

# A High-Resolution Head and Brain Computer Model for Forward and Inverse EEG Simulation

Alexandra Warner, Chris R. Johnson, Brett Burton, and Jess Tate  
Scientific Computing and Imaging Institute, sci.utah.edu  
University of Utah, Salt Lake City, UT

January 28, 2019

## Abstract

To conduct computational forward and inverse EEG studies of brain electrical activity, researchers must construct realistic head and brain computer models. Constructing such models is challenging and time consuming. The availability of realistic head models and corresponding imaging data is limited in terms of both imaging modalities and patient diversity. In this paper, we describe a detailed head modeling pipeline and provide a high-resolution, multimodal, open-source, female head and brain model. The modeling pipeline specifically outlines image acquisition, preprocessing, registration, and segmentation; three-dimensional tetrahedral mesh generation; finite element EEG simulations; and visualization of the model and simulation results. The dataset includes both functional and structural images and EEG recordings from two high-resolution electrode configurations. The intermediate results and software components are also included in the dataset to facilitate modification to the pipeline. This project will contribute to neuroscience research by providing a high-quality dataset that can be used for a variety of applications and a computational pipeline that may help researchers construct new head models more efficiently.

## 1 Introduction

Many simulation studies in biomedicine are based on a similar sequence of steps; starting from images, creating geometric models, assigning tissue properties, performing numerical simulations, and visualizing the resulting computer model and simulation results. These steps generally describe image-based modeling, simulation, and visualization.<sup>1-5</sup> Image-based modeling is useful for simulating neurological processes and can be used in applications such as; electroencephalographic (EEG) forward simulation studies, EEG source imaging (ESI), and brain stimulator simulations. However, no open-source datasets or pipelines currently exist that include both functional magnetic resonance imaging (fMRI) and diffusion tensor imaging (DTI) data, which are both important in generating models of electrical propagation within the brain.

We developed a comprehensive pipeline to build a complete, high-resolution head model containing both fMRI and DTI data which was specifically used for EEG forward simulation studies, and can be subsequently used in ESI applications. We applied this pipeline to a healthy, female subject

to develop a dataset for open-source distribution. Currently, this is the only female open-source head-modeling dataset.

In this paper, we describe in detail the steps of the pipeline which include image acquisition, preprocessing, registration, and segmentation; finite element mesh generation and simulation; and visualization. We also describe the contents of the open-source dataset which is released in conjunction with this paper. The open-source dataset includes raw and simulated data from the subject, intermediate results from each stage of the pipeline, and the software examples used to perform the simulations. This pipeline and dataset will be a valuable addition to the brain-modeling community because technical, resource, and expertise cost are significant, therefore limiting the number of such datasets.

The images, data, models, and software are available at [www.sci.utah.edu/SCI\\_headmodel](http://www.sci.utah.edu/SCI_headmodel).

## 2 Methods

The complete head model presented here took approximately one year to complete, in part due to the many options in software and techniques, as well as the complexity of the multimodal imaging data which all required specific processing techniques. In this section, we describe in detail all of the steps of the modeling pipeline (Figure 1). We begin with data acquisition of four different images modalities followed by image pre-processing. After the images have been pre-processed, they are segmented into eight tissue layers to create a three-dimensional tetrahedral mesh. All of the image modalities are registered to a common coordinate space and used for forward problem simulations with processed EEG data. We describe the pipeline as we applied it to a female subject.

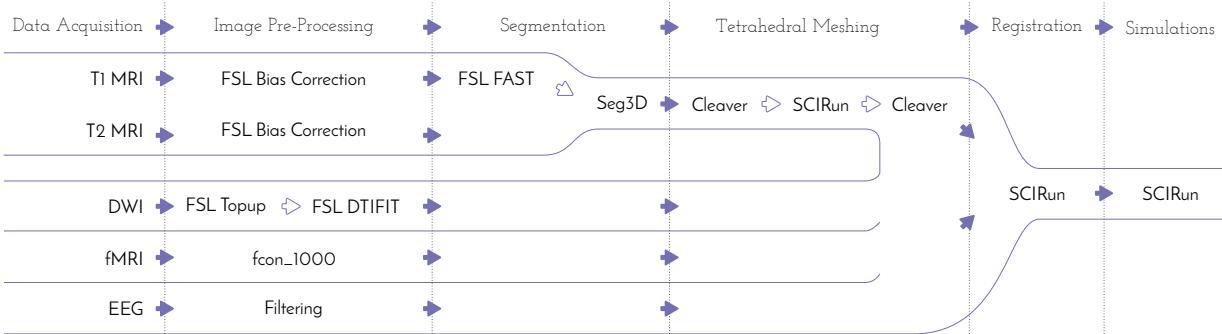


Figure 1: Comprehensive head/brain model pipeline. Data sources are shown on the left with software packages used for creating the head/brain model in the middle and the simulation software on the right.

### 2.1 Data Acquisition

To construct a high-resolution, personalized, anisotropic volume conductor whole-head model,  $T_1$ -,  $T_2$ - weighted, diffusion weighted, and functional magnetic resonance images (MRI) were acquired on a healthy female subject, 23 years of age, on a Skyra 3T full-body scanner (Siemens Medical Solutions, Erlangen, Germany).

The  $T_1$ -weighted scan was performed with a three-dimensional magnetization-prepared, rapid gradient echo (MPRAGE) sequence.<sup>6</sup> The parameters used were as follows: echo time: 3.41ms; repetition time: 2500ms; flip angle: 7 °; resolution matrix size: 256x256 pixels; field of view: 256mm; 208 sagittal slices with a slice thickness of 1mm. Acquisition time was 10:42 minutes.

The  $T_2$ -weighted scan was performed with a SPACE (sampling perfection with application)-optimized contrast using different flip angle evolutions - sequence.<sup>7</sup> The parameters used were as follows: echo time: 406ms; repetition time: 3200ms; resolution matrix size: 256x256 pixels; field of view: 256mm; 208 sagittal slices with a slice thickness of 1mm. Acquisition time was 5:34 minutes. The subject did not move in between the two scans so the scans did not need to be registered.

The diffusion weighted images (DWI) were acquired with multiband, two-dimensional, echo-planar imaging (EPI).<sup>8</sup> Both phase-encoding directions were performed (anterior to posterior and posterior to anterior) with 64 diffusion directions each. Further sequence parameters for each scan were as follows: echo time: 76.8ms; repetition time: 4070ms; flip angle: 90 °; resolution matrix size: 104x104 pixels; field of view: 208mm; 60 slices with 2.5mm slice thickness. Acquisition time was 5:05 minutes each.

The functional MRI (fMRI) scans were acquired with a blood oxygenation level dependent contrast (BOLD) sequence. The following parameters were used: echo time: 76.8ms; repetition time: 780ms; flip angle: 55 °; resolution matrix size: 104x104 pixels; field of view: 210mm; 72 slices with 2mm slice thickness. Acquisition time was 10:32 minutes.

Continuous electroencephalograms (EEGs) were recorded using a 128-channel and 256 channel HydroCel Geodesic Sensor Net that was connected to a NetAmps 400 amplifier and referenced online to a single vertex electrode. Channel impedances were kept at or below 50 kOhms and signals were sampled at 250Hz. The EEGs were recorded while the subject sat quietly in a chair, alternating two-minute epochs of eyes open and eyes closed, for a total of twelve minutes.

All acquisition reports are included with the dataset.

## 2.2 Preprocessing of Images

### 2.2.1 MRI Correction

Bias field signal is a low-frequency, smooth signal that corrupts MRI images due to inhomogeneities in the magnetic fields of the MRI machine by blurring images, thereby reducing the high frequencies of the images, such as edges and contours. The signal changes the intensity values of image pixels so that the same tissue has a different distribution of grayscale intensities across the image.<sup>9</sup> We applied an estimated bias field correction on the  $T_1$  and  $T_2$  MRIs using FMRIB Software Library (FSL) FAST (Figure 2),<sup>10</sup> which will be further described in Section 2.3.

### 2.2.2 DWI Distortion Correction

DWIs performed with EPI sequences are prone to distortions from rapid switching of diffusion weighting gradients, movement from the scanning table, and movement from the subject. The diffusion data was collected with reversed phase-encoded blips (anterior to posterior (AP) and posterior to anterior (PA)),

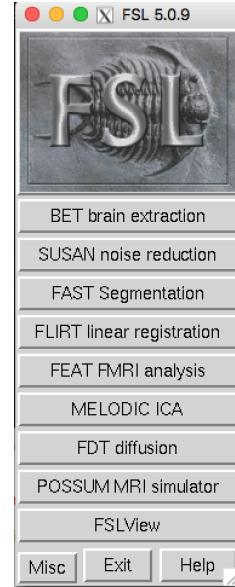


Figure 2: FMRIB software library user interface.

resulting in pairs of images with distortions in opposite directions. From these pairs, we estimated the susceptibility-induced off-resonance field using a method<sup>11</sup> similar to what is currently implemented in FSL.<sup>12</sup> We then combined the two images into a single corrected one using FSL’s topup and eddy command line tools. Details on this process are described in the Appendix in Section A.1.

### 2.2.3 Diffusion Tensor Images

After we preprocessed the DWI images, we calculated diffusion tensor images (DTI) using FSL’s DTIFIT toolbox<sup>13</sup> and SCIRun,<sup>14</sup> a problem solving environment for modeling, simulation, and visualization of scientific problems. We used the output from DTIFIT, the eigenvalues and eigenvectors of the DTI data, as inputs in a SCIRun network (Figure 32) to build and visualize the conductivity tensor field. Details on using DTIFIT are described in the Appendix in Section A.2.

### 2.2.4 fMRI

We preprocessed the fMRI data using the 1000 Functional Connectomes (fcon) Project pipeline scripts,<sup>16</sup> which performed anatomical preprocessing, functional preprocessing, registration to the  $T_1$  MRI, segmentation, and nuisance signal regress. The outline pipeline used on this fMRI dataset, specific to the University of Utah, can be found at <https://bitbucket.org/UtahBrainNetworksbase-prep>, which includes instructions for installation, compilation, and usage. We then converted the preprocessed fMRI data from a four-dimensional dataset to a two-dimensional dataset to be visualized in SCIRun. Details on this data conversion is described in the Appendix in Section A.3.

### 2.2.5 EEG

A 60Hz notch filter and its harmonics<sup>17</sup> were applied to the EEG data, and we created an EEG data matrix. The rows of the matrix corresponded to the channels of the EEG and the columns corresponded to the time step. We removed the last two rows of the EEG signals matrix as these were control rows for the experiment. We also removed several columns at the beginning and the end of the matrix, because these columns corresponded to taking the EEG net on and off the subject’s head. Details on building the EEG data matrix is described in the Appendix in Section A.4.

### 2.2.6 Registration

Since the subject did not move in between the  $T_1$  and  $T_2$  MRI, registration was not necessary before segmentation and mesh generation. We generated the tetrahedral mesh from the segmentation and registered the cortical surface mesh to the DTI coordinate space with a manual, rigid registration,(Figure 3 - *(left)*). We included the registration transformation matrix to the DTI coordinate space in the dataset.

Registration of the fMRI dataset required spatial preparation to ensure a more accurate rigid registration. For each fMRI time step, we mapped the vector of fMRI data onto a lattice volume, rotated it 180 degrees about the  $z$ -axis, smoothed with a median filter, thresholded to remove noise,

and clipped any excess. After we applied this rigid registration to the cortical surface mesh, we continued the registration manually to ensure the most accurate registration. (Figure 3 - *(right)*).

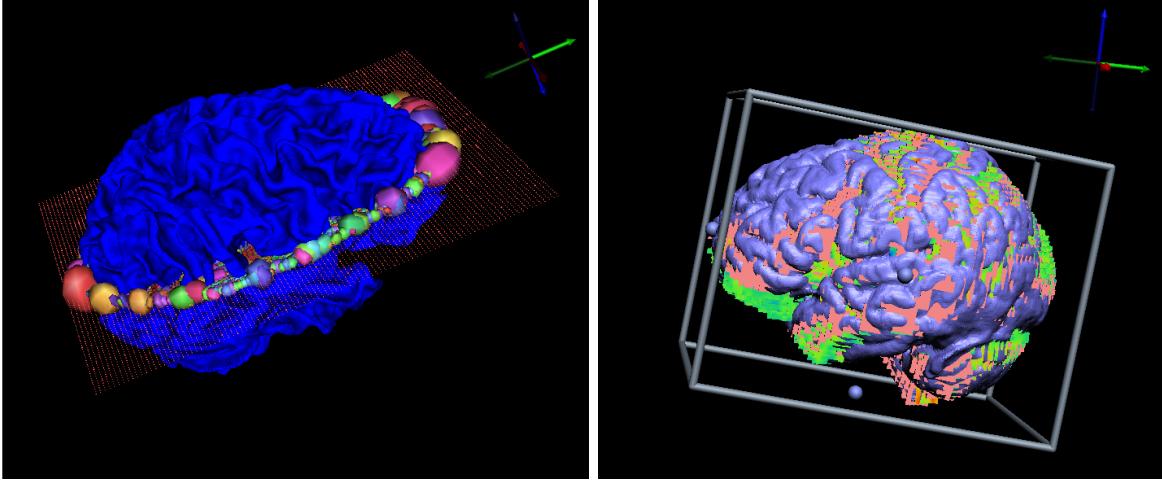


Figure 3: SCIRun manual registrations: mesh to DTI registration (*left*), fMRI to mesh registration (*right*).

The EEG geometry data from the ESI system consisted of dipole source location (4800) and electrode locations (128 or 256). To register the electrode positions to the head mesh, we rotated them 180 degrees about the z-axis, applied a rigid registration, and projected the points onto the mesh (Figure 4). To register the dipoles to cortical mesh, we simply applied a rigid registration.

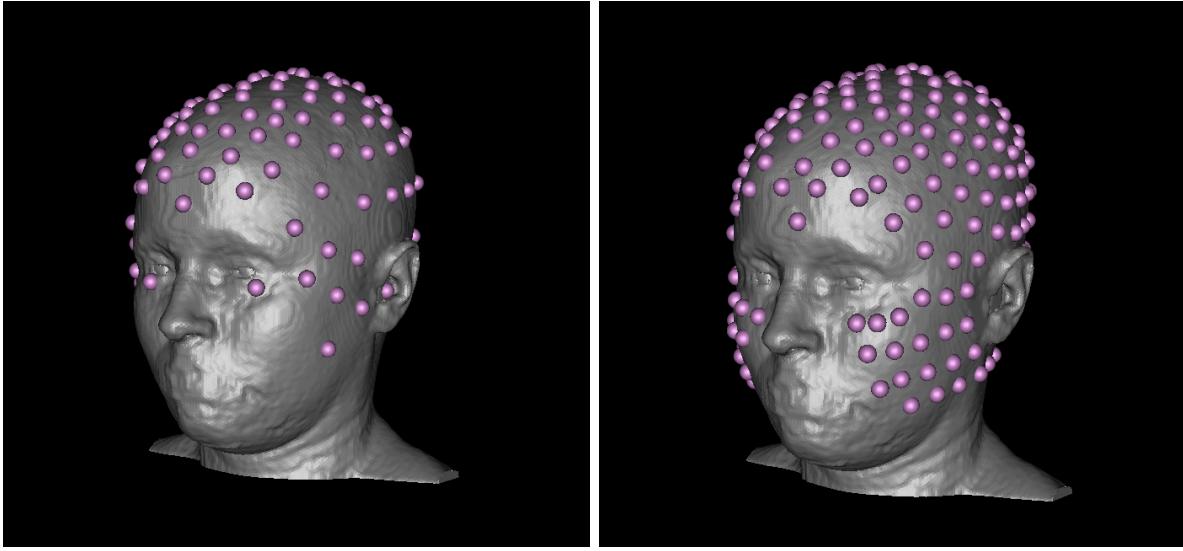


Figure 4: SCIRun manual registration of physical electrodes to mesh: 128 electrodes (*left*), 256 electrodes (*right*).

The SCIRun networks for registration are included in the Appendix in Section A.5 in Figures 33 - 36.

## 2.3 MRI Segmentation of Tissues

Segmentation of the head tissues proved to be the most time-consuming portion of the pipeline. We segmented the head volume using FSL and Seg3D, a free volume segmentation and processing tool,<sup>18</sup> into external air, cerebral spinal fluid (CSF), white matter, gray matter, skull, sinus, eyes, and scalp. We used FSL for initial segmentation and Seg3D for all further segmentation including thresholding and manual editing.

We generated the brain initial segmentation with FSL from the  $T_1$  MRI by stripping the skull with the brain extraction tool (BET)<sup>19</sup> (Figure 11), and then segmenting with FAST segmentation (Figure 5). FSL FAST outputs grayscale probability images of CSF, white matter, and gray matter layers (Figure 6) as well as a bias-corrected  $T_1$  MRI (Figure 7).

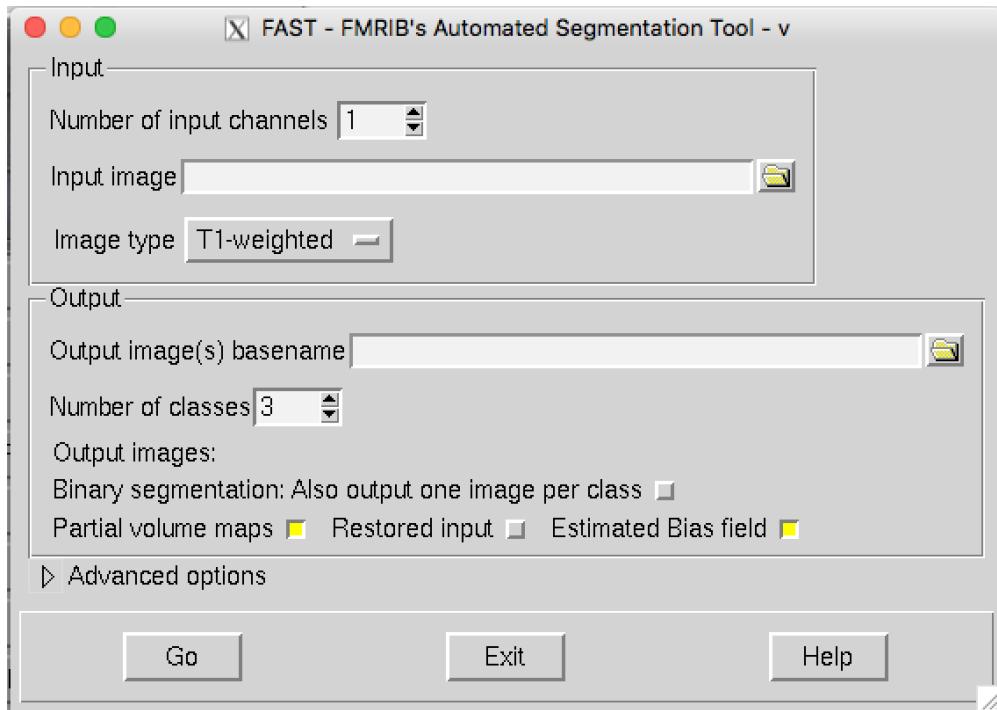


Figure 5: FSL FAST user interface.

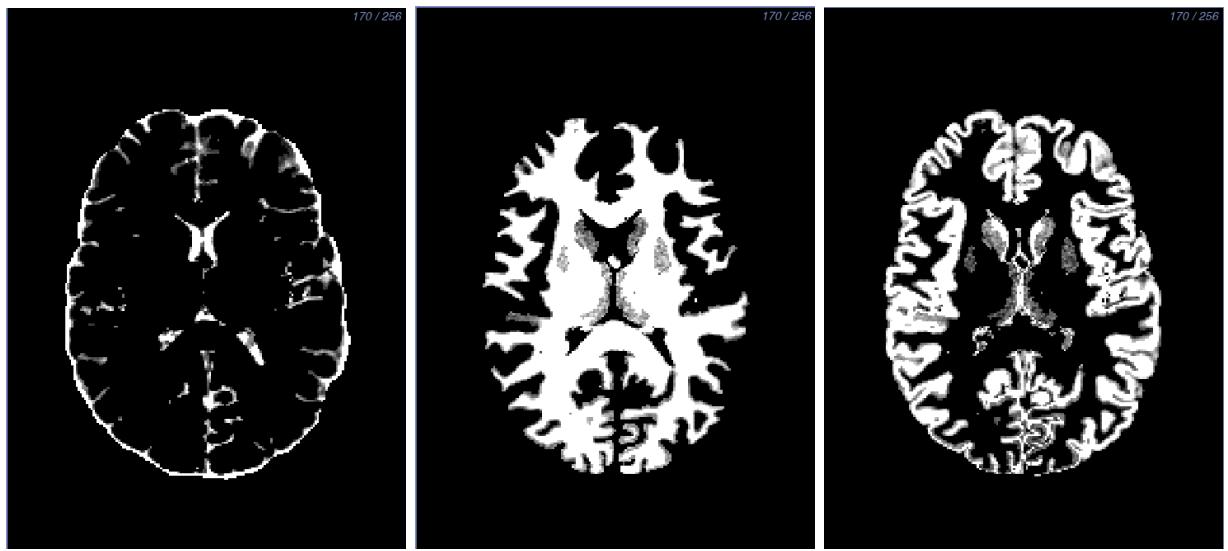


Figure 6: FSL FAST output: CSF (*left*), white matter (*center*), gray matter (*right*).

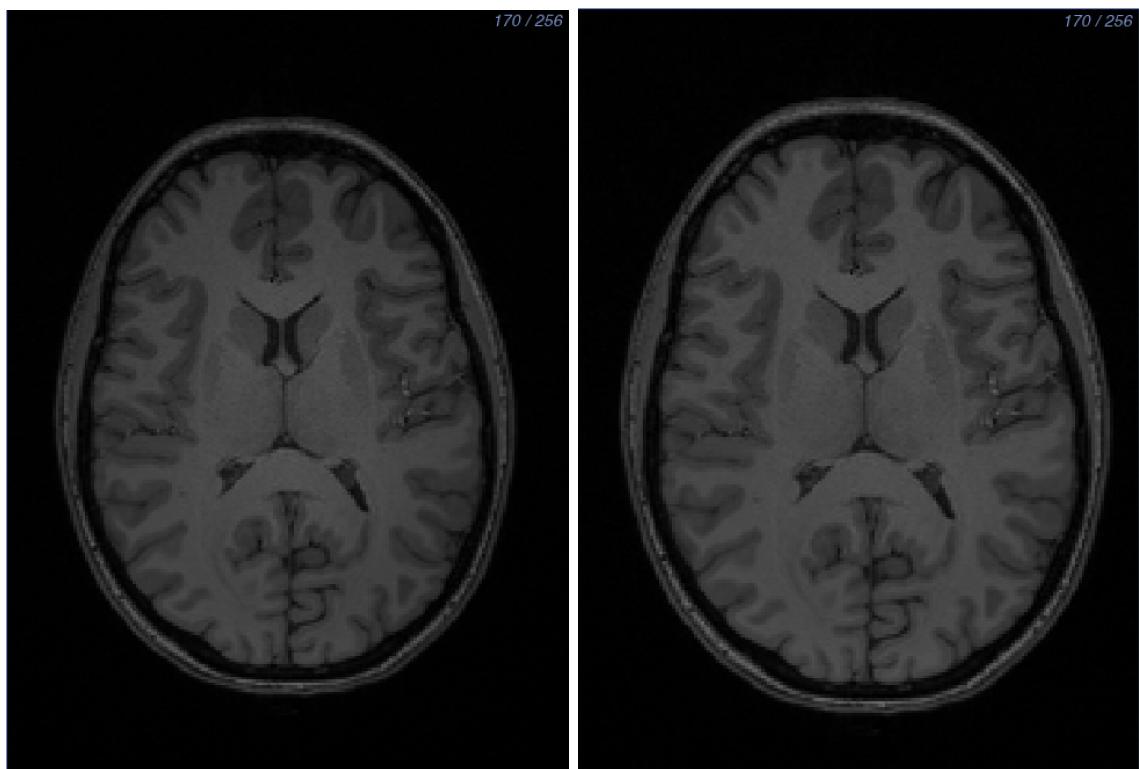


Figure 7: FSL FAST output:  $T_1$  MRI image before bias correction (*left*), and after bias correction (*right*).

We created a segmentation from the FSL FAST output by thresholding the probability images, then inspecting and manually editing each slice to remove any crossover between the layers, to add more detail, and smooth out noisy areas. We started with the white matter layer because it is the innermost layer. First, we created a threshold layer of the FSL FAST output. We then inspected and manually edited each slice in every direction to add more detail or to clean up noise (Figure 8).

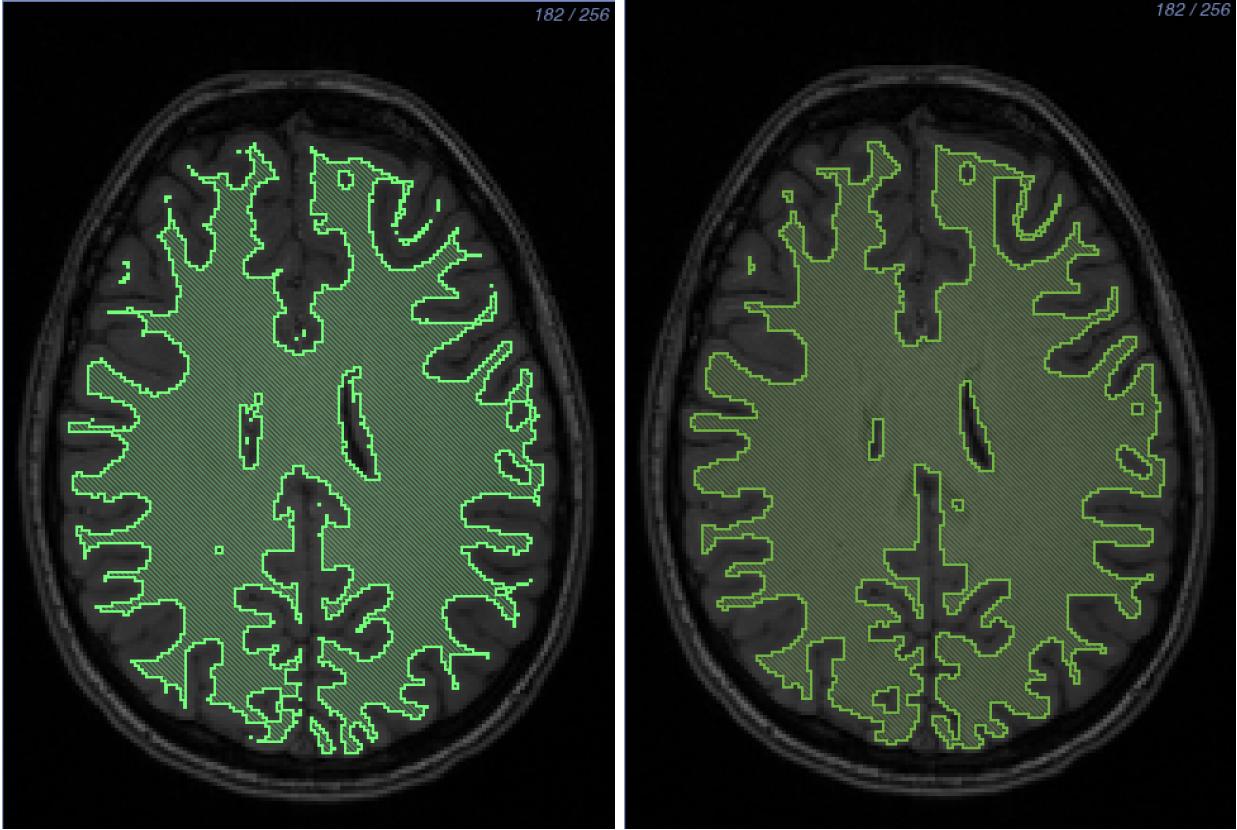


Figure 8: White matter segmentation: Before (*left*) and after (*right*) manual segmentation. The hook feature in the upper right-hand corner is a notable change between the two layers. The layer is more full and has less noise.

After we segmented the white matter, we created a threshold layer from the FSL FAST output for the gray matter. We inspected and manually edited each slice in every direction of the gray matter. We then removed the white matter layer from the gray matter using a Boolean remove mask filter to ensure no overlap between the layers. Lastly, we added a gray matter nuclei segmentation to the gray matter layer. The thresholding algorithms in Seg3D produced noise around these nuclei because of the similarities of the grayscale values. To fix this noise, we segmented the nuclei area manually using the paintbrush tool in Seg3D. Then, we added the nuclei to the gray matter layer using a Boolean OR mask filter, and removed any overlap from the white matter layer using a Boolean remove mask filter. We manually filled any holes between the two layers that could be assigned to white or gray matter layers (Figure 9).

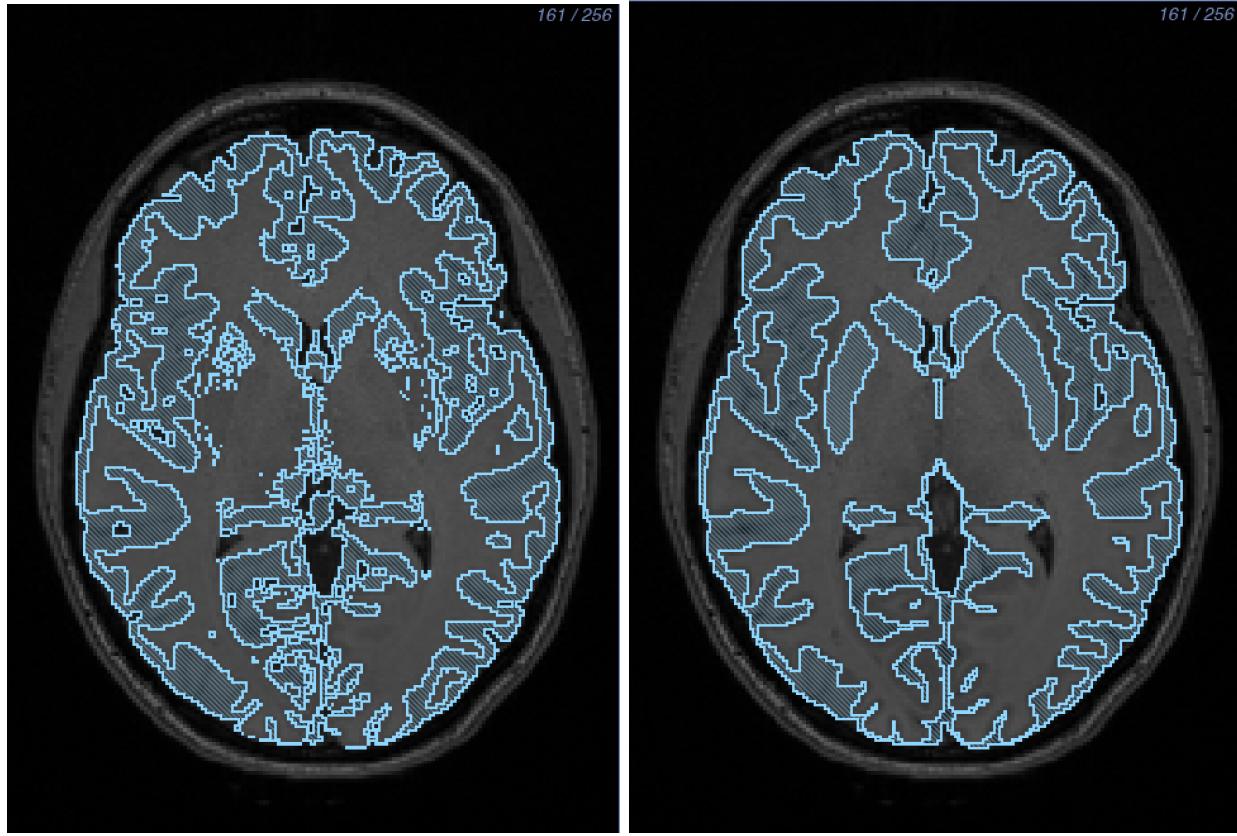


Figure 9: Gray matter segmentation: Before (*left*) and after (*right*) manual segmentation. Gray matter nuclei located in the center of the brain were segmented manually.

an

After we completed the segmentation of the gray and white matter layers, we made the CSF layer by creating a solid threshold layer for the entire brain and removing the white and gray matter layers using a Boolean remove mask filter (Figure 10). We then checked the white matter, gray matter, and CSF layers for holes, both on the surface and inside the segmentation between layers. We also performed a manual quality check on the layers to ensure that they were at least two pixels wide throughout. This thickness criteria helped to ensure a tetrahedral mesh without holes.

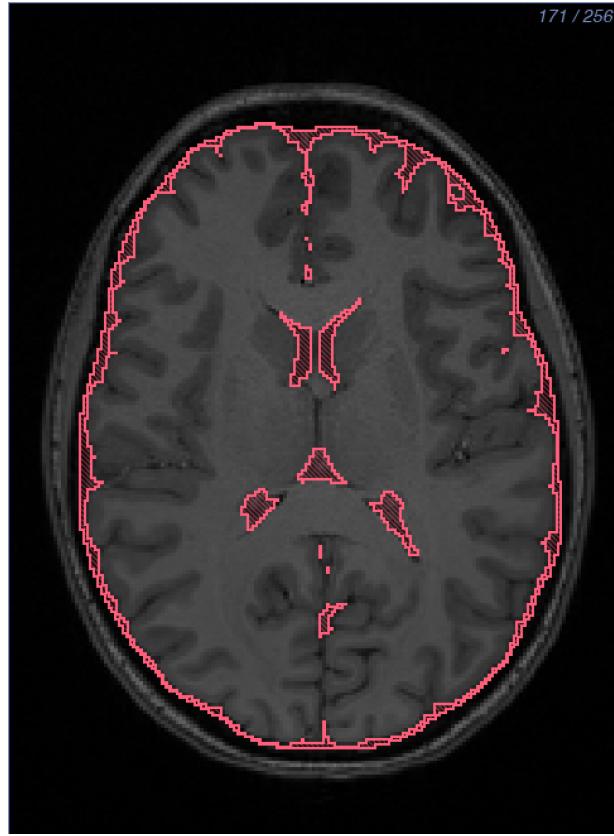


Figure 10: CSF segmentation.

The skull and the sinus layers were the most difficult to segment using only an MRI, because they both appear black in the MR image, and the subject’s data did not include a computed tomography (CT) scan. For our first attempt to create a bone layer, we used FSL’s BET2 tool (Figure 11) to extract a skull surface. We then thresholded the  $T_1$  MRI to create the remainder of the bones in Seg3D, and connected the bones to the skull made from FSL using a Boolean OR mask filter. Although this approach gave an adequate segmentation for the skull, it did not include important tissues, such as the sinus layer. As a second method, we estimated the skull segmentation from an MR-based synthetic pseudo-CT image (Figure 12). We used an improved iterative version of the patch-based method as described by Torrado-Carvajal et al.<sup>23</sup> that takes the  $T_1$  and  $T_2$  images as input and synthesizes the pseudo-CT based on both images, providing more refined and accurate bone boundaries.

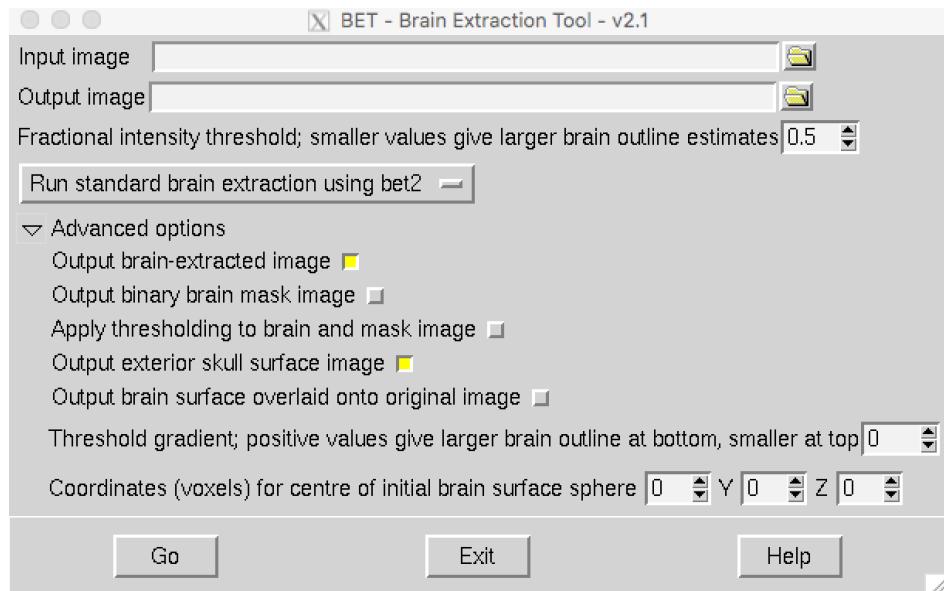


Figure 11: FSL's BET2 tool.

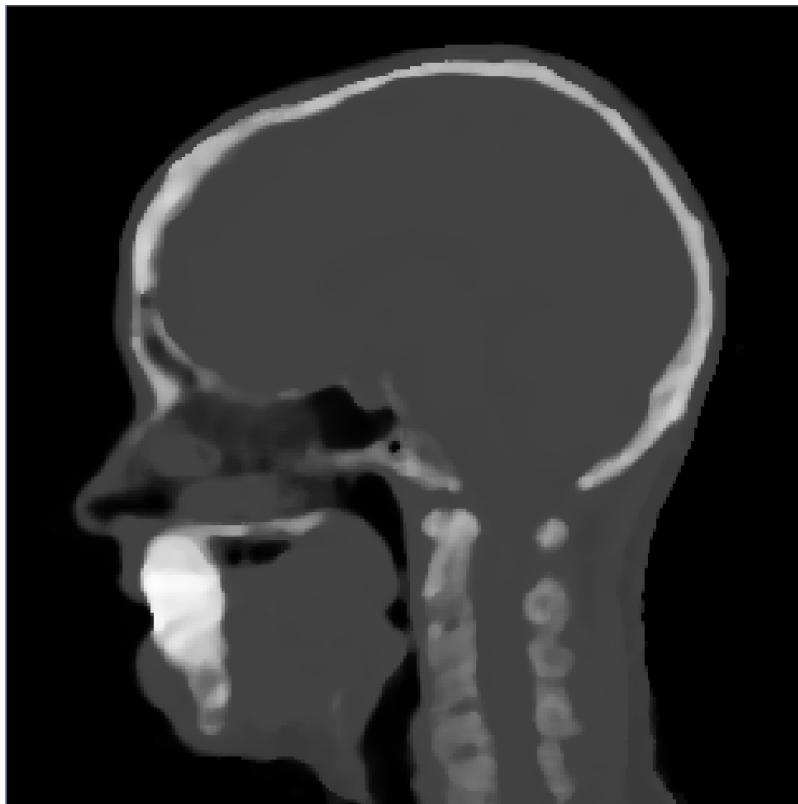


Figure 12: Pseudo-CT image.

The pseudo-CT method provided an image that was a more straight forward starting point for skull segmentation, but we still manually edited the layer to add detail. After we applied a median filter with a one-pixel radius and thresholded the pseudo-CT image, we manually edited each slice in every direction to add detail and to smooth noise. Since the subject had a permanent retainer in her mouth that created all black pixels, we segmented the mouth as solid bone, which we deemed acceptable because the EEG cap used did not cover the subject’s mouth. We were able to provide a segmentation of the internal air, including the sinuses, esophagus, and ear canals, from the pseudo-CT image by thresholding the black pixels and then manually editing each slice in every direction. We also performed a quality check on both layers to ensure they did not contain holes, have any layer overlap, and were at least two pixels thick.

The eyes, skin, and air layers were the least time consuming to segment. We segmented the eyes by thresholding the  $T_2$  MRI. We segmented the skin layer by thresholding the entire head volume and removing all previous segmentation layers using a Boolean remove mask filter. We performed a quality check on the skin layer to ensure that it was at least two pixels thick. The areas that required significant correction were the bridge of the nose, the bottom of the chin, and the sides of the head. Lastly, we included pixels not previously assigned as the external air layer. We checked through the entire segmentation to ensure that there weren’t any holes between layers to assure a quality mesh and accurate simulation results.

The segmentation of the image data took approximately 100 hours which was mostly dedicated to manual editing. A breakdown of time per segmentation layer is shown in Table 1.

Table 1: Segmentation Time

Segmented Tissue	Amount of Work (hrs)
White Matter	40
Gray Matter	20
CSF	4
Skull and Sinus	35
Eyes, Scalp, & Air	8

## 2.4 Mesh Generation

We used our full-head segmentation to generate realistic three-dimensional geometries for use in subsequent finite element simulations. We generated a smooth, linear, subject-specific, boundary-conforming, tetrahedral mesh using the Cleaver software<sup>24</sup> on a Late 2013 Mac Pro with a 2.7 Ghz 12 Core Intel Xeon E5 processor with 64 GB of RAM and an AMD FirePro graphics card. Cleaver is a multimaterial meshing package that produces structured meshes of tetrahedral elements with guaranteed minimum element angles, resulting in quality meshes that require fewer computational resources. We generated a high-resolution mesh without holes using the parameters listed in Table 2.

Table 2: Clever Settings (High Resolution)

scaling factor	0.6
size multiplier	1.0
lipschitz	0.2
padding	0
element sizing method	adaptive

We created indicator functions, functions describing the location of the surface with sub grid accuracy, by calculating inverted distance maps of each layer in the full-head segmentation in Seg3D using the distance map filter. To reduce the size of the mesh, we first generated a new mesh, changing only the scaling factor parameter to 1.0 from the parameters in Table 2. We exported the computing sizing field from Cleaver and manipulated it in SCIRun by changing how quickly the elements increased in size. We input the changed sizing field into Cleaver with the same indicator functions and successfully cleaved a new, smaller mesh without holes. Both of these sizing fields are included in the dataset.

## 2.5 Mathematical Modeling

The forward and inverse EEG problems are governed by a generalized Poisson equation (1). We used the head mesh, with associated inhomogeneous and anisotropic conductivity regions, as a volume conductor to solve the following boundary value problem:

$$\nabla \cdot \sigma \nabla \Phi = -I_V \quad \text{in } \Omega, \quad (1)$$

where  $\Phi$  is the electrostatic potential,  $\sigma$  is the electrical conductivity tensor, and  $I_V$  is the current per unit volume defined within the solution domain,  $\Omega$ . For the forward EEG problem, we solved equation 1 for  $\Phi$  with a known description of  $I_V$  and the Neumann boundary condition:

$$\sigma \nabla \Phi \cdot \mathbf{n} = 0 \quad \text{on } \Gamma_H, \quad (2)$$

which says that the normal component of the electric field is zero on the surface interfacing with air (here denoted by  $\Gamma_H$ ). For testing purposes, we used dipoles for the current source. We calculated the electrical and potential fields everywhere within the head model.<sup>2</sup>

### 2.5.1 Electrical Conductivity Preparation

All electrical conductivities were homogeneous for each tissue with the exception of the white matter when using DTI data. The isotropic conductivities<sup>25</sup> we used are shown in Table 3.

Table 3: Isotropic Tissue Conductivity

Tissue Type	Isotropic Conductivity ( $S/m$ )
White Matter	0.1429
Gray Matter	0.3333
Cerebrospinal Fluid (CSF)	1.79
Skull	0.001
Skin	0.4346
Sinus	1e-6
Eyes	0.5051

When we added the DTI tensor data, we used two approaches to convert the tensor data to conductivities. The first was scaling the data:<sup>26</sup>

$$\sigma_{aniso} = \frac{\sigma_{iso}}{\sqrt[3]{d_1 d_2 d_3}} D, \quad (3)$$

where  $D$  is the diffusion data,  $d_i$  is the  $i$ th eigenvalue of  $D$ , and  $\sigma_{iso}$  is the white matter isotropic conductivity. The second method gave the white matter a fixed ratio of conductivity:

$$\sigma_{aniso} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ W \end{bmatrix}, W = \begin{bmatrix} \sigma_{iso} \\ \frac{\sigma_{iso}}{10} \\ \frac{\sigma_{iso}}{10} \end{bmatrix}, \quad (4)$$

where  $v_i$  is the  $i$ th eigenvector of  $D$ ,  $W$  is the white matter ratio vector, and the ratio is 10 : 1.

We implemented both methods into the SCIRun networks for anisotropic forward problems (Figure 34).

### 2.5.2 Numerical Methods

We computed solutions to equation 1 using the finite element method. By applying Green's theorem to equation 1, we generated the following weak formulation,

$$\langle \sigma \nabla \phi, \nabla \bar{\phi} \rangle = -\langle I_v, \bar{\phi} \rangle \quad (5)$$

where  $\bar{\phi}$  is an arbitrary test function, which can be thought of physically as a virtual potential field. By applying the Galerkin approximation to equation 5, we can represent the finite element approximation as:

$$\sum_{i=0}^N \varepsilon_i \langle \sigma_{ij} \nabla \psi_i, \nabla \psi_j \rangle = -\langle I_v, \psi_i \rangle \quad j = 0, \dots, N, \quad (6)$$

subject to the Dirichlet boundary condition. Then the finite element approximation of equation 1 can equivalently be expressed as a system of  $N$  equations with  $N$  unknowns  $\varepsilon_1 \dots \varepsilon_N$  (e.g., the electrostatic potentials). In matrix form, the above system can be written as  $A\varepsilon = b$ , where  $A = (a_{ij})$  is called the global stiffness matrix and has elements  $(a_{ij}) = (\sigma_{ij} \nabla \psi_i, \nabla \psi_j)$ , and  $b_i = -(I_v, \psi_i)$  is usually termed the load vector. For volume conductor problems,  $A$  contains all the geometry and conductivity information of the model.<sup>2</sup>

We used SCIRun to apply parameters and to solve equation 6 numerically using linear basis functions for tetrahedral elements. Within the SCIRun environment, we applied isotropic and anisotropic conductivity tensors to the tetrahedral mesh as well as to inhomogeneous regions. We applied Dirichlet and Neumann boundary conditions to compute potentials using a conjugate gradient method with a Jacobi preconditioner.

## 2.6 Simulations and Visualizations

We ran all simulations and visualizations in SCIRun. All networks are shown in the Appendix in Section A.5 - Figures 32 - 36.

### 2.6.1 Forward Problem

Solving systems from a known source to the EEG electrodes, described in Section 2.5, is known as a forward problem. The opposite action, solving systems from the EEG data to a unknown source, is an inverse problem. In this project, we built SCIRun example networks to solve forward problems with known sources and to write a lead field matrix for use in future inverse problems. We solved forward problems with isotropic and anisotropic conductivities, using DTI data for the direction of anisotropic conductivities.

The required inputs for the isotropic example network were the tetrahedral head mesh, isotropic conductivities, the head segmentation, the physical electrode locations, and dipole sources. The SCIRun network allowed the user to choose a dipole as a current source from the dataset for a forward simulation. The network also removed fiducial electrodes from the dataset. We registered the mesh to the head segmentation using a rigid registration previously described in Section 2.2.6. We then register the electrodes and the dipoles to the mesh coordinate space, as described in Section 2.2.6. After we removed any flat tetrahedra from the mesh, we mapped the conductivities to their respective tissues. We registered the electrodes and dipoles to the head mesh using the same transform. We solved the system with the mapped data and the chosen dipole sources. We then extracted the solution onto the head surface and the electrode locations for visualization. We also used streamlines and isopotential lines for visualization.

For the anisotropic example, the network was similar with the exception of the scaled diffusion tensor dataset used as conductivities for white matter, and the DTI to mesh transform was necessary as input. We registered the head mesh, electrodes, and dipoles to the DTI space with a rigid registration, as described in Section 2.2.6; the head segmentation was not needed.

### 2.6.2 fMRI

We visualized the fMRI data one time step at a time using the two-dimensional matrix, described in Section 2.2.4, as input by extracting one column at a time. We registered the fMRI to the tetrahedral mesh as described in Section 2.2.6. After registration, we mapped the smoothed fMRI data onto the mesh using a mapping matrix with a linear interpolation basis.

### 2.6.3 EEG

We visualized EEG data on the physical electrode locations, which we registered to the mesh space with a rigid registration, as described in Section 2.2.6, after removing the fiducials from the

electrode dataset. We displayed the filtered EEG signals one time step at a time by extracting the column of the matrix. We then placed the electrodes onto the mesh and mapped the EEG data onto the electrodes one time step at a time. An example EEG network is shown in Figure 36. Each EEG dataset has their own SCIRun network.

## 3 Results

All SCIRun networks used to generate results are included in an open-source dataset for research use at [www.sci.utah.edu/SCI\\_headmodel](http://www.sci.utah.edu/SCI_headmodel). Details of this dataset are described in Table ??.

### 3.1 Segmentation

For this project, we segmented the head into eight detailed layers, listed in Section 2.3. We used this segmentation to create an inhomogeneous three-dimensional tetrahedral mesh.

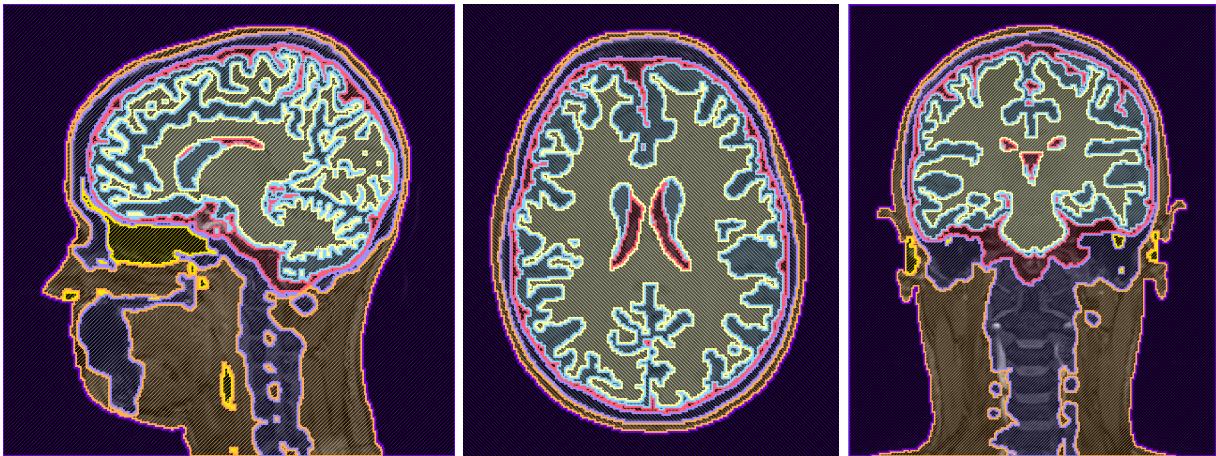


Figure 13: A high-resolution, eight-layer, full head segmentation made with Seg3D.

The two bone segmentation methods, discussed in Section 2.3, produced geometries that were similar in some areas, but different in others. The segmentation created using FSL and Seg3D was rough and had a clear line of where the two segmentations were connected. It also didn't include a chin or a sinus layer. The segmentation created from the pseudo-CT scan was smooth and included a chin and a sinus layer. This segmentation did include the inside of the mouth as bone, but as mentioned before this wasn't concerning for simulations results.

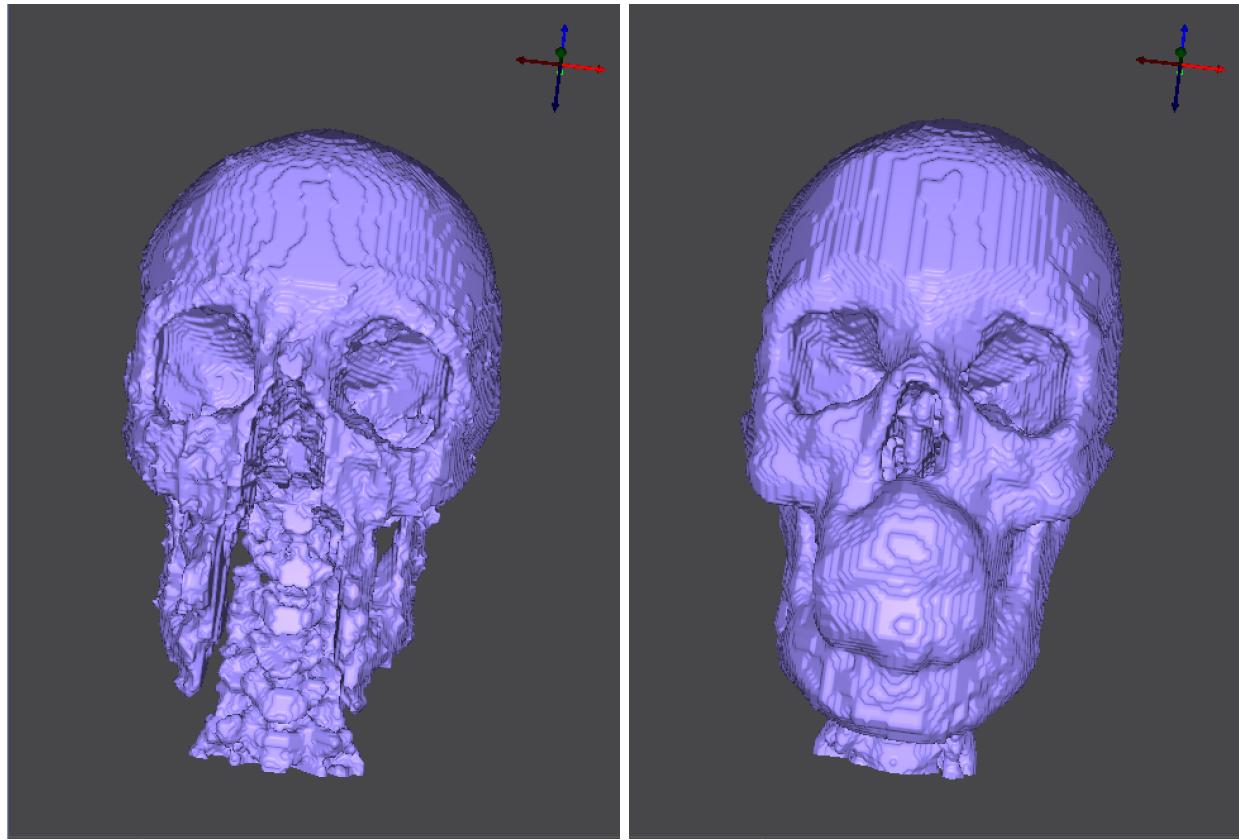


Figure 14: Skull segmentation comparison: Created with BET and thresholding (*left*) and with pseudo-CT (*right*). Both segmentations were made using Seg3D.

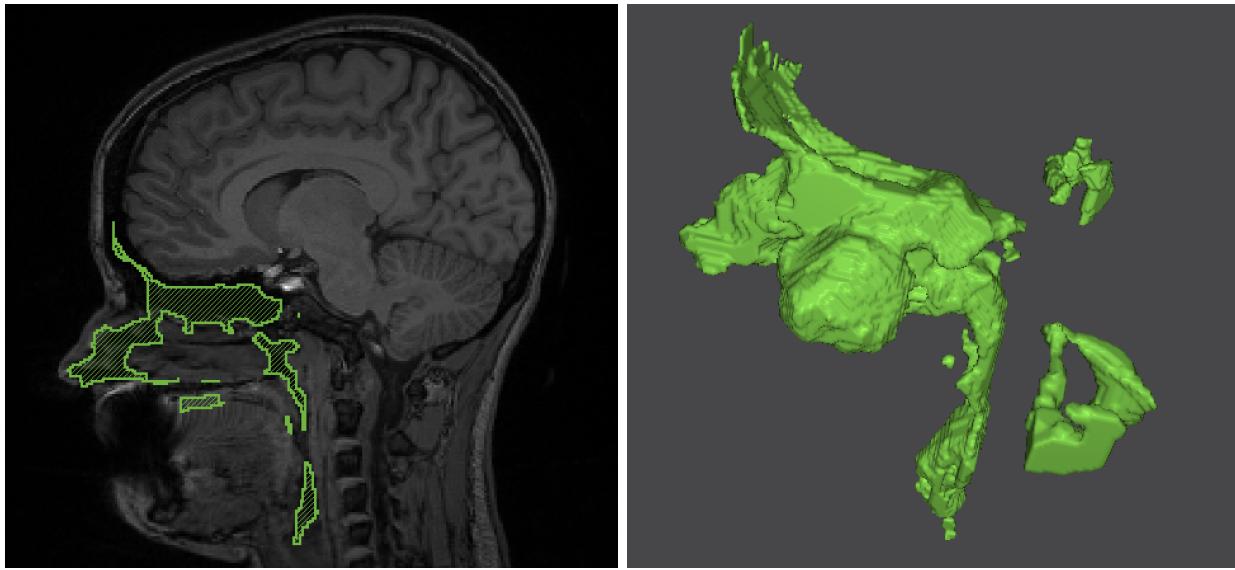


Figure 15: Sinus segmentation.

During MRI imaging, the subject was on her back, which caused the brain to shift to the back of the head, resulting in thin segmented sections on the back of the head as well as on the side of the subject's head, the bridge of the nose, and the bottom of the chin. We made these sections at least two pixels thick to ensure a mesh without holes.



Figure 16: Thin segmentation sections: side of the head (*left*), bridge of the nose (*middle*), bottom of the chin (*right*).

Segmentation of the brain was difficult because of the similar grayscale intensities across different

tissues; thresholding the image produced noisy and incomplete layers. Segmentation of the sinuses and skull was also difficult because they are represented by only black pixels, with no clear tissue boundaries.

This method, compared with Freesurfer,<sup>20</sup> Statistical Parametric Mapping through Matlab (SPM),<sup>21</sup> Atlas Based Classification through 3D Slicer,<sup>22</sup> and Seg3D methods alone, produced the most qualitatively accurate initial brain segmentation results for this data.

### 3.2 Finite Element Meshes

The highest resolution mesh we generated with the settings listed in Section 2.4 had 60.2 million elements and 10.3 million nodes. This mesh was large because of the complexity of the segmentation, including small features, thin sections, and several layers touching at once. The simulations performed slowly when using this mesh due to its size and required at least 32GB of RAM.

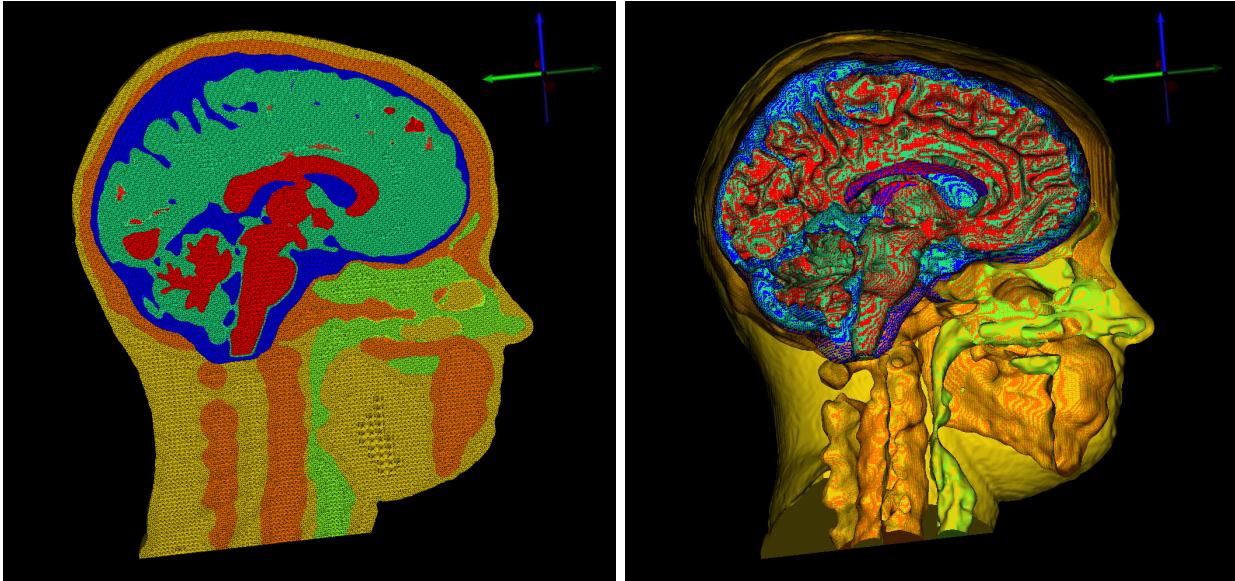


Figure 17: 60.2 M element mesh: tetrahedral mesh (*left*), surface mesh (*right*).

We attempted to generate smaller meshes to be able to run quicker simulations, but many of the meshes contained holes. After we manually changed the sizing field described in Section 2.4, we generated a mesh with 15.7 million elements and 2.7 million nodes without holes. However, this mesh contained one flat tetrahedra, which we later removed in a SCIRun network. This issue is currently being investigated by Cleaver software developers.

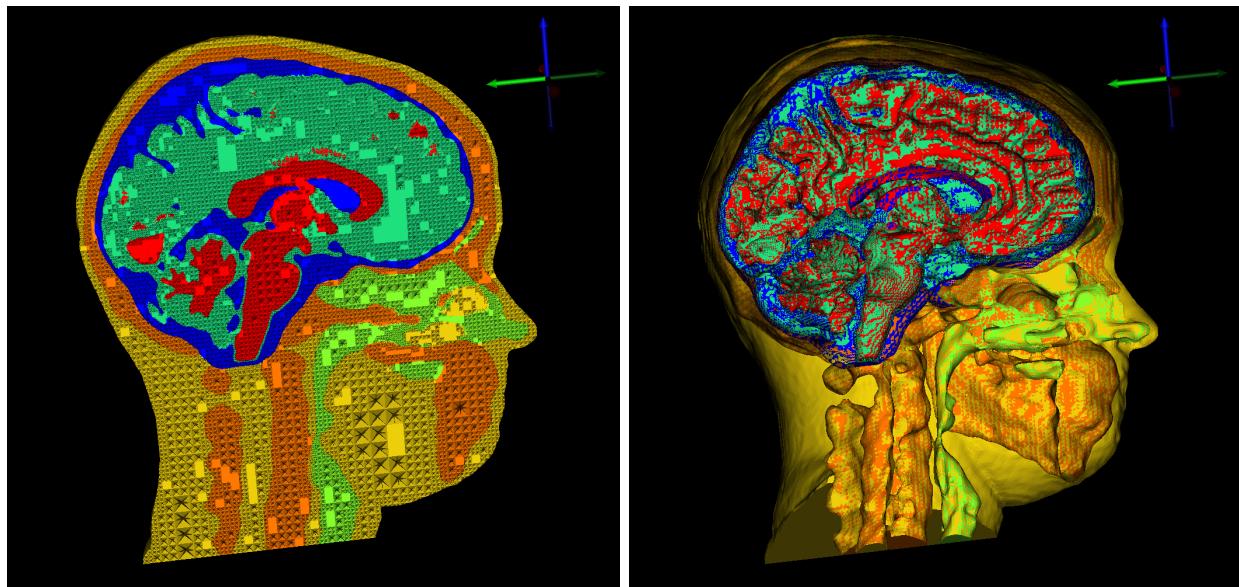


Figure 18: 15.7 M element mesh: tetrahedral mesh (*left*), surface mesh (*right*).

### 3.3 Diffusion Tensor Images

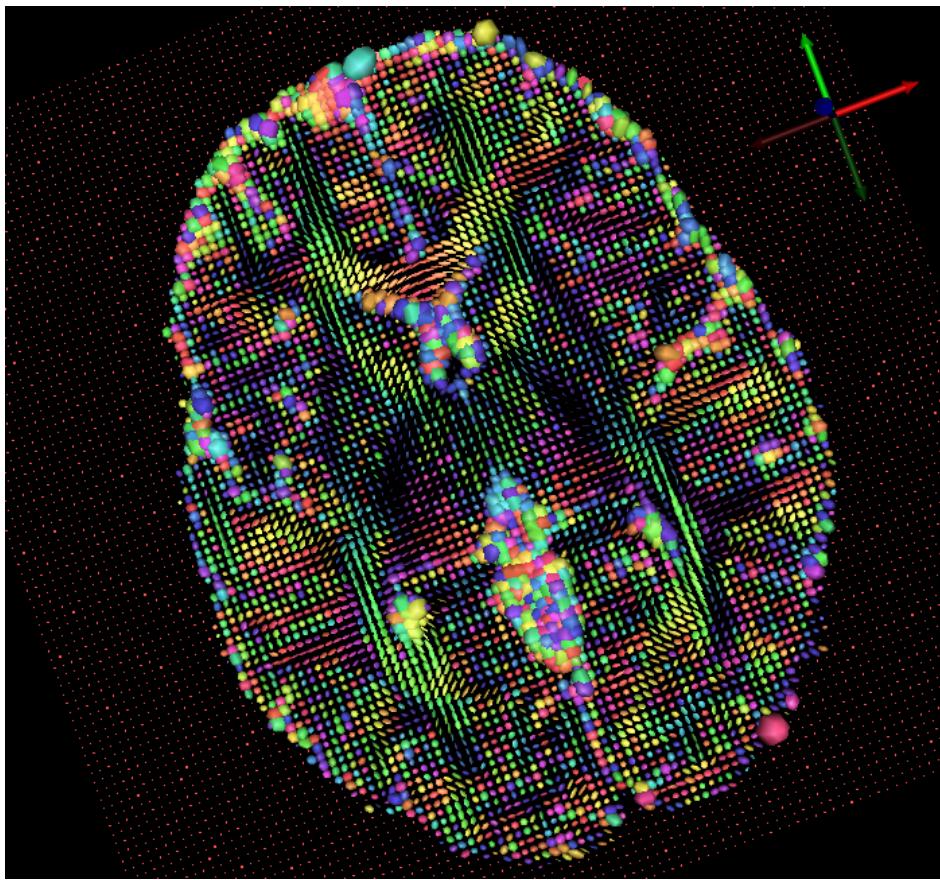


Figure 19: Diffusion tensor visualization using SCIRun.

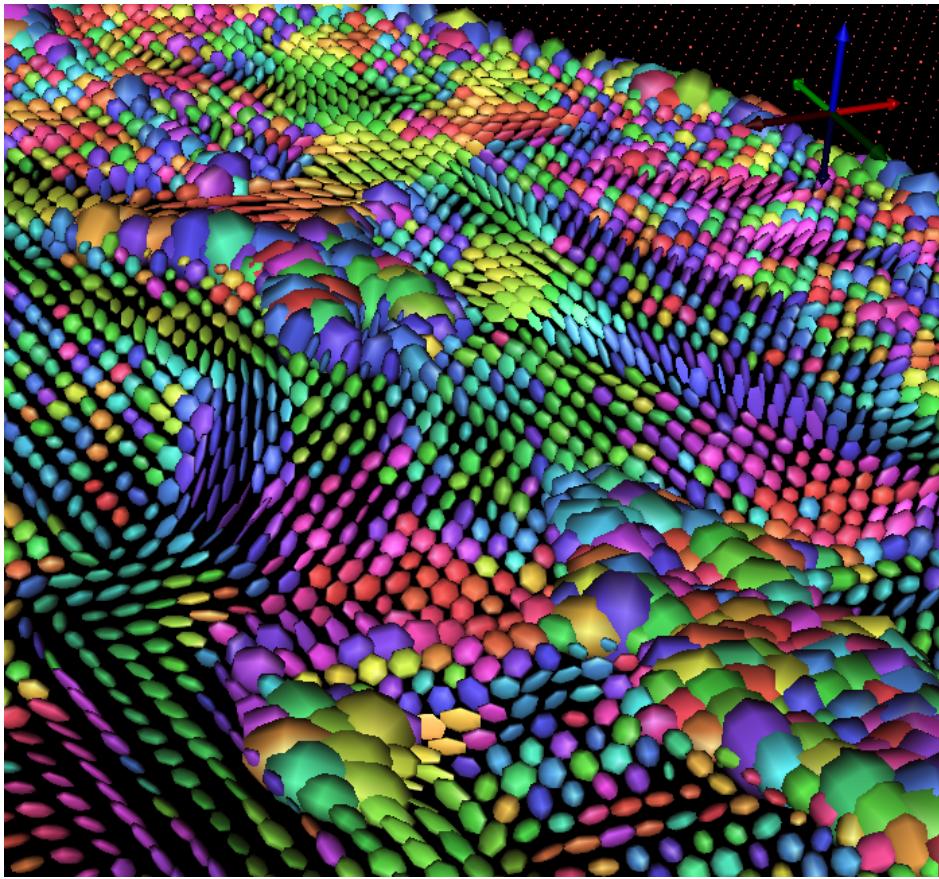


Figure 20: Diffusion tensor visualization using SCIRun.

We built the tensor field in SCIRun rather than in 3D Slicer<sup>15</sup> or FSL DTIFIT because the output data had a different orientation and could not be easily registered with the mesh in SCIRun.

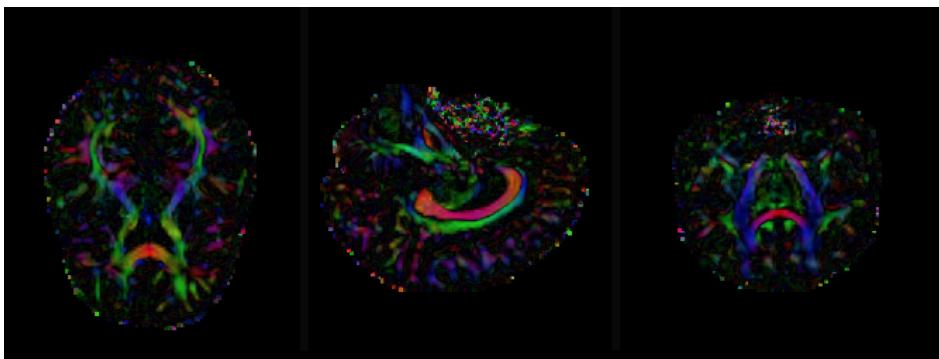


Figure 21: Example of difference in orientation between SCIRun and 3D Slicer data.

### 3.4 Forward Problem

#### 3.4.1 Isotropic

An isotropic, inhomogeneous head model is expected to have largely spherical propagation of electrical signals. We generated three-dimensional streamlines as well as isopotential lines to visualize this propagation and to compare isotropic and anisotropic conductivity. The simulations showed spherical propagation and acceptable registration of electrodes and dipoles to the mesh space.

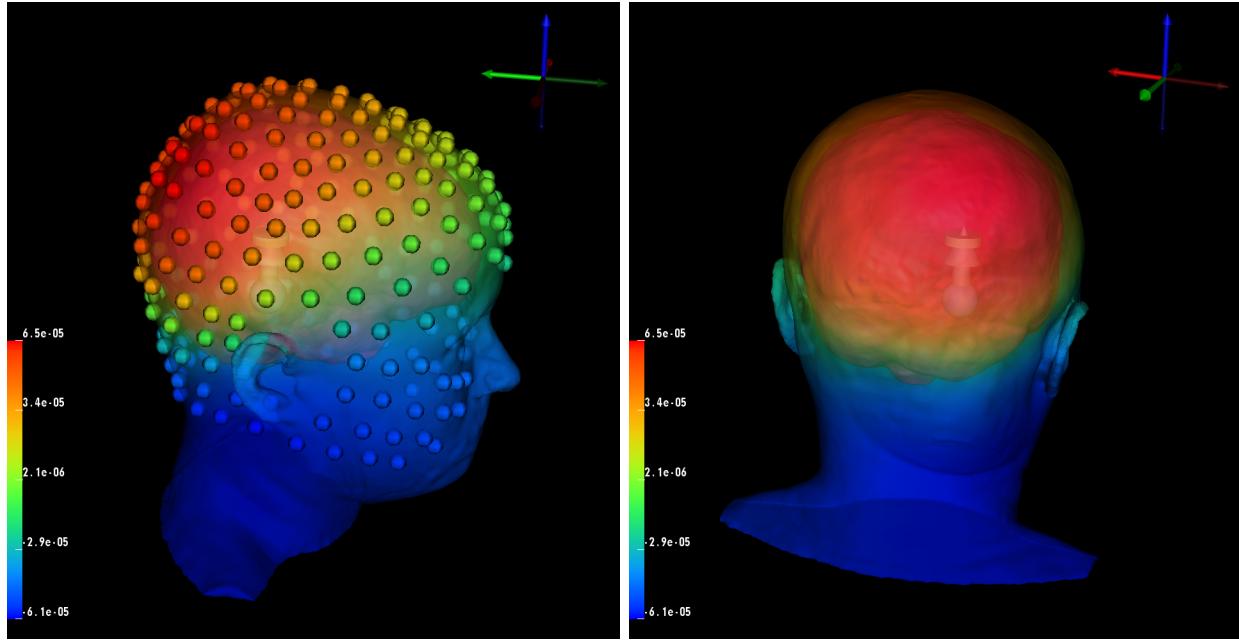


Figure 22: Isotropic forward problem solution with dipole source and data mapped onto the head surface and electrodes.

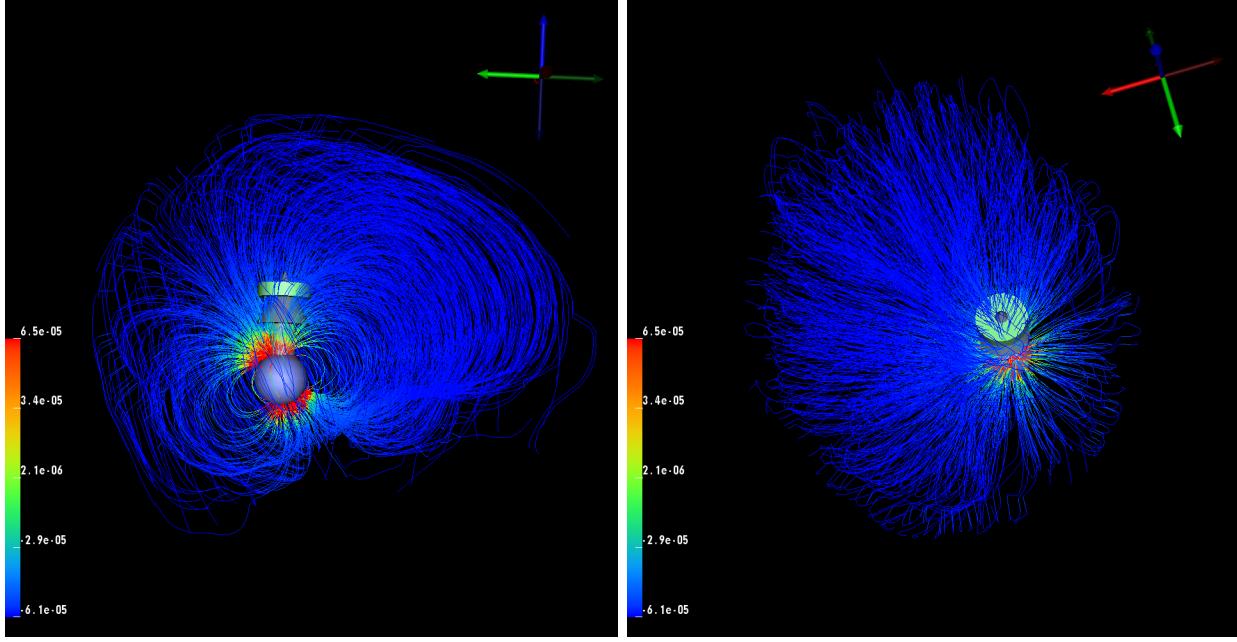


Figure 23: Isotropic streamlines with dipole source.

### 3.4.2 Anisotropic

The expectation for an anisotropic, inhomogeneous head model is to have nonspherical propagation, which can be seen with the streamline and isoline visualizations. As discussed in Section 2.5.1, two methods are used for implementing diffusion tensor data, both being available in the SCIRun network shown in Figure 34. For Figures 24 - 26, we used the scaling method (equation 3). The simulations showed nonspherical propagation and acceptable registration of the electrodes, dipoles, and mesh into the diffusion tensor space.

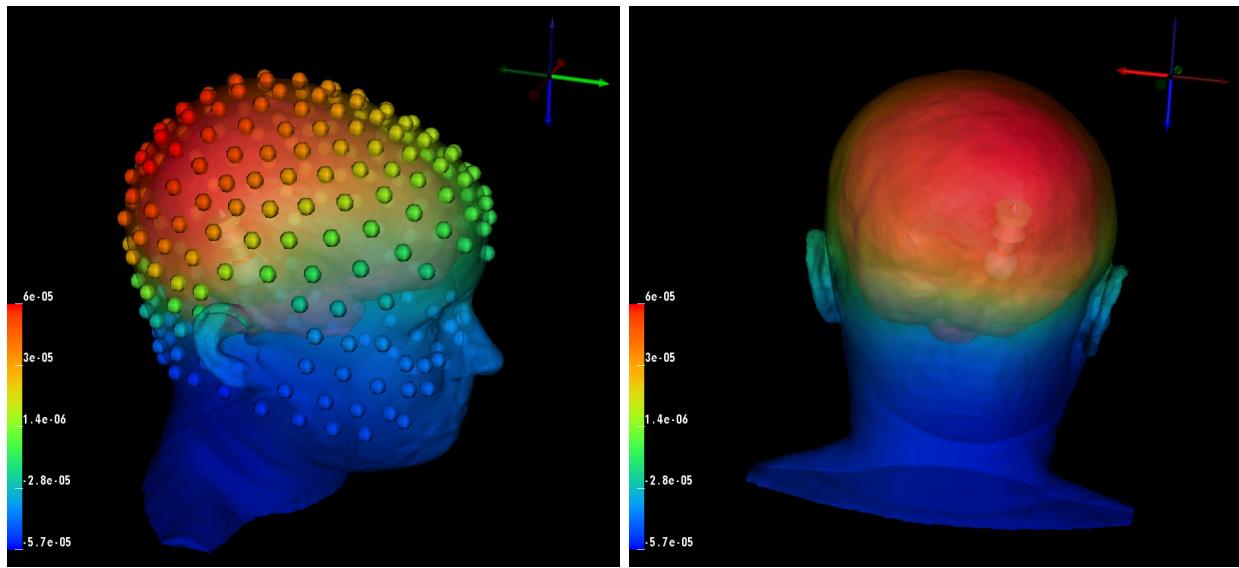


Figure 24: Anisotropic forward problem solution with dipole source and data mapped onto the head surface and electrodes.

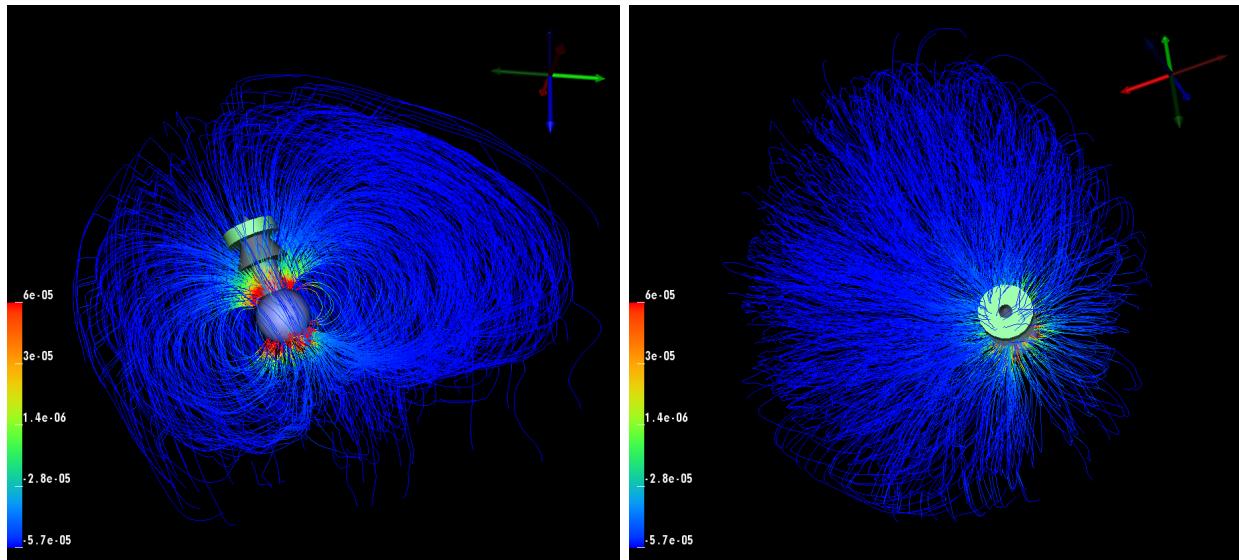


Figure 25: Anisotropic streamlines visualization with dipole source.

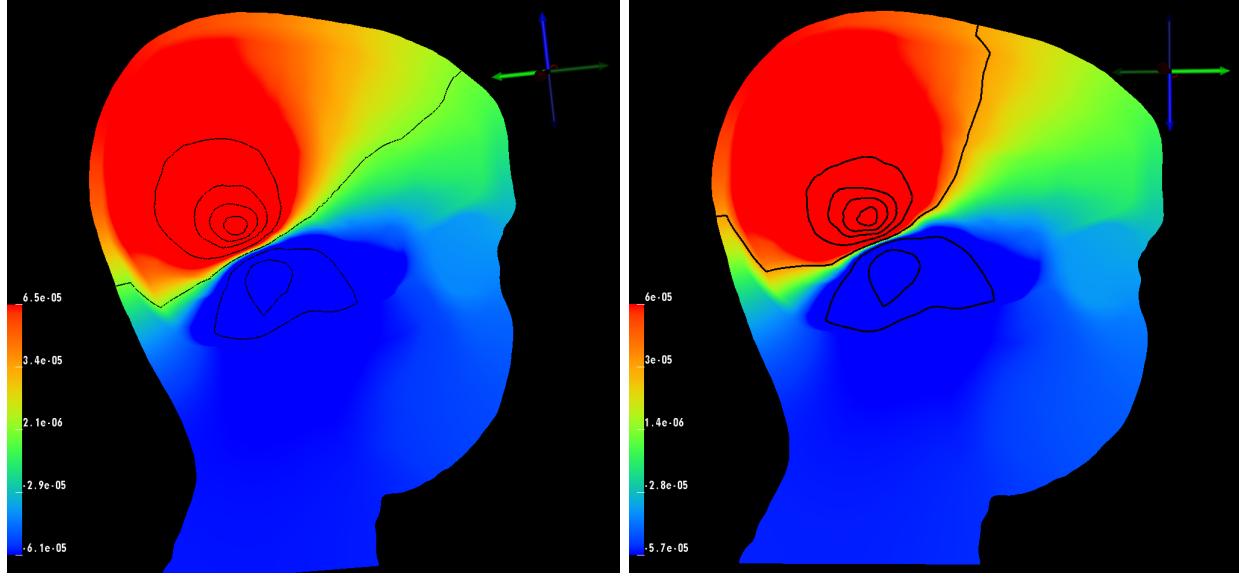


Figure 26: Isopotential lines comparison: isotropic white matter conductivity (*left*), anisotropic white matter conductivity (*right*).

### 3.5 fMRI Visualization

fMRI data was a novel imaging datatype for SCIRun. We successfully mapped and visualized the fMRI data onto the cortical surface with a rigid and manual registration to the mesh coordinate space. This mapping network allows for future use of fMRI data in simulations using the SCIRun software package.

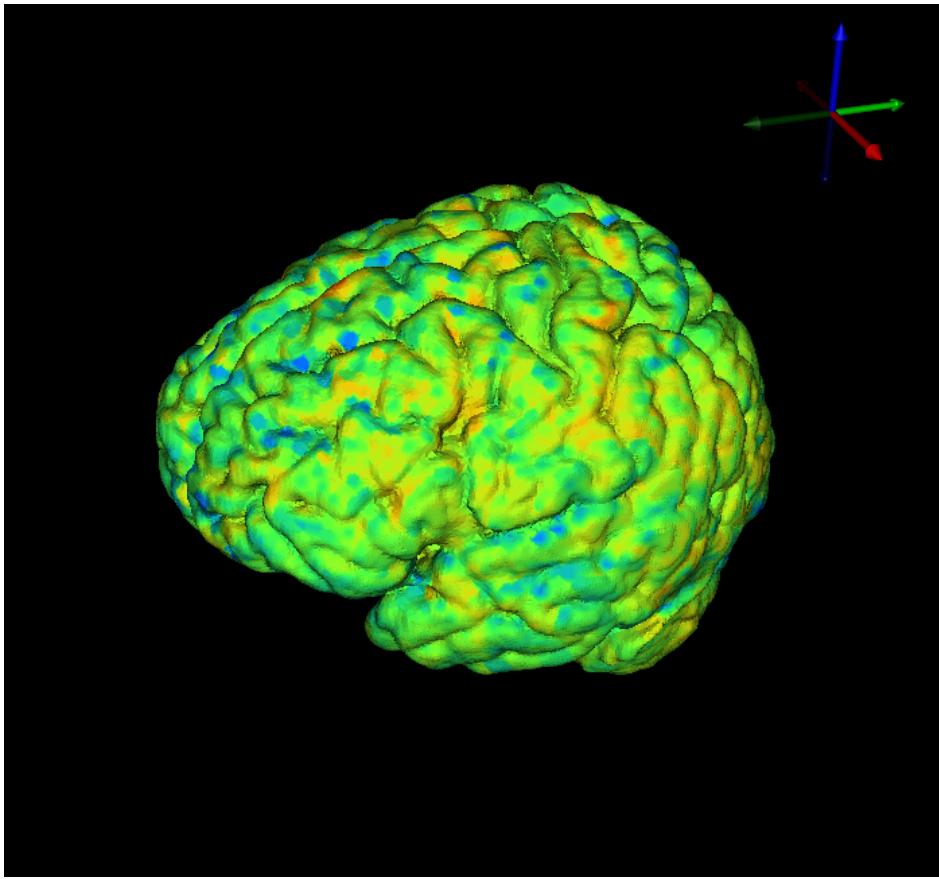


Figure 27: fMRI data mapped onto cortical surface mesh.

### 3.6 EEG Visualization

When using EEG data, the particular application dictates if further processing, filtering, or cutting of the data is necessary. This EEG dataset, taken with 256 electrodes, contained electrodes that require further processing, specifically around the eyes, which was possibly due to the blinking or rolling of the subject's eyes. The bad leads can be corrected or removed with further specific processing or trilinear interpolation.

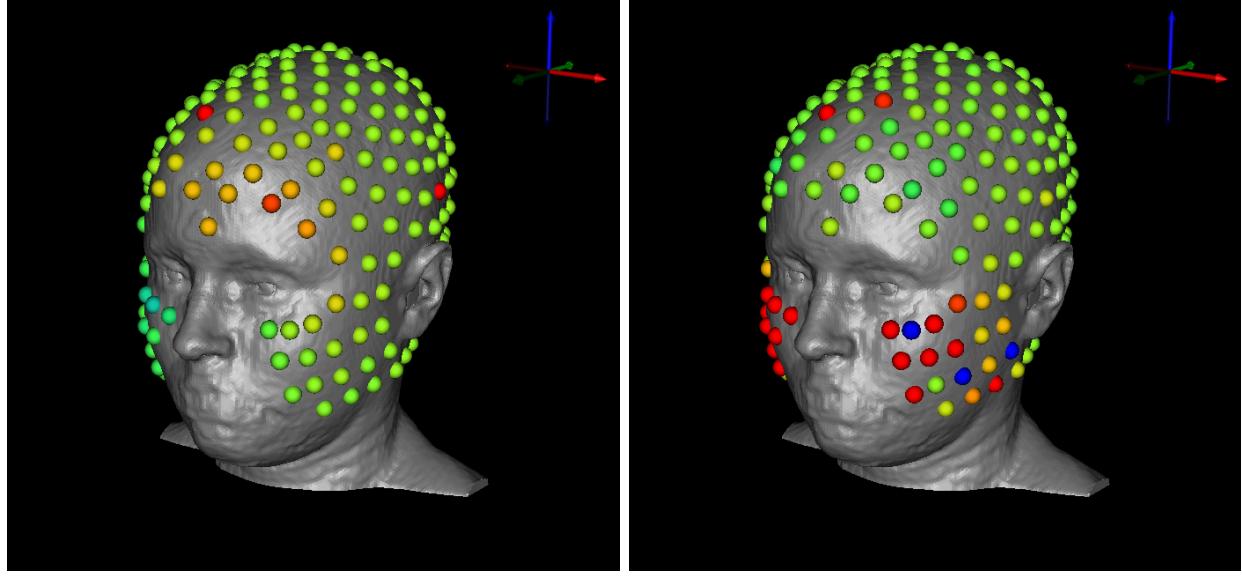


Figure 28: EEG signal visualization with 256 electrodes. Examples of electrodes that require further processing for specific applications - shown in red.

The second EEG dataset, taken with 128 electrodes, still has electrodes that require further processing around the eyes. Since the electrodes don't go as far down the cheeks as the other dataset, we do not see as many electrodes that would need further processing. However, the quality of some time steps were too poor for practical use.

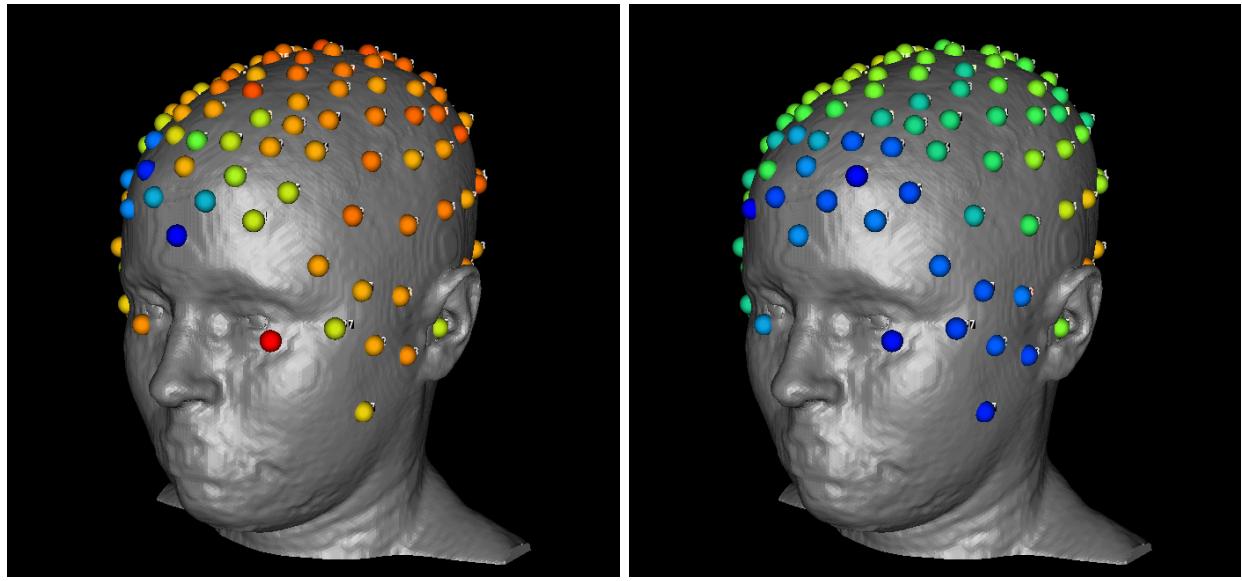


Figure 29: EEG signal visualization with 128 electrodes. Examples of electrodes that require further processing for specific applications - shown in red.

## 4 Conclusion

In this project, we have described a comprehensive pipeline to build an inhomogeneous, anisotropic head and brain model based on human data of multiple image modalities for use in electroencephalography with an emphasis on forward and inverse problem research, as well as visualizations of functional MRI and EEG data. Along with the pipeline, we have released the example subject data as open-source to enable other scientists to have a starting point and a straightforward path for further research. The high-resolution, multi-image dataset is available in parts for those who want to use only specific aspects of the project.

Future investigations based on this pipeline include finding more appropriate decimation algorithms for three-dimensional tetrahedral finite element meshes to further reduce the mesh size; more exact sinus and skull segmentation methods to improve the appearance and accuracy of these layers; and more robust registration techniques, which will provide a better transformation matrix for moving images to DTI coordinate space, especially for fMRI data. Additions to this dataset could include more methods to incorporate functional MRI data into source localization simulations and more specific processing of EEG data for different applications, resulting in more accurate and realistic visualizations.

## 5 Acknowledgements

Imaging and electroencephalography for this project were done at Electrical Geodesics Inc. at the University of Oregon. This project was supported by the National Institute of General Medical Sciences of the National Institutes of Health under grant number P41 GM103545-18.

## A Appendix

### A.1 DWI Distortion Correction

#### A.1.1 FSL Total Readout Time

Two parameters are frequently required to calculate and apply field maps to correct distortions: the effective echo spacing and the total readout time for an EPI sequence. We used “effective” echo spacing, rather than the actual echo spacing, in order to include the effects of parallel imaging, phase oversampling, etc. We defined “effective” echo spacing as:

$$\text{Effective Echo Spacing (s)} = 1 / (\text{BandwidthPerPixelPhaseEncode} * \text{MatrixSizePhase})$$

The total readout time (FSL definition) was:

$$\text{Total readout time (FSL)} = (\text{MatrixSizePhase} - 1) * \text{EffectiveEchoSpacing}$$

Total readout time is a necessary input for using FSL’s topup and eddy tools for DWI image correction, and can be obtained by using MRICorrect.

### A.1.2 MRIConvert

The software package MRIConvert provided acquisition information about the dicom series, as well as converted the MRI to a NiFTI format, including effective echo spacing and total readout time.<sup>27</sup>

To obtain this information, we loaded the dicom series for either DWI acquisition. We selected “Options” to ensure the DWI was saved in NiFTI format. We then selected “Convert All” to save all the files into the output directory specified upon opening MRIConvert. The text file included the FSL-defined total readout time, which was contained in the acquisition parameter file in seconds as a unit. MRIConvert also output the b-values and b-vectors files, which were the same for both the DWI AP and DWI PA scans. The last input file required was an “index.txt” file, which contained one column with 65 rows (for 64 directions plus the b0 image) of 1’s.

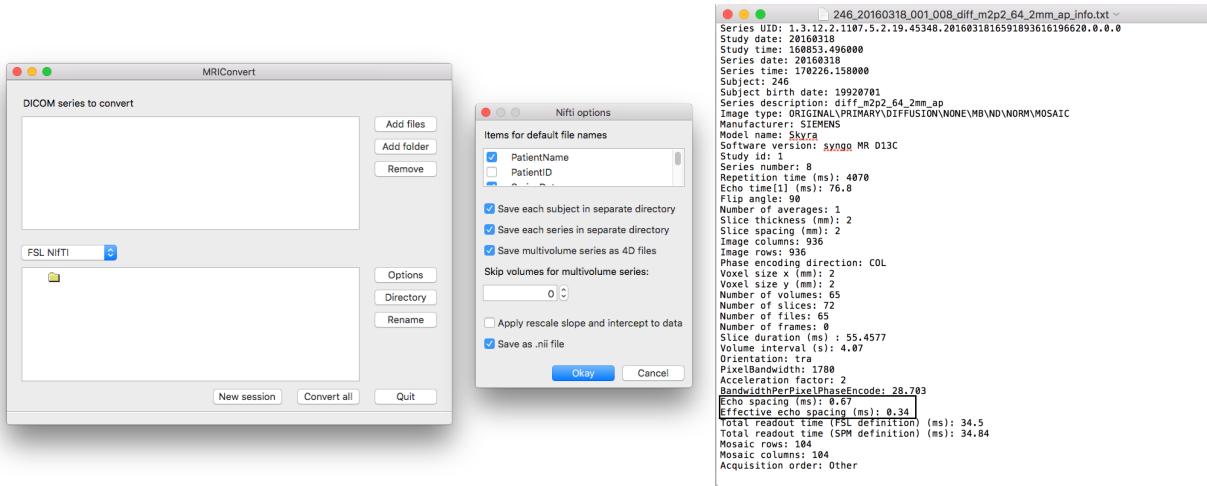


Figure 30: MRIConvert (*left*) with options (*middle*) & output (*right*).

```
0 -1 0 0.0345
0 1 0 0.0345
```

Figure 31: Acquisition parameters text file.

### A.1.3 FSL’s Topup and Eddy Command Line Tools

We made a separate folder for topup results that included the following files: the acquisition parameters file, the index file, b-values, b-vectors, and the DWI AP and DWI PA files. To run topup, we renamed the DWI AP image DWI\_up and the DWI PA image DWI\_down. We renamed the b-values and b-vectors dwi.bval and dwi.bvec, respectively. After all files were in place, we executed the following command line commands:

```
fslroi DWI\_up b0\_up 0 1
fslroi DWI\_down b0\_down 0 1

fslmerge -t both\_b0 b0\_up b0\_down

topup --imain=both\_b0 --datain=acq_params.txt --config=mine.cnf --out=topup\_results
applytopup --imain=b0\_up,b0\_down --inindex=1,2 --datain=acq_params.txt
```

```

--topup=topup\_results --out=b0\_hifi

bet b0\_hifi b0\_hifi\_brain -m -f 0.2
eddy --imain=DWI\_up --mask=b0\_hifi\_brain\_mask --index=index.txt --acqp=acq_params.txt
--bvecs=dwi.bvec --bvals=dwi.bval --fwhm=0 --topup=topup\_results --flm=quadratic
--out=eddy\_unwarped

```

By running these commands, we first obtained the b0 image, which is the baseline image used for calculating field maps for both encoding directions. Then the two b0's were merged together into one file, topup and eddy were applied for distortion correction, and ‘bet’ was applied for brain extraction. The distortion corrected file was named “eddy\\_unwarped.nii.”

## A.2 DTIFIT

To use DTIFIT we first opened FSL, and then we chose “FDT Diffusion,” followed by “DTIFIT Reconstruct diffusion tensors” in the drop-down menu and input files manually; Table 4 lists the files selected.

Table 4: DTIFIT Input Files

Diffusion weighted data	eddy_unwarmed.nii
BET binary brain mask	b0_hifi_brain_mask.nii
Output basename	desired output location
Gradient directions	dwi.bvec
b values	dwi.bval

DTIFIT output the eigenvalues (named L1, L2, and L3) and the eigenvectors (named V1, V2, and V3) for the diffusion tensor field. We converted the files from NiFTI format to nrrd format using ITK-SNAP.<sup>28</sup> These are the input files for the SCIRun network in Figure 32.

## A.3 NiFTI Toolbox for fMRI Preprocessing

After running the fMRI data through the fcon pipeline, we converted the preprocessed fMRI data from four-dimensional data to two-dimensional data. We opened “rest.nii” in Matlab using the “load\_nii(‘rest.nii’)” function within the NiFTI toolbox.<sup>29</sup> We then resized the four-dimensional “img” variable into a two-dimensional variable for use in SCIRun using Matlab’s “resize” function.

## A.4 EEG Data Matrix in MATLAB

The EEG data was output in an .edf file format. We calculated the EEG signals matrix using a Matlab script called “edfRead.m.”<sup>30</sup> To run this script, we used “[hdr, record] = edfread(fname).” The variable ‘record’ contained the EEG signals.

## A.5 Networks

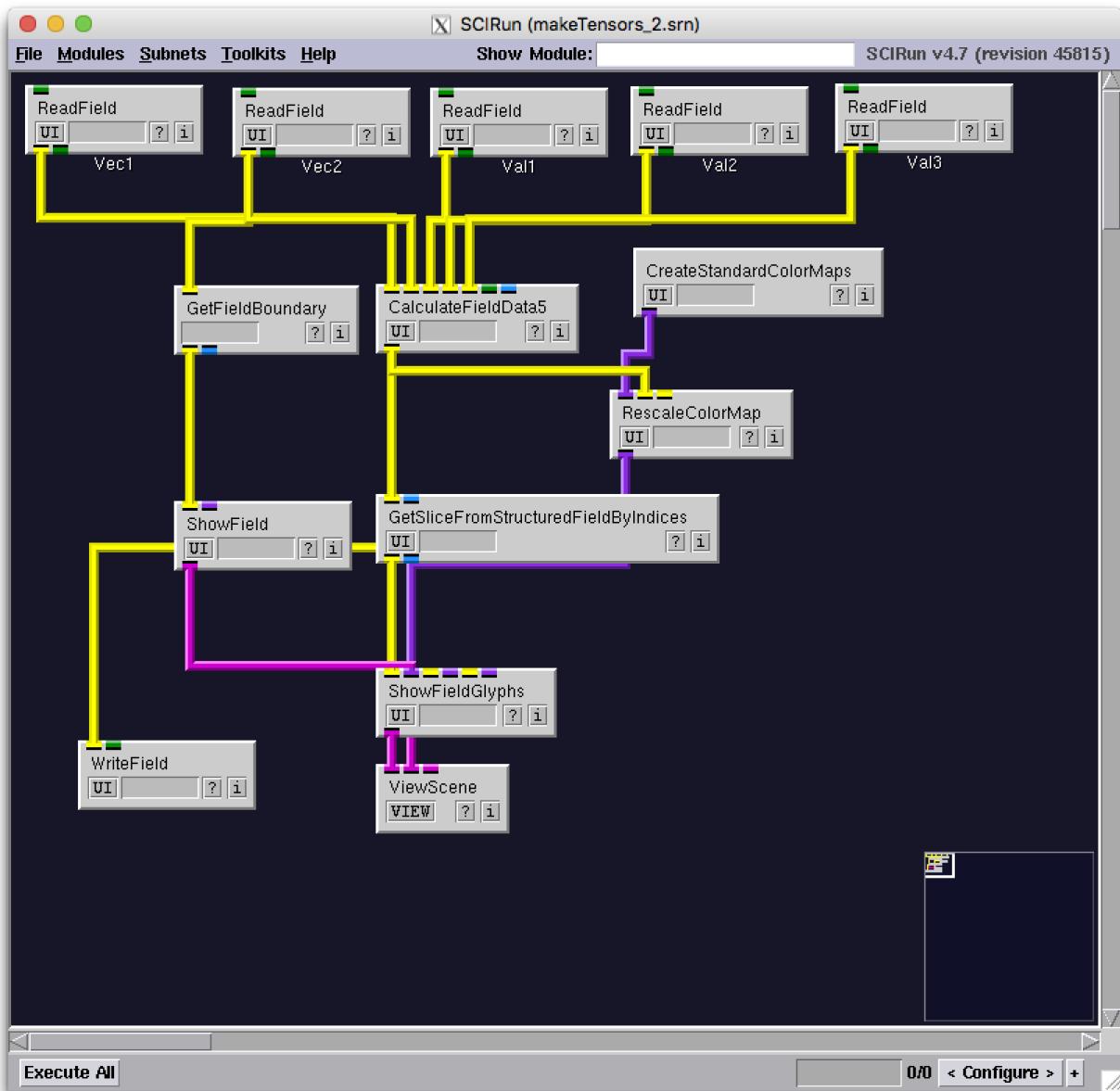


Figure 32: SCIRun network to build diffusion tensor dataset.

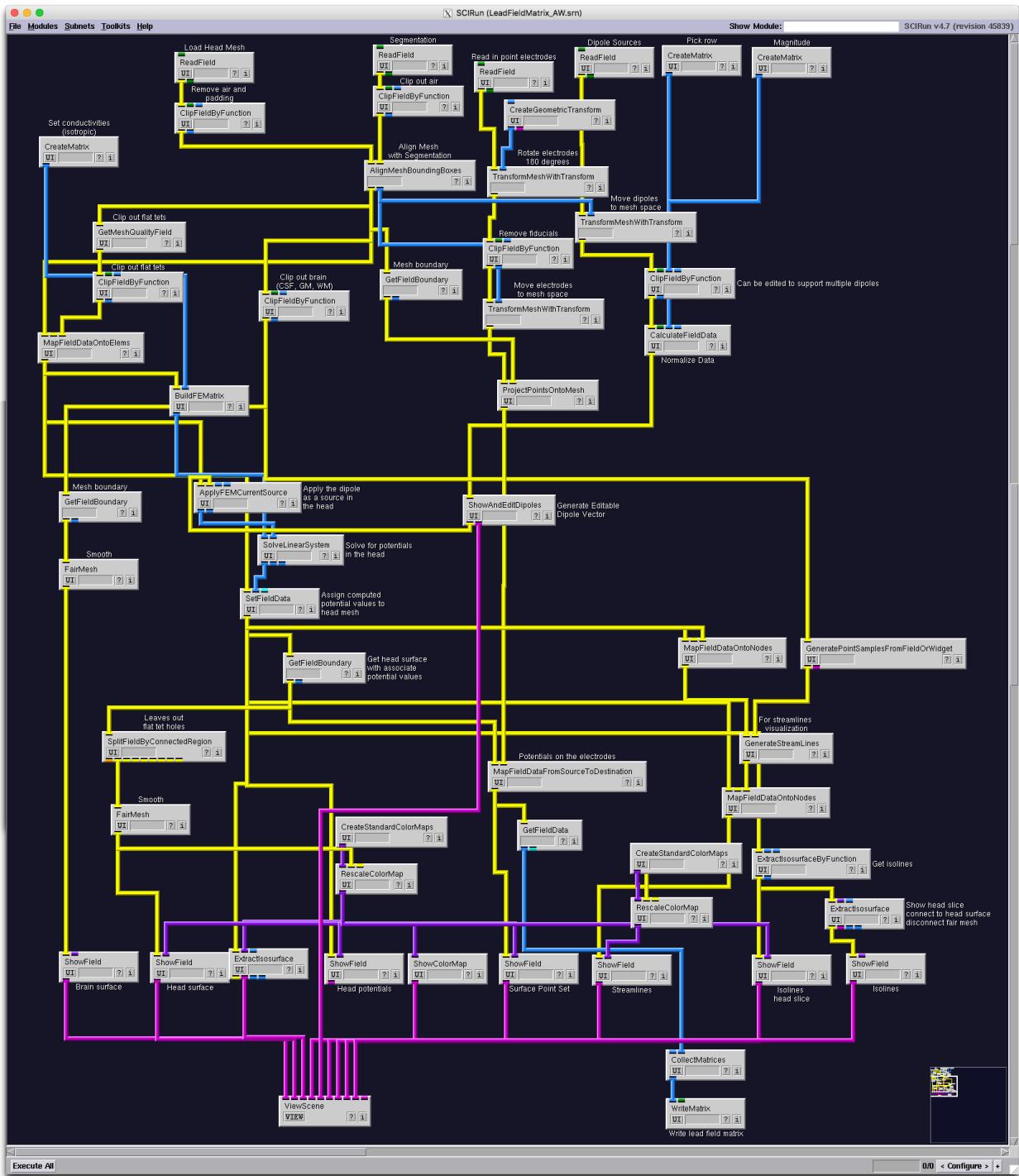


Figure 33: SCIRun network for isotropic forward problem.

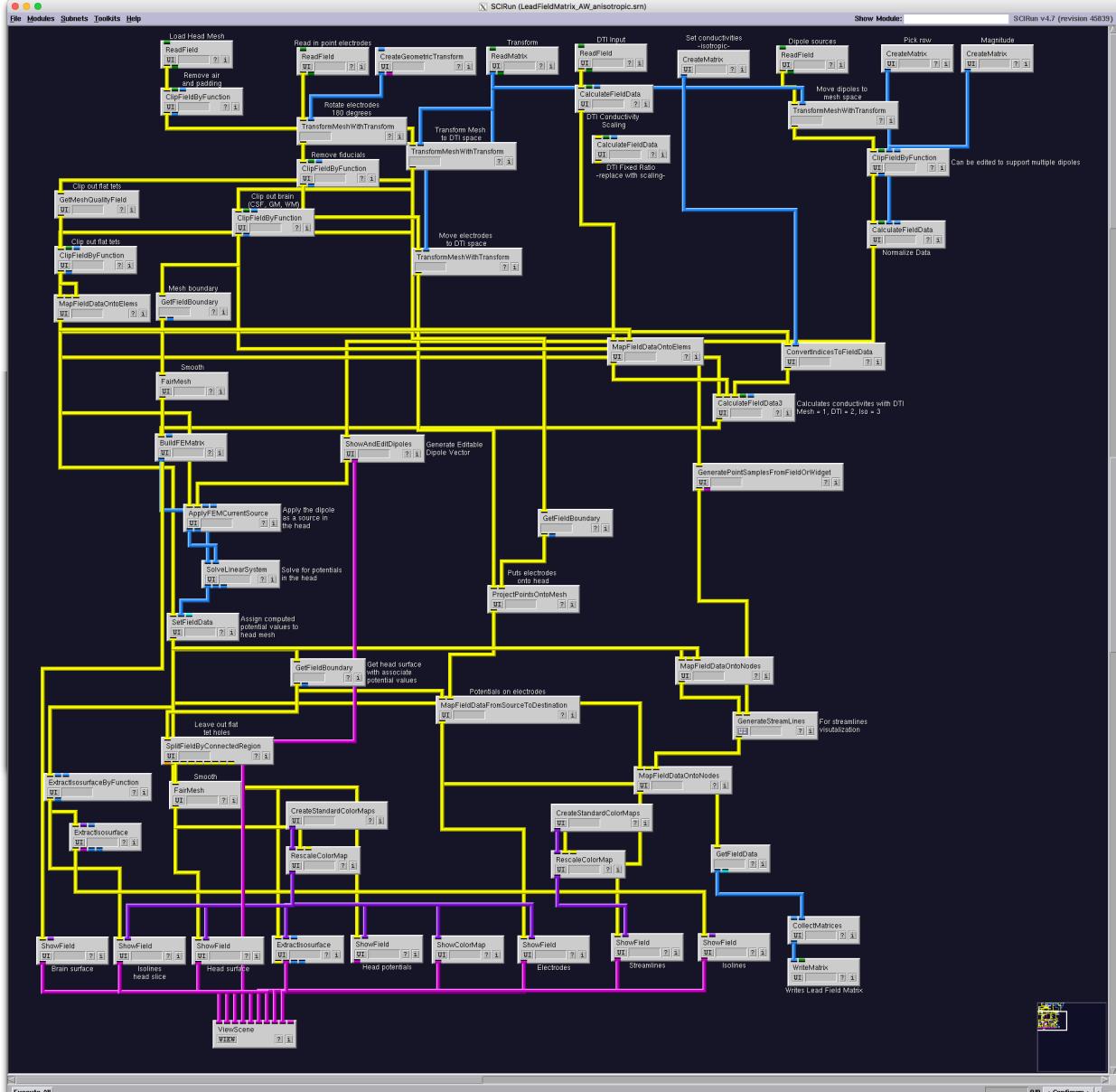


Figure 34: SCIRun network for anisotropic forward problem.

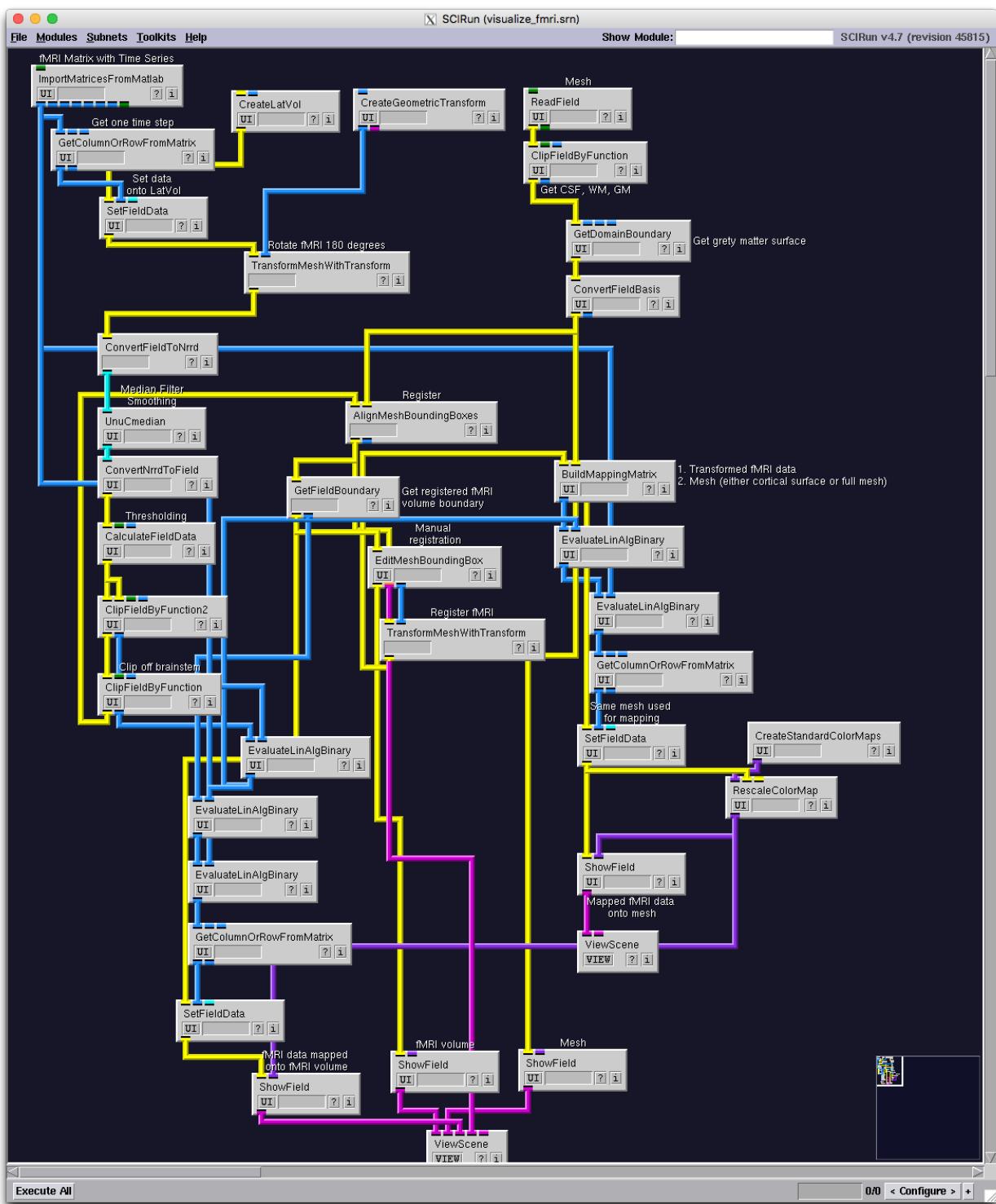


Figure 35: SCIRun network for fMRI visualization.

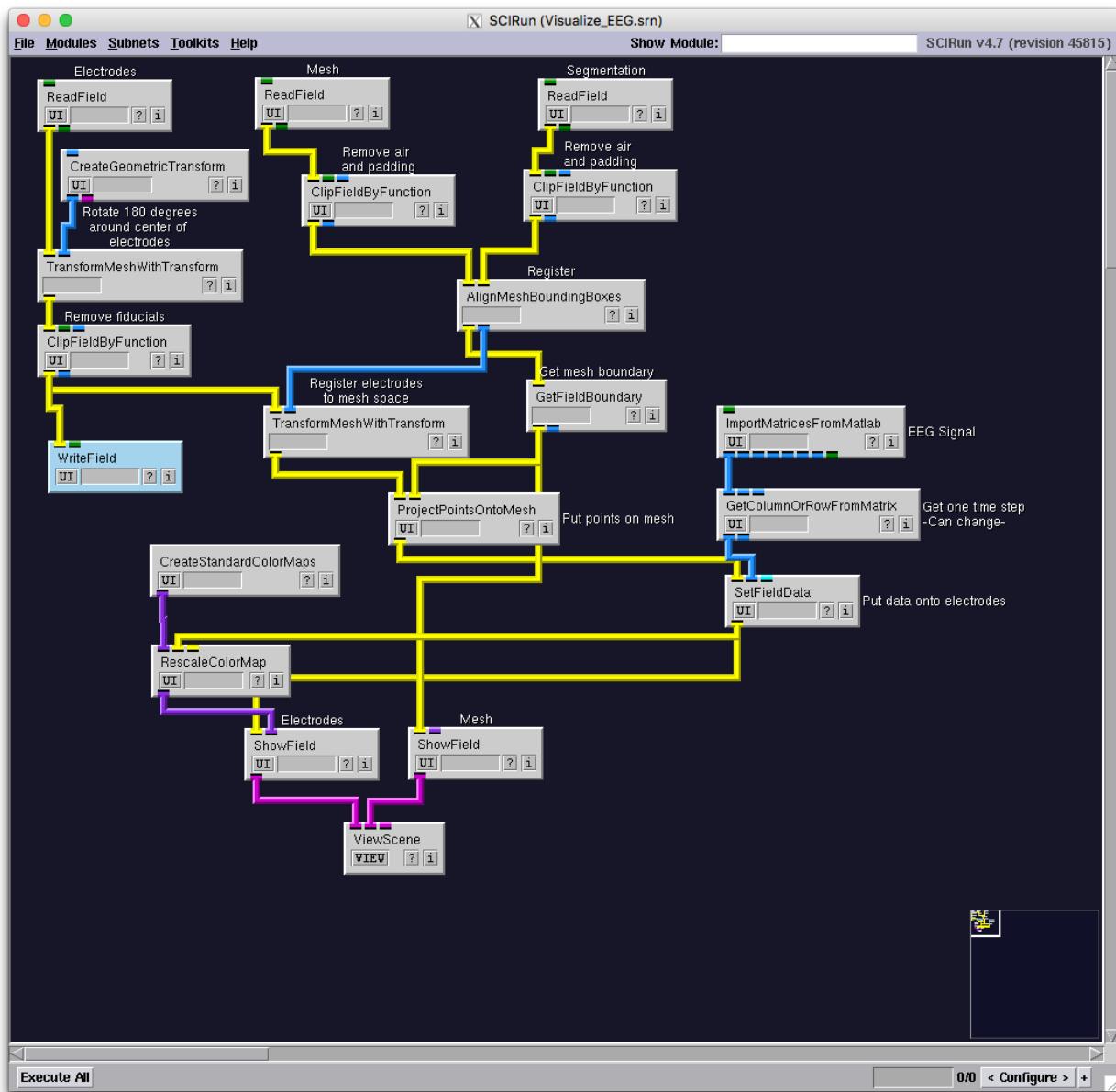


Figure 36: SCIRun network for EEG visualization.

## References

- [1] R.S. MacLeod, J.G. Stinstra, S. Lew, R.T. Whitaker, D.J. Swenson, M.J. Cole, J. Krüger, D.H. Brooks, and C.R. Johnson. Subject-specific, multiscale simulation of electrophysiology: a software pipeline for image-based models and application examples. *Philosophical Transactions of The Royal Society A, Mathematical, Physical & Engineering Sciences*, 367(1896):2293–2310, 2009.
- [2] C.R. Johnson. *Computational Methods and Software for Bioelectric Field Problems*, volume 1, chapter 43, pages 1–28. CRC Press, Boca Raton, FL, USA, 4 edition, 2015.
- [3] C.R. Johnson. Biomedical visual computing: Case studies and challenges. *IEEE Computing in Science and Engineering*, 14(1):12–21, 2012.
- [4] C.R. Johnson and D.M. Weinstein. Biomedical computing and visualization. In Vladimir Estivill-Castro and Gill Dobbie, editors, *Proceedings of the Twenty-Ninth Australasian Computer Science Conference (ACSC2006): Conferences in Research and Practice in Information Technology (CRPIT)*, volume 48, pages 3–10, 2006.
- [5] C.R. Johnson, R.S. MacLeod, S.G. Parker, and D.M. Weinstein. Biomedical computing and visualization software environments. 47(11):64–71, 2004.
- [6] J.P. Mugler and J.R. Brookeman. Three-dimensional magnetization-prepared rapid gradient-echo imaging (3d mp rage). *Magn. Reson. Med.*, 15(1):152–157, July 1990.
- [7] M.P. Lichy, B.M. Wietek, J.P. Mugler, W. Horger, M.I. Menzel, A. Anastasiadis, K. Siegmann, T. Niemeyer, A. Konigsrainer, B. Kiefer, F. Schink, C.D. Claussen, and H.P. Schlemmer. Magnetic resonance imaging of the body trunk using a single-slab, three-dimensional, t2-weighted turbo-spin-echo sequence with high sampling efficiency (space) for high spatial resolution imangiing: initial clinical experiences. *Invest. Radiol.*, 40(12):754–760, December 2005.
- [8] T.G. Reese, O. Heid, R.M. Weisskoff, and V.J. Wedeen. Reduction of eddy-current-induced distortion in diffusion mri using a twice-refocused spin echo. *Magn. Reson. Med.*, 49(1):177–182, 2003.
- [9] J. Juntu, J. Sijbers, D.V. Dyck, and J. Gielen. Bias field correction for mri images. In: Kurzyński M., Puchała E., Woźniak M., żołnierk A. (eds) *Computer Recognition Systems. Advances in Soft Computing*, vol 30. Springer, Berlin, Heidelberg, (543–551), 2005.
- [10] Y. Zang, M. Brady, and S. Smith. Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Trans Med Imag*, 20(1):45–57, 2001.
- [11] J.L.R. Anderson, S. Skare, and J. Ashburner. How to correct susceptibility distortions in spin-echo echo-planar images; applications to diffusion tensor imaging. *NeuroImage*, 20(2):870–888, 2003.
- [12] S.M. Smith, M. Jenkinson, M.W. Woolrich, C.F. Beckmann, T.E.J. Behrens, H. Johansen-Berg, P.R. Bannister, M. De Luca, I. Drobniak, D.E. Flitney, R. Niazy, J. Saunders, J. Vickers,

- Y. Zhang, N. De Stefano, J.M. Brady, and P.M. Matthews. Advances in functional and structural mr image analysis and implementation as fsl. *NeuroImage*, 23(S1):208–219, 2004.
- [13] FSL. Fsl dtifit. <https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/FDT/UserGuide> Section 2.4.
- [14] SCIRun: A Scientific Computing Problem Solving Environment, Scientific Computing and Imaging Institute (SCI), Download from: <http://www.scirun.org>.
- [15] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F.M. Fennessy, M. Sonka, J. Buatti, S.R. Aylward, J.V. Miller, S. Pieper, and R. Kikinis. 3d slicer as an image computing platform for the quantitative imaging network. *Magn. Reson. Imaging*, 30(9):1323–31, November 2012.
- [16] NITRC. 1000 functional connectomes project. [http://fcon\\_1000.projects.nitrc.org/](http://fcon_1000.projects.nitrc.org/).
- [17] U. Aydin, J. Vorwerk, P. Kupper, M. Heers, H. Kugel, A. Galka, L. Hamid, J. Wellmer, C. Kellinghaus, S. Rampp, and C.H. Wolter. Combining eeg and meg for the reconstruction of epileptic activity using a calibrated realistic volume conductor model. *PLOS ONE*, 9(3), March 2014.
- [18] Seg3D: Volumetric Image Segmentation and Visualization. Scientific Computing and Imaging Institute (SCI), Download from: <http://www.seg3d.org>.
- [19] S.M. Smith. Fast robust automated brain extraction. *Human Brain Mapping*, 17(3):143–155, November 2002.
- [20] A.M. Dale, B. Fischl, and M.I. Sereno. Cortical surface-based analysis. i. segmentation and surface reconstruction. *NeuroImage*, 9:179–194, 1999.
- [21] Wellcome Trust Centre for Neuroimaging. Statistical parametric mapping, May 2016. <https://www.mathworks.com/matlabcentral/fileexchange/56963-statistical-parametric-mapping>.
- [22] Koen Van Leemput, Frederik Maes, Dirk Vandermeulen, and Paul Suetens. Automated model-based tissue classification of mr images of the brain. *IEEE transactions on medical imaging*, 18(10):897–908, 1999.
- [23] A. Torrado-Carvajal, J.L. Herranz, E. Alcain, A.S. Montemayor, L. Garcia-Cañamaque, J.A. Hernandez-Tamames, and . . . & N. Malpica. Fast patch-based pseudo-ct synthesis from t1-weighted mr images for pet/mr attenuation correction in brain studies. *Journal of Nuclear Medicine*, 57(1):136–143, 2016.
- [24] Cleaver: A MultiMaterial Tetrahedral Meshing Library and Application. Scientific Computing and Imaging Institute (SCI), Download from: <http://www.sci.utah.edu/cibc/software.html>.
- [25] M. Rullmann, A. Anwander, M. Dannhauer, S.K. Warfield, F.H. Duffy, and C.H. Wolters. Eeg source analysis of epileptiform activity using a 1 mm anisotropic hexahedra finite element head model. *NeuroImage*, 44:399–410, 2009.
- [26] D. Gullmar, J. Haueisen, and J. R. Reichenbach. Influence of anisotropic electrical conductivity in white matter tissue on the eeg/meg forward and inverse solution. a high-resolution whole head simulation study. *NeuroImage*, 51:145–163, 2010.

- [27] X. Li, P.S. Morgan, J. Ashburner, J.C. Smith, and C. Rorden. The first step for neuroimaging data analysis: Dicom to nifti conversion. *Journal of Neuroscience Methods*, 264:47–56, 2016.
- [28] P.A. Yushkevich, J. Piven, H.C. Hazlett, R.G. Smith, S. Ho, J.C. Gee, and G. Gerig. User-guided 3d active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *NeuroImage*, 31(3):1116–28, July 2006.
- [29] Jimmy Shen. Tools for nifti and analyze image, Jan 2014. <https://www.mathworks.com/matlabcentral/fileexchange/8797-tools-for-nifti-and-analyze-image>.
- [30] Brett Shoelson. edfread, September 2016. <https://www.mathworks.com/matlabcentral/fileexchange/31900-edfread>.