

BiImage Suite Web

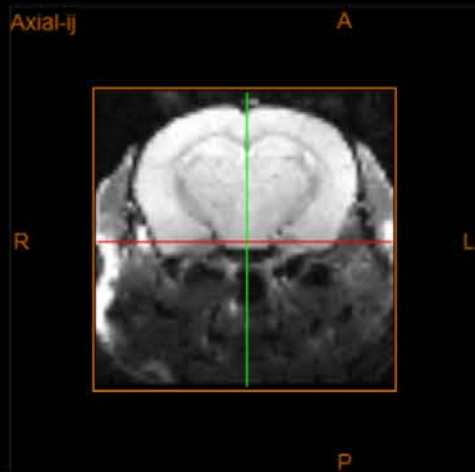
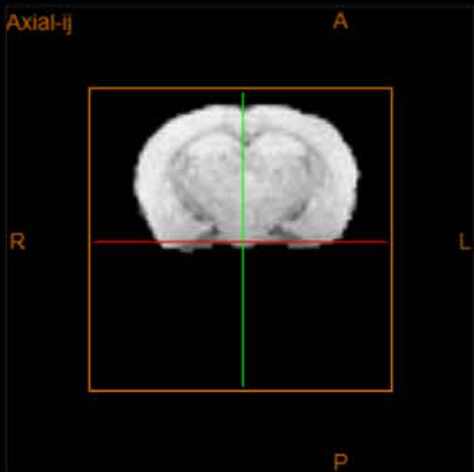
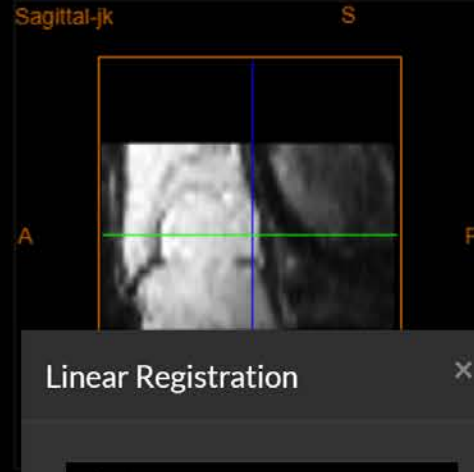
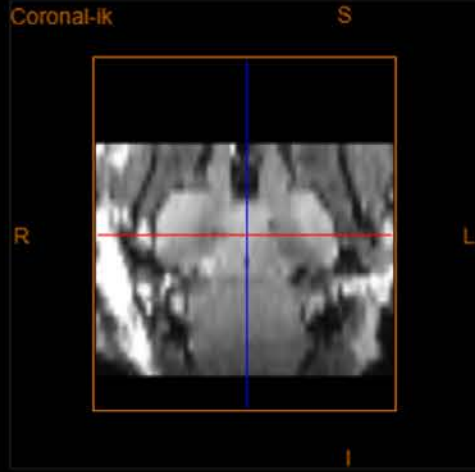
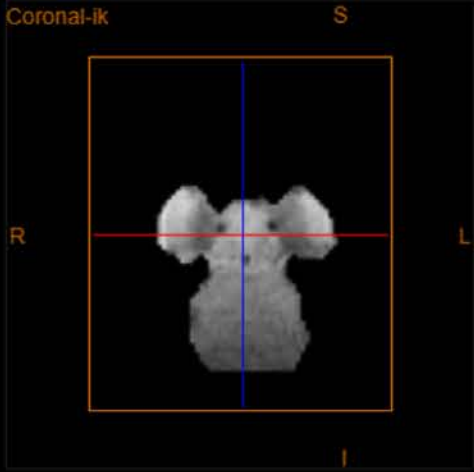
Xenophon (Xenios) Papademetris and Dustin Scheinost

Dept. of Radiology & Biomedical Imaging and Biomedical Engineering

Yale School of Medicine

Note

This Slides are adapted by a presentation made at the NIH Brain Initiative Annual Meeting in April 2018.



Linear Registration

Inputs

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

Parameters

- Levels: 3
- Iterations: 10
- mode: Rigid
- ResIn Factor: 1.5

Advanced

Run Undo Redo More

Viewer 1 Controls

Core

Mode: Slices

I-Coord: 32

J-Coord: 32

K-Coord: 37

Labels:

Disable Mouse:

Image Color Mapping

Overlay Color Mapping

Opacity: 0.2

Reset Slices Z- Z+ ?

Viewer 2 Controls

Viewer Snapshot

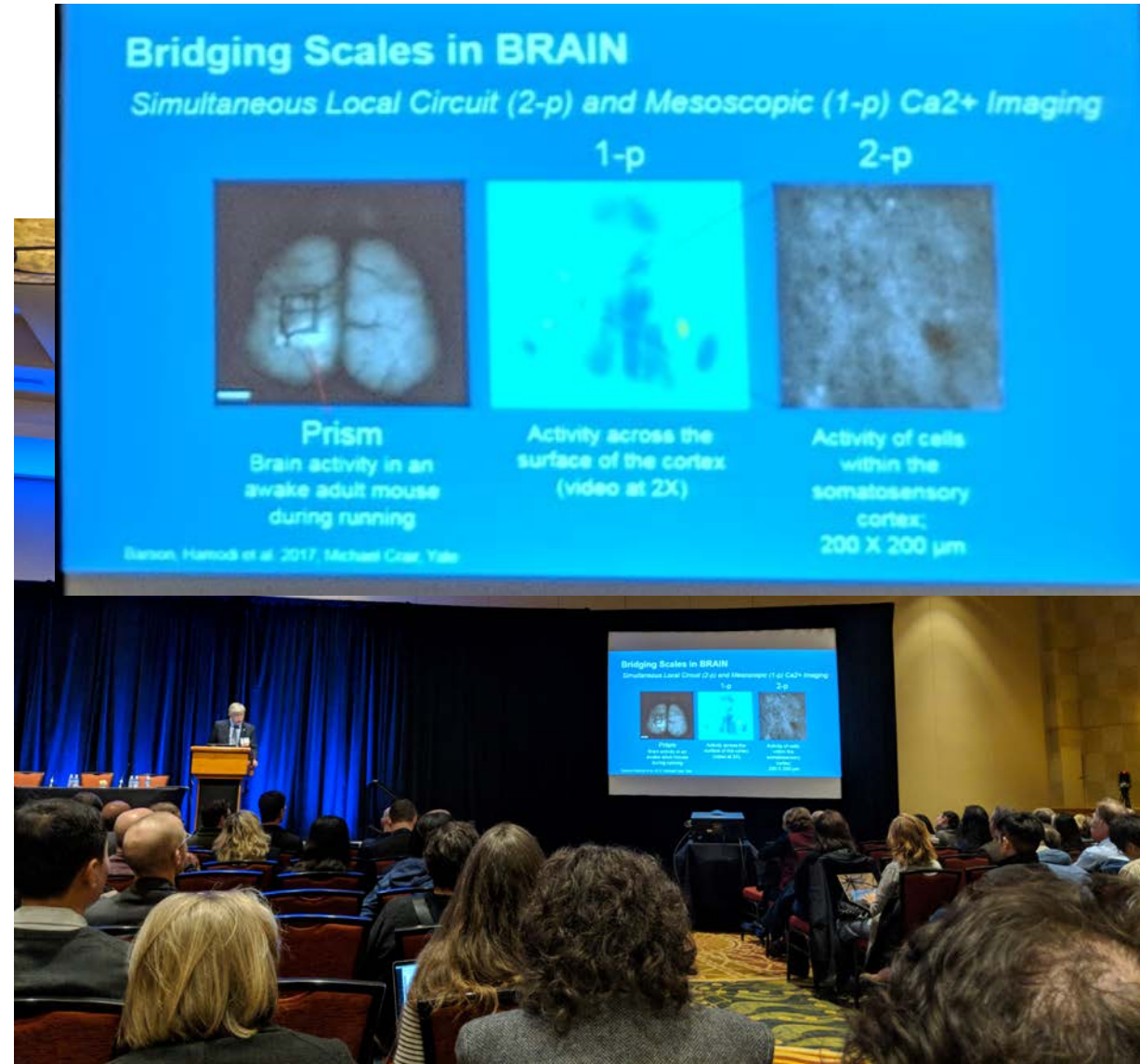
Transformation Manager

V1:(32,32,37,0) =8498

V2:(32,32,37,0) =5741

Collaborators/Partners/Beta Testing Labs

- Radiology & Biomedical Imaging
 - Todd Constable (Human fMRI)
 - Fahmeed Hyder (Small Animal fMRI)
- Neuroscience
 - Michael Crair (Mesoscale Calcium Optical Imaging)
 - Jessica Cardin & Michael Higley (Two Photon Optical Imaging)



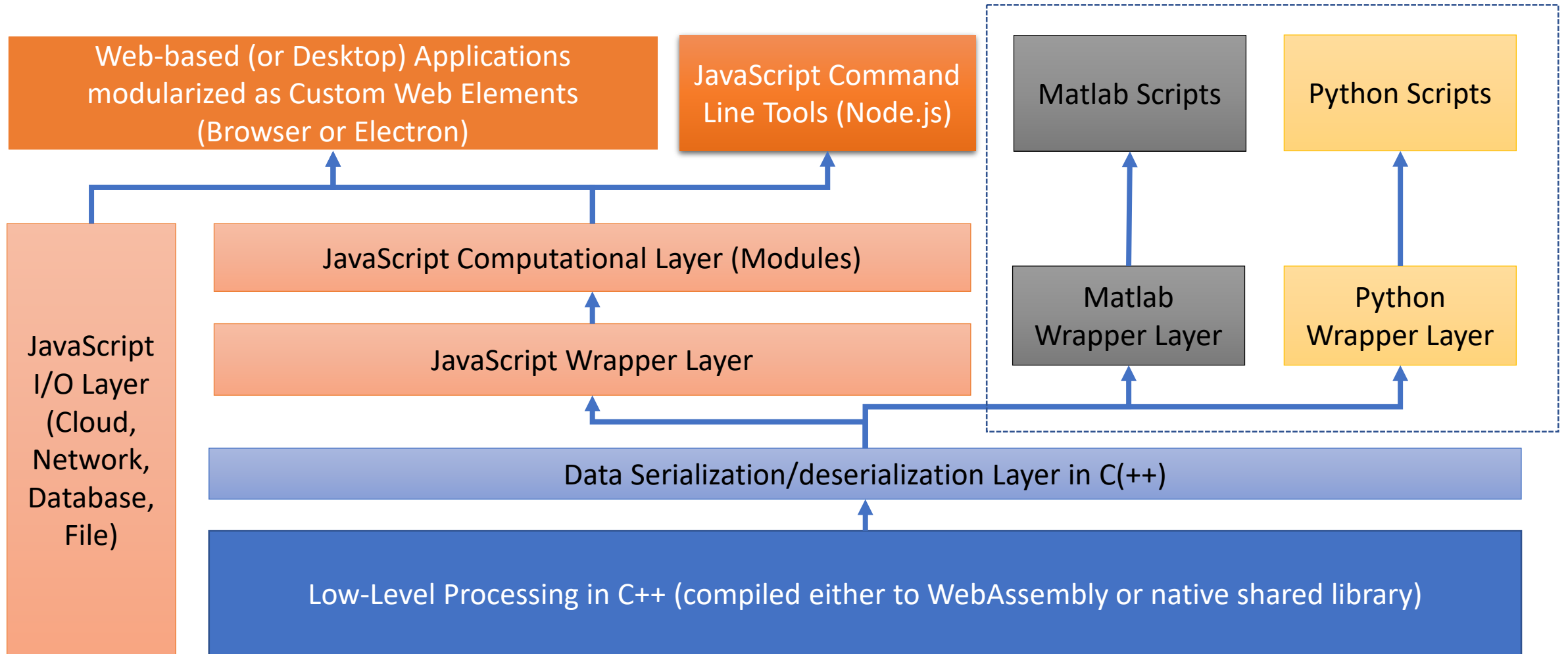
Aims

- Algorithms for Image Analysis of Multimodal/Multispecies neuroimaging data
 - *Registration (linear, nonlinear, 2D → 3D, motion correction)*
 - *Connectome Processing (Prefiltering, parcellation, connectivity ...)*
 - Atlas Integration
- Software Architecture
 - *JavaScript/Web-based with cloud integration*
 - *Data Provenance and file storage*
- Testing
 - *Regression*
 - Formalized human testing
- Documentation & Dissemination

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.

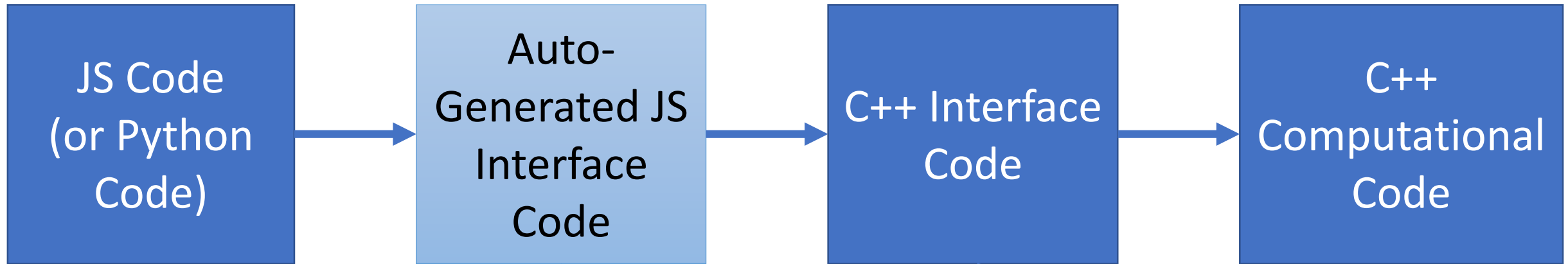
Software Architecture



C++/Web Assembly

- ~18,000 lines of C++ code (per wc) (much of this adapted from Biolmage Suite but stripped of dependencies to VTK/ITK/..)
- Only external dependency is Eigen numerical library
- Compile into bytecode using Emscripten
- Serialization/Transfer layer
 - JS \leftrightarrow 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
```

AkamasXPS:~/javascript/biscplib/js/bin>node bisweb.js linearRegistration

```

....
.... Using node.js version 8.10.0 (OK)
....
''''
'''' Setting forcing orientationOnLoad to: LPS (from LPS), None
'''' bisweb commandline user preferences loaded from C:\Users\xpapa\.bisweb
'''' {"orientationOnLoad":"LPS","snapshotscale":2,"snapshotdowwhite":true}
''''
---- Not enough arguments passed to run this tool

```

Usage: bisweb.js linearRegistration [options]

Options:

```

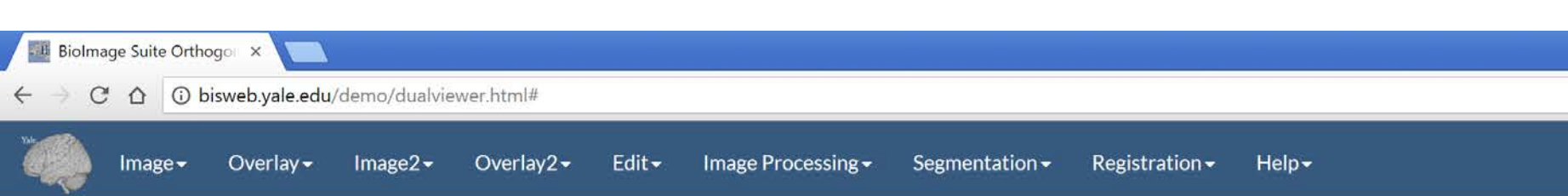
-V, --version          output the version number
--doreslice [s]       If true also output a resliced targeted image using the
--norm [s]            If true normalize input intensities by saturating usi
--intscale [n]        Determines the intensity scaling post image normaliza
--numbins [n]         Number of bins in joint histogram
--extrasmoothing [n] Amount of extra smoothing to perform (values of 0 or
--metric [s]          Metric to compare registration
--optimization [s]    Optimization Method
--stepsize [n]        Step size for gradient computation
--levels [n]          Number of levels in multiresolution optimization
--iterations [n]      Number of iterations (per level and step)
--resolution [n]      Factor to reduce the resolution prior to registration
--debug [s]           Toggles debug logging
--steps [n]           Number of steps in multiresolution optimization
--mode [s]            registration mode, one of [ Rigid Similarity Affine9
-r --reference <s>    The reference image
-t --target <s>      The image to register
--initial [s]        (optional) The initial transformation (optional)
-o --output <s>      The output transformation
--resliced [s]       (optional) The resliced image
--paramfile [s]      Specifies that parameters should be read from a file as opposed to parsed from the command lin
e.
--silent              Run in silent mode (no output on the console)
-h, --help            output usage information
AkamasXPS:~/javascript/biscplib/js/bin>

```

Command line
Modules

38 Modules,
1.1 MB tar.gz file
Works on all major
platforms, just add
node.js

Web-based Applications



```
usr/bin/bash --login -i
kamasXPS:~/javascript/biscplib/js/bin>node bisweb.js linearRegistration
... Using node.js version 8.10.0 (OK)
...
''' Setting forcing orientationOnLoad to: LPS (from LPS), None
''' bisweb commandline user preferences loaded from C:\Users\xpapa\.bisweb
''' {"orientationOnLoad":"LPS","snapshotscale":2,"snapshotdownwhite":true}
'''
--- Not enough arguments passed to run this tool

Usage: bisweb.js linearRegistration [options]

Options:
  -V, --version                output the version number
  --doreslice [s]             If true also output a resliced targeted image using the current transform
  --norm [s]                  If true normalize input intensities by saturating using cumulative histogram
  --intscale [n]              Determines the intensity scaling post image normalization
  --numbins [n]               Number of bins in joint histogram
  --extrasmoothing [n]       Amount of extra smoothing to perform (values of 0 or less will perform no smoothing)
  --metric [s]                Metric to compare registration
  --optimization [s]         Optimization Method
  --stepsize [n]              Step size for gradient computation
  --levels [n]                Number of levels in multiresolution optimization
  --iterations [n]           Number of iterations (per level and step)
  --resolution [n]           Factor to reduce the resolution prior to registration
  --debug [s]                 Toggles debug logging
  --steps [n]                 Number of steps in multiresolution optimization
  --mode [s]                  registration mode, one of [ Rigid Similarity Affine9 Affine ]
  -r --reference <s>          The reference image
  -t --target <s>             The image to register
  --initial [s]               (optional) The initial transformation (optional)
  -o --output <s>             The output transformation
  --resliced [s]              (optional) The resliced image
  --paramfile [s]            Specifies that parameters should be read from a file as opposed to parsed from the command line

  --silent                    Run in silent mode (no output on the console)
  -h, --help                  output usage information
kamasXPS:~/javascript/biscplib/js/bin>
```

Linear Registration

Inputs

- Reference Image
- Target Image
- Initial Xform:
- Resliced Image

Parameters

- Levels:
- Iterations:
- mode:
- ResIn Factor:

Advanced

Run Undo Redo More

Viewer 1 Controls

Core

- Mode:
- I-Coord:
- J-Coord:
- K-Coord:
- Labels:
- Disable Mouse:

Image Color Mapping

Overlay Color Mapping

- Opacity:

Reset Slices Z- Z+ ?

Viewer 2 Controls

Viewer Snapshot

[Transformation Manager](#)

Web-based Applications

Coronal-ik S
R L
I

Sagittal-jk S
A P
I

Coronal-ik S
R L
I

Sagittal-jk S
A P
I

Axial-ij A
R L
P

Axial-ij A
R L
P

Linear Registration

- Inputs
 - Reference Image
 - Target Image
 - Initial Xform: identity
 - Resliced Image
- Parameters
 - Levels: 3
 - Iterations: 10
 - mode: Rigid
 - ResIn Factor: 1.5
- Advanced

Run Undo Redo More

V1:(32,32,37,0) =8498

V2:(32,32,37,0) =5741

Viewer 1 Controls

- Core
 - Mode: Slices
 - I-Coord: 32
 - J-Coord: 32
 - K-Coord: 37
 - Labels:
 - Disable Mouse:
- Image Color Mapping
- Overlay Color Mapping
 - Opacity: 0.2

Reset Slices Z- Z+ ?

Viewer 2 Controls

Viewer Snapshot

Transformation Manager

Web Components

```
<body oncontextmenu="return false;" ondragstart="return false;" 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="algorithmelement"
  bis-viewerid="#viewer">
</bisweb-simplealgorithmcontrollerelement>

<bisweb-painttoolelement
  id="painttool"
  bis-layoutwidgetid="#viewer_layout"
  bis-viewerid="#viewer"
  bis-algorithmcontrollerid="#algorithmelement">
</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 false;" ondrop="return false;">
```

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

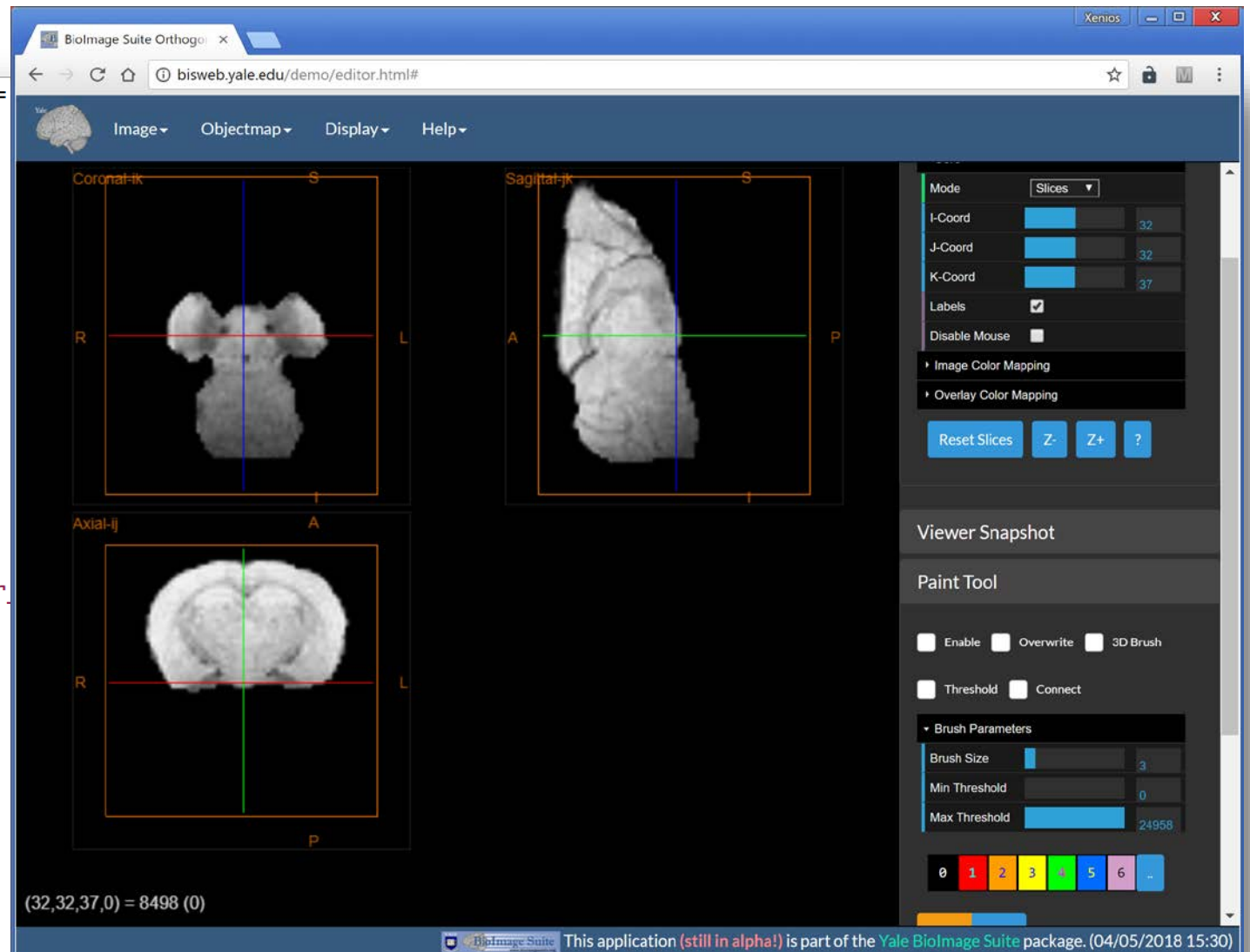
```
<div id="viewerwidget">
```

```
<bisweb-viewerlayoutelement  
  id="viewer_layout"  
  bis-sidewidth="310"  
  bis-coreopen="false"  
  bis-wholescreen="1"  
  bis-defaulttext="">  
</bisweb-viewerlayoutelement>
```

```
<bisweb-colormapcontrollerelement id="viewer_cm"  
  </bisweb-colormapcontrollerelement>
```

```
<bisweb-orthogonalviewer  
  id="viewer"  
  bis-layoutwidgetid="#viewer_layout"  
  bis-colormapeditorid="#viewer_cm">  
</bisweb-orthogonalviewer>
```

```
<bisweb-snapshotelement  
  bis-layoutwidgetid="#viewer_layout"  
  bis-dowhite="false"  
  bis-viewerid="#viewer">  
</bisweb-snapshotelement>
```



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/biscomp
        lib/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
```

```

  "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\":[0,0,0,0,0,0]}",
      "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__mot.json",
  "comments": [
    {
      "ModuleOutput": {
        "command": "/usr/bin/node /home/xenios/javascript/biscompilib/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/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/biscompilib/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/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
}
```

Data Format Example

```
{
  "bisformat": "BisDataObjectCollection",
  "filename": "testdata/test_motion_correction__test_motion_correction__mot.json",
  "comments": [
    {
      "ModuleOutput": {
        "command": "/usr/bin/node /home/xenios/javascript/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/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/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/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
}
```

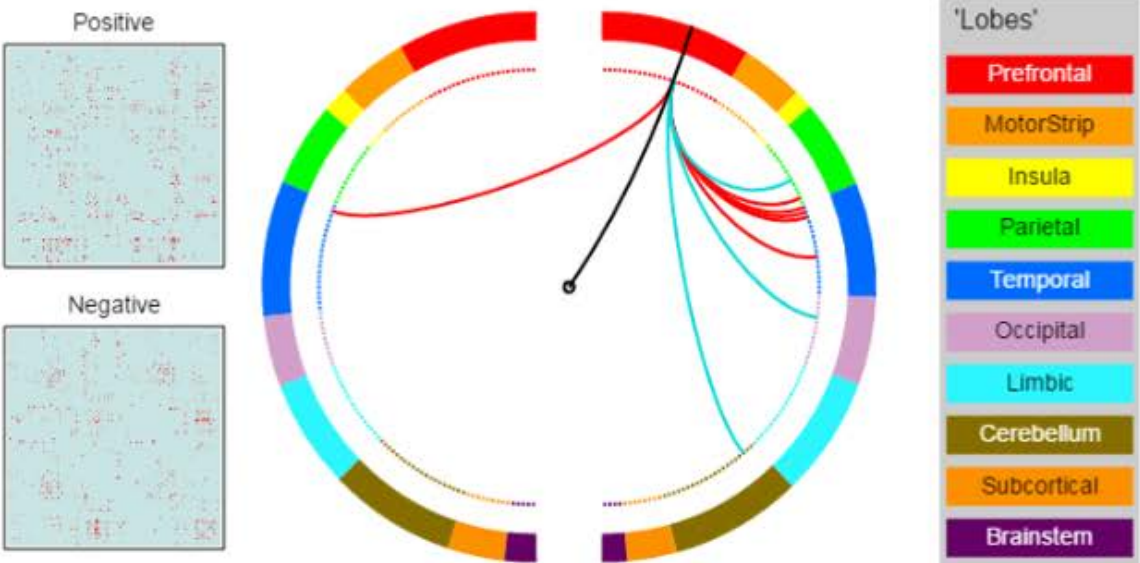
```
      "metadata": {
        "frame": 3
      }
    }
  ],
  "numitems": 5
}
```

Web-based Connectome Visualization

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

Positive

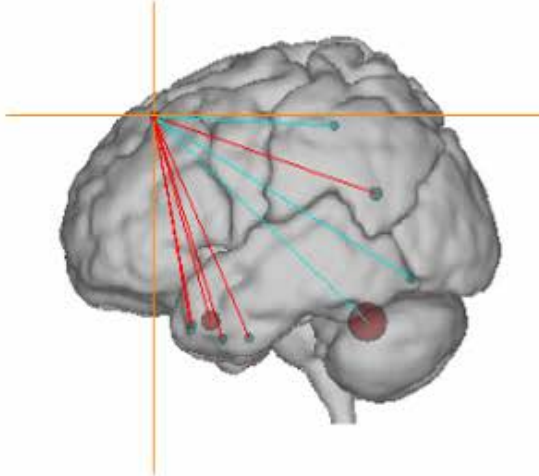
Negative



'Lobes'

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

Node: 148 (L-Prefrontal, Default Mode, BA8).
MNI=(-11.2,34.3,51.5), (Degree: p=6, n=3, s=9) (sorted=148)



Sagittal-jk

Coronal-ik

Axial-ij

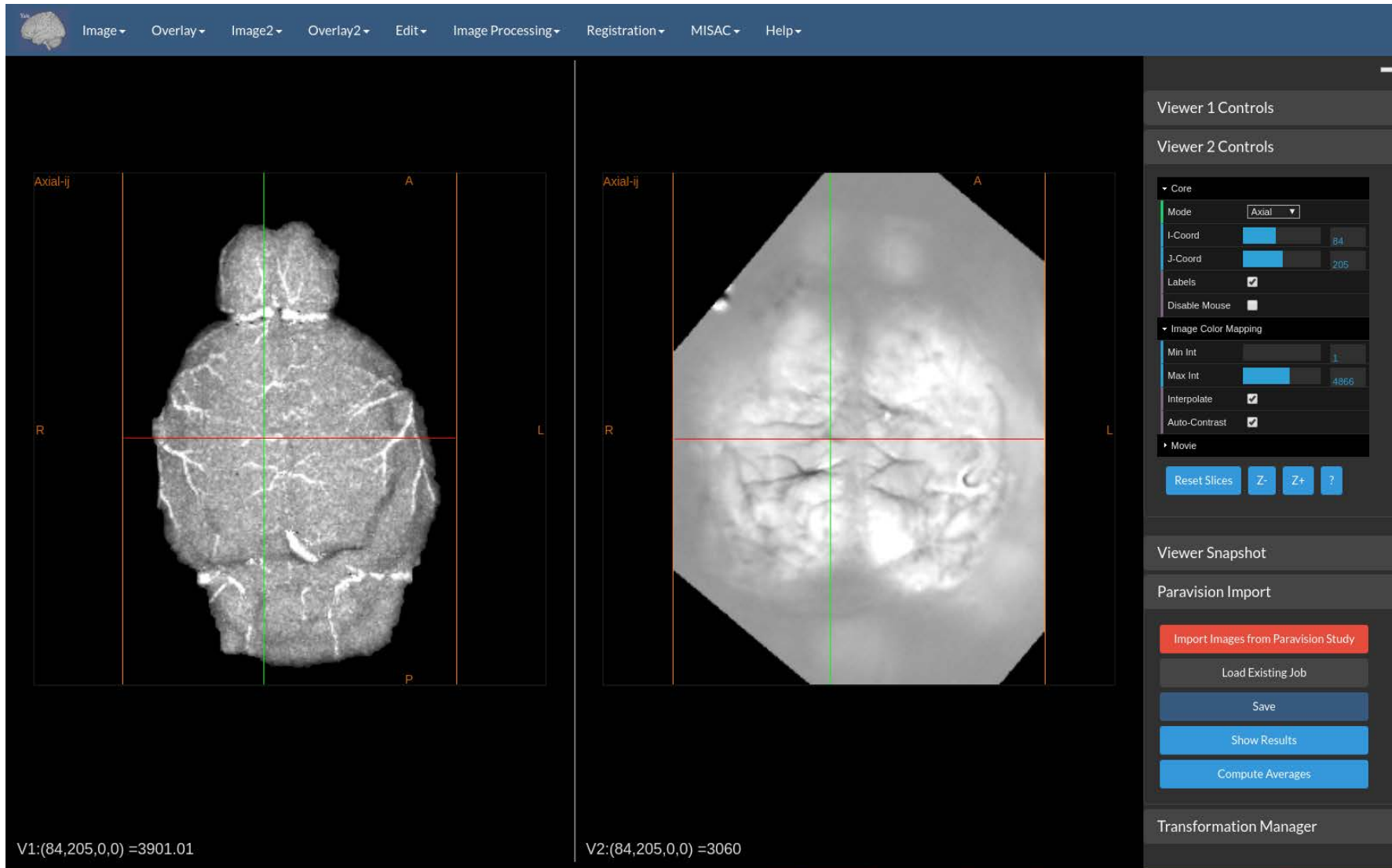
Viewer Controls

Viewer Snapshot

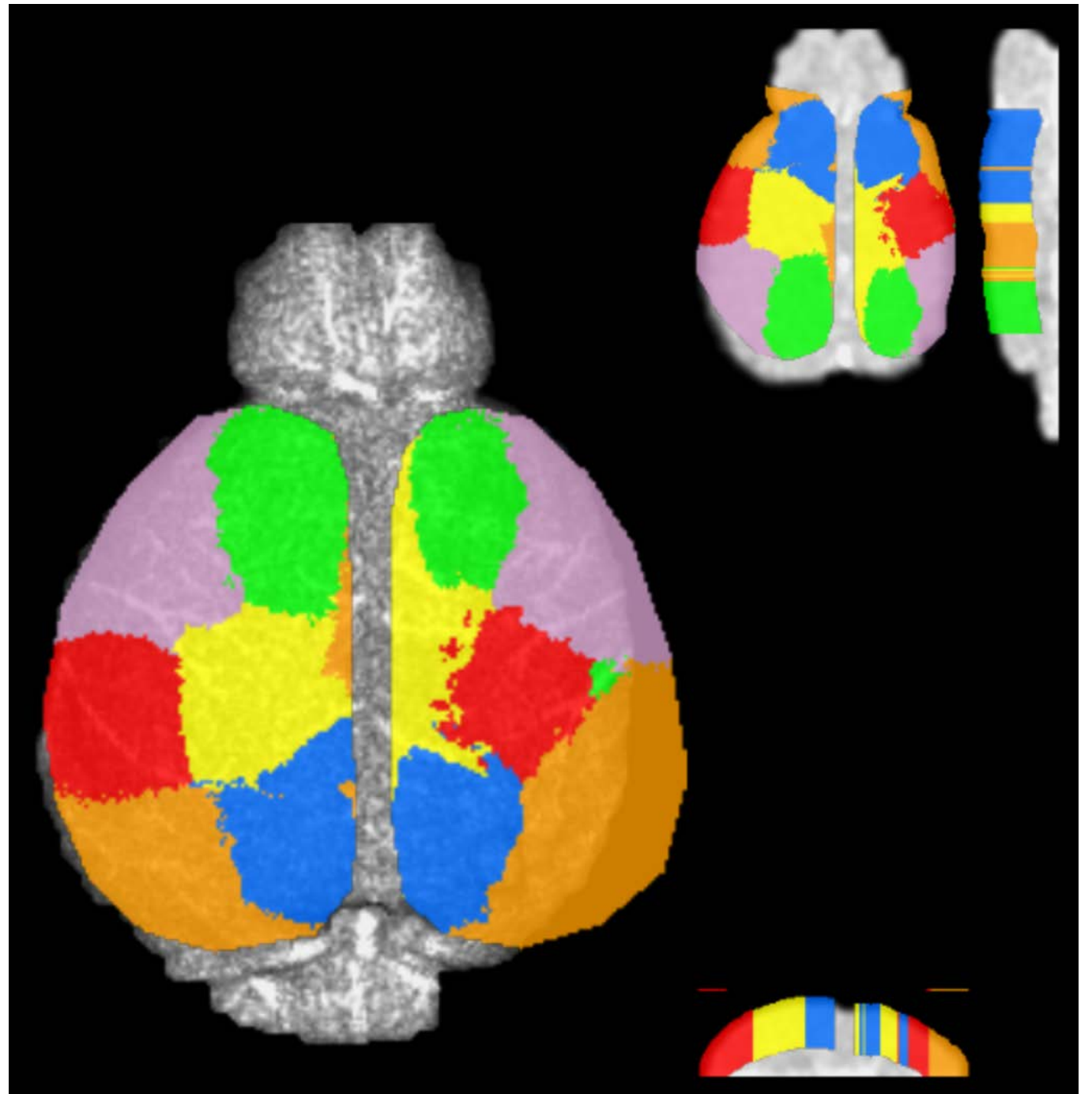
Connectivity Control

- Core
 - Mode:
 - Node:
 - Lobe:
 - Network:
 - Degree Threshld:
 - Lines to Draw:
- Display
 -
 -
 -
 -

Mapping 2D Optical to fMRI (via Angio MRI)



Mapping Parcellations from Optical to MRI



Testing

- Almost all modules (~35) have one or more module regression tests
 - Implement module regression tests as part of module development
 - Subset of tests also in Python (2/3 of the modules have Python versions)
 - Testing is the “fourth context” (commandline, desktop, web) with its own adaptor framework

```
{  
  "command" : "butterworthFilter -i testdata/newtests/but_low_inp.matr --type high --tr 1.0 --high 0.03 --debug true",  
  "test" : "--test_target testdata/newtests/but_high_out.matr --test_type matrix --test_comparison ssd --test_threshold 0.01",  
  "result" : true,  
  "dopython" : true  
},
```

- We have another 100+ unit tests (in about 30 scripts) to test lower level code (e.g. C++ to JS data transfer)
- Manual user testing
- Initial applications for the Yale MISAC Project (talk yesterday)

Status

- Beta-ish version online bisweb.yale.edu/demo
- Source code coming end of this month (BSD style license)
 - Code is build using a combination of cmake and gulp
 - Emscripten used for the WebAssembly
 - Delay is administrative (need to finish build documentation)
- Low level code is basically almost finished
- Working on documentation (including YouTube channel with direct links to the software)

Acknowledgments

- Zach Saltzman
- John Onofrey
- Cheryl Lacadie
- Haley Garbus
- Michelle Lim
- Noah Amsel
- Pranav Chandrasekaran
- Users, testers, ...
- Eigen
- CMake/CTest/CPack
- Emscripten
- Node.js
- Gulp
- Webpack
- Bootstrap
- JQuery
- dat.gui
- Three.js
- Electron
- Electron-Packager