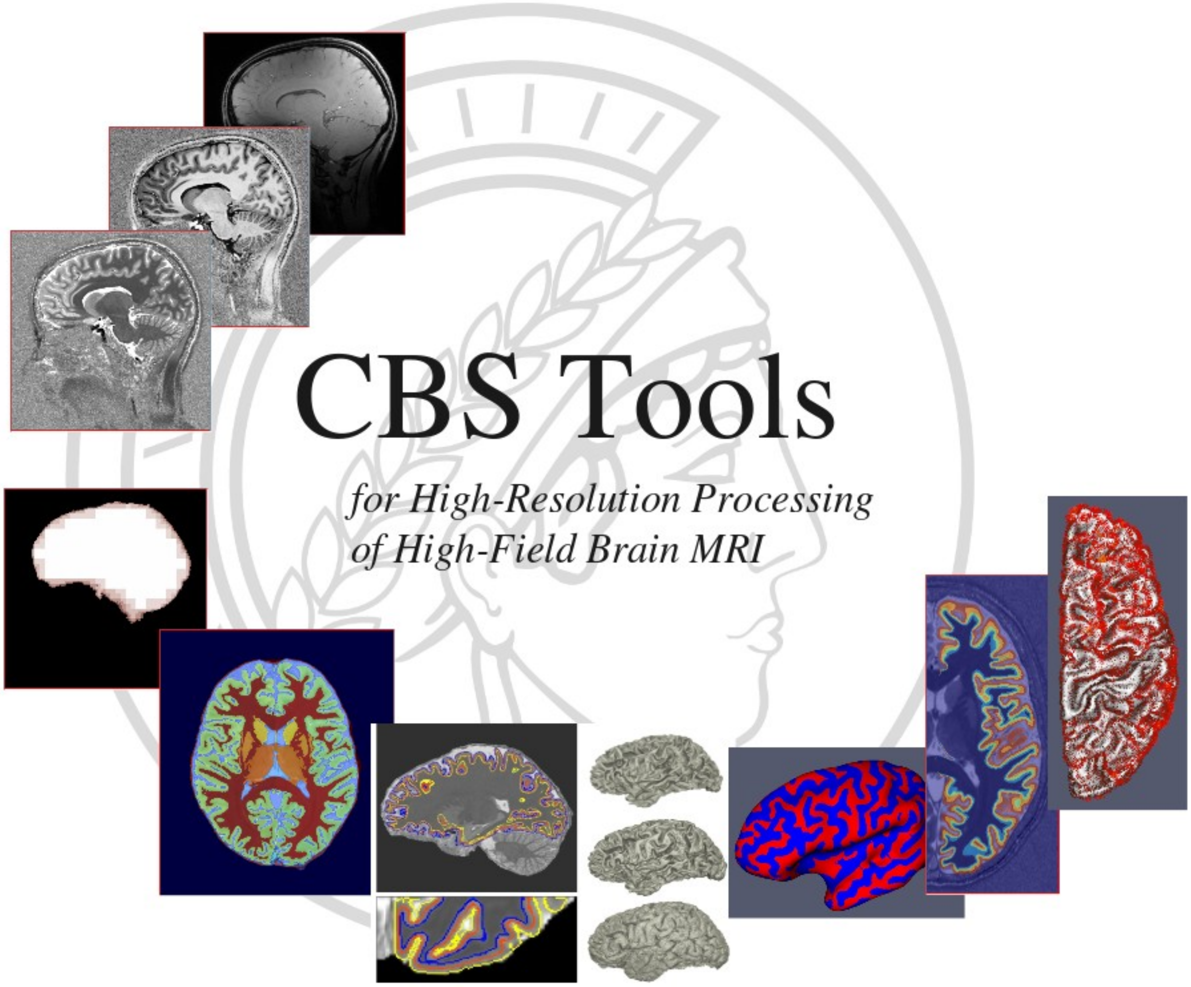
CBS Tools for MIPAV & JIST

Quick Installation and Processing Guide



Foreword

This guide describes some freely available software tools for MR image processing in neuroscience developed at the Max Planck Institute for Human Cognitive and Brain Sciences (MPI-CBS). The software, documentation and associated data are property of the MPI-CBS and governed by the license terms within the package. The algorithms included in the software package are described in the corresponding articles (see references below); this guide is aimed at helping you install and run the software in a standard way to perform high-resolution brain and cortex segmentation and laminar analysis from high field MRI.

The procedure described below have been tested with the following configuration:

- MIPAV release 5.4.4
- JIST-CRUISE nightly build of Oct. 31st, 2012
- CBS Tools release 2.0

The MIPAV, JIST, TOADS-CRUISE and CBS Tools packages are maintained by separated groups, so some temporary instabilities can occur when updating to new versions of MIPAV not supported by JIST, or versions of JIST not supported by the other tools. A continuous build system is offered by the MASI laboratory at Vanderbilt University for MIPAV, JIST and all the open-source plug-ins and tools associated with the software: http://masi.vuse.vanderbilt.edu/hudson/.

Installation

1. **Download the required software packages**:

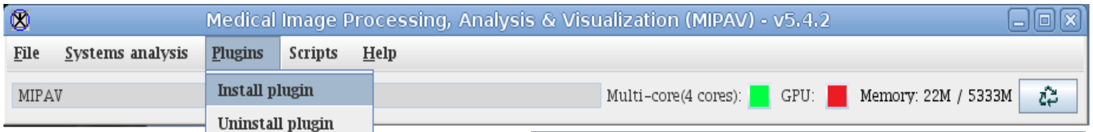
   to use the CBS Tools, you will need a copy of MIPAV, JIST and some tools from the TOADS-CRUISE package. The JIST resources tools also include many useful tools to help with general data processing. The latest version of these packages can be accessed from the following web sites (and/or the continuous build server above):

CBS Tools	http://www.cbs.mpg.de/institute/software/
MIPAV	http://mipav.cit.nih.gov/
JIST	http://www.nitrc.org/projects/jist/
TOADS-CRUISE	http://www.nitrc.org/projects/toads-cruise/
JIST resources	http://www.nitrc.org/projects/jhumipavplugins/

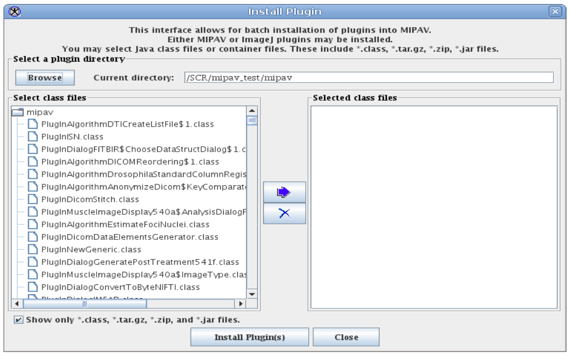
   For your convenience, we also include with the **CBS Tools Bundle** a copy of the JIST, TOADS-CRUISE and JIST resources packages which has been tested for compatibility. Bundles of packages need to be uncompressed first to get **individual JAR files**. To avoid conflicts between versions, we recommend that you use the bundled jar files (if you already have a working version of JIST and TOADS-CRUISE, you only need to install the CBS Tools). We recommend you to use the latest MIPAV version as new features and improvements are being continuously added to the software.

2. **Install MIPAV**: simply run the MIPAV installer downloaded from the website (see there for details if needed).

   Run MIPAV. From the main MIPAV menu bar, select *'Plugins-> Install plugin'*.



   The program may ask you to restart MIPAV, in which case you close MIPAV, restart, and repeat. You will get the *'Install Plugin'* window:



This interface allows for batch installation of plugins into MIPAV.
Either MIPAV or ImageJ plugins may be installed.
You may select Java class files or container files. These include *.class, *.tar.gz, *.zip, *.jar files.

Select a plugin directory
Browse

Current directory: /SCR/mipav_test/mipav

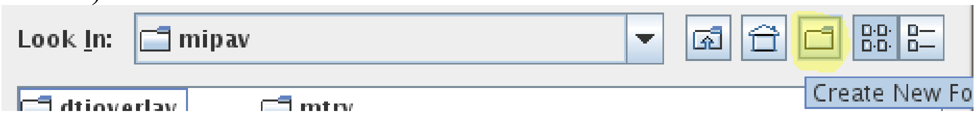
Select class files
mipav
PlugInAlgorithmDTICreateListFile$1.class
PlugInISN.class
PlugInDialogFITBIR$ChooseDataStructDialog$1.c
PlugInAlgorithmDICOMReordering$1.class
PlugInAlgorithmDrosophilaStandardColumnRegis
PlugInAlgorithmAnonymizeDicom$KeyComparat
PlugInDicomStitch.class
PlugInMuscleImageDisplay540a$AnalysisDialogF
PlugInAlgorithmEstimateFociNuclei.class
PlugInDicomDataElementsGenerator.class
PlugInNewGeneric.class
PlugInDialogGeneratePostTreatment541f.class
PlugInMuscleImageDisplay540a$ImageType.clas
PlugInDialogConvertToByteNIFTI.class
Selected class files

☑ Show only *.class, *.tar.gz, *.zip, and *.jar files.

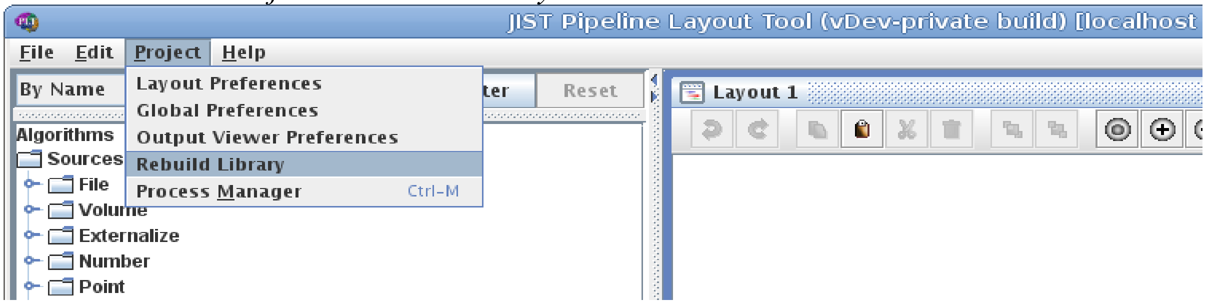
Next, you need to **select the directory** where you downloaded the CBS, JIST and TOADS-CRUISE packages with the *'Browse'* button. The packages will appear in the *'Select class files'* list, and you need to **select the packages** and press the arrow to add them to the *'Selected class files'* list. Then you can press the *'Install Plugin(s)'* button. After a few seconds, you should get a message window listing all the installed plug-ins (if some plug-ins are missing, the pessimistic message *'No plugins were installed, please select valid MIPAV plugin files.' appears*, although most plug-ins were in fact installed. In that case you need to restart MIPAV to see the new plugins). You can now *'Close'* the *'Install Plugins'* window. If you look again under the *'Plugins'* menu bar, you will see several new sub-menus including JIST, TOADS-CRUISE and CBS Tools. The installation is complete!

JIST Set-up

There are a few additional steps in order to use the JIST pipeline environment. First, select *'Plugins-> JIST-> JISTLayoutTool'* in the main menu bar of MIPAV. For this first time, a browser will appear and request you to select which directory to use to **store the JIST module definitions**. We recommend that you create a **new directory** for this called 'modules', for instance inside the 'mipav' directory of your home folder (to create a new folder directly from the browser, you can use the third button of the toolbar shown below).



When you select the new directory and press *'Open'*, the JIST layout tool will appear. The second thing to do now is to select *'Project-> Rebuild library'* in the menu:



A dialog will ask if you wish to delete all existing module definitions, which you can accept by default (it ensures the module definitions are kept clean if you do multiple installations). **Once the library has been rebuilt**, you should see that the module list on the left now has many *'Algorithms'* modules, including the CBS tools under *'CBS'*.

Finally, we recommend you to look at the ***'Global Preferences'*** (under *'Project'*) and select your preferred data format and compression (we suggest 'nii' and 'gzip', which produces compressed NIFTI files) as well as the layout format (we recommend layout if you need to encapsulate external tools, but LayoutXML may be preferable if you exchange layout over different computer systems). It is also recommended to keep the *'Debug level'* to 0 or 1 unless there are particular problems with some modules, as the debug procedures can slow down the computations.

Running the CBS Tools in MIPAV

The main CBS tools are directly accessible from the *'Plugins-> CBS Tools'* menu. Other tools included in the list of JIST modules can also be accessed from the *'Plugins-> JIST-> JISTPluginSelector'* application. Running these tools directly in MIPAV is convenient **for testing purposes**, but the preferred way to use them is within the JIST pipeline environment. Note that it is also possible to **call modules within a script** with the JIST command-line interface (see detailed instructions on: http://www.nitrc.org/plugins/mwiki/index.php/jist:CommandLineInterface).

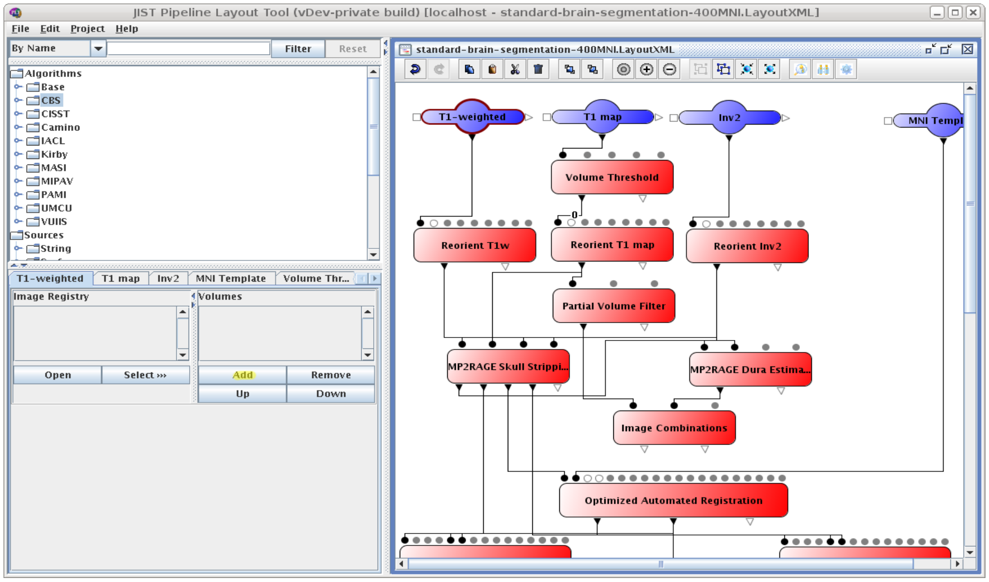
Running the CBS Tools in JIST

The JIST pipelining environment is a powerful system to perform complex image processing tasks on large data sets. To perform a series of computations, one simply line up the desired modules, connect their input and outputs and set their parameters. Each separate data set to process can be then be added to source modules, and the completed layout can then be saved and passed to the *'JIST Process Manager'* for actual processing.

In the following, we will describe the minimal steps needed to perform high resolution brain segmentation and cortical extraction from 7T MP2RAGE data with the included layout *'brain-segmentation-MP2RAGE-7T-400MNI-space.layout'*. Other included layouts can be used, and new layouts can easily be created for customized data processing, as described in the JIST documentation.

1. Opening the layout in the JIST layout manager: start the JIST layout manager from *'Plugins-> JIST-> JISTLayoutTool'*. Open the layout from *'File-> Open'* inside the JIST layout manager.

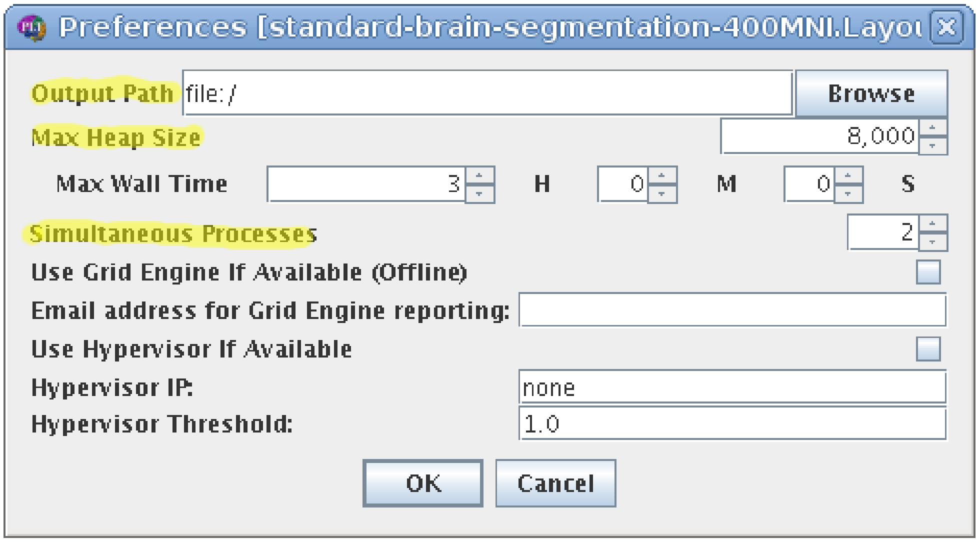
2. Selecting the data to process:



once the layout is loaded, you need to **add the data** you want to process to the source modules (in blue). When **multiple data sets** are loaded in the different source modules, they will be processed jointly line by line. A set of inputs can be reused multiple times by specifying a **parent/child relationship**: for each of the parent data set, the entire set of children data sets are to be processed (see the JIST documentation for more details and other ways of manipulating layouts).

For this layout, there are two **standard data sets** that need to be specified, apart from the user's data: the target template which defines the template space in which to perform the computations and the atlas file in the MGDMBrainSegmentation module. Both files can be found under *mipav/plugins/atlases/* in the user's home directory:
- the template is *'MNI-template/MNI152_T1_04mm_brain.nii.gz'*, a copy of the MNI152 template resampled at 0.4mm isotropic,
- the atlas file is *'brain-segmentation-prior/brain-atlas.txt'*, a statistical atlas based on Brainweb and the IBSR v2.0 segmentation database.

3. Setting up the output of the layout: the last thing to set up is to choose the destination directory for the JIST output. Because **JIST generates output data for every step** of the way, it is usually recommended to create an empty directory for that output. Once the processing is done, one can simply copy the desired results out of the JIST processing tree and into a specific destination, either manually or with the **JIST destination** modules.
   To set up the output, select *'Project-> Layout Preferences'* in the JIST layout manager and specify the output path. It is also recommended to set there the **maximum amount of memory** to be used by a single module (*'Max Heap Size'*, in MB) and the **maximum number of processes** to run in parallel (*'Simultaneous Processes'*). These numbers depend on the computer's hardware. For our high-resolution processing algorithms, large amounts of memory are required at times, and we recommend a max heap size of **15-24GB**. Having multiple processes reduces computation times, but it can slow down or crash some processes if the total amount of memory used goes beyond the computer's memory size.



Output Path file:/

Max Heap Size 8,000

Max Wall Time 3 H 0 M 0 S

Simultaneous Processes 2

Use Grid Engine If Available (Offline) □

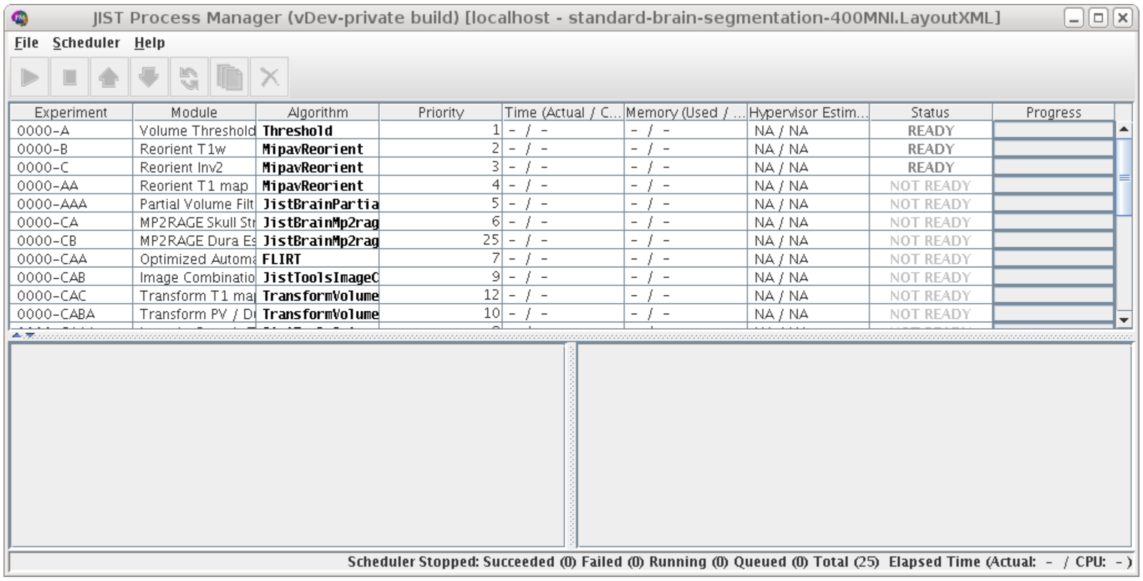
Email address for Grid Engine reporting:

Use Hypervisor If Available □

Hypervisor IP: none

Hypervisor Threshold: 1.0

4. Starting the pipeline: finally, the layout can be saved and is ready to run. You can open the JIST process manager (from *'Project-> Process Manager'* in the JIST layout manager or from *'Plugins-> JIST-> JIST Process Manager'* in MIPAV) and load the layout (it is loaded automatically if you call the process manager from the layout manager).



JIST Process Manager (vDev-private build) [localhost - standard-brain-segmentation-400MNI.LayoutXML]

Experiment	Module	Algorithm	Priority	Time (Actual / C...	Memory (Used / ...	Hypervisor Estim...	Status	Progress
0000-A	Volume Threshold	Threshold	1	- / -	- / -	NA / NA	READY	
0000-B	Reorient T1w	MipavReorient	2	- / -	- / -	NA / NA	READY	
0000-C	Reorient Inv2	MipavReorient	3	- / -	- / -	NA / NA	READY	
0000-AA	Reorient T1 map	MipavReorient	4	- / -	- / -	NA / NA	NOT READY	
0000-AAA	Partial Volume Filt	JistBrainPartia	5	- / -	- / -	NA / NA	NOT READY	
0000-CA	MP2RAGE Skull Str	JistBrainMp2rag	6	- / -	- / -	NA / NA	NOT READY	
0000-CB	MP2RAGE Dura Es	JistBrainMp2rag	25	- / -	- / -	NA / NA	NOT READY	
0000-CAA	Optimized Automa	FLIRT	7	- / -	- / -	NA / NA	NOT READY	
0000-CAB	Image Combinatio	JistToolsImageC	9	- / -	- / -	NA / NA	NOT READY	
0000-CAC	Transform T1 ma	TransformVolume	12	- / -	- / -	NA / NA	NOT READY	
0000-CABA	Transform PV / Di	TransformVolume	10	- / -	- / -	NA / NA	NOT READY	

Scheduler Stopped: Succeeded (0) Failed (0) Running (0) Queued (0) Total (25) Elapsed Time (Actual: - / CPU: -)

The process manager will list all the experiments, i.e. the steps performed by each module for each data set. For each of them, it displays the expected inputs and outputs (bottom window) and the processing status. **To start a single process**, select it and press the *'run'* button (multiple selections with the usual key combinations are possible). **To start the entire pipeline**, select instead *'Scheduler-> Start Scheduler'*. Note that the JIST layout manager is not required to run the pipeline, once the layout has been saved. If the layout is modified, the process manager may be halted and needs to be reloaded. Results that have been obtained with an earlier version of the layout are labeled as *'out of sync'* and must be recomputed (with the *'restart'* button) or overridden (left-click of the mouse on the selected experiment).

Additional layouts

The other layouts included with the CBS Tools follow the same principles as above. It is important to keep in mind the memory requirements of the main computational modules, which will also vary depending on the processing resolution. A computer with about **8GB of memory** should be able to handle data processed at up to ***0.8 mm*** resolution, while a ***0.4 mm*** resolution may require **24GB** or more. Computation times are also variable, between a few minutes and a few hours (the average computation time for the *'MGDM Brain Segmentation'* module varies from under 1h to about 6h for *0.8 mm* and *0.4 mm* resolutions, respectively).

For those layouts that use the *SPECTRE2010* algorithm for skull-stripping (a better option at 1.5T and 3T, because of the differences in signal outside of the brain region), you can find the SPECTRE atlas under *mipav/plugins/Atlas/spectre/oasis-3-v2.txt* in the user's home directory. A list of the **included atlases and other important files** is given below.

A new group of layouts for **version 2.0** of the CBS Tools include our newest **cortical analysis** methods that allow to sample and analyze data along realistic cortical profiles at ultra-high resolution. Measurements of **cortical thickness, surface geometry, and intensity profile statistics** are now possible based on the cortical surfaces generated with our segmentation method. We also included a layout to process data based on surfaces obtained with the popular FreeSurfer algorithm as well as a basic wrapper to run FreeSurfer inside a JIST pipeline (with some necessary editing to link to your local installation settings, see below for details).

Additional tools

MIPAV, JIST and CRUISE already come with a large number of tools for medical image processing and brain MRI analysis such as registration and normalization tools, advanced algorithms for skull stripping, inhomogeneity correction, tissue classification, cortical extraction and mapping, DWI data processing, etc. See the publications in the references section for examples and details.
However, it is also possible to **encapsulate external tools** into a JIST module using a simple wrapper, as long as the external tools can be called via a bash script (this has been primarily tested on Linux operating systems). See details on how to create your own external wrappers here: http://www.nitrc.org/plugins/mwiki/index.php/jist:ExecutableWrapper.

Troubleshooting

Many problems can occur when applying a complex processing pipeline to a new set of data; and MIPAV, JIST and the CBS Tools are no exceptions. Upon encountering an unsolvable problem, please send the following information along with the bug report:

- version of MIPAV, version of JIST, version of the CBS Tools,
- amount of memory given to the software,
- dimensions and resolutions of the data.

A copy of the actual **layout** and the **output and error messages** generated by JIST (in *'View debugging information'*) and/or the terminal running MIPAV are usually very helpful as well. Note also that the CBS development team may not be able to solve problems related to MIPAV or JIST, but not specific to the CBS Tools.

Location of atlases and other important files

Several CBS Tools and CRUISE modules use external data sets as atlases or templates for normalization or segmentation. They are included in the following folders inside *[my home directory]/ mipav/plugins/*:

- *atlases/brain-segmentation-prior*: the probabilistic atlas used by the *MGDM Whole Brain Segmentation, Extract Brain Region* and modules,
- *atlases/MNI-templates*: the standard MNI152 T1-weighted template as well as ROIs for the left and right cerebral hemisphere from the Harvard-Oxford atlas, all resampled at high resolution,
- *atlases/freesurfer-adapters*: a plug-in wrapper to run FreeSurfer inside a JIST pipeline as well as label euqivalence files to convert its output.
- *Atlas/spectre*: the brain atlas used by the SPECTRE2010 module to perform skull stripping.

Other atlases and templates that may be needed by other modules are included in these two directories.

References

A computational pipeline for subject-specific, ultra-high resolution cortical analysis at 7 Tesla.
Bazin, Weiss , Dinse, Schäfer, Trampel & Turner.
Meeting of the OHBM, 2012.

System for Integrated Neuroimaging Analysis and Processing of Structure.
Landman, Bogovic, Carass, Chen, Roy, Shiee, Yang, Kishore, Pham, Bazin, Resnick & Prince.
Neuroinformatics, 2012.

The Java Image Science Toolkit (JIST) for Rapid Prototyping and Publishing of Neuroimaging Software. Lucas, Bogovic, Carass, Bazin, Prince, Pham & Landman.
Neuroinformatics, 2010.