

BioImage Suite Web

Xenophon Papademetris and Dustin Scheinost

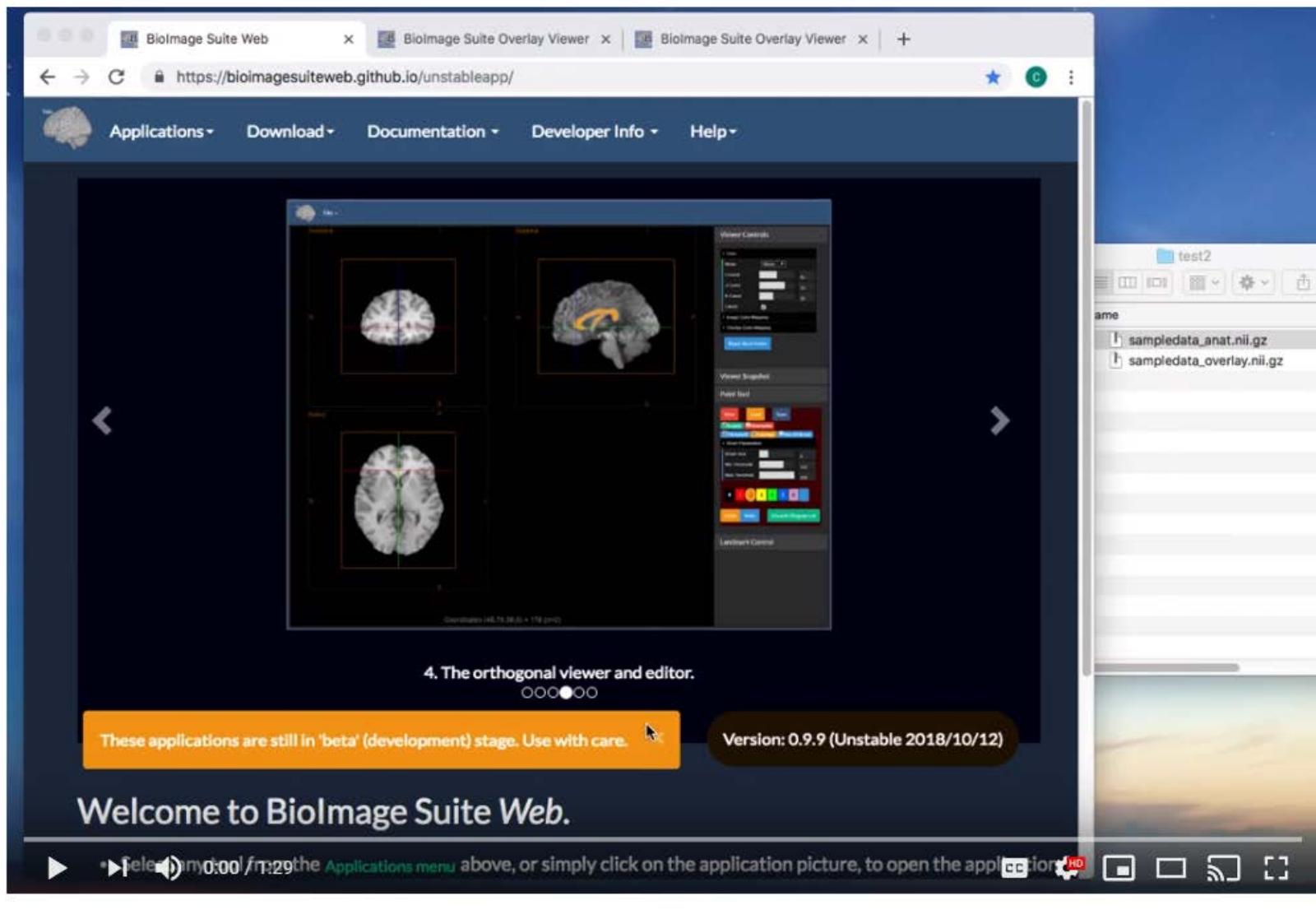
Dept. of Radiology & Biomedical Imaging

Yale School of Medicine

Adapted from a talk given at Yale, October 2018

What is It?

- BioImage Suite ported to run in a browser
- No SERVER – everything is running locally
- Desktop version (with essentially embedded browser) also available
- Ported over some of the functionality from the “legacy” version



<https://youtu.be/zWS68aOWUBQ>

Video: Cheryl Lacadie & Haley Garbus

BioImage Suite Web

www.bioimagesuite.org

Applications Download Documentation Developer Info Help

BioImage Suite Dual View

localhost:8080/main/doubleviewer.html

File Objectmap File 2 Objectmap 2

Coronal-R S Sagittal-R S Coronal-R S Sagittal-R S

Axial-R A L P Axial-R A L P

(54,17,43,0) = 167 (c=116) (54,17,43,0) = 167

-139 -71 -3 71 140

Viewer Controls Second Viewer Controls

Mode: Slides
I-Coord: 54
J-Coord: 17
K-Coord: 43
Labels:
Image Color Mapping
Min Int: 34.4
Max Int: 167
Interpolate:

Reset Slave View Z Z+

Viewer Snapshot

6. A dual registration viewer with linked cursors.

These applications are still in 'beta' (development) stage. Use with care. ×

Version: 0.9.9 (2018/10/12)

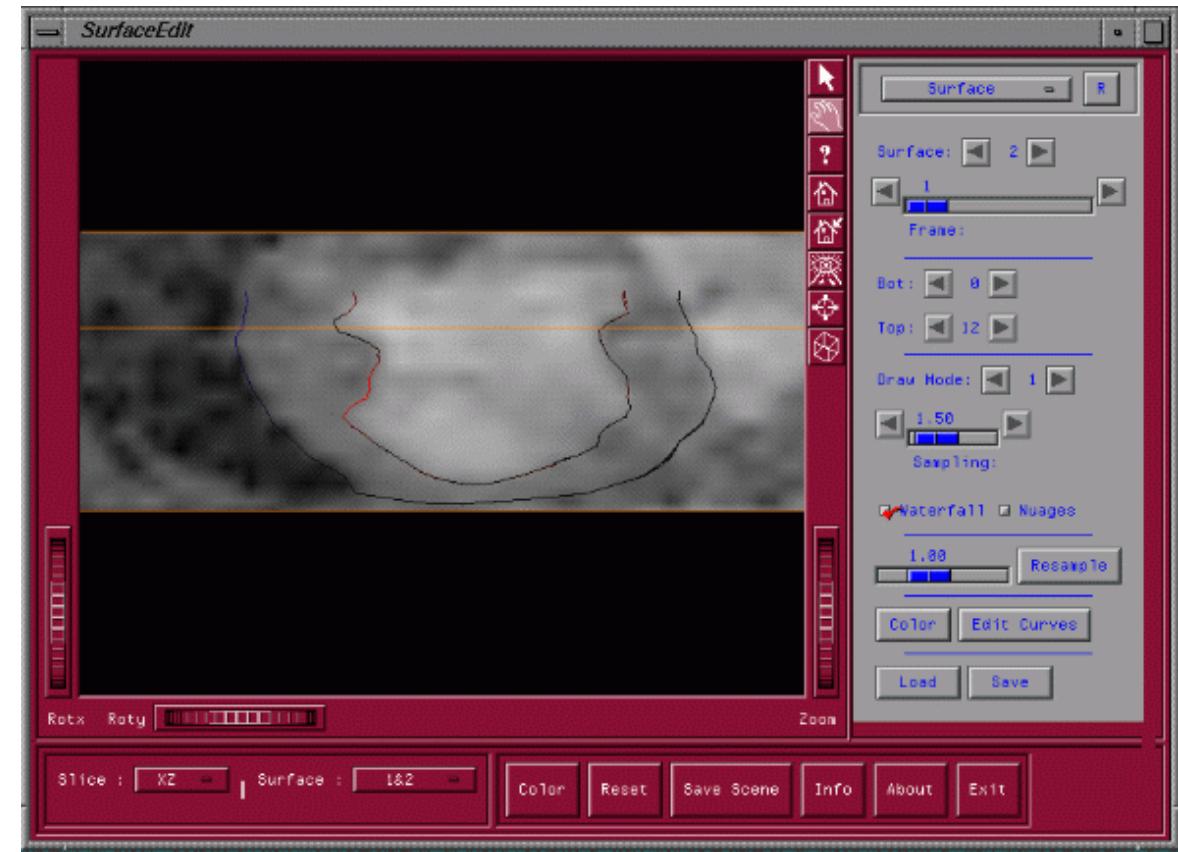
Welcome to BioImage Suite Web.

- Select any tool from the [Applications menu](#) above, or simply click on the application picture, to open the application. Please visit [our download site](#) for commandline and desktop versions of the software.
- This software is not approved for clinical use. Use at your own risk.**
- BioImage Suite Web is written in Javascript and has been mostly tested on Google Chrome. All applications are running on your local machine. **No data** is ever uploaded to any server.
- We gratefully acknowledge support from the [NIH Brain Initiative](#) under grant R24 MH114805 (*Papademetris X. and Scheinost D. PIs, Dept. of Radiology and Biomedical Imaging, Yale School of Medicine.*)

If you are looking for the old desktop based BioImage Suite software you may download it from the [Legacy BioImage Suite Webpage](#).

Biolimage Suite Versions

- 1995 – “Surface Editor” – cardiac segmentation on Silicon Graphics (MOTIF, Open Inventor)
- 2001 – Moved to multiplatform and adapted to brain use (with Todd) (VTK, Tcl)
- 2009 – 2017 : Many failed attempts to port to web
- 2017 – Biolimage Suite Web (JavaScript/WebGL)



Why Web?

- Nothing to install – just navigate to www.bioimagesuite.org
- Works on almost anything (best on Chrome) (try it on your phone!)
- Software is always up-to-date
- But web based does not preclude desktop – we provide
 - Command line apps
 - Desktop Apps
 - Web Applications
- Same code, (almost) same functionality

What can it do?

- Provide six applications (plus two hidden ones!)
- Overlay images to create image visualizations
- Connectivity Visualizations
- Interactive Image Segmentation
- Run various image processing, segmentation and registration algorithms
- MNI2TAL Conversions



MNI \longleftrightarrow Talairach Converter with Brodmann Areas (1.3)

A

MNI Y=0

R

B

MNI X=0

L

MNI Z=0

R

L

C**D**

Show Brodmann areas overlay

E MNI:

0 0 0

Go

F TAL:

0 -2 3

Go

G Area:

Outside defined BAs

Reset

TAL \rightarrow MNI Batch ConvertMNI \rightarrow TAL Batch Convert**H** About this application**I**

MNI \longleftrightarrow Talairach Converter with Brodmann Areas (1.3)

A

MNI Y=-21

R

MNI X=-12

L

MNI Z=4

R

L

Show Brodmann areas overlay

MNI: -12 -21 4 Go

TAL: -11 -22 7 Go

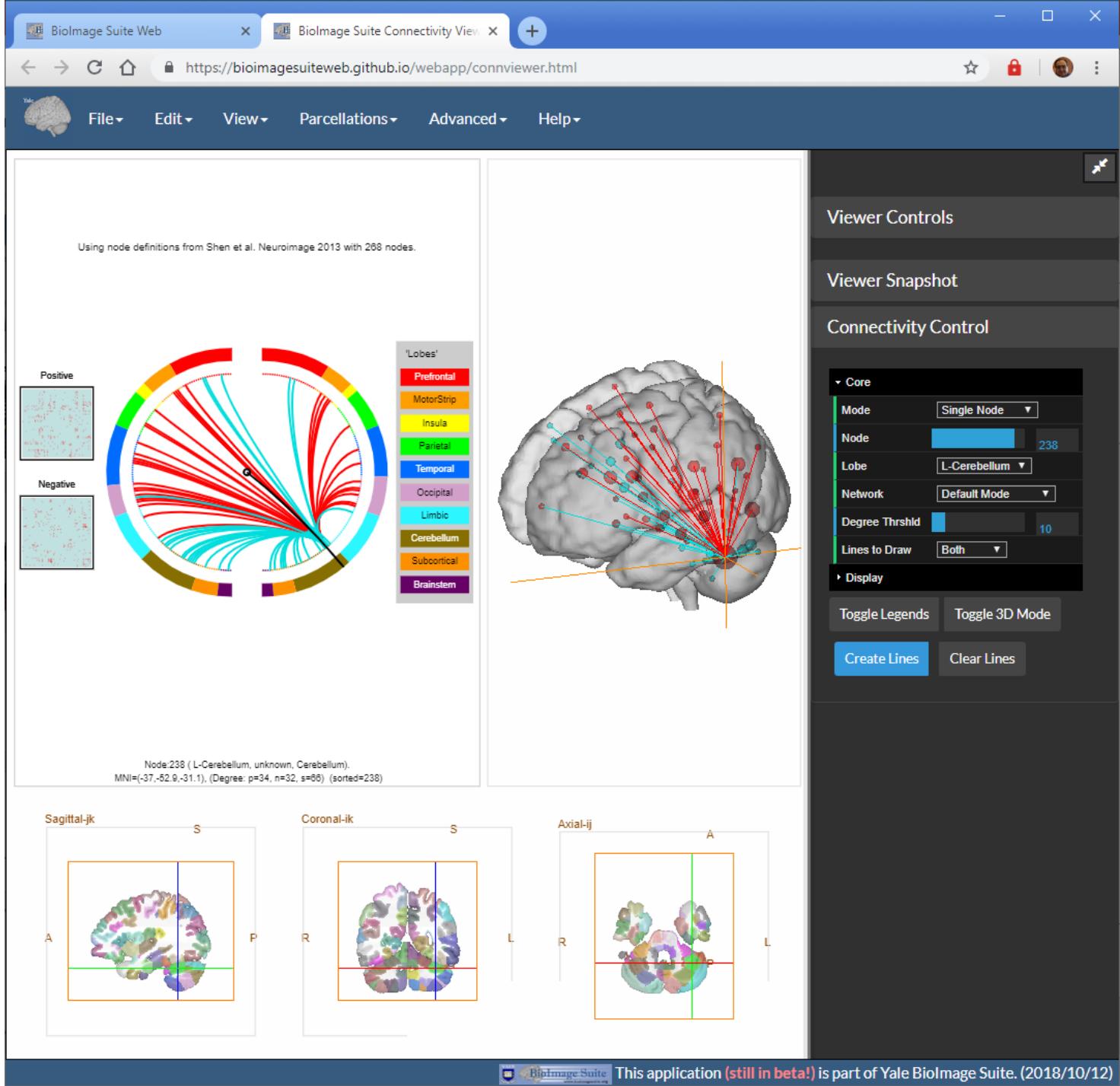
Area: Left-Thalamus (50) Reset

TAL \rightarrow MNI Batch Convert MNI \rightarrow TAL Batch Convert

About this application

Connectivity Viewer

Inputs= Parcellation +
Matrices



Overlay Viewer

Inputs= Anatomical+ Functional Images

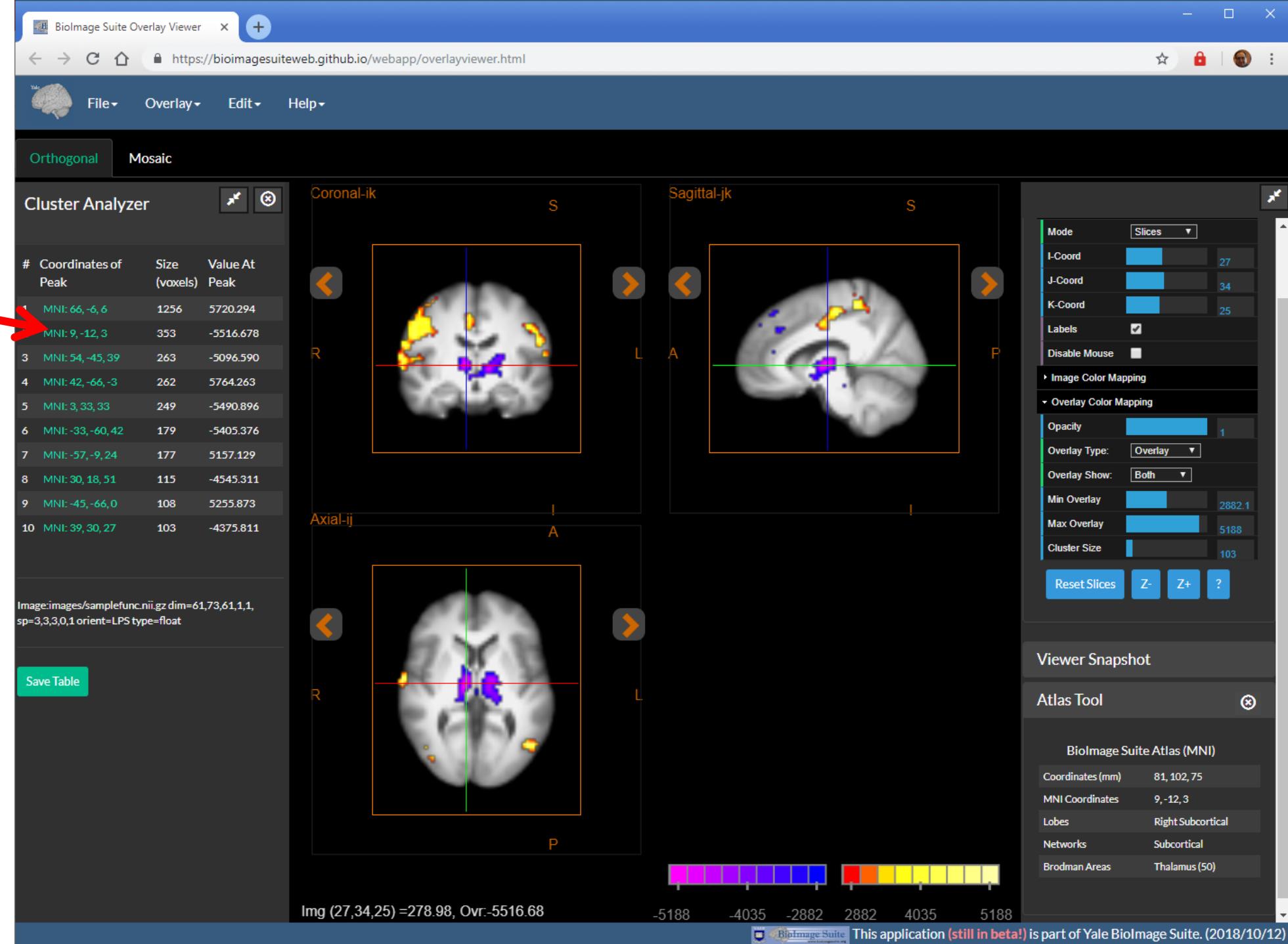


Image Editor

Interactive Segmentation

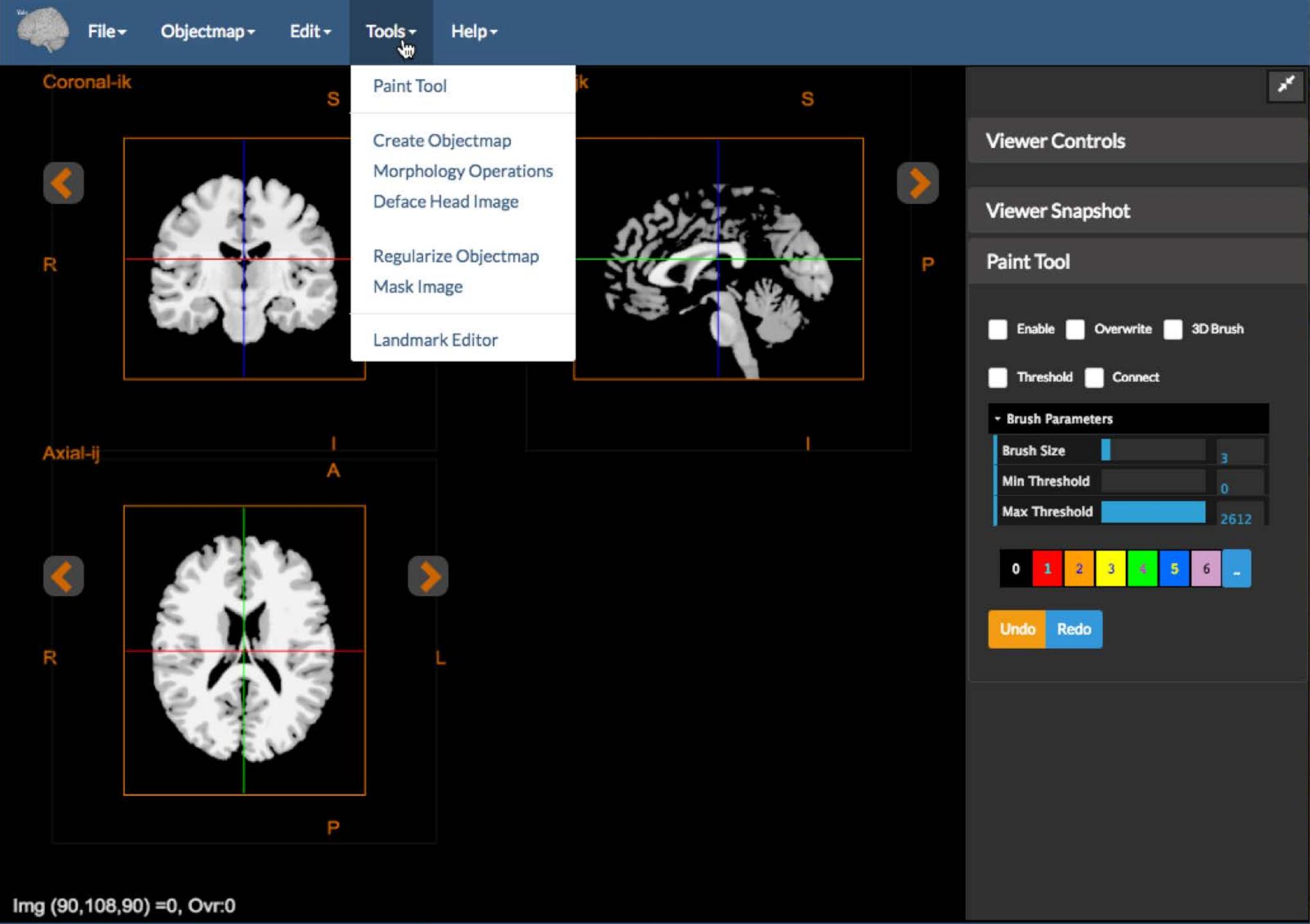


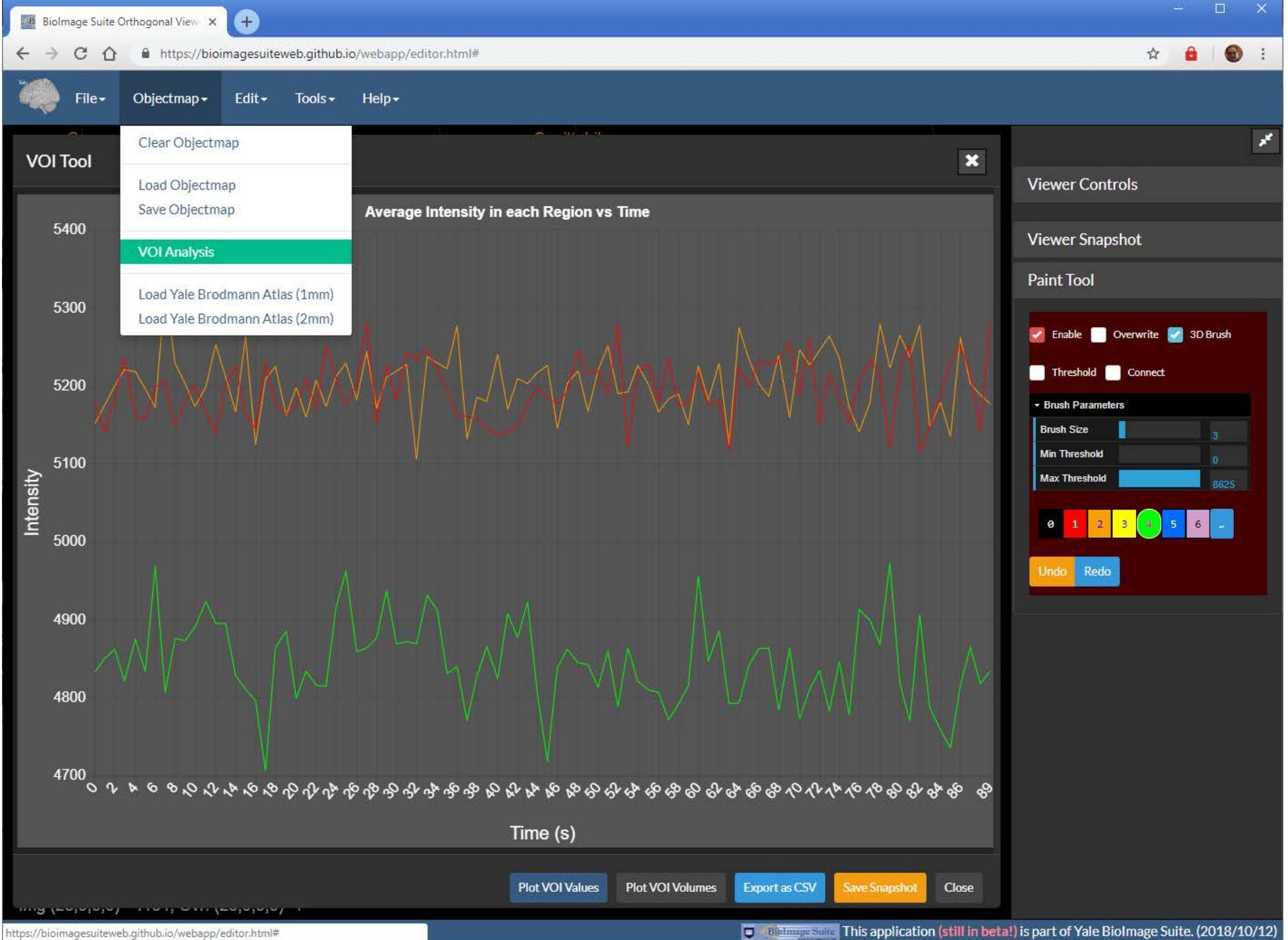
Image Editor II

VOI Analysis



Image Editor II

VOI Analysis



Single Viewer

Image Display
+ Processing

BioImage Suite Viewer Tool

https://bioimagesuiteweb.github.io/unstableapp/viewer.html

File Overlay Edit Image Processing Segmentation Help

Coronal-ik Sagittal-jk Axial-ij

R A S P L I

Smooth Image Normalize Image Threshold Image Cluster Threshold Correct Bias Field

Resample Image Shift+Scale(+Cast) Image Reorient Image Flip Image

Crop Image

Blank Image Extract Frame

Combine Images Process 4D Image

Viewer Controls

Core Mode Slices I-Coord J-Coord K-Coord Labels Disable Mouse

Image Color Mapping Overlay Color Mapping

Reset Slices Z- Z+ ?

Viewer Snapshot

Crop Image

Inputs Outputs Parameters

I-Start I-End J-Start J-End K-Start K-End

Advanced

Crop Undo More

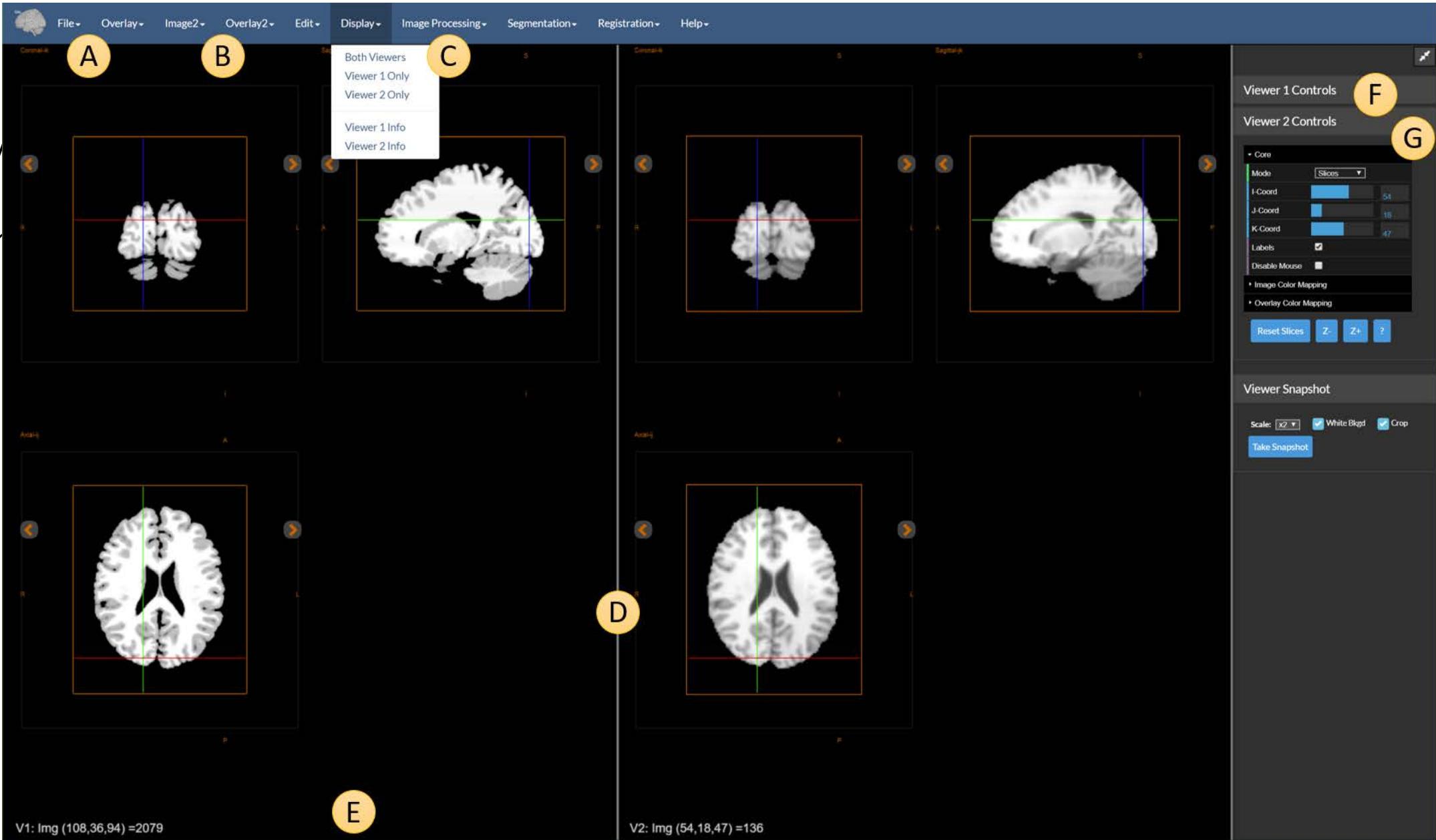
Img (45,54,45) =67

This application (still in beta!) is part of Yale BioImage Suite. (2018/10/22)

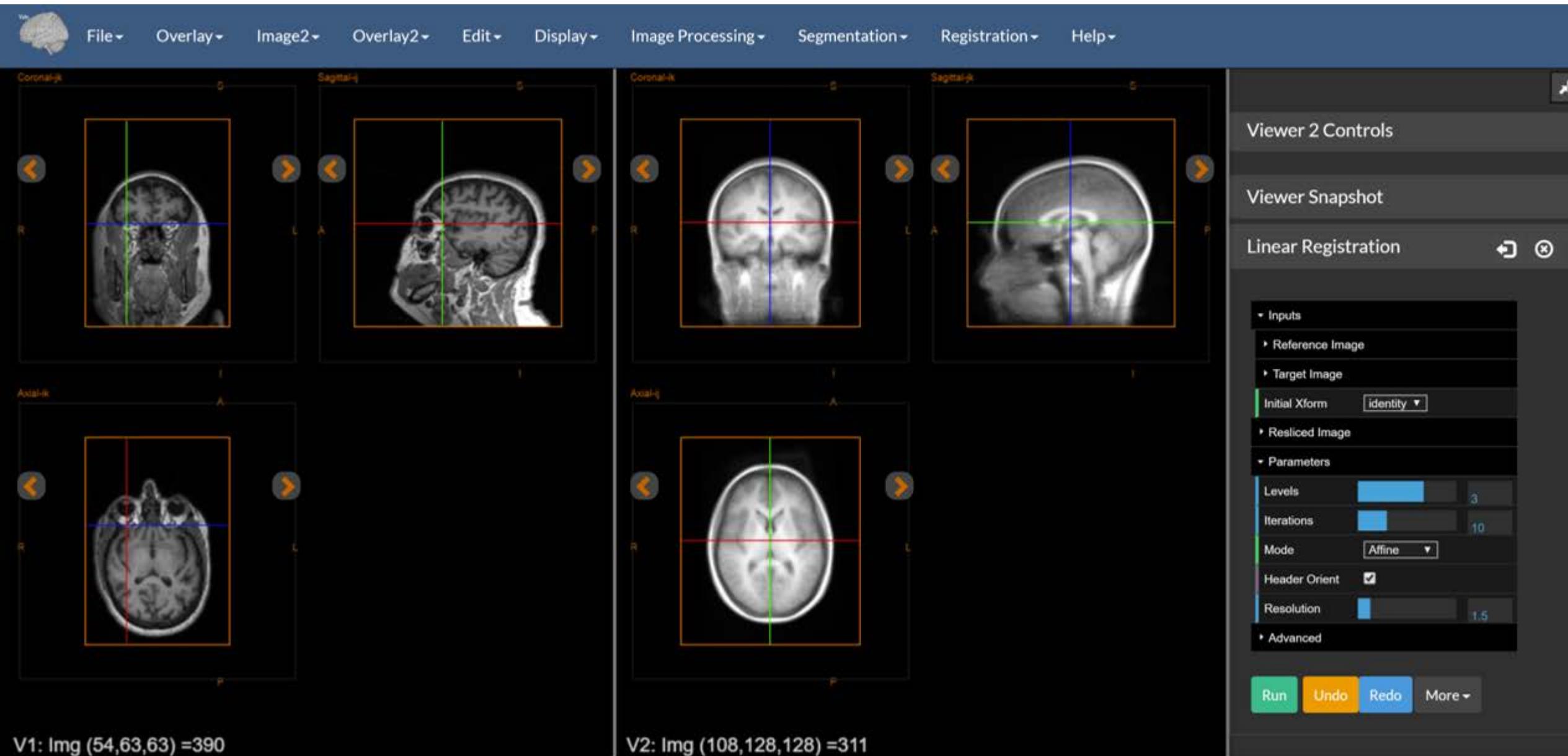
A screenshot of the BioImage Suite Viewer Tool interface. The main area displays three 3D brain volume slices: Coronal-ik, Sagittal-jk, and Axial-ij. Each slice has orientation markers (R, A, S, P, L, I). A context menu is open over the Coronal slice, with 'Crop Image' highlighted in green. A large red arrow points from this menu item down to the 'Parameters' section of the 'Crop Image' dialog box on the right. The 'Parameters' box shows numerical values for crop coordinates: I-Start=0, I-End=90, J-Start=0, J-End=108, K-Start=0, K-End=90.

Dual Viewer

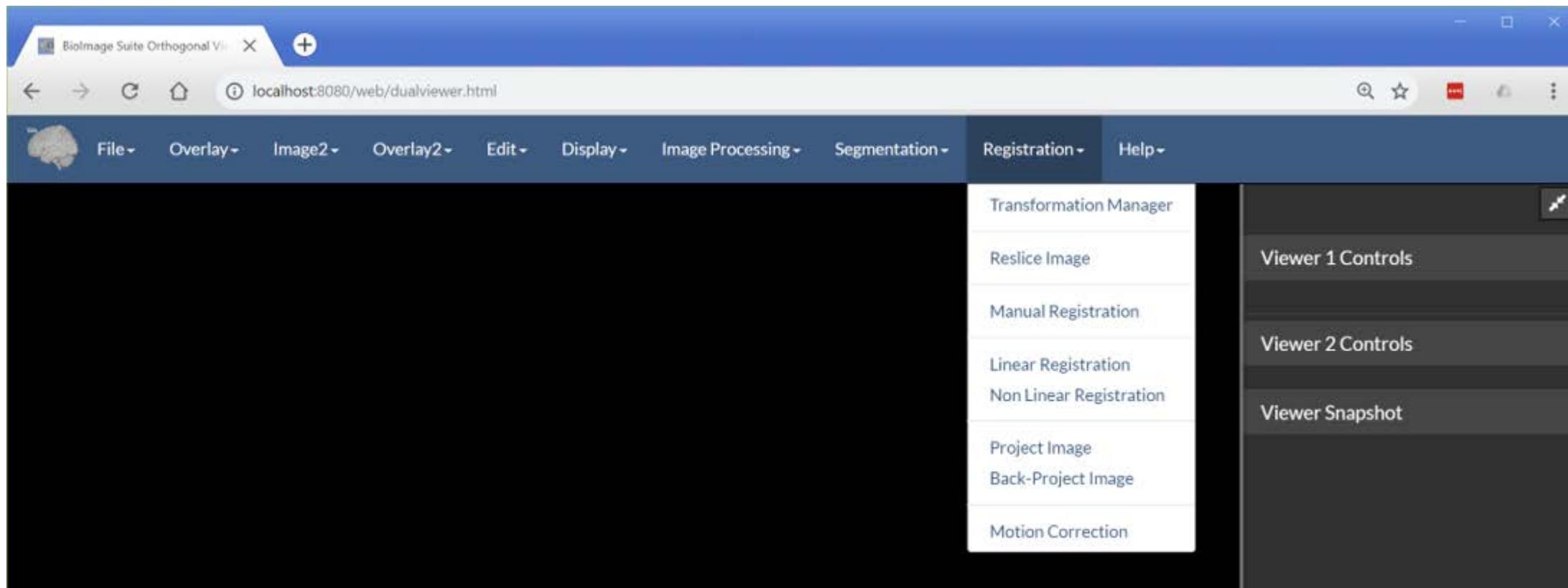
Image Display
+ Processing
+ Registration
+ ...



Registration I



Registration II



Registration III

File ▾ Overlay ▾ Image2 ▾ Overlay2 ▾ Edit ▾ Display ▾ Image Processing ▾ Segmentation ▾ Registration ▾ Help ▾

Viewer 2 Controls

Viewer Snapshot

Linear Registration

Inputs

- Reference Image
- Target Image
- Initial Xform identity
- Resliced Image

Parameters

- Levels 3
- Iterations 10
- Mode Affine
- Header Orient
- Resolution 1.5

Advanced

Run Undo Redo More ▾

```
++ Beginning to compute linear registration at level=1, numsteps=1, tolerance=0.001 centeronrefonly=1
++ ++++++ * * * * *
++ Initializing level r=1.500, rate=2.000 resol= 1.500
++ ++++++ * * * * *
++ Preprocessing Step1 linear registration:level_ref_image:
++ Extracting frame => dim=(128,128,88) spa=(2.000,2.000,2.000) rng=(0.000:886.000)
++ Smoothing (1.274,1.274,1.274) --> dim=(128,128,88) spa=(2.000,2.000,2.000) rng=(0.637,0.637,0.637)
++ Resampling (3.000,3.000,3.000) --> dim=(85,85,59) spa=(3.000,3.000,3.000) rng=(0.017:728.377)
++ Normalizing (0.819:0.990) 63 --> dim=(85,85,39) spa=(3.000,3.000,5.000) rng=(0.000:63.000) robust 1:99 %
info@0.817,554.142 numbins=64
++ ++++++ * * * * *
++ Preprocessing Step1 linear registration:level_targ_image:
++ Extracting frame => dim=(216,256,256) spa=(1.000,0.977,0.977) rng=(0.000:646.000)
++ Smoothing (0.622,0.622,0.622) --> dim=(216,256,256) spa=(1.000,0.977,0.977) rng=(0.000:631.582) vx:sigma=(0.622,0.637,0.637)
++ Resampling (1.465,1.465,1.465) --> dim=(147,170,170) spa=(1.465,1.465,1.465) rng=(0.000:626.669)
++ Normalizing (0.810:0.990) 63 --> dim=(147,170,170) spa=(1.465,1.465,1.465) rng=(0.000:63.000) robust 1:99 %
info@0.800,495.492 numbins=64
++ ++++++ * * * * *
++ Beginning level=1 resolution=3 numof=12 current_step=3
++ +
++ In step = 1. Iterations = 10, optimization=2, current=3.000
~~~ CG: Beginning (it=10), tol=0.001, stepsize=3
~~~ CG:Init 0 : (-13.99936 12.95625 -6.92582 -0.80411
-1.06547 3.39585 180.00000 108.00000 108.00000 0.00000 0.00000 )
-0.13928
~~~ CG:Lmin 1 : (-15.58428 11.68796 -5.72948 0.68189
-0.85405 3.88345 99.37441 98.55619 102.74436 -0.01089 0.01987
-0.14200
~~~ CG:Lmin 2 : (-16.36783 10.19819 -6.51256 0.67115
0.20496 3.39537 99.28657 97.13298 105.01382 -0.03828 0.04152
-0.14416
~~~ CG:Lmin 3 : (-16.37877 10.02462 -6.72674 0.40214
0.33843 2.89729 99.54964 97.00649 105.08165 -0.04760 0.04458
-0.14416
~~~ CG:Done 4 : (-16.37077 10.02462 -6.72674 0.40214
0.33843 2.89729 99.54964 97.00649 105.08165 -0.04760 0.04458
-0.14416
~~~ CG Stats: Nfunc=25, Ngrad=3, v=-0.14450
++ ++++++ * * * * *
++ +
++ ++++++ * * * * *
++ Stats : total_time 8
++ ++++++ * * * * *
```

V1: Img (54,63,63) =390, Ovr:297
V2: Img (108,129,129) =298

Testing and Verification

BioImage Suite Web

https://bioimagesuiteweb.github.io/unstableapp/

Yale Applications Download Documentation Developer Info Help

Main Repository
Plugin Repository

Run Module Regression Tests
Run Display Regression Tests
Run Display Regression Tests 2

Brodmann Areas (1.09)

MNI_Z=-3

MNI_Z=34

R L

Show Brodmann areas overlay

MNI: -3 -33 34 Go

TAL: -2 -31 32 Go

Area: Left-BA23 Reset

About this application

3. The MNI 2 Talairach Converter program.

These applications are still in 'beta' (development) stage. Use with care.

Version: 0.9.10 (Unstable 2018/10/22)

Welcome to BioImage Suite Web.

- Select any tool from the [Applications menu](#) above, or simply click on the application picture, to open the application. Please visit our [download site](#) for commandline and desktop versions of the software.
- This software is not approved for clinical use. Use at your own risk.**
- BioImage Suite Web is written in Javascript and has been mostly tested on Google Chrome. All applications are running on your local machine. **No data** is ever uploaded to any server.
- We gratefully acknowledge support from the [NIH Brain Initiative](#) under grant R24 MH114805 (*Papademetris X. and Scheinost D. Pls, Dept. of Radiology and Biomedical Imaging, Yale School of Medicine.*)

the software you may download it from the [Legacy BioImage Suite Webpage](#).

Module Testing I

The screenshot shows the BioImage Suite Web Regression Test Runner interface. At the top, there is a navigation bar with a brain icon, the title "BioImage Suite Web Regression Test Runner", and several buttons labeled A through E. Buttons A, C, D, and E are in blue circles, while B is in a red circle. Below the navigation bar, there is a search bar with fields for "First: 0", "Last: 87", and "Testname". A dropdown menu is open over the "Testname" field, showing a list of test names. To the right of the dropdown, there are three checkboxes: "WebWorker", "FileServer", and "Run Tests". The "Run Tests" button is highlighted with a red border. The main content area displays a welcome message, instructions for running tests, and a note about developer tools.

BioImage Suite Web Regression Test Runner

First: 0 Last: 87 Testname List (but not run) tests

A

Welcome to the BioImage Suite Web R

Select the tests you would like to run from the co

► Additional Help

Note: You may, before running tests, like to open developer Tools (on Chrome simply press 'Control-Shift-I' or 'Apple-Option-I') to observe the print outs from the algorithms.

B

C

D

E

Test all modules
approximateField
backProjectImage
binaryThresholdImage
blankImage
butterworthFilter
clusterThreshold
combineImages
computeCorrelation
computeGLM
computeROI
cropImage
defaceImage
displacementField
extractFrame
extractSlice
flipImage
linearRegistration
manualRegistration
maskImage
morphologyFilter
motionCorrection
motionReslice
nonlinearRegistration
normalizeImage
prepareRegistration
process4DImage
projectImage
regressGlobal
regressOut
regularizeObjectmap
reorientImage
resampleImage
resliceImage
segmentImage
shiftScaleImage
sliceBiasCorrect
smoothImage
thresholdImage

Run Tests

Run Memory Test

Module Testing II

https://bioimagesuiteweb.github.io/webapp/biswebtest.html

BioImage Suite Web Regression Test Runner

First: 0 Last: 87 Testname: computeCorrelation ▾ WebWorker **Run Tests**

Running Tests

Executing tests 0:87 (Max index=87). Only running tests with name=computeCorrelation

Test 3: computeCorrelation

- Command: computeCorrelation -i testdata/ButterWorthOutput.csv --zscore false
- Test details: --test_target testdata/newtests/goldcorrelation.matr --test_type matrix --test_comparision ssd --test_threshold 0.01
- Should pass: true

.... test execution time=0.21s
.... WASM memory size=16 MB.

Test completed, now checking results.

Comparing matrix using ssd and threshold=0.01 Module computeCorrelation test passed.
deviation (ssd) from expected: 2.537576312709809e-7 < 0.01

► Details

Tests for version=2018/09/03: completed=1/88, passed=1/88, failed=0/88, skipped=87/88

```
"command" : "computeCorrelation -i testdata/ButterWorthOutput.csv --zscore false",
"test"    : "--test_target testdata/newtests/goldcorrelation.matr --test_type matrix"
```

Image Display Testing

BioImage Suite Display Test <https://bioimagesuiteweb.github.io/unstableapp/biswebdisplaytest.html>

File Overlay Edit Help

BioImage Suite Web Display Regression Test Runner

First: 0 Last: 8 Run Single Test Run Multiple Tests

Orthogonal Mosaic

Viewer Controls

Core

- Plane Axial
- Rows
- Columns
- First
- Increment 1
- Labels

Image Color Mapping

Overlay Color Mapping

Reset Slices Z- Z+ ?

Viewer Snapshot

Scale: x1 White Bkgd Crop

Take Snapshot

Gold 8 Result 8

Test:8 cc=1.000 PASSED

Starting Test 5

Reading app state from images/testdata/overlay.biswebstate

Reading result from: images/testdata/test5.png

Result: {"testresult":true,"value":0.9999741777130562,"metric":"cc"}

Starting Test 6

Reading app state from images/testdata/overlay2.biswebstate

Reading result from: images/testdata/test6.png

Result: {"testresult":true,"value":0.9359878311099219,"metric":"cc"}

Starting Test 7

Reading app state from images/testdata/overlay2.biswebstate

Reading result from: images/testdata/test7.png

Result: {"testresult":true,"value":0.9911937795834109,"metric":"cc"}

Starting Test 8

Reading app state from images/testdata/overlay2.biswebstate

Reading result from: images/testdata/test8.png

Result: {"testresult":true,"value":0.9999736178960362,"metric":"cc"}

Tests for version=2018/10/22: completed=9/9, passed=9/9, failed=0/9

Connectivity Display Testing

BioImage Suite Display Test 2

https://bioimagesuiteweb.github.io/unstableapp/biswebdisplaytest2.html

File ▾ Edit ▾ View ▾ Parcellations ▾ Advanced ▾ Help ▾

BioImage Suite Web Display Regression Test Runner (2)

First: 1 Last: 5 Run Single Test Run Multiple Tests

Using node definitions from Shen et al. Neuroimage 2013 with 268 nodes.

Positive Negative

Lobes:

- Parietal
- MotorStrip
- Insula
- Precuneus
- Temporal
- Occipital
- Limbic
- Cerebellum
- Subcortical
- Brainstem

Node: 108 (R-Cerebellum, unknown, Cerebellum). MNi(16.2,-47.2,-52.3), (Degree: p=1, n=2, s=3) (sorted=108)

Viewer Controls

Viewer Snapshot

Connectivity Control

Core

- Mode: Single Node
- Node: 108
- Lobe: R-Cerebellum
- Network: Default Mode
- Degree Threshld: 10
- Lines to Draw: Both

Display

Toggle Legends Toggle 3D Mode

Create Lines Clear Lines

Tests Completed 2: completed=1/1, passed=1/1, failed=0/1

Gold 0 Result 0

Test:0 cc=0.994 PASSED

Starting Test 0

Reading app state from images/testdata/connstate.biswebstate

Reading result from: images/testdata/octest1.png

Result: {"testresult":true,"value":0.993808477333613,"metric":"cc"}

Manual Testing (Cheryl, Dana, Haley)

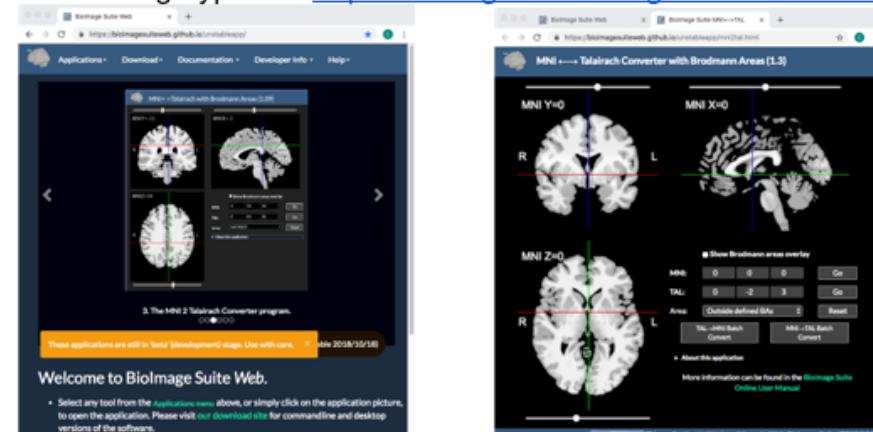
Testing Protocol for Bioimage Suite: MNI2TAL

Background: Imaging of the brain is a relatively new technique in medicine, neuroscience, and psychology. The Montreal Neurological Institute (MNI) template is used to identify reference spaces in imaging studies to identify structures and regions within the brain. The Talairach (TAL) Atlas is a mapping technique used to reduce the intersubject anatomic variability in mapping studies and allows for comparisons between coordinate-based data. Each point in the brain has both a set of MNI and Talairach coordinates.

About Bioimage Suite: Bioimage Suite is a collection of image analysis programs available as a software and online that can be used to view brain regions. One of the applications in Bioimage Suite, MNI2TAL, is used to provide the MNI and Talairach coordinates of a given point in the brain. Given any MNI coordinate, the program outputs the Talairach equivalent coordinate, and for any Talairach coordinate, the program provides the corresponding MNI.

Testing MNI2TAL on a Mac:

1. Open the following hyperlink <http://bioimagesuiteweb.github.io/unstableapp/>,



- a. Find and click on "3. The MNI 2 Talairach Converter program"

10/19/2018	Cheryl	MNI2TAL	9.1 - 10/18/18	MNI2TAL	Mac	Chrome	All	error	running TAL > MNI batch conversion yields the wrong coordinates. MNI > TAL batch conversion works fine.	Fixed it was using x
------------	--------	---------	----------------	---------	-----	--------	-----	-------	---	----------------------

Documentation

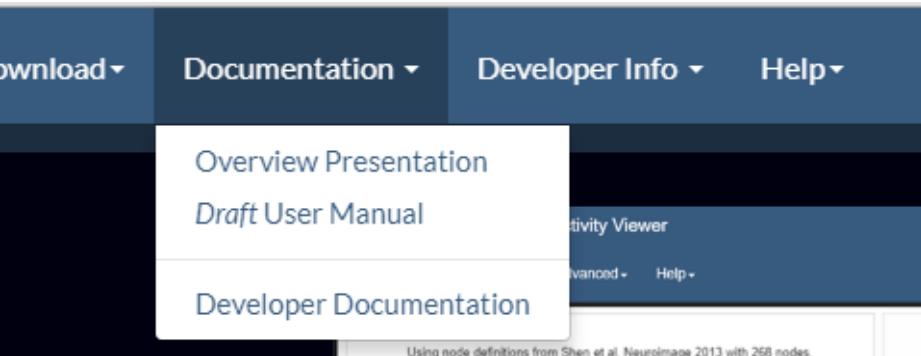


Table Of Contents | bisweb-manual X +
<https://bioimagesuiteweb.github.io/bisweb-manual/>

BiolImage Suite Web User Documentation

Table Of Contents

First we cover some core information in this document:

- [Starting BiolImage Suite Web](#) – how to get and run the software.
- [Some Key Information](#) – information about configuring your browser and default image orientations.

The following documents describe the different applications in BiolImage Suite Web.

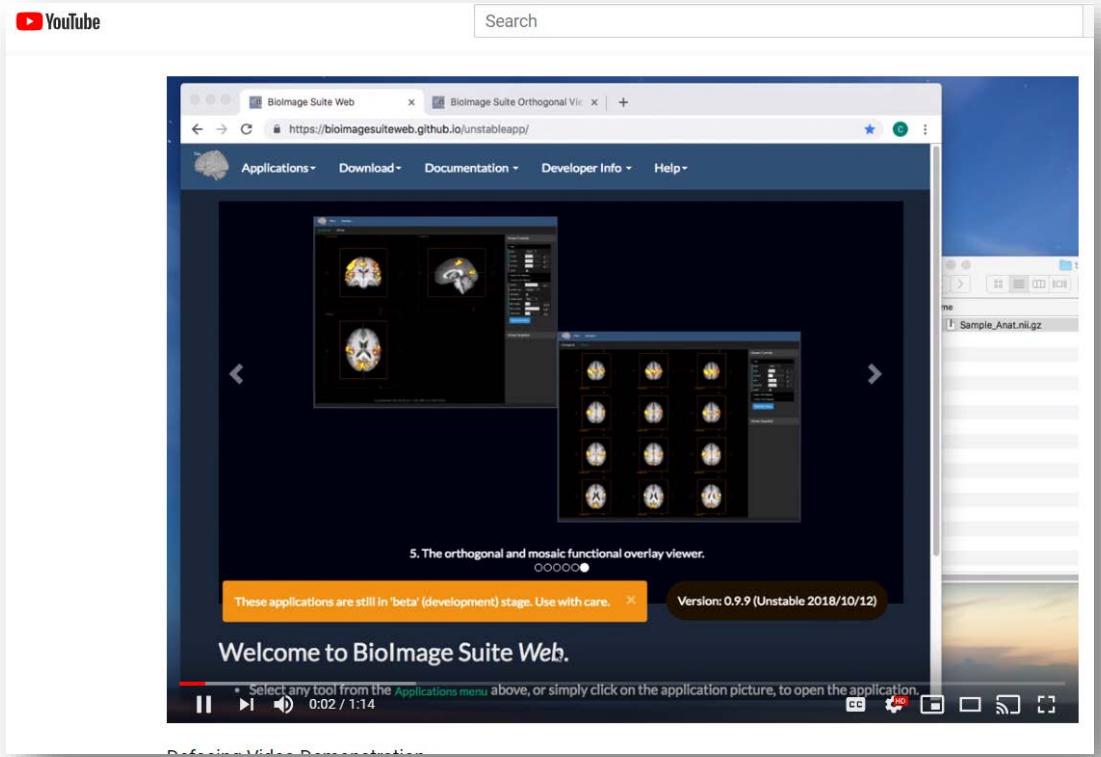
- [Displaying Images](#) – we discuss here how the core viewer in BiolImage Suite web works (including colormapping etc.). This is shared by almost all the applications.
- [The Overlay Viewer Application](#) – an application optimized for displaying functional overlays, including a brief description of the Mosaic Viewer Tool included here.
- [The Image Editor Application](#) – an application that can be used for interactive segmentation and VOI analysis of images.
- [The Connectivity Visualization Tool](#) – an application to visualize connectome matrices.
- [The MNI2TAL Tool](#) – an application to map MNI to Talairach coordinates and vice-versa.
- [The Dual Viewer Tool](#) – an advanced dual viewer application. This is intended to be primarily used for image registration tasks.

These documents describe how to perform specific tasks in one or more of the applications in the software.

- [Image Processing Tasks](#) – here we describe how to perform basic image processing tasks (e.g. image thresholding, cropping, resampling) using the tools in BiolImage Suite Web.
- [Image Segmentation](#) – this describes image segmentation functionality in our software.
- [Image Registration](#) – the image registration tools are described in the document describing the

Text + Video (Videos in Progress)

The screenshot shows a web browser window with the title "Image Defacing | bisweb-manual". The URL is <https://bioimagesuiteweb.github.io/bisweb-manual/tools/defacing.html>. The main content is titled "BiolImage Suite Web User Documentation" and "Image Defacing". The text describes the procedure for anonymizing MR brain images by masking faces. It mentions the OpenfMRI Project as a source. A note at the bottom suggests seeing the intro to viewers for help. A link at the bottom points to "Method 1 — Simple Version".

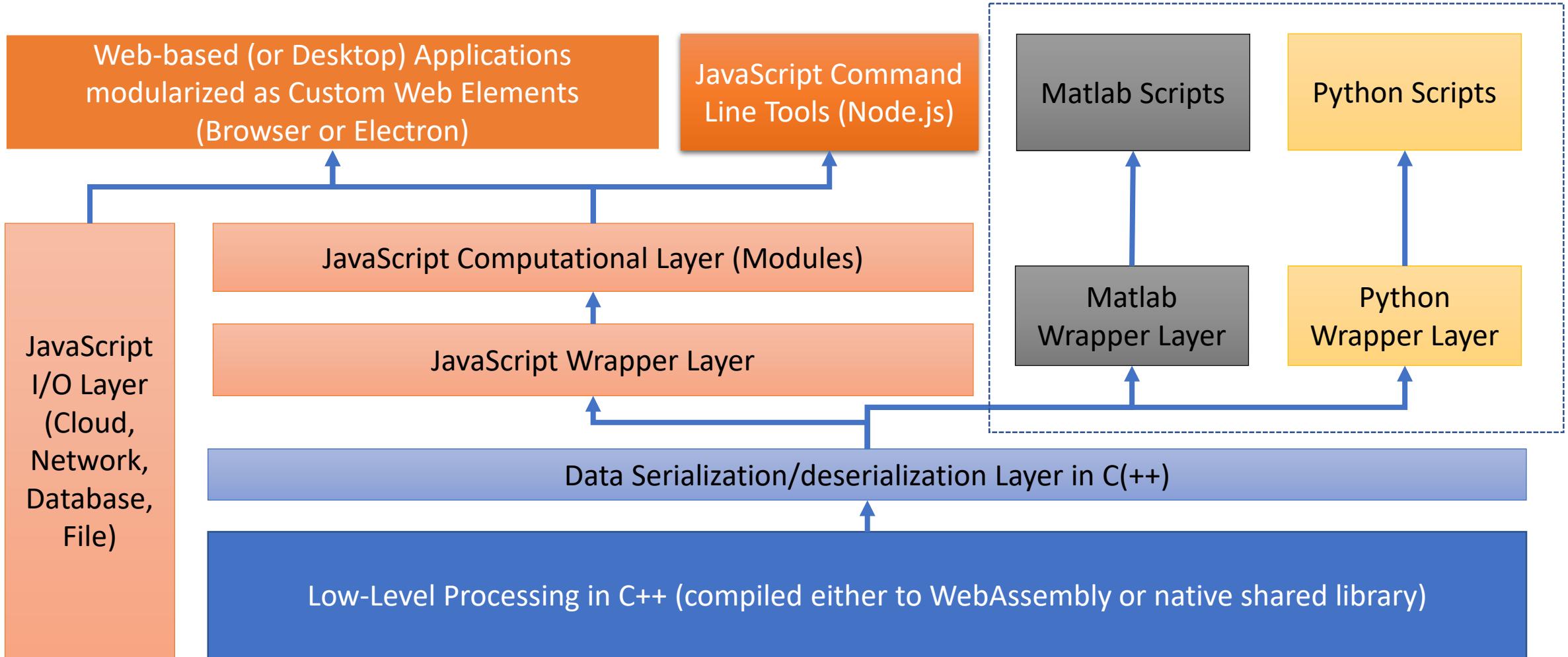


<https://youtu.be/a8Cbs4SJABs>
<https://bioimagesuiteweb.github.io/bisweb-manual/tools/defacing.html>

Key Technological (Software) Innovations

- Multi-context: same code commandline (via Node.js), desktop (via Electron) and Web
 - Each context has unique strengths
 - Tune to technical sophistication of the user and her needs
- Formal module architecture for automatic GUI and command line implementations of algorithmic modules.
- WebAssembly implementation of computationally expensive code (C++ compiled to WASM) e.g. Non Linear Registration ~90% native performance
- Custom Web Elements to modularize the applications
- No server, all computation is done in the client (your browser)
- A lot of this work builds on our previous experience with the Yale BioImage Suite (www.bioimagesuite.org) software package → This is really the web version of this.

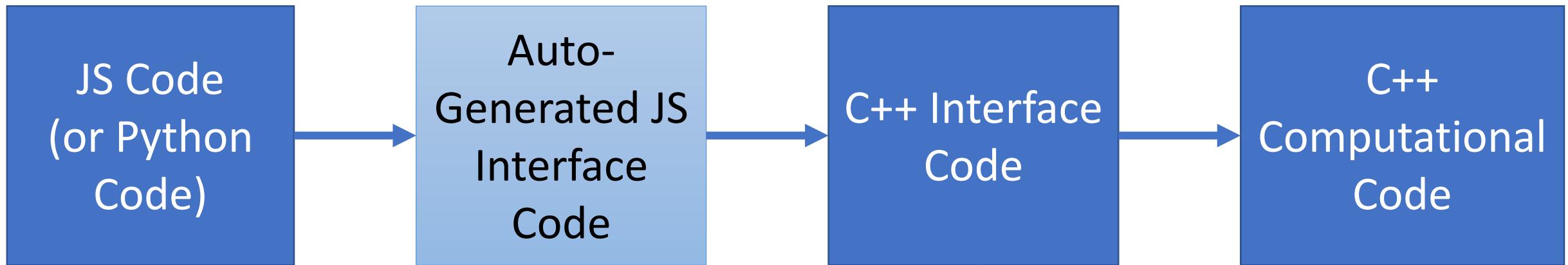
Software Architecture



C++/Web Assembly

- ~18,000 lines of C++ code (per wc) (much of this adapted from BioImage Suite but stripped of dependencies to VTK/ITK/..)
- Only external dependency is Eigen numerical library
- Compile into bytecode using Emscripten
- Serialization/Transfer layer
 - JS $\leftarrow \rightarrow$ C++ data transfer is restricted to simple “C”-style objects, essentially numbers and arrays

C++ JS Integration



```
/** Compute butterworthFilter output
 * @param input the input matrix to filter (time = rows)
 * @param jsonstring the parameters { 'type': "low", "cutoff": 0.15, 'sampleRate': 1.5 };
 * @param debug if > 0 print debug messages
 * @returns a pointer to the filtered matrix (rows=frames,cols=rois)
 */
// BIS: { 'butterworthFilterWASM', 'Matrix', [ 'Matrix', 'ParamObj', 'debug' ] }
BISEXPORT unsigned char* butterworthFilterWASM(unsigned char* input,const char* jsonstring,int debug);
```

JS Code

- 45,000 lines of JS Code
- Abstraction of File I/O
- Computational Modules
 - Automatic user interface generation
 - Automatic commandline generation
- User Interface packaged as Custom Web Elements

Formal Module Architecture

(Inputs, Outputs, Parameters specified using JSON)

JS

```
class SmoothImageModule extends BaseModule {  
  
    constructor() {  
        super();  
        this.name = 'smoothImage';  
    }  
  
    execute(vals) {  
        console.log('oooo executing: smoothImage with vals',vals);  
        return new Promise( (resolve, reject) => {  
            let input = this.inputs['input'];  
            let s = parseFloat(vals.sigma);  
  
            biswrap.initialize().then(() => {  
                this.outputs['output'] =  
                    biswrap.gaussianSmoothImageWASM(input, {  
                        "sigmas": [s, s, s],  
                        "inmm": super.parseBoolean(vals.inmm),  
                        "radiusfactor": parseFloat(vals.radiusfactor)  
                    }, super.parseBoolean(vals.debug));  
                resolve();  
            }).catch( (e) => {  
                reject(e);  
            });  
        });  
    }  
}
```

Python

```
class smoothImage(bis_basemodule.baseModule):  
  
    def __init__(self):  
        super().__init__();  
        self.name='smoothImage';  
  
    def execute(self,vals):  
        print('oooo executing: smoothImage with vals', vals);  
  
        input = self.inputs['input'];  
        s = (vals['sigma']);  
  
        try:  
            self.outputs['output']=  
                libbis.gaussianSmoothImageWASM(input,  
                    paramobj={  
                        "sigmas": [s, s, s],  
                        "inmm": self.parseBoolean(vals['inmm']),  
                        "radiusfactor": vals['radiusfactor'],},  
                        debug=self.parseBoolean(vals['debug']))  
        except:  
            return False  
  
        return True
```

Web Components

```
<body oncontextmenu="return false;" ondragstart="return fa
lse;" ondrop="return false;">>

<bisweb-topmenubar id="viewer_menubar">
</bisweb-topmenubar>

<div id="viewerwidget">
  <bisweb-viewerlayoutelement
    id="viewer_layout"
    bis-sidewidth="310"
    bis-coreopen="false"
    bis-wholescreen="1"
    bis-defaulttext="">
  </bisweb-viewerlayoutelement>
  <bisweb-colormapcontrollerelement id="viewer_cmap">
    </bisweb-colormapcontrollerelement>
  <bisweb-orthogonalviewer
    id="viewer"
    bis-layoutwidgetid="#viewer_layout"
    bis-colormapeditorid="#viewer_cmap">
  </bisweb-orthogonalviewer>
  <bisweb-snapshotelement
    bis-layoutwidgetid="#viewer_layout"
    bis-dowhite="false"
    bis-viewerid="#viewer">
  </bisweb-snapshotelement>
  <bisweb-simplealgorithmcontrollerelement
    id="algoelement"
    bis-viewerid="#viewer">
  </bisweb-simplealgorithmcontrollerelement>
  <bisweb-painttoolelement
    id="painttool"
    bis-layoutwidgetid="#viewer_layout"
    bis-viewerid="#viewer"
    bis-algorithmcontrollerid="#algoelement">
  </bisweb-painttoolelement>
  <bisweb-console id="bisconsole"></bisweb-console>
  <bisweb-viewerapplication
    bis-menubarid="#viewer_menubar"
    bis-painttoolid="#painttool"
    bis-consoleid="#bisconsole"
    bis-viewerid="#viewer">
  </bisweb-viewerapplication>
</div>
<bisweb-botmenubar></bisweb-botmenubar>
</body>
</html>
```

Web Components

```
<body oncontextmenu="return false;" ondragstart="return fa
lse;" ondrop="return false;">>

<bisweb-topmenubar id="viewer_menubar">
</bisweb-topmenubar>

<div id="viewerwidget">
  <bisweb-viewerlayoutelement
    id="viewer_layout"
    bis-sidewidth="310"
    bis-coreopen="false"
    bis-wholescreen="1"
    bis-defaulttext="">
  </bisweb-viewerlayoutelement>
  <bisweb-colormapcontrollerelement id="viewer_cmap">
    </bisweb-colormapcontrollerelement>
  <bisweb-orthogonalviewer
    id="viewer"
    bis-layoutwidgetid="#viewer_layout"
    bis-colormapeditorid="#viewer_cmap">
  </bisweb-orthogonalviewer>
  <bisweb-snapshotelement
    bis-layoutwidgetid="#viewer_layout"
    bis-dowhite="false"
    bis-viewerid="#viewer">
  </bisweb-snapshotelement>
  <bisweb-simplealgorithmcontrollerelement
    id="algoelement"
    bis-viewerid="#viewer">
  </bisweb-simplealgorithmcontrollerelement>
  <bisweb-painttoolelement
    id="painttool"
    bis-layoutwidgetid="#viewer_layout"
    bis-viewerid="#viewer"
    bis-algorithmcontrollerid="#algoelement">
  </bisweb-painttoolelement>
  <bisweb-console id="bisconsole"></bisweb-console>
  <bisweb-viewerapplication
    bis-menubarid="#viewer_menubar"
    bis-painttoolid="#painttool"
    bis-consoleid="#bisconsole"
    bis-viewerid="#viewer">
  </bisweb-viewerapplication>
</div>
<bisweb-botmenubar></bisweb-botmenubar>
</body>
</html>
```

Data Provenance

- Store data provenance information as JSON-extensions to files
- Who generated this file, when, where, with what and how?
- For images embed the JSON info as extensions to NIFTI Headers
- For matrices and transformations we are formalizing new JSON based file formats
- Same applies to higher level outputs (e.g. motion parameters)

Data Format Example

```
{  
    "bisformat": "BisDataobjectCollection",  
    "filename": "testdata/test_motion_correction_test_motion_correction.  
_mot.json",  
    "comments": [  
        {  
            "ModuleOutput": {  
                "command": "/usr/bin/node /home/xenios/javascript/biscp  
plib/js/bin/bisweb.js motionCorrection -t testdata/test_motion_correctio  
n.nii.gz -r testdata/test_motion_correction.nii.gz --doreslice true -o /  
home/xenios/winhome/Desktop/motionparams.json --resliced /home/xenios/wi  
nhome/Desktop/motioncorrect.nii.gz",  
                "output": "output",  
                "parameters": {  
                    "doreslice": true,  
                    "norm": true,  
                    "intscale": 1,  
                    "numbins": 1024,  
                    "extrasmoothing": 0,  
                    "metric": "CC",  
                    "optimization": "Hillclimb",  
                    "stepsize": 0.25,  
                    "levels": 3,  
                    "iterations": 32,  
                    "resolution": 1.01,  
                    "debug": false,  
                    "steps": 4,  
                    "refno": 0  
                },  
                "systeminfo": {  
                    "os": "linux",  
                    "arch": "x64",  
                    "hostname": "z230pc",  
                    "user": "xenios",  
                    "date": "2018-04-02T20:13:52.377Z",  
                    "nodeversion": "v8.9.4",  
                    "biswebversion": "04/02/2018"  
                }  
            }  
        }  
    ],  
    "numitems": 5,  
    "itemlist": [  
        {  
            "type": "transform",  
            "data": "{\"bisformat\": \"BisLinearTransformation\", \"filen  
ame\": \"identity.matr\", \"comments\": [], \"matrix\": [[1,0,0,0],[0,1,0,0  
,[0,0,1,0],[0,0,0,1]], \"parameters\": [0,0,0,0,0]}",  
            "metadata": {  
                "frame": 0  
            }  
        },  
        {  
            "type": "transform",  
            "data": "{\"bisformat\": \"BisLinearTransformation\", \"filen  
ame\": \"identity.matr\", \"comments\": [], \"matrix\": [[0.9049245715141296  
,-0.425572007894516,0,61.54688262939453],[0.425572007894516,0.904924571  
5141296,0,-39.15327453613281],[0,0,1,0],[0,0,0,1]], \"parameters\": [2.65  
1249885559082,-1.7674999237060547,0,0,0,-25.186874389648438]}",  
            "metadata": {  
                "frame": 1  
            }  
        },  
        {  
            "type": "transform",  
            "data": "{\"bisformat\": \"BisLinearTransformation\", \"filen  
ame\": \"identity.matr\", \"comments\": [], \"matrix\": [[0.9049245715141296  
,-0.425572007894516,0,-39.15327453613281],[-0.425572007894516,0.90492457  
15141296,0,61.54688262939453],[0,0,1,0],[0,0,0,1]], \"parameters\": [-1.7  
674999237060547,2.651249885559082,0,0,0,25.186874389648438]}",  
            "metadata": {  
                "frame": 2  
            }  
        },  
        {  
            "type": "transform",  
            "data": "{\"bisformat\": \"BisLinearTransformation\", \"filen  
ame\": \"identity.matr\", \"comments\": [], \"matrix\": [[0.9081797003746033  
,-0,-0.41858047246932983,34.94537353515625],[0,1,0,0],[0.418580472469329  
83,0,0.9081797003746033,-43.92396545410156],[0,0,0,1]], \"parameters\": [  
0.8837499618530273,0,-1.7674999237060547,0,-24.744998931884766,0]}",  
            "metadata": {  
                "frame": 3  
            }  
        }  
    ]  
}
```

Data Format Example

```
{  
    "bisformat": "BisDataobjectCollection",  
    "filename": "testdata/test_motion_correction__test_motion_correction.json",  
    "comments": [  
        {  
            "Moduleoutput": {  
                "command": "/usr/bin/node /home/xenios/javascript/bisplib/js/bin/bisweb.js motionCorrection -t testdata/test_motion_correction.nii.gz -r testdata/test_motion_correction.nii.gz --doreslice true -o /home/xenios/winhome/Desktop/motionparams.json --resliced /home/xenios/Desktop/motioncorrect.nii.gz",  
                "output": "output",  
                "parameters": {  
                    "doreslice": true,  
                    "norm": true,  
                    "intscale": 1,  
                    "numbins": 1024,  
                    "extrasmoothing": 0,  
                    "metric": "CC",  
                    "optimization": "Hillclimb",  
                    "stepsize": 0.25,  
                    "levels": 3,  
                    "iterations": 32,  
                    "resolution": 1.01,  
                    "debug": false,  
                    "steps": 4,  
                    "refno": 0  
                },  
                "systeminfo": {  
                    "os": "linux",  
                    "arch": "x64",  
                    "hostname": "z230pc",  
                    "user": "xenios",  
                    "date": "2018-04-02T20:13:52.377Z",  
                    "nodeversion": "v8.9.4",  
                    "biswebversion": "04/02/2018"  
                }  
            }  
        }  
    ],  
    "numitems": 5  
}
```

```
        "Moduleoutput": {  
            "command": "/usr/bin/node /home/xenios/javascript/bisplib/js/bin/bisweb.js motionCorrection -t testdata/test_motion_correction.nii.gz -r testdata/test_motion_correction.nii.gz --doreslice true -o /home/xenios/winhome/Desktop/motionparams.json --resliced /home/xenios/Desktop/motioncorrect.nii.gz",  
            "output": "output",  
            "parameters": {  
                "doreslice": true,  
                "norm": true,  
                "intscale": 1,  
                "numbins": 1024,  
                "extrasmoothing": 0,  
                "metric": "CC",  
                "optimization": "Hillclimb",  
                "stepsize": 0.25,  
                "levels": 3,  
                "iterations": 32,  
                "resolution": 1.01,  
                "debug": false,  
                "steps": 4,  
                "refno": 0  
            },  
            "systeminfo": {  
                "os": "linux",  
                "arch": "x64",  
                "hostname": "z230pc",  
                "user": "xenios",  
                "date": "2018-04-02T20:13:52.377Z",  
                "nodeversion": "v8.9.4",  
                "biswebversion": "04/02/2018"  
            }  
        }  
    ]},  
    "0.8837499618530273,0,-1.7674999237060547,0,-24.744998931884766,0]}],  
    "metadata": {  
        "frame": 3  
    }  
}
```

Data Format Example

```
{  
    "bisformat": "BisDataobjectCollection",  
    "filename": "testdata/test_motion_correction__test_motion_correction.json",  
    "comments": [  
        {  
            "Moduleoutput": {  
                "command": "/usr/bin/node /home/xenios/javascript/bisplib/js/bin/bisweb.js motionCorrection -t testdata/test_motion_correction.nii.gz -r testdata/test_motion_correction.nii.gz --doreslice true -o /home/xenios/winhome/Desktop/motionparams.json --resliced /home/xenios/Desktop/motioncorrect.nii.gz",  
                "output": "output",  
                "parameters": {  
                    "doreslice": true,  
                    "norm": true,  
                    "intscale": 1,  
                    "numbins": 1024,  
                    "extrasmoothing": 0,  
                    "metric": "CC",  
                    "optimization": "Hillclimb",  
                    "stepsize": 0.25,  
                    "levels": 3,  
                    "iterations": 32,  
                    "resolution": 1.01,  
                    "debug": false,  
                    "steps": 4,  
                    "refno": 0  
                },  
                "systeminfo": {  
                    "os": "linux",  
                    "arch": "x64",  
                    "hostname": "z230pc",  
                    "user": "xenios",  
                    "date": "2018-04-02T20:13:52.377Z",  
                    "nodeversion": "v8.9.4",  
                    "biswebversion": "04/02/2018"  
                }  
            }  
        }  
    ],  
    "numitems": 5  
}
```

```
        "Moduleoutput": {  
            "command": "/usr/bin/node /home/xenios/javascript/bisplib/js/bin/bisweb.js motionCorrection -t testdata/test_motion_correction.nii.gz -r testdata/test_motion_correction.nii.gz --doreslice true -o /home/xenios/winhome/Desktop/motionparams.json --resliced /home/xenios/Desktop/motioncorrect.nii.gz",  
            "output": "output",  
            "parameters": {  
                "doreslice": true,  
                "norm": true,  
                "intscale": 1,  
                "numbins": 1024,  
                "extrasmoothing": 0,  
                "metric": "CC",  
                "optimization": "Hillclimb",  
                "stepsize": 0.25,  
                "levels": 3,  
                "iterations": 32,  
                "resolution": 1.01,  
                "debug": false,  
                "steps": 4  
            },  
            "systeminfo": {  
                "os": "linux",  
                "arch": "x64",  
                "hostname": "z230pc",  
                "user": "xenios",  
                "date": "2018-04-02T20:13:52.377Z",  
                "nodeversion": "v8.9.4",  
                "biswebversion": "04/02/2018"  
            }  
        },  
        "metadata": {  
            "frame": 3  
        }  
    }  
}
```

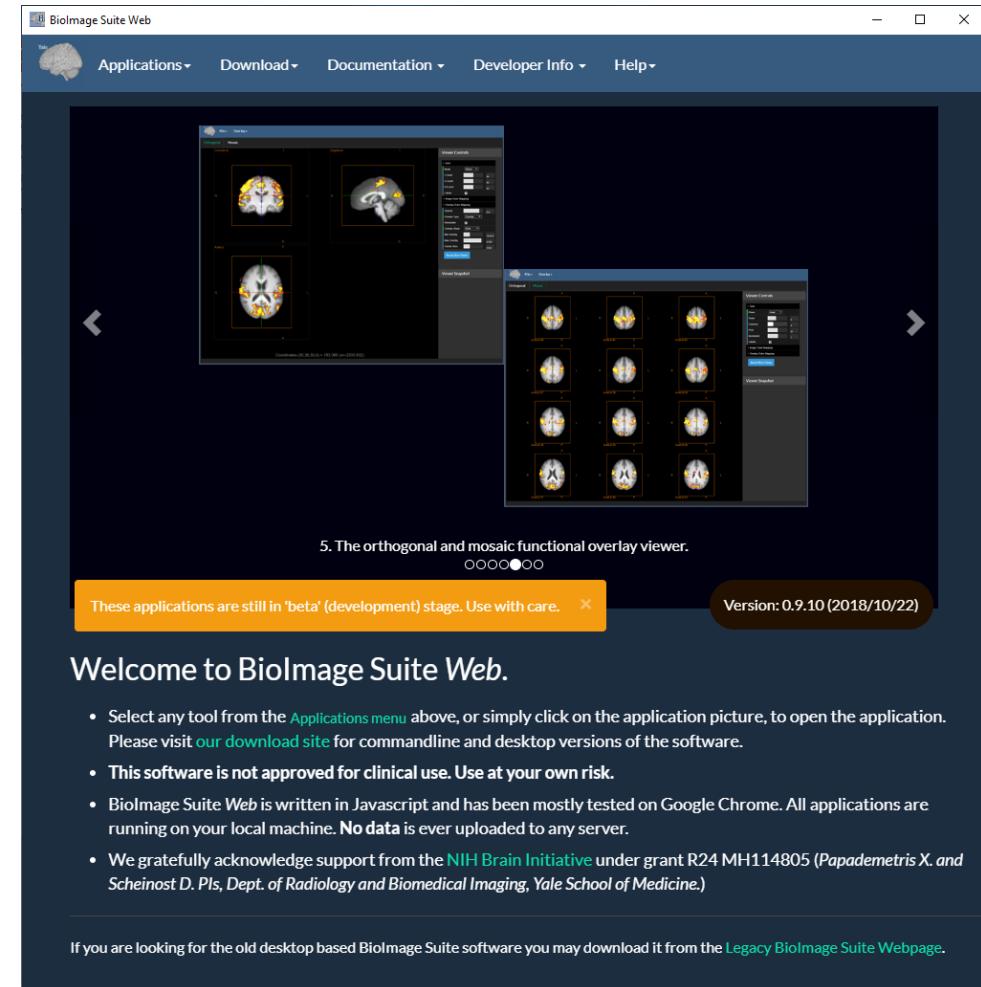
Some Limitations

- There are some cases were the web (alone) is not enough
- Reading multiple files from disk at the same time is not easy to do in a browser
 - Each file|open and file|save operation must be approved by the user (security)
- Solution I – Cloud Storage (Amazon S3 and Local Server)
 - We provide a simple script to allow browser to access your filesystem directly
- Solution II – Desktop Version
 - Electron-based Desktop Apps

Desktop Apps

- Must be downloaded + installed
- Essentially package a version of Chrome (Electron) and our code
- Can access filesystem directly
- Looks more or less same as web-based version

Desktop Version



Command Line Apps

- Some times it is useful to “batch” process data
- We provide command line applications of the same modules as in the web-based version (same exact code)
- Download and Install
- Also need node.js (commandline JS interpreter)

Bash on Ubuntu on Windows

```
bw_thresholdImage -h
```

.... Using node.js version 8.11.2 (OK)
.... This program is part of the commandline suite of tools from BioImage Suite Web.
.... See <https://github.com/bioimagesuiteweb/bisweb> for more information.
....
..... Setting forcing orientationOnLoad to: None (from None), None
.... bisweb commandline user preferences loaded from /home/xenios/.bisweb
.... {"orientationOnLoad":"None","snapshotScale":2,"snapshotDowhite":true,"filesource":"local","showwelcome":true,"favoriteFolders":[],"internal":false}
....
Usage: bisweb.js thresholdImage [options]
Options:
-V, --version output the version number
--low [n] The threshold below which values will be classified as 'out'
--high [n] The value above which values will be classified as 'in'
--replacein [s] If true, values classified as 'in' will be replaced by 'out'
--replaceout [s] If true, values classified as 'out' will be replaced by 'in'
--inval [n] Value to replace 'in' values with
--outval [n] Value to replace 'out' values with
--outtype [s] Output Type
--debug [s] Toggles debug logging
-i --input <s> The image to be processed
-o --output <s> Save the output image
--paramfile [s] Specifies that parameters should be read from file instead of parsed from the command line.
--silent Run in silent mode (no output on the console)
-h, --help output usage information

Threshold Image

Inputs Outputs Parameters Advanced

Low Threshold 21 High Threshold 208

Replace 'in' Replace 'out'

'in' Value 1 'out' Value 0

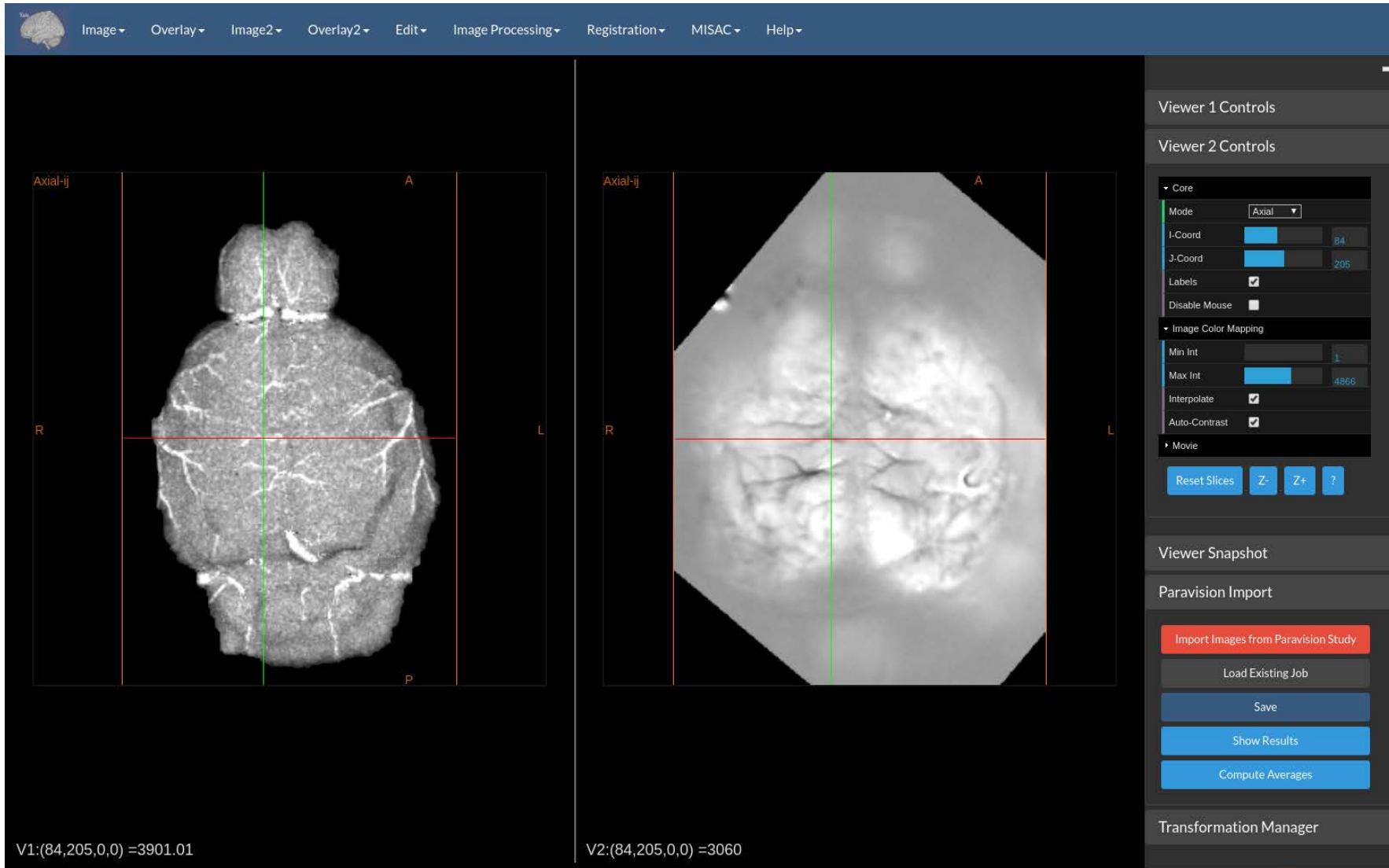
Output Type Same Debug

Threshold Undo More ▾

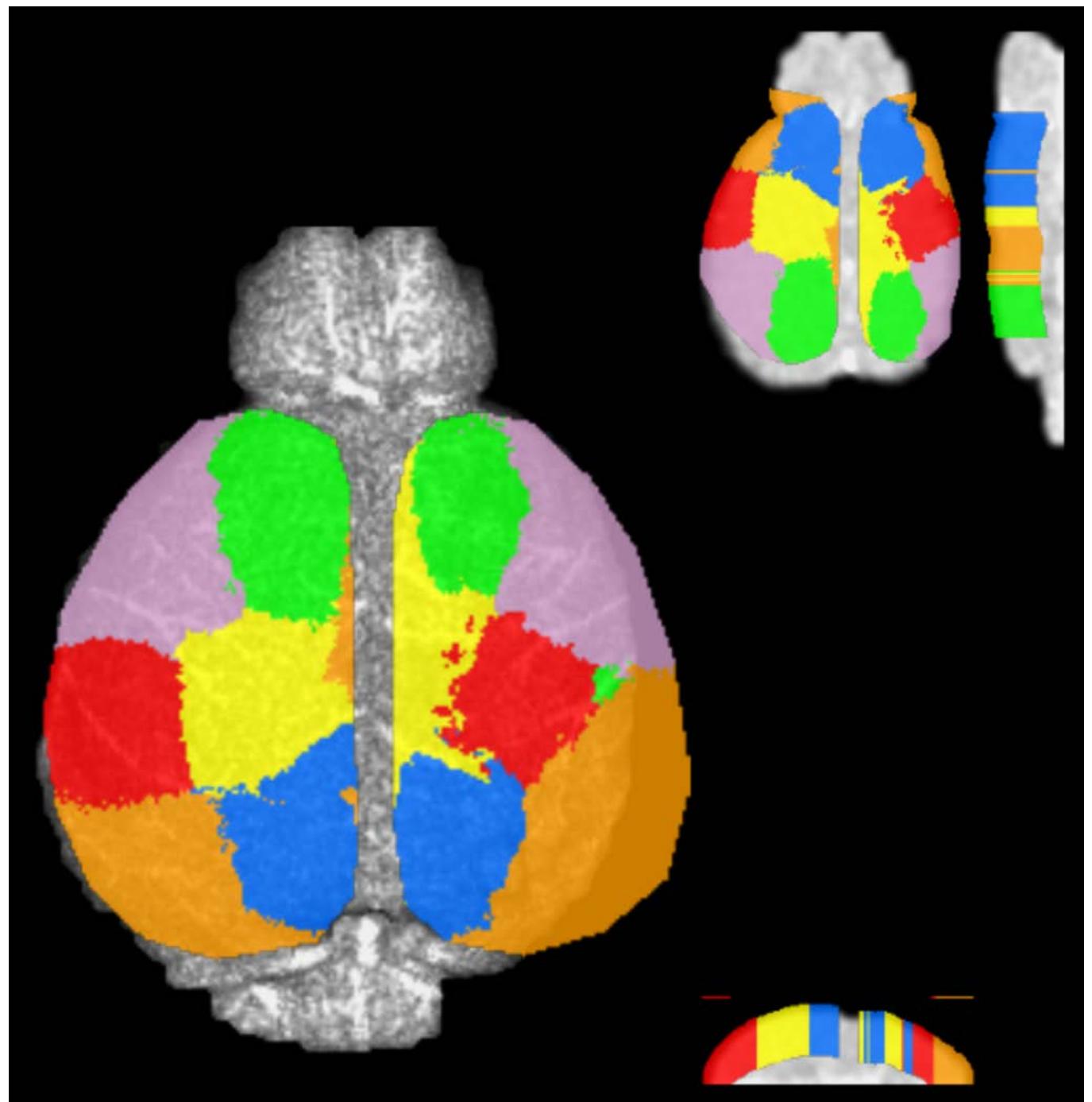
Some Cool Figures

- Simultaneous fMRI Ca(2+) Imaging Project (Crair, Constable PIs)
- Images using data from Eve Lake and Xinxin GE
- Custom code to merge 2D Projected Images and 3D MRI

Mapping 2D Optical to fMRI (via Angio MRI)



Mapping Parcellations from Optical to MRI



Lots more

- Google Drive and Dropbox integration (read only)
- Progressive Web Application (offline use via caching)
- All code on github
- Lots of programmer documentation on JS/WebAssembly etc. on Github (under docs).
- Python and Matlab (early) integration of computational code.
- Paravision Import functionality for small animal MRI
- Port of old diff-SPECT code for Epilepsy
- Landmark placement and editing
- Interactive Registration
- Copy & Paste Viewers across applications (via browser cache)
- Arbitrary Image Orientation and Registration (plus force “on load” if desired)

Current State

- Software is available (web, desktop, source code etc.)
- Closing in on v1.0 release (0.99!) – probably next week
 - Focus has been on getting low level functionality in place
- Next
 - Data organization (DICOM, BIDS)
 - Processing pipeline for resting state fMRI and optical Ca 2+ images
 - Better cloud integration
 - Machine learning via tensorflow.js
 - Form a basis for all future development work

Acknowledgments

- Zach Saltzman
- John Onofrey
- Cheryl Lacadie
- Haley Garbus
- Dana Casetti
- Daniel Coman
- Basav Ganganna
- Pranav Chandrasekaran
- Users, testers, ...
- Eigen
- CMake/CTest/CPack
- Emscripten
- Node.js
- Gulp
- Webpack
- Bootstrap
- JQuery
- dat.gui
- Three.js
- Electron
- Electron-Packager

We gratefully acknowledge support from the [NIH Brain Initiative](#) under grant R24 MH114805