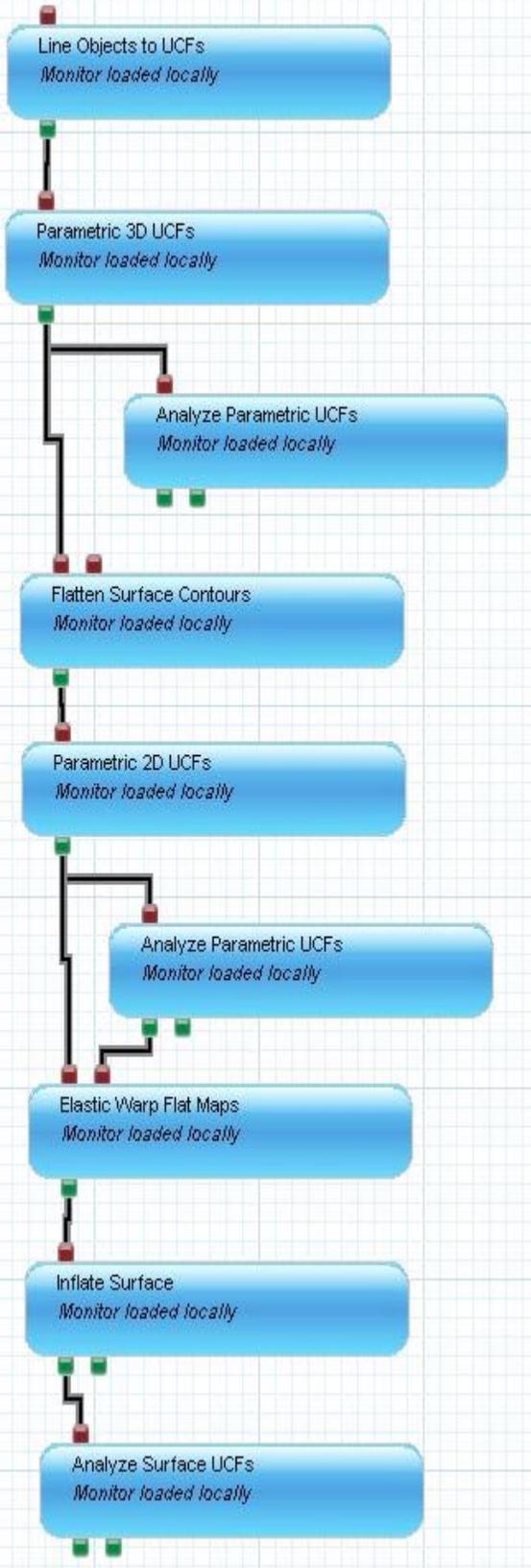
### Processing steps:

The module “Line Objects to UCFs” takes as input the directory in which your sulcal line object files are stored. It converts all of the line files into UCFs and outputs them into the specified directory. This directory is then passed to “Parametric 3D UCFs” where the lines are re-parametrized to have the same number of levels as well as points at each level and then outputted to the specified directory.

If elected, these lines can be passed to “Analyze Parametric UCFs” which uses as input the directory where your parameterized lines are located. This module uses the line numbers specified according to Dr. Sowel’s convention, and creates averages of each line number, as well as the variability of each line. This node outputs the newly created 4D UCFs into the directories specified by you, where the 4th dimension of the UCF file stores the variability found within each line.

After parameterization, the lines are ready to be passed to “Flatten Surface Contours” which takes as input the directory where the 3D lines exist, along with the color coded 3D object model, and outputs 2D lines with a reduced number of points and flat object models, ie, the flatmaps discussed above, all in 2D, into the specified directory, where the lines are reparameterized with a call to “Parametric 2D UCFs,” followed by the averaging of each line with a call to “Analyze Parametric UCFs.”

At this point, we warp the case specific 2D sulci to the group’s average by using the “Elastic Warp Flat Maps” module, from which, the “Inflate Surface” node yield the cortical warped 3D UCF file into the directory specified, from which cortical analyses can be derived. The cortical UCF can also be passed to “Analyze Surface UCFs” which takes as input the directory in which multiple cortical UCFs exist in order to make group averages and gauge variability.



Key Variables: (Suggestions in parenthesis)

### Line Objects to UCFs

Input: Object Directory (*dir file*)  
Output: UCFs Directory (*dir file*)

### Parametric 3D UCFs:

Input: UCFs Directory (*dir file*)  
Output: Parametric UCFs Directory (*dir file*)

### Analyze Parametric UCFs:

Input: Parametric UCFs Directory (*dir file*)  
Output: Average UCFs Directory (*dir file*)  
Output: Variability UCFs Directory (*dir file*)

### Flatten Surface Contours:

Input: Parametric UCFs Directory (*dir file*)  
Input: Colored Surface Objects (*obj file*)  
Output: Flattened Directory (*dir file*)

### Parametric 2D:

Input: Flat UCFs Directory (*dir file*)  
Output: Parametric UCFs Directory (*dir file*)

### Analyze Parametric UCFs:

Input: Parametric UCFs Directory (*dir file*)  
Output: Average UCFs Directory (*dir file*)  
Output: Variability UCFs Directory (*dir file*)

### Elastic Warp Flat Maps:

Input: Reference Flats Directory (*dir file*)  
Input: Target Flat UCFs Directory (*dir file*)  
Output: Results Directory (*dir file*)

### Inflate Surface:

Input: Precise UIFs Directory (*dir file*)  
Output: Color Surface UCFs Directory (*dir file*)

### Analyze Surface UCFs:

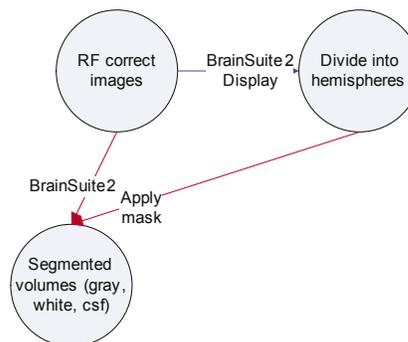
Input: Surface UCFs Directory (*dir file*)  
Output: Average UCFs Directory (*dir file*)  
Output: Variability UCFs Directory (*dir file*)



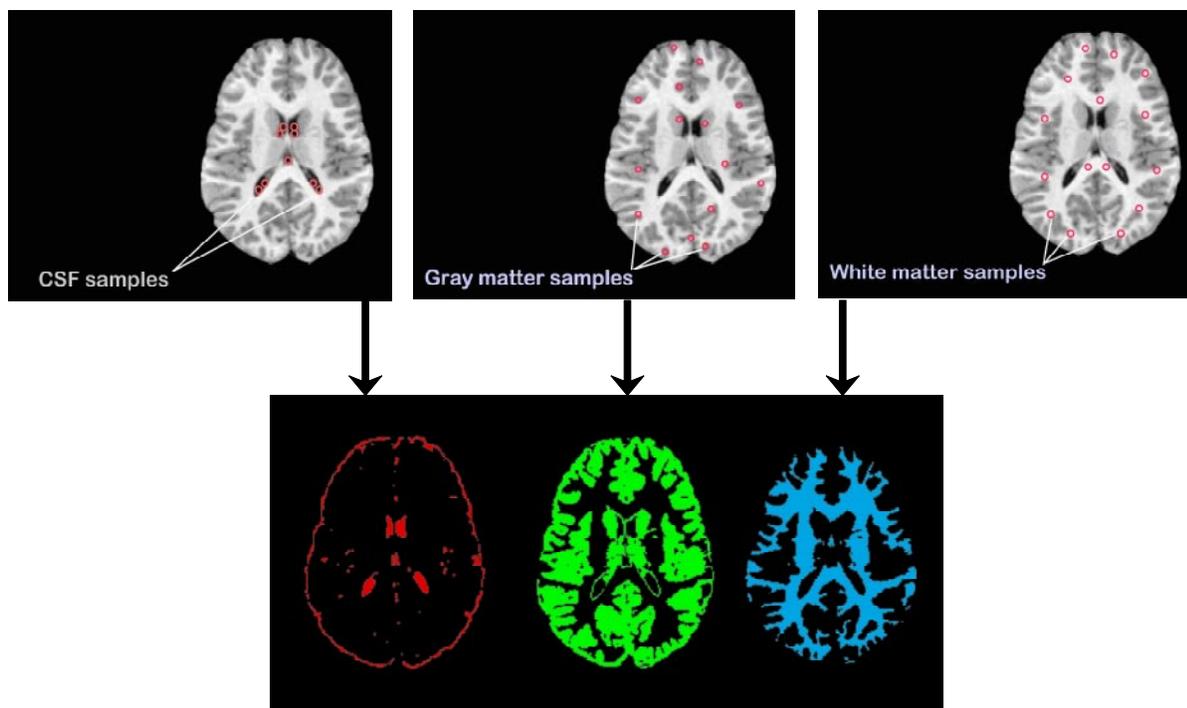
## 3.6 Separation of Gray Matter, White Matter, and CSF (Segmentation)

Segmentation, or tissue classification, involves defining a volume by tissue type (gray matter, white matter, and CSF). This is useful for determining the volume of a given tissue type in the brain, and also for mapping gray matter information, e.g., density or thickness, onto the cortical surface.

Typically we use a 3-class tissue segmentation, which simply divides the brain into regions of gray matter (GM), white matter (WM) and cerebral spinal fluid (CSF). It is important to note, however, that we are capable of partial volume, or 5 class segmentations, which additionally identify regions of GM/WM overlap and GM/CSF overlap (see PVC section below for more details). This can be useful in specific cases.



Although different tissue segmentation programs exist, partial volume classifier (PVC) of Brainsuite is the one most commonly used here in the lab. Simply put, signal values are evaluated according to frequency, while measuring differences across the whole brain and classifying the signal values accordingly to tissue type.



While you may plan on hemispheric analysis using tissue classified files, it is highly recommended that you segment your data on the entire skull stripped brain, before applying the hemisphere masks. This gives the segmentation program more data to use for classifications, and keeps an individual subject's classification as consistent as possible.

## A. Automatic with PVC

While the resolution and contrast of your MR data is highly important in tissue classification, you may be able to separate out the gray matter, white matter, and CSF automatically using PVC on most data sets (using a 3 class or a 5 class separation). The 3 class segmentation, which is most commonly used, labels different tissue types with the following number values: CSF = 0, GM = 1, WM = 2, and background = 8. PVC also allows for a 5 class separation, which additionally distinguishes the areas where gray and white matter overlap, and where the gray and CSF overlap.

PVC can be run on a command line, within in the Pipeline, or in BrainSuite2.

*Image type:* Analyze image format

*Required preprocessing steps:* Skull-stripping

*Suggested preprocessing steps:* RF correction

### Running PVC using BrainSuite2

You can use PVC directly in BrainSuite2. Open your file in BrainSuite2. If you have not already stripped the skull and RF corrected the images, you can do that whole process using BrainSuite2 (see Brainsuite2).

1. Open the PVC dialog box from the BrainSuite menu.
2. Be sure that you have checked the box that reads “three class” if you want a 3 class segmentation. Note that the program will perform a 5 class segmentation if this box is left unchecked.
3. Click Do All. For more information, see the BrainSuite2 tutorial.
4. To save the tissue classified file, go to File and select Save Labels As
  - Type a filename for the file (filename.label.img), and click Save.

### Command line

You can also use the following command in your UNIX terminal to generate a 3 class segmentation (if you have compiled pvc for your system):

```
pvc -3 -i input_filename, output_filename
```

You may need to convert your file back to Minc format if you want to examine it using Display, but you may view your results directly in Brainsuite2 in analyze image format.

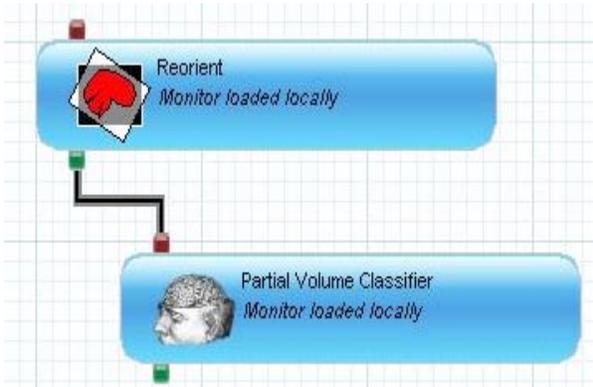
### Running PVC in the Pipeline

You can also call the PVC module in the Pipeline BrainSuite2 package to classify skull-stripped MRI images. It is good practice to run your MR data through Reorient, without exercising any reorienting options, as this will set a series of fields in the MR's header file, which PVC requires. PVC also works on either 8 or 16 bit data, so you may have to change 16 bit data to 8 bit data by using the module Reorient 8bit from the AIR package, or visa versa.

*General tips before beginning:* Partial Volume Classifier (PVC) has two options that you need to choose depending on your data. One is tissue prior weight value, usually the default doesn't need to be changed. The other is three class labeling flag should you want to perform a 3 class segmentation (otherwise the program will perform a 5 class segmentation).



*Processing steps:* If you plan to use Display to view this, the label images need to be converted to MINC images before they can be opened in Display to view and manually edit them. Otherwise, you may be able to view them in BrainSuite2.



Key Variables: (Suggestions in parenthesis)

Reorient 8bit:  
Volume: Input\_file.img  
Reoriented Volume: 8bit\_file.img

Partial Volume Classifier:  
Input volume: 8bit\_file.img  
Tissue Prior Weight: (0.1)  
Three Class Labeling flag:   
Label Volume: \_tissue\_labeled\_image.img

## B. Manual

Generally PVC will do an acceptable job of tissue classification, so we recommend that you use PVC first and use the manual method only as a last resort. The manual method is substantially more time and labor intensive, and also takes great practice to gain reliability. Moreover, the manual method is only designed to do a 3 class separation.

The main principal underlying manual segmentation involves tagging voxels that represent each of four regions of the scan. A total of 120 tag points are selected (40 for white matter, 40 for gray matter, 20 for CSF, and 20 for background). To best capture the variability of voxel intensities for these different regions, the tags are selected throughout the entire brain. You want to select points in different areas of the scan so long as they are distinct, and can be clearly identified as one of the four regions noted above (GM, WM, CSF, or background).

A summary of the tags is shown in the table below.

<b>Tag Points</b>	1 to 40	41 to 80	81 to 100	101 to 120
<b>Label Value</b>	1	2	3	4
<b>Tissue Type</b>	White matter	Gray matter	CSF	Background



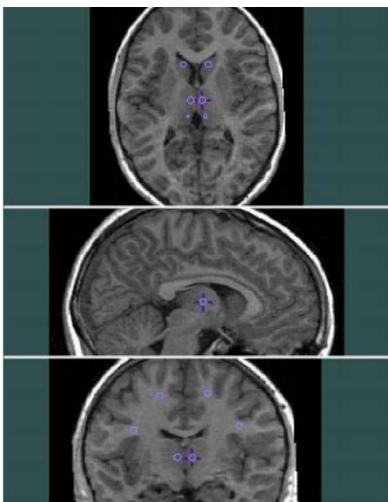
*Input image type:* Minc format

*Required preprocessing steps:* Skull-stripping

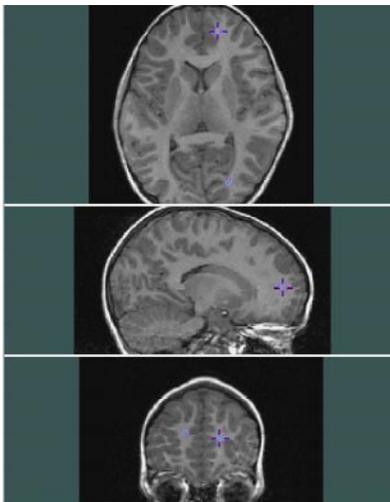
*Suggested preprocessing steps:* RF correction

### Registration

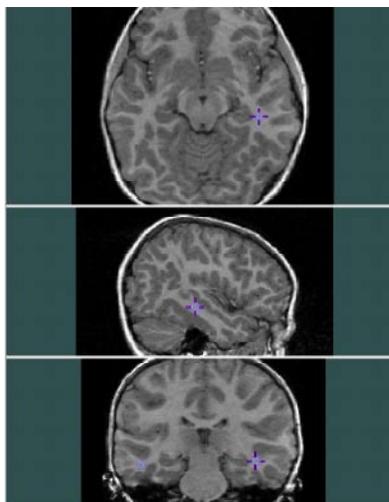
- Processing steps:*
1. Open the raw minc in the program register  
`register input_file_raw.mnc`
  2. Adjust the intensity bar to ensure good contrast between tissue types
  3. Select 40 white matter samples from the scan according to the following guidelines:
    - Do not take samples in the corpus callosum, or any region close to CSF
    - Be sure to include:
      - a. 1 left and 1 right hemisphere temporal lobe sample in a coronal section near the anterior extent of the superior colliculi (Figure 1)
      - b. 1 left and 1 right hemisphere sample in the posterior 1/10th of the brain in the occipital poles (Figure 2)
      - c. 1 left and 1 right hemisphere sample in the anterior 1/10th of the brain in the frontal poles (Figure 3)



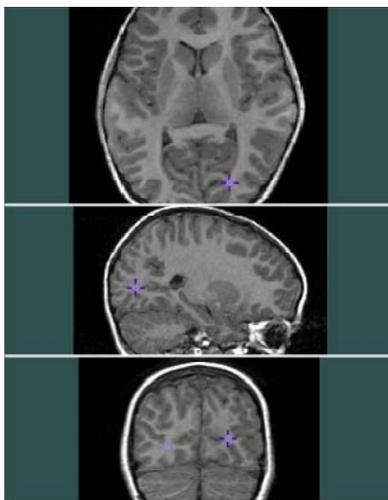
4.6 Figure 1



4.6 Figure 2



4.6 Figure 3

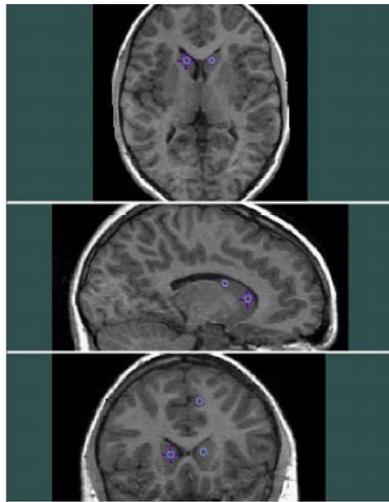


4.6 Figure 4

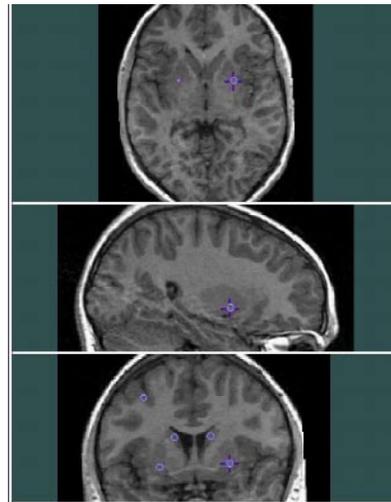
- Many samples in the middle 8/10ths of the brain will be taken in regions of the centrum semiovale extending from the parietal to the frontal lobes. Many other samples should be taken superior to the lateral fissure and parietal operculum across the extent of the parietal and into the frontal lobe. (Figure 4)
- **After the 40th sample type the number 1 in the far right window of the register program for later editing purposes.**



4. Select 40 gray matter samples from the scan according to the following guidelines:
  - Be sure to include the following 12 subcortical samples first:
    - a. 1 left and 1 right hemisphere sample in the caudate nucleus in the coronal section where the anterior commissure crosses the midline (Figure 5)
    - b. 1 left and 1 right hemisphere sample in the caudate nucleus in the most anterior coronal section where the putamen is still present (Figure 6)

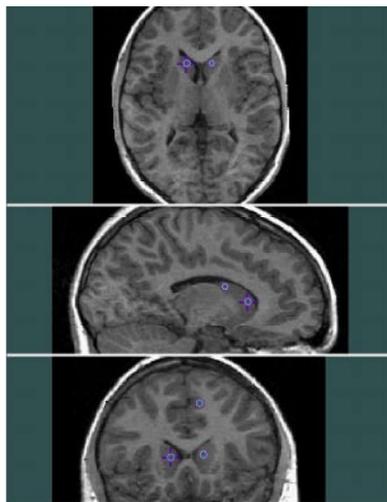


4.6 Figure 5

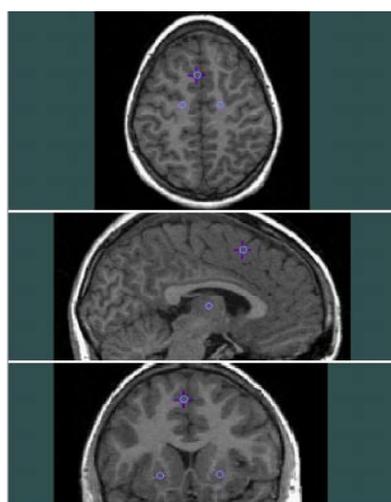


4.6 Figure 6

- c. 1 left and 1 right hemisphere sample in the putamen in the same section where the posterior caudate samples were taken at the decussation of the anterior commissure. Take the sample more inferior and lateral where the most robust gray matter signal can be visualized (Figure 7 below)
  - d. 1 left and 1 right hemisphere sample in the putamen more anteriorly when the subcallosal gyrus can first be distinguished (by a white matter tract between the gyrus and the nucleus acumbens) from the basal forebrain gray matter. Take the sample in the center of the oval shaped form of the putamen at this level (Figure 8 below)

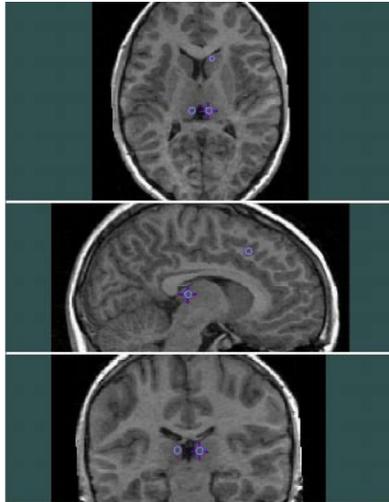


4.6 Figure 7

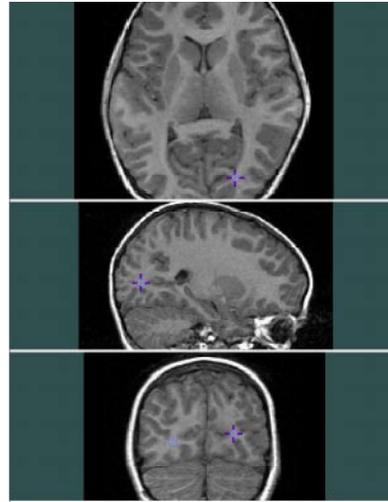


4.6 Figure 8

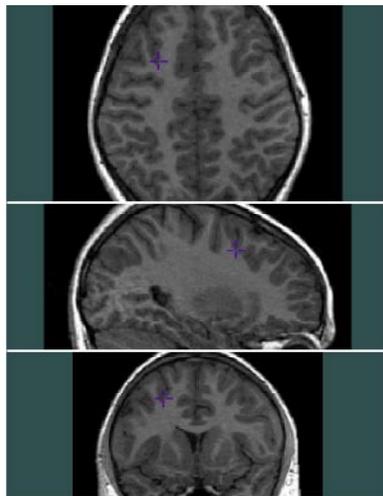
- e. 1 left and 1 right hemisphere sample in the pulvinar of the thalamus just anterior to the anterior extent of the superior coliculli. NOTE: keep these samples relatively medial where the most robust gray signal can be found in this structure (Figure 9 below)
- f. 1 left and 1 right hemisphere sample more anteriorly in the dorsomedial nucleus of the thalamus in roughly the center of the thalamus as viewed in the midsagittal plane in the section where the pulvinar sample was taken. NOTE: keep these samples relatively medial where the most robust gray signal can be found in this structure (Figure 10 below)



4.6 Figure 9

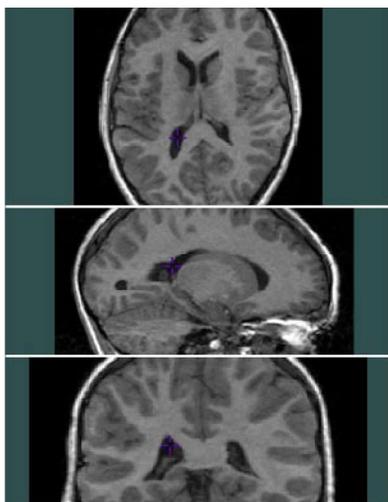


4.6 Figure 10



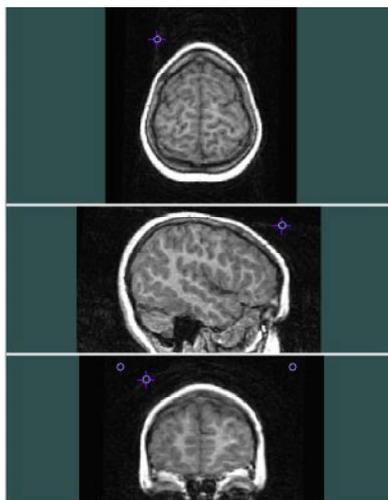
4.6 Figure 11

- Start taking cortical samples in the posterior 1/10th of the brain, taking the first in the most posterior section where the cortical gray ribbon can be clearly distinguished from white matter
- Cortical samples should tend to be evenly distributed across approximately 7 coronal sections evenly spaced across the sagittal extent of the brain.
- Be especially careful to examine the sagittal and axial views to ensure fully volumed samples. Also avoid samples in the very center of a gyrus where there is inevitably a sulcus containing CSF. Figure 11 on the left shows an inadequate sample.
- **After the 40th sample type the number 2 in the far right window of the register program for later editing purposes.**



4.6 Figure 12

5. Select 20 CSF samples from the scan according to the following guidelines:
  - The CSF samples should be, to the extent possible, evenly distributed from back to front, top to bottom, and left to right. The samples taken in the lateral ventricles should be, for the most part, taken from posterior to anterior.
  - Take the first ten samples in the lateral ventricles, and the last ten in other subcortical CSF locations.
  - Make sure not to take samples in the choroid plexus within the lateral ventricles.
  - Generally make sure to take samples in places where the lowest signal value can be seen in the center of the tag circle in all 3 orthogonal views (this will help avoid choroid plexus which has a signal value closer to gray matter) (Figure 12)
  - All of the samples should be taken in the lateral ventricles and the subcortical spaces (e.g., the 4th ventricle, cerebral aqueduct, etc.).
  - **After the 20th sample type the number 3 in the far right window of the register program for later editing purposes.**



4.6 Figure 13

6. Select 20 background samples from the scan according to the following guidelines:
  - Make sure to lower contrast to 0 or below to ensure visualization of noise in background around the brain. This is the only time the contrast goes under 0.
  - Zoom out on the image for visualization of the entire background space (Figure 13 at right)
  - Choose samples in approximately 5 coronal sections equally distributed across the coronal extent of the brain. Take samples from top left, top right, bottom left and bottom right.
  - The signal value for each sample cannot be above 4 (This value might differ depending on scan intensity).
  - No more than 4 of the 20 samples should be taken in the 0 background.
  - **After the 20th sample type the number 4 in the far right window of the register program for later editing purposes.**

7. Once you have finished selecting all 120 tag points, save the tag file. From the register program, enter a file name (filename.tag) in the box above the “save tags” button on the left side of the screen, hit enter, and then click the “save tags” button.

8. The tag file now needs to be edited to fill in the 1s, 2s, 3s, and 4s that were not filled in. Open the .tag file in a text editor such as nedit:
 

```
nedit input_file_seg.tag
```

- Highlight all of the numbers until you see a “1” on the far right side. Every row above the row containing the “1” should be highlighted. Go to the search menu and select replace. In the search for box type “”, and in the replace with box type “1”. Select “replace in selection”, and there should now be ones on the far right of the first 40 lines. Repeat this procedure

9. Now, the tag file may be used to create the segmented brain file using the following command:

```
classify -verbose -min -tag input_file_seg.tag input_file_raw.mnc input_file_seg.mnc
```





## 3.7 Cortical Analyses

### 3.7.1 Volume of Gray Matter, White Matter, and CSF

#### A. Calculating Volumes from Segmented Images

Once the MR data has been segmented and labeled (this is done during the segmentation process), the volume of each tissue type can be determined, that is, volumes for the gray matter, white matter, csf, and total brain matter. Clearly, volumes can be generated for the entire brain, or if masks of ROIs exist, for just those regions (see section D for details on masking).

The number of voxels identified as a given tissue type, along with voxel dimensions are used to calculate volumes. The pipeline shown in this section will generate a table that contains volumetric information for each tissue type, as well as the whole brain.

*Image type:* Minc file

*Required preprocessing steps:* Skull stripping  
RF correction  
Segmentation of gray and white matter and CSF

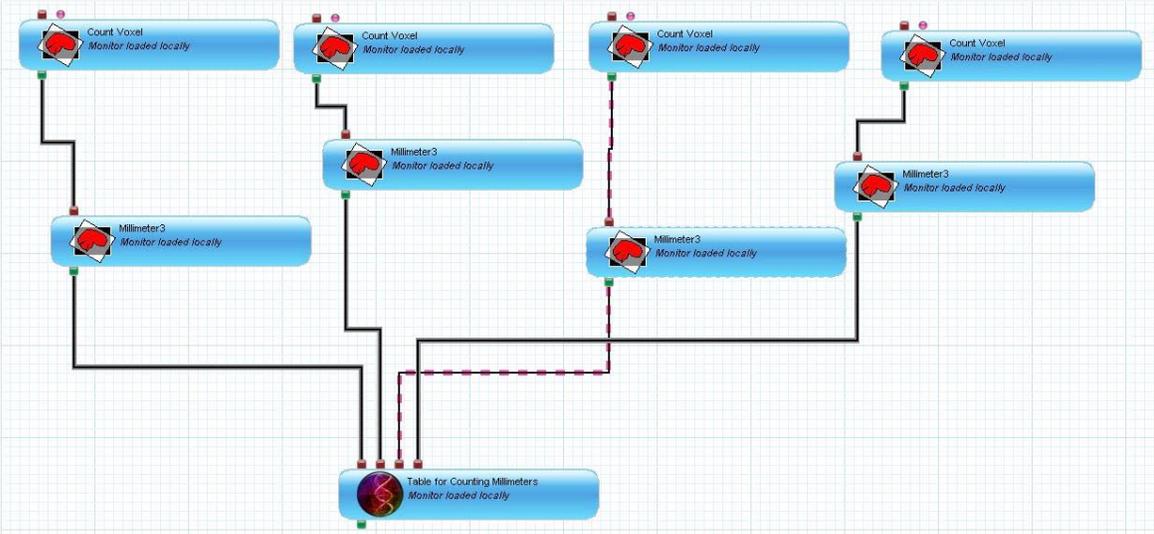
*Suggested preprocessing steps:* None

*General tips before beginning:* None

*Processing steps:* Using the skull stripped tissue classified file, “Count Voxel” modules 1-4 use the specified range for each tissue type and count the number of voxels that fall under each range. For example, PVC classifies gray matter as 1. Thus, if you specify a range of 0.5 to 1.5, the module will count the number of voxels classified as gray matter.

This value is then sent to the module “Millimeter3 From Counting Voxel,” which identifies the voxel dimensions, and outputs the number of voxles by the dimensions, generating a final volume calculation.

The pipeline on the next page repeats this process four times, for total inter-cranial volume (ICV), gray matter, white matter, and CSF (if it was included in the original skull stripped minc). The results are then processed to the “Table for Counting Millimeter3” module which outputs a text table holding the volumes for each of the four classifications.



Key Variables: (Suggestions in parenthesis)

Count Voxel #:

Segmented input volume (Minc File) or list of input volumes (input\_file.list)  
Range of Voxels: Specify the range to be counted e.g. -0.5 2.5 for all voxels.

Output Voxel File (the number of voxels in the specified range)

Millimeter3 From Counting Voxel #:

Input Text file (the numerical value of the number of voxels)  
Output Voxel Volume (in millimeters cubed)

Table for Counting Millimeter3:

Input Case Number (so as to allow for different entries into the table)  
Input Total Millimeter Number (for total ICV)  
Input Gray Matter Millimeter Number (for total GM volume)  
Input White Matter Millimeter Number (for total WM volume)  
Input CSF Millimeter Number (for total CSF volume)

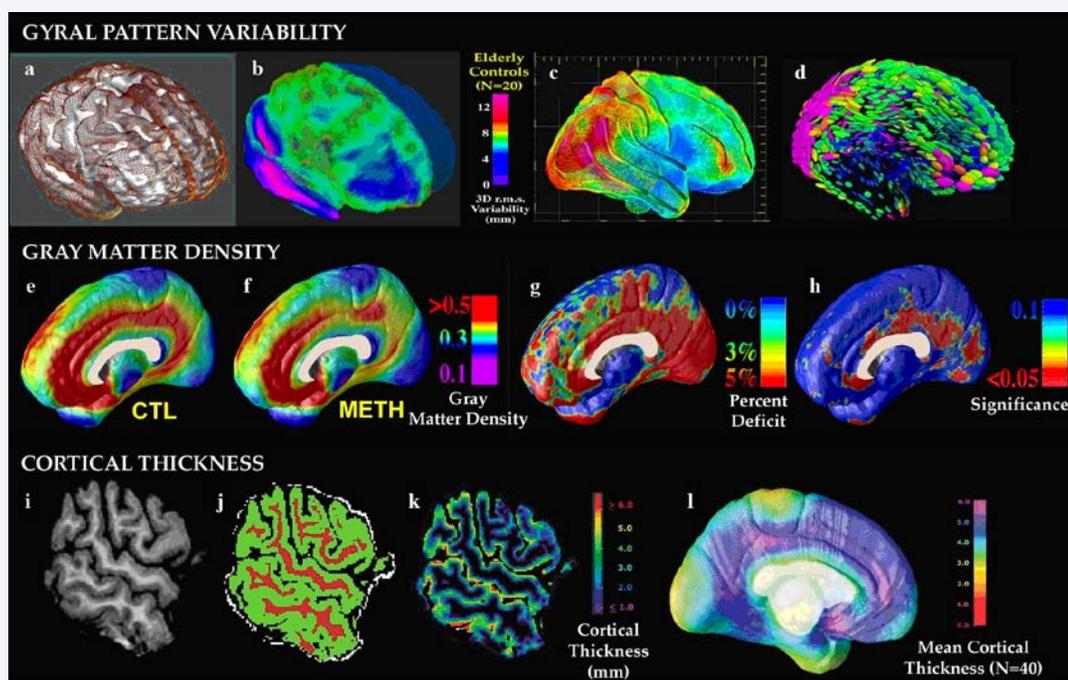
Output Table Txt File (the .txt file which includes all the volumes for each tissue type)



## 3.7.2 Cortical Surface Analyses

Nearly all cortical surface analyses at LONI requires cortical pattern matching, which is a nonlinear registration method for pooling data across subjects as described in the section E3 (“Warping, Flatmapping, and Re-inflation”). The cortical pattern matched data may be coupled with information on cortical thickness, gray matter density, functional MRI signals, or many other cortical measures. Practically, this means that differences in cortical measures/anatomy can be compared. Methods for comparison across groups are discussed further in part 3 or this section.

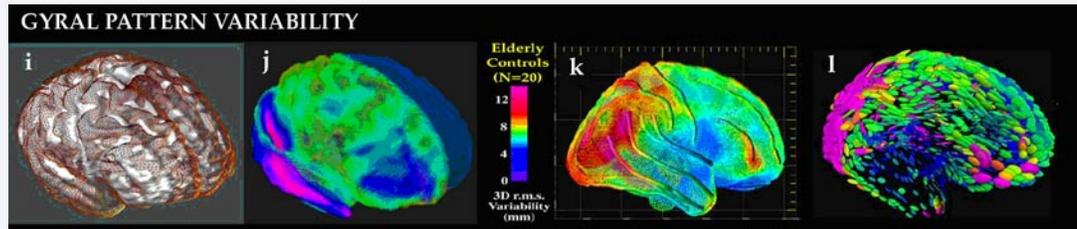
This section discusses many of the potential analyses you may want to perform on the cortex, along with any necessary/suggested preprocessing steps. The figure below briefly summarizes some of these cortical analyses.



(from 2004 neuroimage) Statistical Maps of Cortical Structure. A variety of maps can be made that describe different aspects of cortical anatomy. These include maps of gyral pattern variability [(a)-(d)], gray matter density [(e)-(h)], and cortical thickness [(i)-(l)].

## A. Sulcal Analysis (gyral pattern variability)

Anatomical variability can be studied by comparing gyral patterns based on the sulcal delineations described in section E2. Note that the example shown below is a slightly more complex version of the analysis described in this section.



Sulcal Analysis is a general term used to describe statistical methods that quantify sulcal differences between various groups or individuals. After sulcal lines have been drawn, converted into UCFs, redigitized, and resliced, one may begin a series of statistical analysis on these, examples of which include:

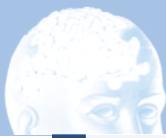
1. Finding group averages of various sulci and the variability of each sulci within the group
2. Once group averages are made, one can analyze differences between the two groups by comparing the average distance of each sulcus from the two groups
3. Asymmetry of sulcal lines between the two hemispheres by putting all lines in the same hemispheric space (preferably the left hemispheric) and examining the distance of the corresponding sulci from each hemisphere.

This section describes how to run a general sulcal analysis that determines the average position and variability of the sulci of a group that you determine by placing them into the same directory.

**Required preprocessing steps:** Registration to standard space  
Skull stripping  
RF correction  
Divide into hemispheres  
Cortical sulcal lines drawn

**Suggested preprocessing steps:** It is paramount that all lines have been named properly and have been drawn consistently and in the same direction.

**General tips before beginning:** Typically, lines are saved as objects, with an .obj extension. These object files must be converted into parametric .ucfs prior to finding variability in one group or across groups.



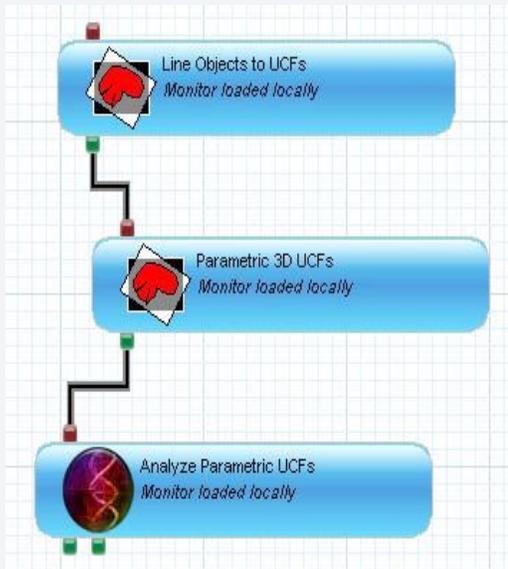
**Processing steps:** Line Objects to UCFs converts lines from object format to uc format. This node takes as an input the directory where the line objects are kept. These can be compiled as a group average, or done on a case by case basis, but if you do this, put all your output files in the same directory to analyze later.

Parametric 3D UCFs redigitizes and reslices the ucf's so that all lines have the same number of points and the same number of levels so as to allow for comparison of lines which differ in size and length. This saves parameterized ucf's to a specified directory.

Analyze Parametric UCFs takes as an input the directory of the parameterized ucf's and it derives both an average of the input lines and a variability of the input lines. It achieves this by looking for lines numbers in the name of the files found in the directory as specified on the following web sites: [http://www.loni.ucla.edu/~esowell/elevel/new\\_sulcvar.html](http://www.loni.ucla.edu/~esowell/elevel/new_sulcvar.html) and <http://www.loni.ucla.edu/~esowell/elevel/MedialLinesProtocol.htm>.

*Other tips or troubleshooting:* None

*Specific information:* None



Key Variables: (Suggestions in parenthesis)

Line Objects to UCFs:  
Objects Directory (dir file)  
UCFs Directory (dir file )

Parametric 3D UCFs:  
UCFs Directory (dir file)  
Parametric UCFs Directory (dir file)

Analyze Parametric UCFs:  
Parametric UCFs Directory (dir file)  
Average UCFs Directory (dir file)  
Variability UCFs Directory (dir file)

## B. Complexity of Cortical Surface

Cortical complexity is a measure that quantifies the “degree of folding” of the cortical surface. The analysis may be applied to the entire cortical surface, or to functionally relevant cortical regions of interests (ROIs). Cortical complexity is defined as the rate at which the surface area increases relative to increases in the spatial frequency (detail) in the surface. That is, a flat surface would not increase in area with more detail, but a really convoluted one would.

The rate of area increase is computed by gridding the surface at many different resolutions. To obtain measures of cortical complexity, the logarithmic (log) least squares regression of the surface area is plotted against the log of spatial frequency. The slopes of these regression plots are derived and added to 2.00 resulting in complexity values between 2.00 (for a planar or flat surface) and 3.00 (for a surface with immense numbers of convolutions and extensive folding).

*Image type:* Reinflated 3D UCF

*Required preprocessing steps:* Aligned and registered  
Create 3D projections  
Draw sulcal lines  
Surface mesh models  
(warped and reinflated 3D UCF)

*Suggested preprocessing steps:* None

*General tips before beginning:* None

*Processing steps:* Using the reinflated 3D UCF of each subject, run the following command:

```
ucfmeasure -fractal inputfile.ucf >> outputfile.txt
```

The output text file, lists all the complexity measures for the specified subjects.

Note that it is also possible to separate the 3D UCF into regions of interest (ROIs) in the event that you are interested in comparing the complexity values in a specific region of the brain. This can be accomplished by first using the following code to apply a binary minc file (a mask of the specified region of interest) to the surface mesh:

```
/data/ad/mass3/users/PAULS_SURFACE_CODE/SGI/MINC/UCF4D_TO_MINC/ucf_propagate_label_write_surf 2 0.5 1.5 4 5 6 wholebrain.ucf.roi.mnc > tmp.ucf
```

Before performing any actual analyses on the parcellated ucf, it is necessary to reverse the order of the points in that ucf as follows:

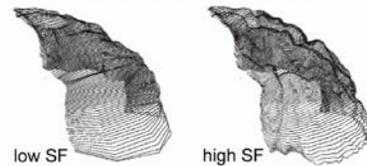
```
~thompson/SEG/SGI/hack_transfmucf 1 2 3 4 5 6 tmp.ucf > roi.ucf
```

You may now calculate the complexity for the particular region of interest, following the command described above:

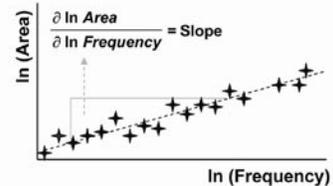
```
ucfmeasure -fractal roi.ucf >> outputfile.txt
```

To determine the cortical complexity of a 3D surface:

- (1) Modeling the surface mesh at different spatial frequencies (SF)



- (2) Plotting the mesh's surface area against the spatial frequency



- (3) Adding 2 to the slope of the regression plot.



## C. Local brain size

### Distance from Center/Zero

Distance from center (or zero) is used as a measure of local brain size. If one brain is larger than another, the distance from the center of the brain (0, 0, 0 in the coordinate system) to the surface will be greater in the larger brain. This measure can be obtained at all of the surface points in the warped ucfs.

The pipeline module “Distance from Zero” measures the distance from the center of the brain to each cortical surface point. The 3 dimensional distance from the center of the brain to the cortex at each location is recorded as the fourth dimension of the output.



**Distance From Zero:**  
Input: Subject's 3D cortical UCF  
Output: 4D UCF, in which the fourth dimension is the distance from the center

### DFCH (Distance from the Center of the Hemisphere)

This is similar to the analysis noted above, but works well on medial surfaces.

The module “DFCH” can be used to find the center of mass of the average hemisphere (left and right should be combined). For example, the center of the average hemisphere of the YALE 176 group is `/loni/edevl/bin/MEDIAL/center_of_hem.ucf`.

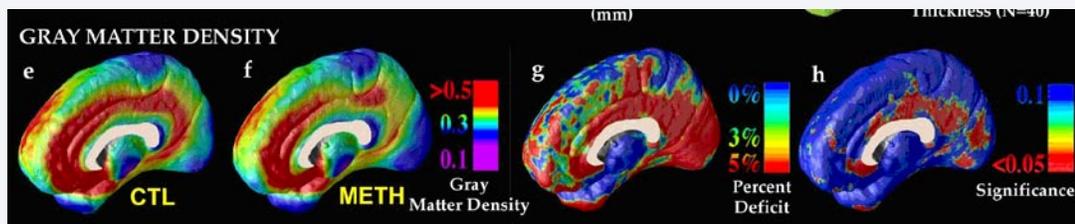


**DFCH:**  
Input: Subject's 3D cortical UCF  
Output: 4D UCF

## D. Gray matter density

At this point your image file should have been completely preprocessed and segmented into gray matter, white matter, and CSF. In addition, a 3D UCF file should have been created, which contains the 3D position information of the cortex. In this analysis, this file is used in conjunction with the segmented or tissue classified file, to assign gray matter density values at every point on the cortex (designated by the 3D UCF file), within a sphere of radius 13-15 mm. The gray matter density is classified as the percent gray matter relative to the other matter within that sphere and that value is stored for that point on the cortex. This value is then stored as the 4th dimension of the UCF file. These values are then averaged at specific homologous cortical locations in each subject.

(from neuroimage 2004)



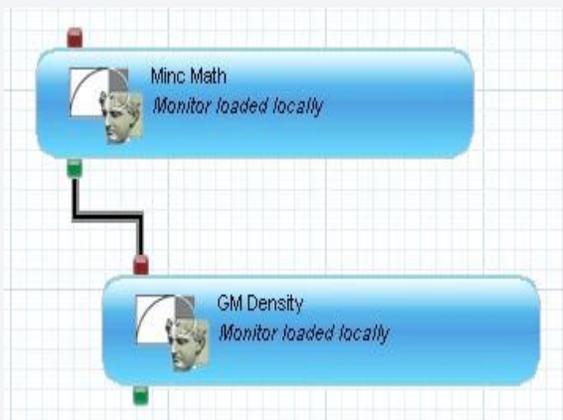
*Image type:* Analyze image

*Required preprocessing steps:* Registration to standard space  
Skull stripping  
RF correction  
Segmentation of gray and white matter and CSF  
Divide into hemispheres  
Cortical sulcal lines drawn  
Cortical flat mapping and warping  
Reinflation of warped flat maps to UCFs

*Suggested preprocessing steps:* None

*General tips before beginning:* None

*Processing steps:* Beginning with a tissue classified skull stripped volume, the gray matter is isolated using “Minc Math” by specifying the range of the gray matter, which if processed using PVC, will be set to ‘1’, hence a range of 0.5 to 1.5 is most appropriate. Using this volume along with the cortical UCF, the two inputs are supplied to the “GM Density” node. Using the 3D ucf, spheres of specified radii are cast at each point of the cortex, which along with the specified range of the gray matter, count the number of voxels filled with voxels of gray matter. Dividing this number by the total number of voxels, a density of GM at each cortical point is calculated, furnishing the fourth dimension of the resultant UCF.



Key Variables: (Suggestions in parenthesis)

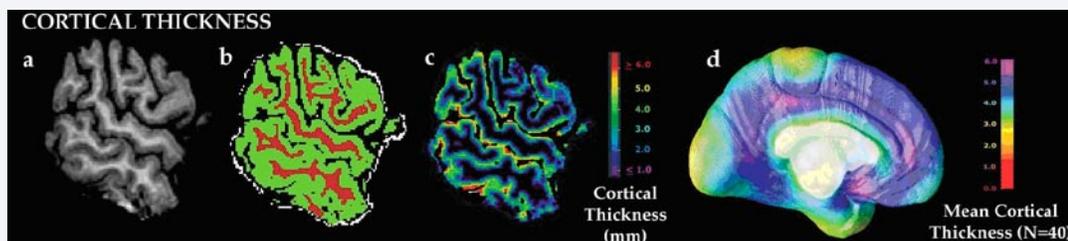
**Minc Math:**

Input: Minc Volume/tissue classified minc.  
 Output: Minc Volume/gray matter tissue only minc.  
 Specifics: Set "Segment" flag.  
 Set "Input Two Constants" flag with the range of your gray matter (.5-1.5)

**GM Density:**

Input: UCF file/3D cortical UCF  
 Minc volume/GM only minc.  
 Sphere radius: 15 mm  
 : 0.5 lowest value of GM  
 : 1.5 max value of GM  
 Output: 4D UCF.

## E. Maps of cortical thickness



from neuroimage 2004

The actual process of creating a map of cortical thickness is very similar to that of determining gray matter density, although an intermediate file is created from the segmented or tissue classified file which is then used in conjunction with the 3D UCF file to assign values to each voxel giving its distance away from the gray matter/white matter boundary. These values are then averaged at specific homologous cortical locations in each subject.

After these are generated for each subject, they can be compared across subjects and averaged at each cortical surface location to give spatially detailed maps of local thickness differences within or between groups.

*Image type:* Analyze image

- Required preprocessing steps:*
- Registration to standard space
  - Skull stripping
  - RF correction
  - Segmentation of gray and white matter and CSF
  - Divide into hemispheres
  - Cortical sulcal lines drawn
  - Cortical flat mapping and warping
  - Reinflation of warped flat maps to UCFs

*Suggested preprocessing steps:* None

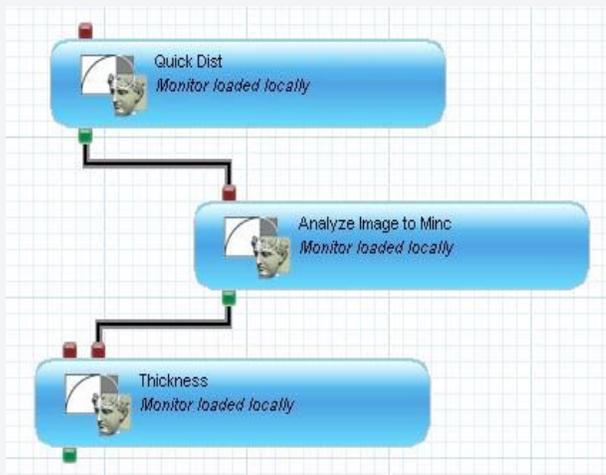
*General tips before beginning:* None



**Processing steps:** To begin quantifying the thickness of the gray matter, the segmented Analyze image file is sent to the Pipeline module Quick\_Dist (developed by Dr. Roger Woods) that determines the distance (in mm) of each voxel of the gray matter from the gray matter/white matter boundary (defined as the thickness of the gray matter). This module needs some information about the image file. This includes the maximum thickness one would expect to find for gray matter (the default is set to 10 mm), and the values of the segmented file assigned to white matter and gray matter. The output of the PVC program assigns the values of 1 to gray matter and 2 to white matter.

For the next step, the image file must be in Minc file format, so we use the Analyze image to Minc Pipeline module for this conversion.

After the thickness at each voxel has been determined, each point on the cortex that is specified in the 3D UCF file, will be assigned a thickness based on the average values of gray matter distance from the gray matter/white matter boundary within a sphere of a given radius (usually 13-15 mm) from that point. This is done by using the cortical coordinates specified by the 3D UCF file and the image output from Quick\_Dist which has a specific value describing the distance for each voxel of gray matter, and the information for each point is stored as the 4th dimension of a 4D UCF.



Key Variables: (*Suggestions in parenthesis*)

Quick Dist:  
Segmented input volume (*input\_file.img*) or list of input volumes (*input\_file.list*)  
Max thickness: 10 mm  
output Analyze image (*output\_image\_file.imgf*)

Analyze Image to Minc:  
Analyze image: ??  
Minc volume: ?.mnc

Thickness:  
Input UCF file: 3D cortical UCF  
Input image volume:

Sphere radius: 13 mm  
: 1 mm  
: 10 mm  
Output file: 4D UCF

*Other tips or troubleshooting:* None

*Specific information:* None



### 3.7.3 Analyzing 4D UCFs

Once each subject's MRI has been quantified as a 4D UCF, several higher order analyses can be performed using these files depending on how you group the files. The 4th dimension of the UCF file will hold the measurement of interest for each 3D location identified in the first 3 dimensions of the file.

#### A. Calculating Group Average Cortical Thickness/Density Files

Once warping has been accomplished and gray matter density/thickness 4D UCFs have been created for specific cases, group averages can be made with the idea to examine deviations between two groups (i.e. disease versus control group). In order to make group averages from a series of 4D UCFs, execute:

```
/data/ad/mass3/users/PAULS_SURFACE_CODE/SGI/4D_UCFS/ARB/arb_surfNstat4D_avg4D_ucfs  
<ALL INPUT 4D UCFs> > <SINGLE OUTPUT 4D UCF>
```

This will result in a single 4D UCF file which will contain in the first 3 columns the average x, y, and z coordinates of all the input UCF files and in its fourth column the average of the fourth column of all the input 4D UCFs. This is a group average for all the input files and can be used in other statistical programs to extract relevant information.

#### B. Calculating Group Variance of Cortical Thickness/Density Files

Once warping has been accomplished and gray matter density/thickness 4D UCFs have been created for specific cases, group variance can be examined between two groups (i.e. disease versus control group). In order to get variance from a series of 4D UCFs, execute one of the following commands:

```
/data/ad/mass3/users/PAULS_SURFACE_CODE/SGI/4D_UCFS/ARB/arb_surfNstat4D_var4D_ucfs  
<ALL INPUT 4D UCFs> > <SINGLE OUTPUT 4D UCF>
```

This will result in a single 4D UCF file which will contain in the first 3 columns the average x, y, and z coordinates of all the input UCF files and in its fourth column the variance of the fourth column of all the input 4D UCFs. This is a group variance for all the input files.

#### C. Calculating Percentage Difference Between Two Group Average UCFs

One of the simplest and most powerful tools available is the ratio test. Once group averages have been calculated, the ratio of one group to another can be calculated. This analysis generates a new 4D UCF with the fourth dimension quantifying the percentage difference between the two groups on a point by point basis of the cortex.

For example, if group averages were made of the gray matter thickness for a group with autism and a control group, the ratio test can show the percentage difference of the thickness between these groups at each point on the cortex. The fourth column is this ratio value, and thus can be converted to a percentage value by multiplying this column by 100 ( $\text{Group2}/\text{Group1}$ ). These values quantify the second group as compared to the first group. Thus, a ratio value less than one indicates that the thickness value is lower in the second group than the first group, and a ratio value greater than one indicates that the thickness value is larger in the second group than the first group.

After computing group averages for two populations, find the ratios between these two groups by executing:

```
/data/ad/mass3/users/PAULS_SURFACE_CODE/SGI/4D_UCFS/ARB/arb_surf2ratio4D_avg4D_ucfs <INPUT GROUP1 4D UCF> <INPUT GROUP2 4D UCF> > <OUTPUT RATIO>
```

This information can then be visualized by converting the 4D UCF to a dx file, which can be displayed by using the program OpenDX. This is the quickest method by which immediate group averages can be visualized.

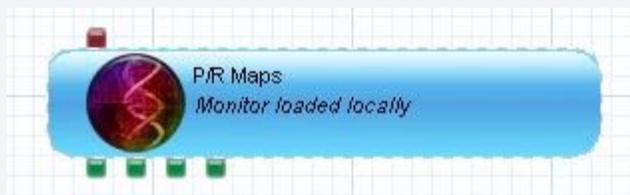
## D. Covariates and P-maps

Once individual 4D UCFs have been created, we can use these files to determine whether certain characteristics of subjects, e.g., age, sex, IQ, or diagnosis, have any direct effect on cortical trends when comparing the correlation between two groups. This is accomplished by comparing the average fourth column of one group to that of another group.

P-maps are powerful statistical tool which allow you to study trends amongst groups begin studied, using the 4D UCF files containing cortical information. As an example, let's presume that you measured the cortical thickness in a group of autistic children and normal children. A region of positive P-map shows that thickness in that area of the cortex increases in the autistic group alongside the normals. A region of negative P-map illustrates that as thickness decreases in one group, the thickness is increasing in the other. It is important to keep in mind that areas of significance always fall under the statistical value of 0.05, although you could set your threshold to lower values if significance is evident at all points of the cortex and you wish to localize it more.

To undertake this analysis, the covariate must be placed in the header of the UCF files being compared, so as to distinguish controls from patients. To run Paul Thompson's covary code, the values [-200.000000 200.000000] found in the header of the UCF file must be replaced with either 0 0 or 1 1, and while this may be undertaken however you wish, the UNIX command sed is useful and could be used in the following way:

```
sed 's/-200.000000 200.000000/1 1/' INPUT_4D.ucf > OUTPUT_4D.ucf
```

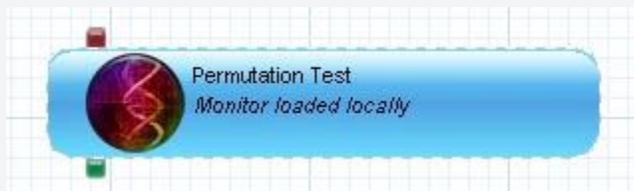


- P/R Maps:  
 Input: Directory of covaried 4D UCF files  
  
 Output: R-map  
 Output: P-map  
 Output: Positive P-map  
 Output: Negative P-map



## E. Permutation Tests

A permutation test is a great way to validate the results of the P-maps or any other test based on 4D UCFs. It may be possible that you derived results which looked promising but are due to a spurious effect. After covering your files by replacing -200 200 with 0 0 and 1 1, you can run the permutation test by executing the following module:



P/R Maps:  
Input: Directory of covaried 4D UCF files  
Output: Text File of P-Value

This module calls code which does a Monte-Carlo permutation test (multiple comparisons correction) on any surface P-map (for displacement, gray matter, asymmetry, etc., and for the effects of any covariate/score). It randomizes all input files 1000 times and gives a CORRECTED P-value for the whole map. It first thresholds the real P-map at  $p=0.01$ , calculates the surface area with p-values less than (i.e. more significant than) the threshold. Then it randomly assigns all the subjects and recalculates the P-map, a total of 1000 times.

At the end of each iteration it determines if that trial's P value was greater or less than the original map and keeps a running tally. It then tabulates a reference distribution and estimates a p-value for the real grouping based on the random groups, leaving you with a final P-value, which again, if under 0.05, signifies that the two groups were statistically different.

