

A Complete Brain Model (fix)

¹Alexandra Warner, ¹Chris Johnson

¹University of Utah, Salt Lake City UT; ²Placeholder in case we need to put more refs

December 5, 2017

Abstract

Here is some abstract text

1 Introduction

Introduction text.

2 Methods

ADD PIPELINE FIGURE HERE

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 (MRI) scans were acquired on a healthy, female volunteer who is 23 years of age on a Skyra 3T full-body scanner (Siemens Medical Solutions).

The T_1 -weighted scan was performed with a 3D magnetization prepared rapid gradient echo (MPRAGE) sequence (Mugler and Brookman, 1990) – get this reference. 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 sampling perfection with application-optimized contrast using different flip angle evolutions (check ref) (SPACE) sequence (Lichy et. al., 2005; Mugler et al., 2000). 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). Both phase encoding directions were performed (anterior to posterior and posterior to anterior) with 64 diffusion directions each. Further sequence parameters for each scan was 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 function MRI (fMRI) scans were acquired with blood oxygenation level dependent contrast (BOLD). 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.

A continuous electroencephalogram (EEG) was recorded using a 256-channel HydroCel Geodesic Sensor Net that was connected to a NetAmps 400 amplifier and referenced online to a single vertex electrode shown in Figure 1. Channel impedances were kept at or below 50 kOhms and signals were sampled at 250Hz. The EEG was recorded while the subject sat quietly in a chair, alternating two minute epochs of eyes open and eyes closed for a total of 12 minutes.

All acquisition reports will be included with the dataset.



Figure 1: 256-channel HydroCel Geodesic Sensor Net on Subject

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 reducing the high frequencies of the images such as edges and contours. It changes the intensity values of image pixels so that the same tissue has a different distribution of grayscale intensities across the image. (ref) An estimated bias field correction on the T_1 and T_2 MRI's was done using FMRIB Software Library (FSL) FAST, and will be further described in the Segmentation section.

2.2.2 DWI Distortion Correction

Diffusion weighted images 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-encode blips (anterior to posterior (AP) and posterior to anterior(PA)), resulting in pairs of images with distortions going in opposite directions. From these pairs the susceptibility-induced off-resonance field was estimated using a method similar to that described in [Andersson 2003] as implemented in FSL [Smith 2004] and the two images were combined into a single corrected one. This was accounted for using FSL's topup and eddy command line tools.

Before running these tools, an acquisition parameters text file needs to be created with the FSL-defined total readout time. There are two parameters that are frequently needed when calculating and applying field maps: the effective echo spacing and the total readout time for an EPI sequence. "Effective" echo spacing was used, rather than the actual echo spacing, in order to include the effects of parallel imaging, phase oversampling, etc. Multiplying this by the size of the reconstructed image in the phase direction will give you the reciprocal of the effective echo spacing:

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

The total readout time (FSL definition) is:

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

MRIConvert (ref) is a software package that provides all of the acquisition information about a dicom series, as well as converts to a NiFTI format, including effective echo spacing and total readout time. To obtain this information, first load in the dicom series for either DWI acquisition. Choose "Options" to ensure that the DWI will be saved as a NiFTI, and click "Convert All." This will save all the files into the output directory specified upon opening MRIConvert. The text file will include the FSL-defined total readout time which should be contained in the acquisition parameter file in seconds. MRIConvert also outputs the b-values and b-vectors files, and they should be the same for both the DWI AP and DWI PA scans. The last input file needed is an "index.txt" file. This text file contains one column with 65 rows (for 64 directions plus the b0 image) of 1's.

(MRIConvert, Options, and text file figures) (acquisition parameters file and index file)(maybe a figure out the outputs in finder)

Create a separate folder for topup results and include the following files: the acquisition parameters file, the index file, b-values, b-vectors, and the DWI AP and DWI PA files. In the following instructions, DWI AP was renamed as DWI_up and DWIAP was renamed to DWI_down. The b-values and b-vectors were renamed to dwi.bval and dwi.bvec, respectively. After all of these files are in place, run 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 --bva
```

These commands first obtain the b0 image, which is the baseline image used for calculating field maps, for both encoding directions. Then the two b0's are merged together into one file. Topup and eddy are applied for distortion correction. Bet is applied for brain extraction. The distortion corrected file is named "eddy_unwarped.nii."

2.2.3 Diffusion Tensor Images

After the DWI images have been corrected, diffusion tensor images (DTI) are calculated using FSL's DTIFIT. Upon opening FSL choose "FDT Diffusion." Then choose "DTIFIT Reconstruct diffusion tensors" in the drop down menu, select to input files manually, and put in the following files. (reference and figure of DTIFIT)

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

DTIFIT will output the eigenvalues (named L1, L2, and L3) and the eigenvectors (named V1, V2, and V3) for the diffusion tensor field. The files were converted from NiFTI format to nrrd format using ITK-SNAP (reference and figure) although there is a loss of precision. The files were then input into SCIRun to build the tensor field using the eigenvalue and eigenvectors. The SCIRun CalculateFieldData module only requires two eigenvectors as input because it calculates the third eigenvector automatically since it should be orthogonal to the first two. This process can be done in either SCIRun 4 or SCIRun 5 and the same results are produced.

(SCIRun figures of building DTI)

The tensor field was built in SCIRun rather than in 3D Slicer (reference if haven't seen yet) or FSL DTIFIT because the output data would be "backwards" and couldn't be registered with the mesh.

2.2.4 fMRI

fMRI data was preprocessed using the 1000 Functional Connectomes Project pipeline scripts (reference) which do 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/UtahBrainNetworks/base_prep and includes specific instructions for installation, compilation, and usage.

After running fMRI data through the pipeline "rest.nii", the preprocessed fMRI file, was opened in Matlab using the "load_nii('rest.nii')" function within the NiFTI toolbox. The 4D "img" variable (x, y, z, t) is then resized to a 2D variable (x*y*z,t) and saved to use in SCIRun. (reference for NiFTI toolbox?)

2.2.5 EEG

The filetype of the EEG recordings in an .edf file. The EEG signal matrix was obtained using a Matlab script called "edfRead.m." To run this script use the following command "[hdr, record] = edfread(fname)." The variable 'record' will contain the signals. The last two rows of the matrix were removed because they did not correspond to EEG electrodes. Also the beginning and end of the experiment were cut out of the matrix when the EEG net is being put on and taken off.

2.2.6 Registration

Since the subject did not move in between the T_1 and T_2 MRI, no registration was necessary before segmentation and meshing. The tetrahedral mesh was generated in its own coordinate space from the segmentation, and was registered to the DTI coordinate space with a rigid registration using SCIRun. (figure of registration network – explain more?) The fMRI data was registered to the mesh coordinate space with a rigid registration using SCIRun. (figure registration network – explain more?) The fMRI data can use same transform to register to DTI later if desired.

2.3 MRI Segmentation of Tissues

Segmentation of the head tissues proved to be the most time consuming section of the pipeline. The head volume was segmented into air, cerebral spinal fluid (CSF), white matter, grey matter, skull, sinus, eyes, and scalp. Segmentation of the brain can be difficult due to the similar intensities of the different tissues, making merely applying a median filter and thresholding the image not enough. (show a figure of this happening?)

The brain was initially segmented by inputting a skull stripped T_1 MRI into FSL FAST Segmentation. This outputs CSF, white matter, and grey matter layers as well as a bias-corrected T_1 MRI. This method, compared with Freesurfer, Statistical Parametric Mapping through Matlab (SPM), Atlas Based Classification through 3D Slicer, and Seg3D methods alone, produced the best initial brain segmentation results for this data due to how well it filled in each tissue. (??) (three panel figure of FSL FAST results)

Although the FSL Fast results were a great improvement compared to the other segmentation trials, manual segmentation still needed to be done on those layers to add more detail and take out any cross over between the layers. Since white matter is the innermost layer, it was worked on first. All manual editing was done using Seg3D software. (ref) First a threshold layer was created from the FSL Fast output. Every slice in every direction was inspected and manually edited whether that was adding more detail that could be seen with the naked eye or cleaning up noise from FSL Fast. This manual editing of the white matter took roughly 40 hours of work. (Figure of before and after editing – hook detail!)

After the white matter was completed, a threshold layer for grey matter was created from the FSL Fast output. Each slice in every direction of the grey matter was inspected as well. The white matter layer was removed from the grey matter using a boolean remove mask filter. Any holes between the two layers were decided manually. More detail was added to the grey matter folds to add to the CSF layer as well. The last part of editing the grey matter was decided to add a grey matter nucleus to the layer. The thresholding algorithms generated a lot of noise around these nuclei so they were segmented by hand, using the paintbrush tool, and added to the grey matter layer with a boolean or mask filter. The nuclei were also removed from the white matter layer using a boolean remove mask filter. The manual editing of the grey matter took roughly 20 hours of work. (figures of grey matter and nuclei)

After the grey and white matter layers were completed, the CSF layer was made by creating a solid threshold layer for the entire brain and removing the white and grey matter layers using a boolean remove mask filter. The white matter, grey matter, and CSF layers were then checked for holes, whether on the surface or the inside of the segmentation. Also a quality check on the layers was performed to ensure that the layers were at least two pixels wide. This is an important note for creating a hole-less tetrahedral mesh. The creation of the CSF layer and the manual editing

and hole checking took roughly 4 hours of work. (??) (Figures of CSF? Figures of all three brain layers together?)

The skull and the sinus layers are the most difficult to segment using only an MRI because they both appear black in the image, and the volunteer did not have a CT scan. (define CT-scan) The first attempt to create a bone layer was first to use FSL's skull stripping using the BET2 tool to create a skull. Then to threshold the remainder of the bones in Seg3D from the T_1 MRI and connect it to the skull from FSL. (figure and compare the two skull layers) Although this gave a decent skull for only having an MRI layer, the method to segment sinus layer was still to be determined. As a second method, the skull was estimated from an MR-based synthetic pseudo-CT. An improved iterative version of the patch-based method was used described by Torrado-Carvajal et al. [ref] that takes the T_1 and T_2 images as input, and synthesize the pseudo-CT based on both images providing more refined and accurate bone boundaries. MR input images were preprocessed to correct for MRI bias inhomogeneities (N4 bias correction tool in the Insight Toolkit)(ref?) prior to computing the pseudo-CT. (figure of the pseudo-CT output) This method gave a good starting place for skull segmentation, but still needed manual editing. After using a median filter (pixel radius?) and thresholding, each slice in each direction was manually edited by hand. (figure of segmentation) Since the volunteer has a permanent retainer in their mouth, the mouth was segmented as solid bone for now. This is not concerning because the EEG cap used did not cover the volunteer's mouth. (picture? explain EEG caps) The pseduo-CT image also provided a segmentation of the sinuses and esophagus by thresholding the black pixels. After the thresholding, the sinus layer was also manually edited. Quality checks were done on both layers to ensure that there were no holes and that the layers were at least two pixels thick. (figure) Manually editing the second skull and the sinuses took roughly 30-40 hours of work.

The eyes, skin, and air layers were simple, in comparison, to segment. The eyes were easily segmented by thresholding the T_2 MRI. (figure of T2 axially) The skin layer was segmented by thresholding the entire volume and removing all of the previous layers using a boolean remove mask filter. A quality check was performed on the skin layer to ensure that it was at least two pixels thick. The important places to check for this is at the bridge of the nose and the bottom of the chin. (sagittal figure) Last, the air was segmented by thresholding the entire image and removing the solid skin layer. (how much time?) There was a check to ensure that the segmentation did not contain any holes between layers after they were removed. (should I explain this procedure?) To create these three layers took roughly 8 hours of work, most of which was ensuring there were no holes in the segmentation. This is imperative to creating a quality mesh.

(Should I make a table instead for the amount of work?)

2.4 Finite Element Mesh Generation

The tetrahedral finite element mesh was generated using Cleaver software (ref and figure) on a Late 2013 Mac Pro with a 2.7 Ghz 12 Core Intel Xeon E5 processor, 64 GB of RAM, and an AMD FirePro graphics card. To make a very high resolution mesh with no holes the following parameters were used: scaling factor: 0.6, size multiplier: 1.0, lipschitz: 0.2, padding: 0, element sizing method: apdative. This produced a mesh with 60.2 million elements and 10.3 million nodes. (figure[s]) This mesh was so large due to the complexity of the segmentation. To reduce the size of the mesh, a mesh was made with a scaling factor of 1.0 with the remainder of the parameters as described before. The computing sizing field was exported from Cleaver and manipulated using SCIRun4 (ref and figure of network) by changing how quickly the elements increase in size. This

was done by multiplying the scaling by 27. The changed sizing field was then input into Cleaver and cleaved a new mesh. This produced a mesh with 15.7 million elements and 2.7 million nodes with no holes. However, this mesh does contain one flat tetrahedra. It is later removed in a SCIRun network, and is currently being investigated by Cleaver software developers.

2.5 Mathematical Modeling

2.5.1 Conductivity Preparation

A *typical* bioelectric volume conductor can be posed as the following boundary value problem:

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

where Φ is the electrostatic potential, σ is the electrical conductivity tensor, and I_V is the current per unit volume defined within the solution domain, Ω . The associated boundary conditions depend on what type of problem one wishes to solve. There are generally considered to be two different types of direct and inverse volume conductor problems.

One type of problem deals with the interplay between the description of the bioelectric volume source currents and the resulting volume currents and volume and surface voltages. Here, the problem statement would be to solve (1) for Φ with a known description of I_V and the Neumann boundary condition:

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

which says that the normal component of the electric field is zero on the surface interfacing with air (here denoted by Γ_T). This problem can be used to solve two well known problems in medicine, the direct EEG (electroencephalography) and ECG (electrocardiography) volume conductor problems. In the direct EEG problem, one usually discretizes the brain and surrounding tissue and skull. One then assumes a description of the bioelectric current source within the brain (this usually takes the form of dipoles or multipoles) and calculates the field within the brain and on the surface of the scalp.

All conductivities were homogeneous with the exception of white matter when using tensor data. These conductivities are as follows: ADD HERE (ref). When DTI tensor data is added, there are two approaches to converting the tensor data to conductivities. The first is scaling the data, and the second is giving the white matter a fixed ratio of conductivity.

SCALING MATH (ref)

RATIO MATH (ref)

2.6 Numerical Methods

Finite element math

END OF ALLY'S TEXT

2.6.1 Experimental Methods

We induced episodes of acute, transient ischemia in anesthetized, open-chest swine ($n = ??$) and canine ($n = ??$) preparations. In each experiment, the heart was exposed and suspended in a pericardial cradle. A portion of the LAD was minimally dissected and fitted with a hydraulic occluder, which could be compressed to restrict coronary blood flow, thereby creating the transient ischemic

condition, and then released to restore normal cardiac perfusion. Experimental protocols consisted of measuring extracellular potentials, both on the epicardial surface and within the myocardium, while applying stepwise increases of ischemic load.⁷ To this end, two basic protocols were used to induce ischemia: 1) LAD blood flow was incrementally reduced by the occluder while maintaining a constant, often elevated, heart rate (supply ischemia) or 2) increases in the heart rate were applied while maintaining a constant, often restricted, LAD blood flow (demand ischemia). For purposes of this study, we do not consider the differences between supply and demand ischemia but rather focus on the size, shape, and location of the zones within the heart that show an electrical response to the ischemic condition. All studies were performed in accordance with the Guide for the Care and Use of Laboratory Animals (NIH Pub. No 85-23, Revised 1996).

Customized sock and needle electrodes were used to acquire electrical recordings of both epicardial and intramural electrical potentials. A 247-electrode, flexible sock array,⁷ with recording electrodes evenly distributed across the ventricles, acquired epicardial electrogram recordings. Twenty-five (25) flexible fiberglass plunge needles⁷ were used to record intramural electrical activity. Plunge needles were constructed with 10 evenly spaced electrodes at 1.6 mm intervals along the shaft. Needles were placed in and around the perfusion bed of the occluded LAD. Sock and needle recordings were made periodically at a 1KHz sampling rate, the combination of which provided a *3-dimensional* electrical representation of the induced ischemic condition.

In addition to the acquisition of electrical recordings, digitized locations of sock electrodes and needle locations were extracted for postexperiment validation studies. After digitization, needle electrodes were removed and replaced with radio-opaque spacers prior to cardiac imaging in order to provide a registration reference.

Electrograms were calibrated, gain adjusted, and baseline corrected against control recordings, which had been taken immediately before each intervention. Poor quality electrograms, caused by broken lead connections or bad contacts, as well as electrograms without positive Q-wave deflections (identified as cavity electrodes) were discarded. The global root mean squared signal was computed from data recordings of both sock and needle electrodes. These signals were used to identify a point that lay at 40% of the distance between the J point and T wave peak (ST40%). Potential difference maps were generated at ST40%, which compared baseline recordings to those obtained during an ischemic episode. Potential differences, taken at ST40% during baseline conditions, were used to generate a threshold by which ischemic regions were identified as values exceeding two standard deviations. Sock recordings were used to validate simulation findings and will be addressed later. Needle electrodes were used to generate subject-specific ischemic zone geometries by identifying regions within a spatial neighborhood that met the above-mentioned ischemic thresholding criteria.⁷

2.7 Simulation Pipeline

A simulation pipeline was implemented in order to produce in silico models that were representative of experimental findings. Electrical data acquired during the experimental process, as well as imaging information acquired post experiment, were used to construct subject-specific, finite element, bioelectric simulations by way of the steps illustrated in Figure 2.

2.7.1 Imaging and Segmentation

After completion of the experimental protocol, each heart was excised and scanned with a 7 tesla MRI scanner using FISP and FLASH MRI sequences. Diffusion tensor images (DTI) were also acquired to determine fiber direction. FISP scans rendered consistent, albeit low, contrast throughout

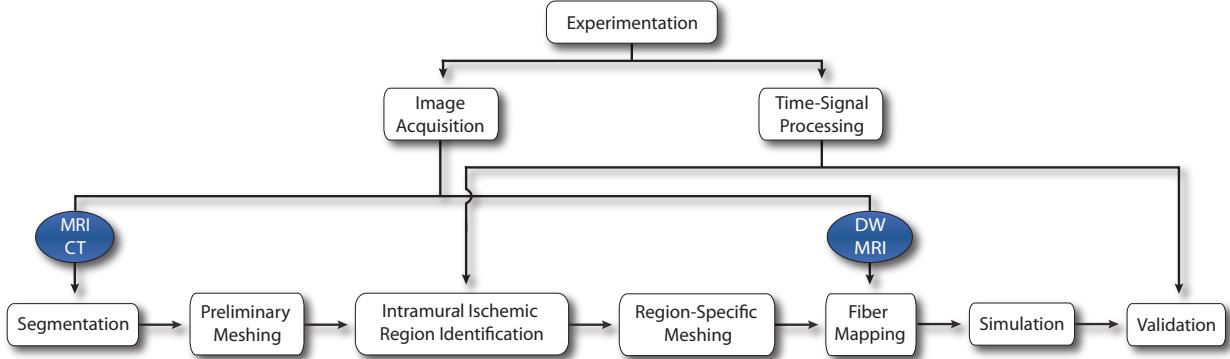


Figure 2: Ischemia Simulation Pipeline. Both image and time signals were extracted from experimental protocols of acute, induced ischemic preparations. Image data was used to generate geometries through segmentation and meshing. Intramural electrical data, recorded from plunge needles, were mapped to these meshes to define ischemic zone location. New meshes were generated with ischemic geometries imposed on which ischemic simulations were generated and validated against the original experimental data recorded from epicardial sock data.

the tissue - preserving edges near the field of view boundaries. FLASH scans, in comparison, provided images of high contrast within the center of the volume, which diminished steeply near the field of view boundaries. The advantages of both FISP and FLASH were combined in order to produce realistic, geometric segmentations of blood, cardiac tissue, and needles, using the Seg3D* open-source software package. Needle locations were readily identified within the scans as dark regions occupied by the radio-opaque spacers that were inserted at the end of the experimental phase. Diffusion-weighted MRI (DW-MRI) images were also obtained from which fiber orientation was derived.

2.7.2 Geometric Processing and Data Mapping

Segmentations were used to generate realistic *3-dimensional* geometries for use in subsequent finite element simulations (See Section 2.7.4). By using two open-source meshing packages, we were able to generate smooth, linear, subject-specific, boundary-conforming, tetrahedral meshes for use in simulations. First, segmentations were ported into the BioMesh3D* software package to generate smoothed/tightened surface representations. BioMesh3D surface representations were used as input indicator functions to a second meshing package, Cleaver*, from which the final mesh was derived. Cleaver is a multimaterial meshing package that produces structured meshes of tetrahedral elements with guaranteed minimum element angles,⁷ resulting in quality meshes that require fewer computational resources. Cleaver, however, does not offer the surface-tightening features of BioMesh3D. As a result, meshes produced by Cleaver from standard MRI segmentations (with no surface tightening) would propagate the stair-stepped surfaces inherent in rasterized, volumetric data. By combining packages, we were able to produce smooth, structured meshes of guaranteed element quality.

Correspondence points derived from known sock and needle locations were used to register

*<http://sci.utah.edu/software.html>

needle and sock electrode geometries within the cardiac mesh using the SCIRun problem-solving environment [†]. Processed data values were mapped to corresponding node locations within the cardiac mesh. Linear interpolation to local nodes was applied but restricted to the geometric convex hull of the needle locations. Extrapolation to outlying regions, not within the scope of needle locations, was not included.

To identify ischemic zones, potential difference maps were generated that removed baseline potentials from those observed during ischemic interventions. Difference map values within the specified needle region that exceeded a value greater than two standard deviations from baseline recordings were identified and labeled as ischemic. These new label masks were used to create a final mesh that contained three defined tissues (blood, healthy cardiac tissue, and ischemic cardiac tissue) that shared conforming surfaces - an important feature in static forward simulations when considering areas in proximity to potential sources.[?]

Finally, subject-specific fiber orientation was applied within the mesh. A vector field was defined by the principle eigenvector from DW-MRI images with all other cross-sectional fiber components regarded as isotropic. This vector field was normalized, aligned, and mapped to the cardiac mesh using weighted-average interpolation to provide a basis for anisotropic conductivity.

2.7.3 Mathematical Modeling

The cardiac mesh, with associated fiber structure and ischemic region, was used to solve the bidomain passive current flow equation:

$$\nabla \cdot (\bar{\sigma}_e + \bar{\sigma}_i) \nabla \phi_e = -\nabla \cdot \bar{\sigma}_i \nabla V_m \quad (3)$$

where σ_e and σ_i represent the extracellular and intracellular conductivity tensors, respectively. ϕ_e is the extracellular potentials, and V_m represents the transmembrane potential.

In this model, it was assumed that the heart was surrounded by a perfect insulator, leading to a Neumann boundary condition on the epicardial surface. The endocardium, in contrast, allowed for extracellular, but not intracellular, current flow into the ventricular blood pool. Initial blood potentials were defined by Cauchy boundary conditions along the endocardial surface as shown in Equation 4.

$$\begin{cases} \nabla \cdot (\bar{\sigma}_e + \bar{\sigma}_i) \nabla \phi_e = -\nabla \cdot \bar{\sigma}_i V_m & x \in \Omega_H \\ \vec{n}_{epi} \cdot (\bar{\sigma}_e + \bar{\sigma}_i) \nabla \phi_e = 0 & x \in \partial\Omega_{H,epi} \\ \vec{n}_{endo} \cdot (\bar{\sigma}_e \nabla \phi_e) = -\vec{n}_b \cdot (\bar{\sigma}_b \nabla \phi_b) & x \in \partial\Omega_{H,endo} \\ \phi_e = \phi_b & x \in \partial\Omega_{H,endo} \\ \phi_i = 0 & x \in \Omega_b \end{cases} \quad (4)$$

where Ω_H represents the cardiac volume; $\partial\Omega_{H,epi}$ and $\partial\Omega_{H,endo}$ the epicardial and endocardial surfaces, respectively; ϕ_e and ϕ_b correspond to potentials in the extracellular space and the blood, respectively. \vec{n} represents the normal, outward unit vector along the epicardial (*epi*), endocardial (*endo*), and blood (*b*) surfaces. The first equation specifies the relationship between transmembrane and extracellular potentials within the cardiac domain. The following four equations define the handling of currents and potentials along the cardiac surfaces. The blood, in all studies, was completely enclosed within the cardiac region and considered to have isotropic conductivity (See Figure 3 and Table 1).

Conductivity values within the tissue, as well as the blood pool, were normalized with respect to extracellular longitudinal values and matched those used in previous studies.^{?, ?, ?} Table 1 shows

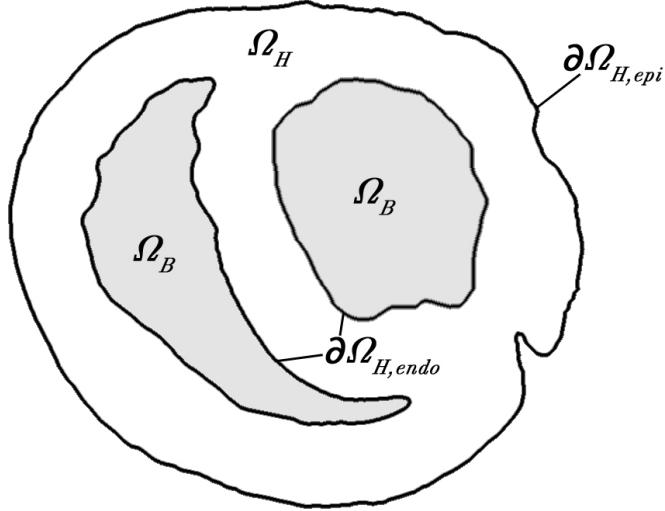


Figure 3: Bidomain equations are defined within the cardiac tissue Ω_H and bounded by epicardial and endocardial boundaries $\Omega_{H,epi}$ and $\Omega_{H,endo}$. Extracellular currents were allowed to flow into the blood volume Ω_B along the endocardial boundary.

Table 1: Ratio applied to tensor conductivity values within healthy and ischemic regions.[?]

Conductivity Labels	Healthy Conductivity Values	Ischemic Conductivity Values
σ_{el}	1	1/2
σ_{il}	1	1
σ_{et}	1/3	1/4
σ_{it}	1/20	1/20
σ_b	3	3

the conductivity values used for this study. These conductivity values were chosen to be consistent with those previously defined by Johnston and Kilpatrick.[?] Conductivities within the ischemic region were reduced with respect to healthy values corresponding to the first 5 - 15 minutes after ischemic onset.

We defined a fixed transmembrane potential value (V_m) of 30mV as the potential source for our forward simulations. The reduced transmembrane potential mimics the delayed activation of diseased tissue under ischemic conditions during the ST-segment. Healthy tissue was assigned a value of 0 mV, typical of the relatively quiescent state of healthy cardiac activation during the ST segment. Between healthy and ischemic tissue, a border zone was defined in which potentials progressed from the diseased to healthy states.

The border zone, the region between healthy and ischemic tissue, has been determined to be a necessary, and problematic, region to include in simulation studies.^{?, ?, ?} In this study, we defined the border zone as a piecewise, continuous function represented by Equation 5 in which a Gaussian function defines the region from the ischemic zone boundary to a specified transition point, S_1 . At

S_1 the border transitions to a linear, decreasing function that reduces to 0 (the value assigned to healthy tissue) at S_2 . It is important to note that not all models, found in the literature, have applied border zones. For the sake of consistency and uniformity, however, we have imposed border regions into all of the forward simulations constructed in this paper. Table 2 shows the values applied to Equation 5 for this study.

$$BZ = \begin{cases} V_m e^{\frac{-d^2}{2\sigma^2}} & d < S_1 \\ \Gamma - \frac{\Gamma}{S_2 - S_1}(d - S_1) & S_1 \leq d < S_2 \\ 0 & d \geq S_2 \end{cases} \quad (5)$$

$$\text{where } \Gamma = V_m e^{\frac{-(S_1)^2}{2\sigma^2}}$$

Table 2: Values Used In Border Zone Determination.

Label	Definition	Value
V_m	Transmembrane Potential	30 mV
σ	Gaussian RMS Width	5 mm
S_1	First Transition Distance	8 mm
S_2	Second Transition Distance	11 mm
d	<i>distance</i>	<i>variable</i>

2.7.4 Numerical Methods

Solutions to Equation 3 were computed using finite element methods. By applying Green's divergence theorem to Equation 3, the following weak formulation is generated

$$\int ((\bar{\sigma}_e + \bar{\sigma}_i) \nabla \phi_e) \cdot \nabla \psi(\bar{x}) d\bar{x} = - \int (\bar{\sigma}_i \nabla V_m) \cdot \nabla \psi(\bar{x}) d\bar{x}, \quad \forall \psi \in \Omega \quad (6)$$

where, Ω (see Section Section 2.7.2) is the linear, finite element mesh, ψ represents the finite element basis functions characterized by local hat functions associated with mesh nodes. By applying this formulation to the finite dimensional mesh, we can reduce Equation 6 to a system of linear equations

$$A\phi_e = -RV_m \quad (7)$$

where A and R represent stiffness matrices defined by $A_{j,k} = \langle \nabla \psi_j, (\bar{\sigma}_e + \bar{\sigma}_i) \nabla \psi_k \rangle_\Omega$ and $R_{j,k} = \langle \nabla \psi_j, \bar{\sigma}_i \nabla V_m \rangle_\Omega$, while ϕ_e and V_m represent extracellular and transmembrane potentials, respectively.⁷

We used the open-source, SCIRun problem solving environment⁷ to apply parameters and to solve Equation 7 numerically. Within the SCIRun environment, fiber orientation and conductivity tensors were applied to the mesh, initial and boundary conditions were defined, and border regions were generated in order to compute extracellular potentials by way of a conjugate gradient method with a Jacobi preconditioner.

3 Results

3.1 Segmentation

3.2 Finite Element Meshes

3.3 Forward Problem

3.3.1 Isotropic

3.3.2 Anisotropic

3.4 fMRI Visualization

3.5 EEG Visualization

4 Conclusion

In this paper we have outlined a comprehensive pipeline to build an inhomogeneous, anisotropic head and brain model based on human data of all image modalities for use in electroencephalography with an emphasis in forward and inverse problem research as well as visualizations of function MRI data and EEG data. Along with the pipeline, the human data will be released as open-source to enable other scientists to have a starting point to continue further research.

Further investigations from building this pipeline include finding better decimation algorithms for 3D tetrahedral finite element meshes. Due to the number of materials included in the mesh, current decimation algorithms have not been able to further simplify the mesh. More inclusive methods for functional MRI data into a source localization head model. More specific processing of EEG data for different applications which will make for better visualization.

5 Acknowledgements

This work could not have been completed without the help of Jess Tate and Brett Burton. This project was supported by the National Institute of General Medical Sciences of the National Institutes of Health under grant number P41 GM103545-18.