NFT & NIST

Neuroelectromagnetic Forward and Inverse Head Modeling Toolbox

Zeynep AKALIN ACAR

EEGLAB Workshop, San Diego

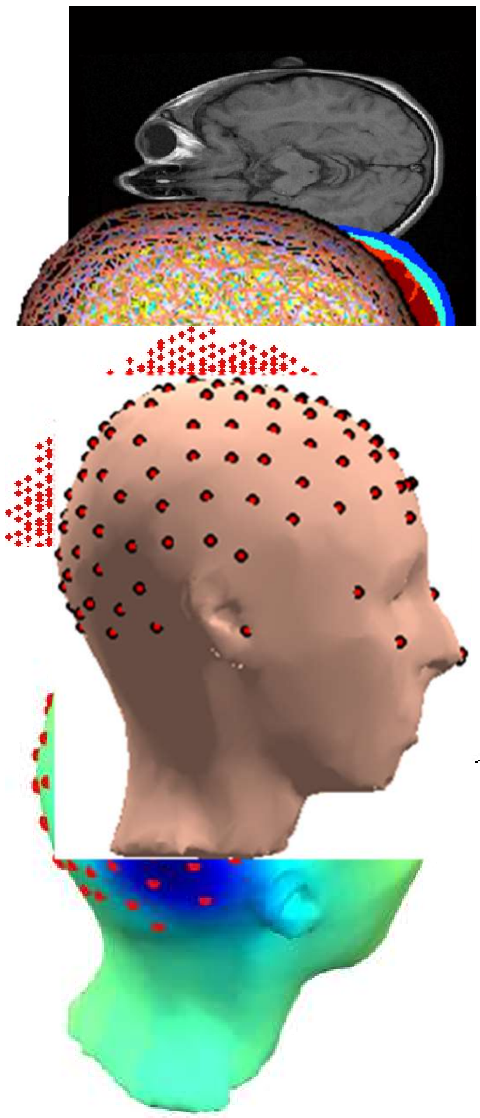
November, 2018

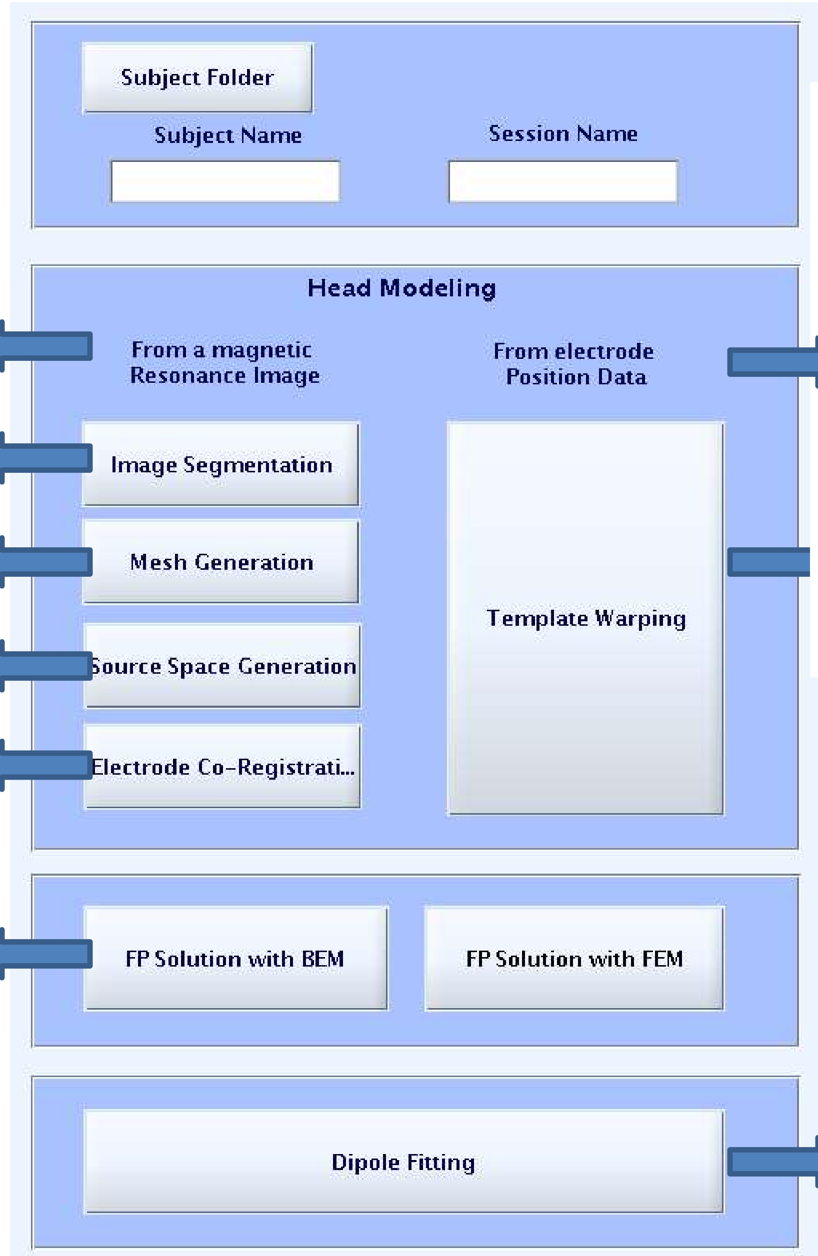
NFT

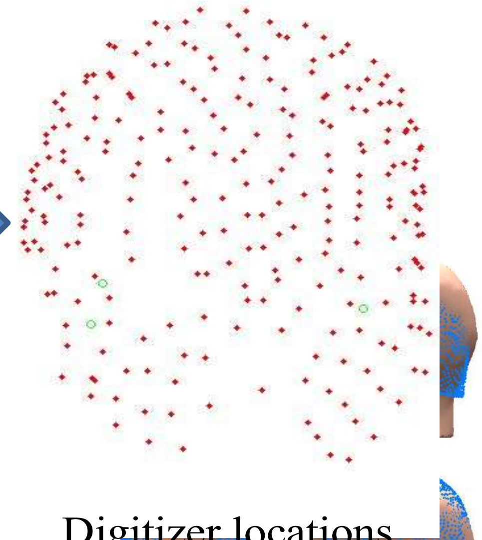
- A complete framework for accurate forward problem solution.
- Easy-to-use MATLAB environment with GUI and command-line functions.
- Ability to use available subject information
  - T1-weighted 3D MR images
  - Digitized sensor (electrode) locations

Comparison with Dipfit

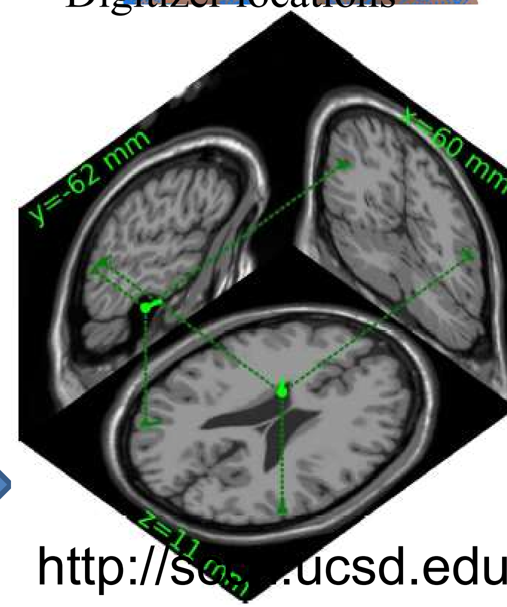
- The realistic model in Dipfit is a three-layer MNI head model represented with 3000 vertices.
  - The forward matrices are pre-calculated, so there is no need for FP calculations.
- NFT generates subject-specific models.
  - NFT does model generation and forward problem calculations.
  - More accurate.





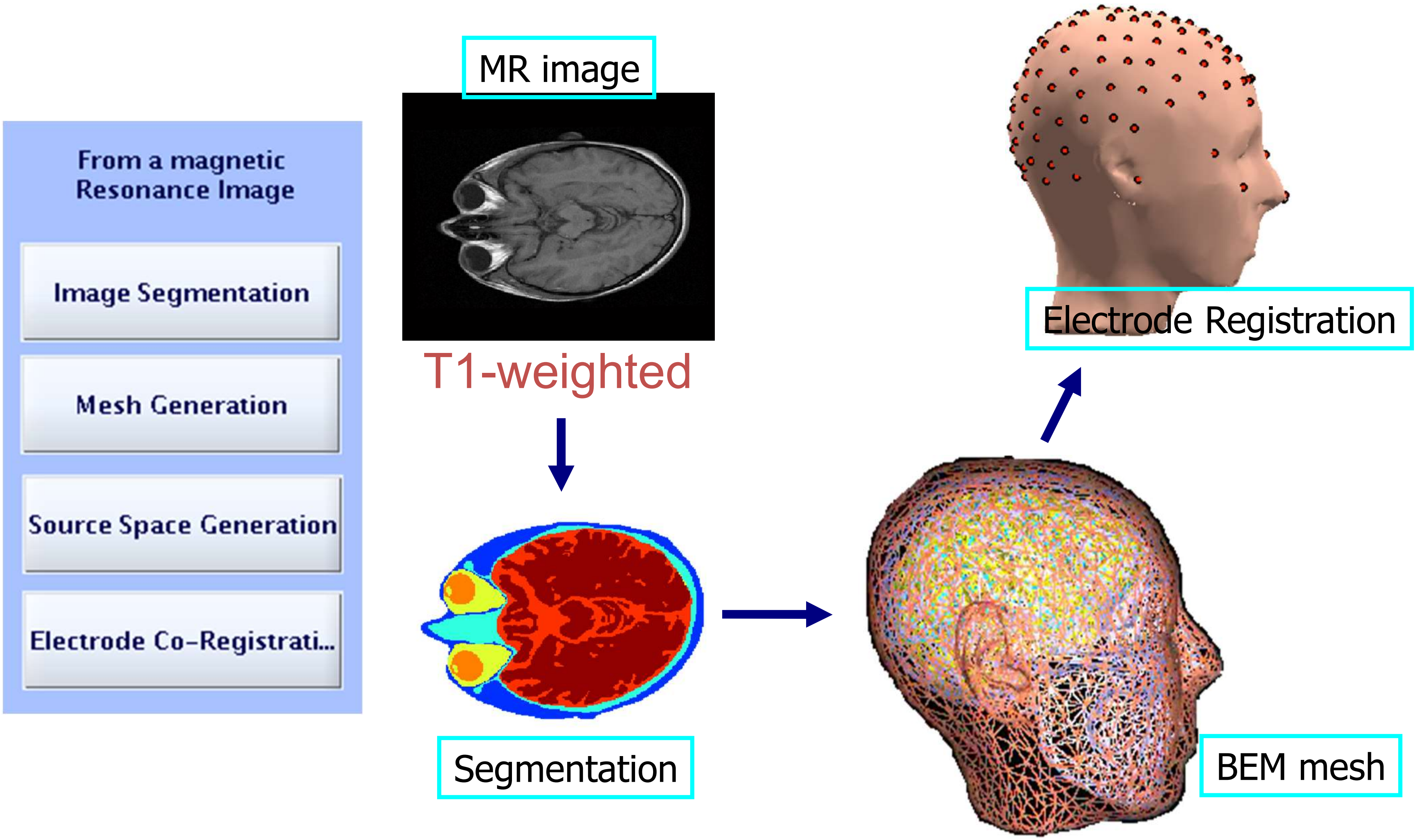


Digitizer locations



http://sc...ucsd.edu/nft

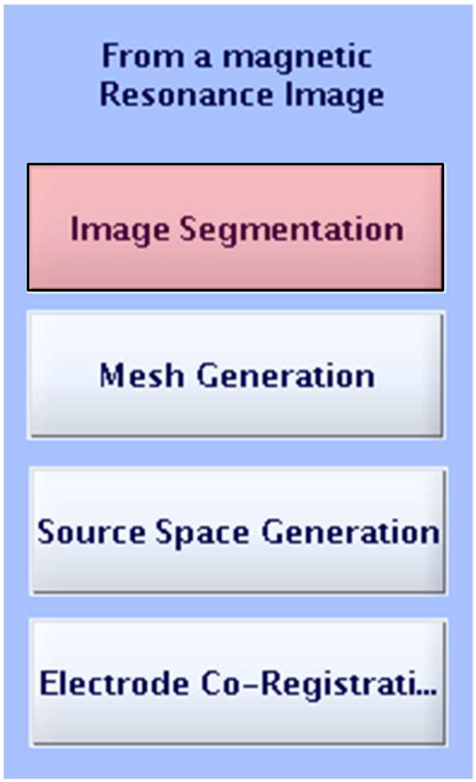
Head modeling from MR images

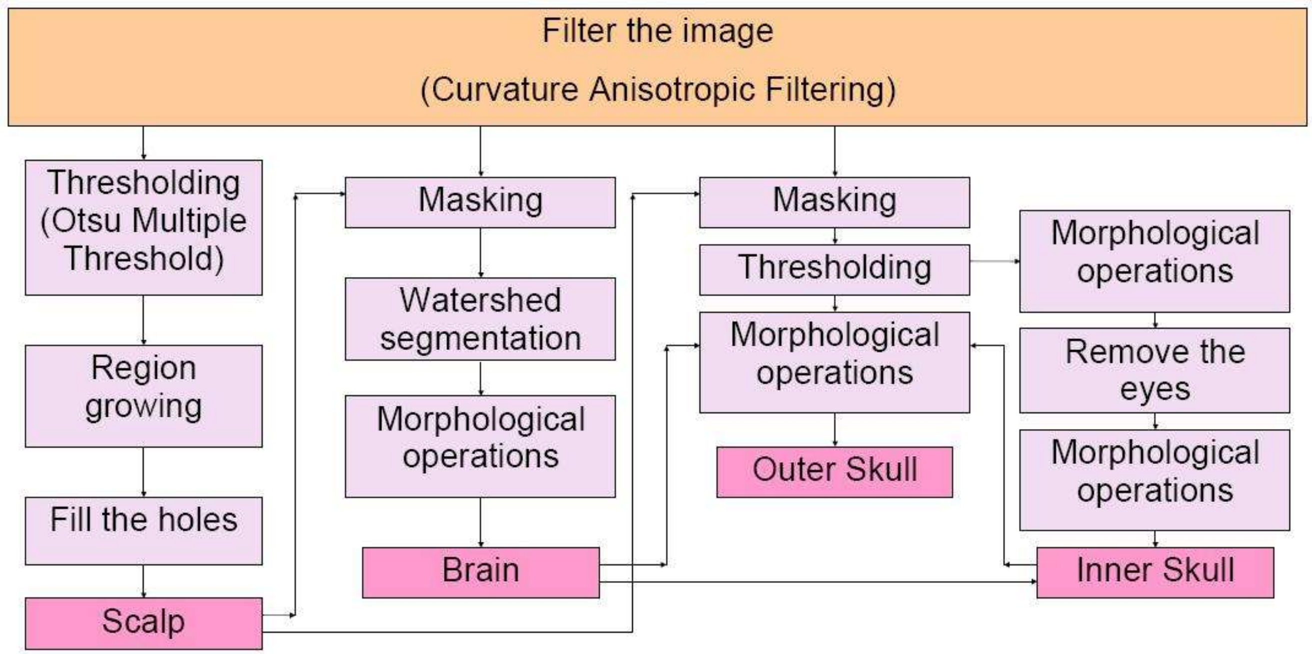


Preparing the MR Image

- Using FreeSurfer
  - Inhomogeneity correction
  - Convert to 1x1x1 volume
  - Arrange direction of the image
  - Save in analyze format

Image Segmentation





Classifies four tissues from T1-weighted images
Scalp, Skull, CSF and Brain

Starting NFT

- To start from EEGLAB

  EEGLAB -> Tools -> NFT

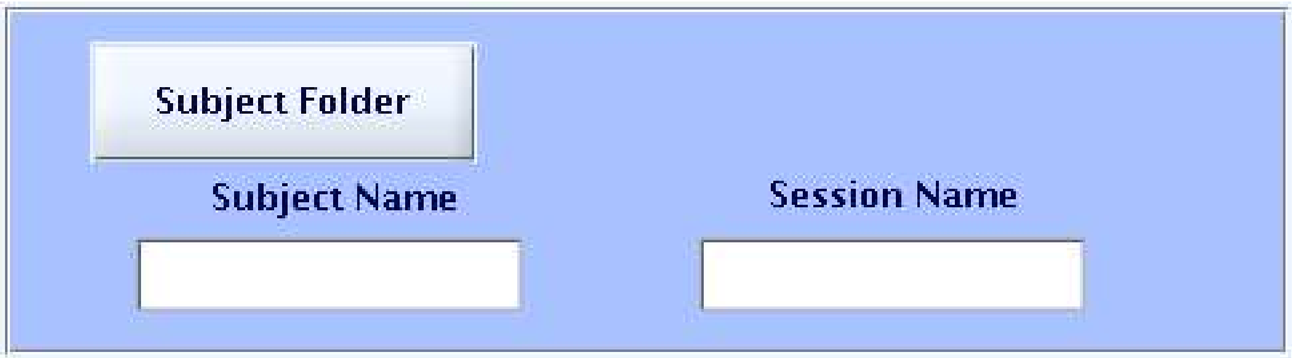
- To start as a standalone toolbox

  addpath NFT directory

  Type 'NFT' in Matlab

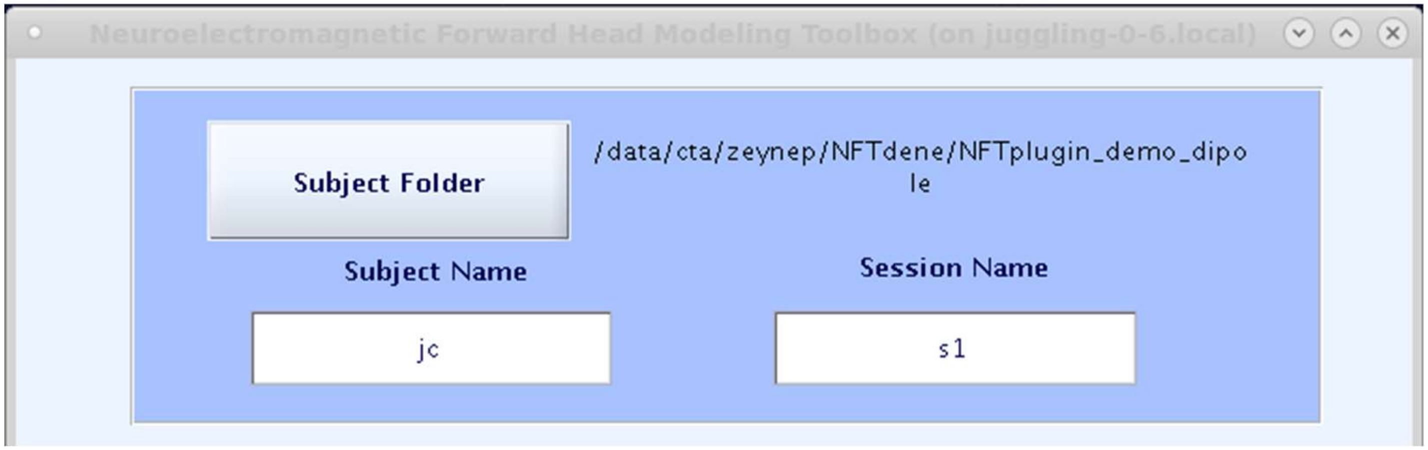
For demo: go to NFT-2.4_demo folder

Subject Selection



- Select subject folder
- Specify subject name
- Specify session name

Subject Selection



Select current folder as subject folder
Enter “jc” as subject name
Enter “s1” as the session name

Image Segmentation

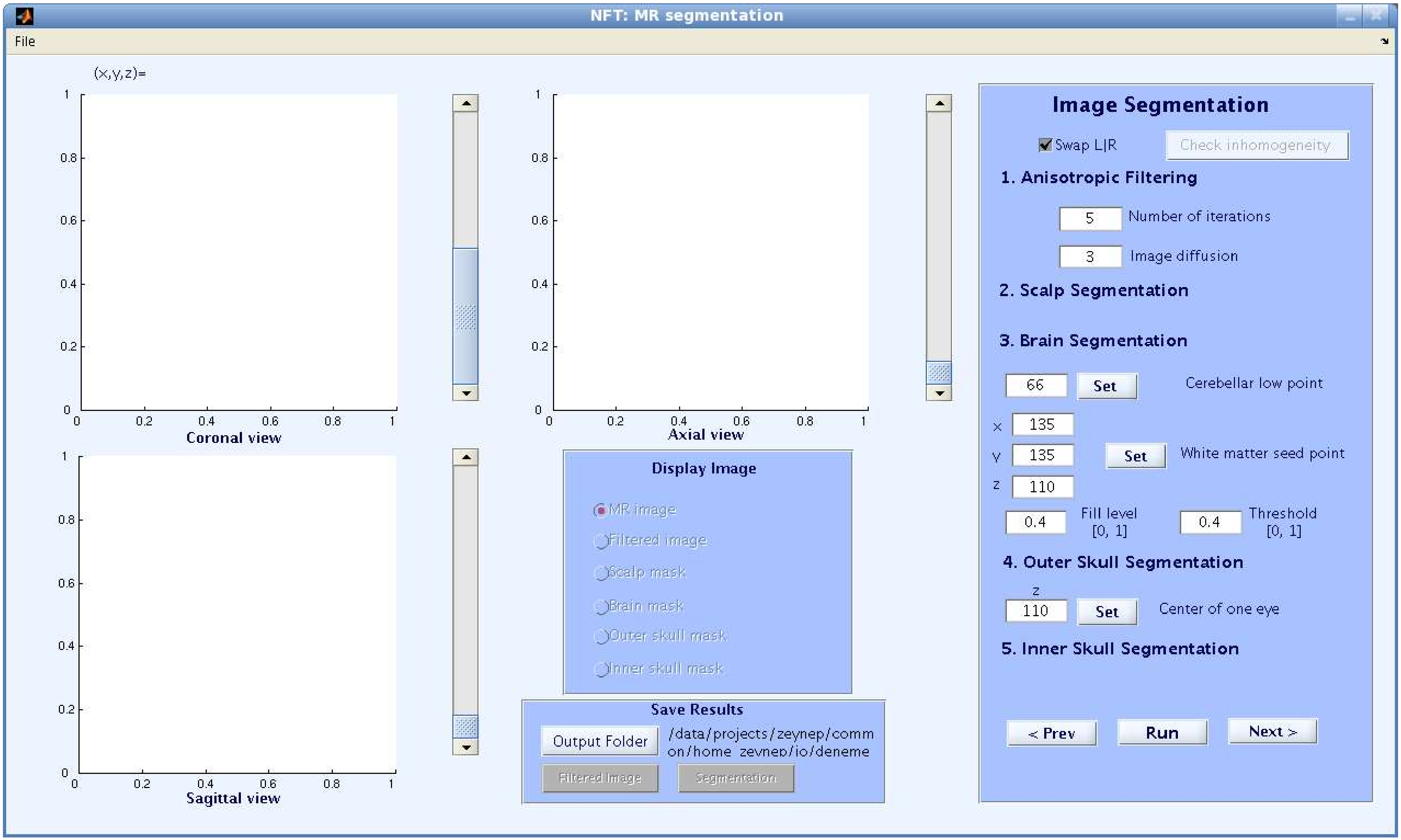
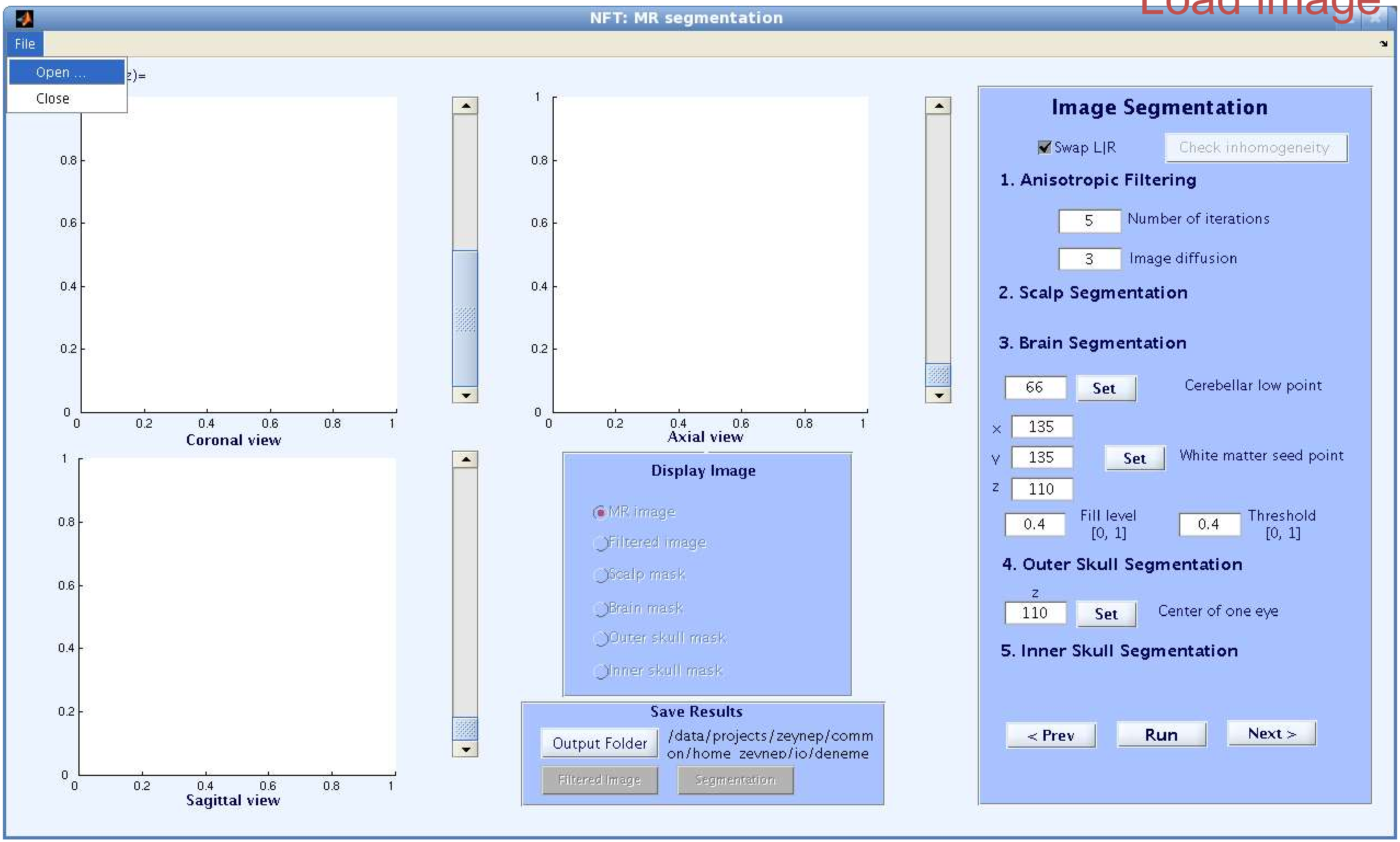


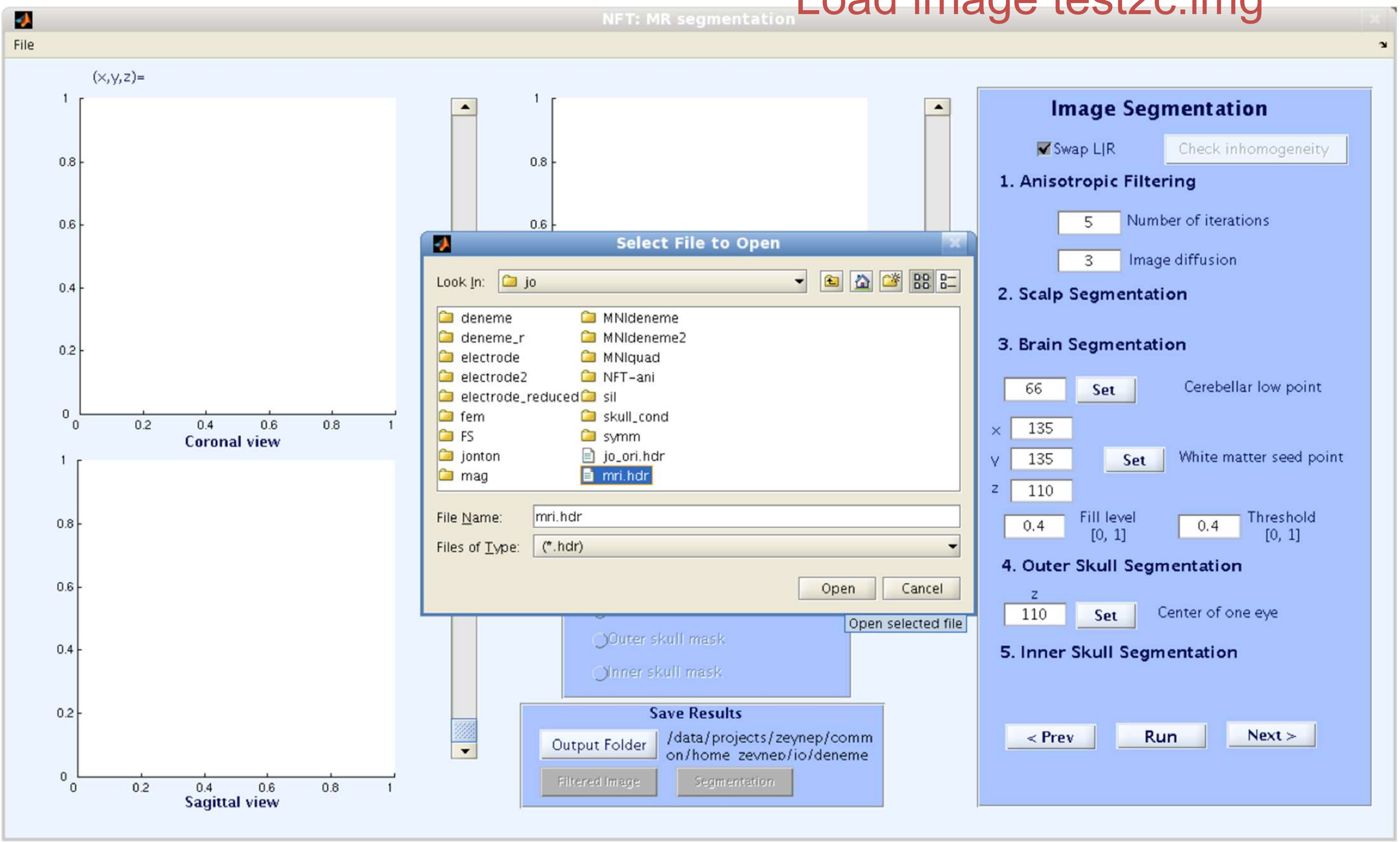
Image Segmentation

Load image



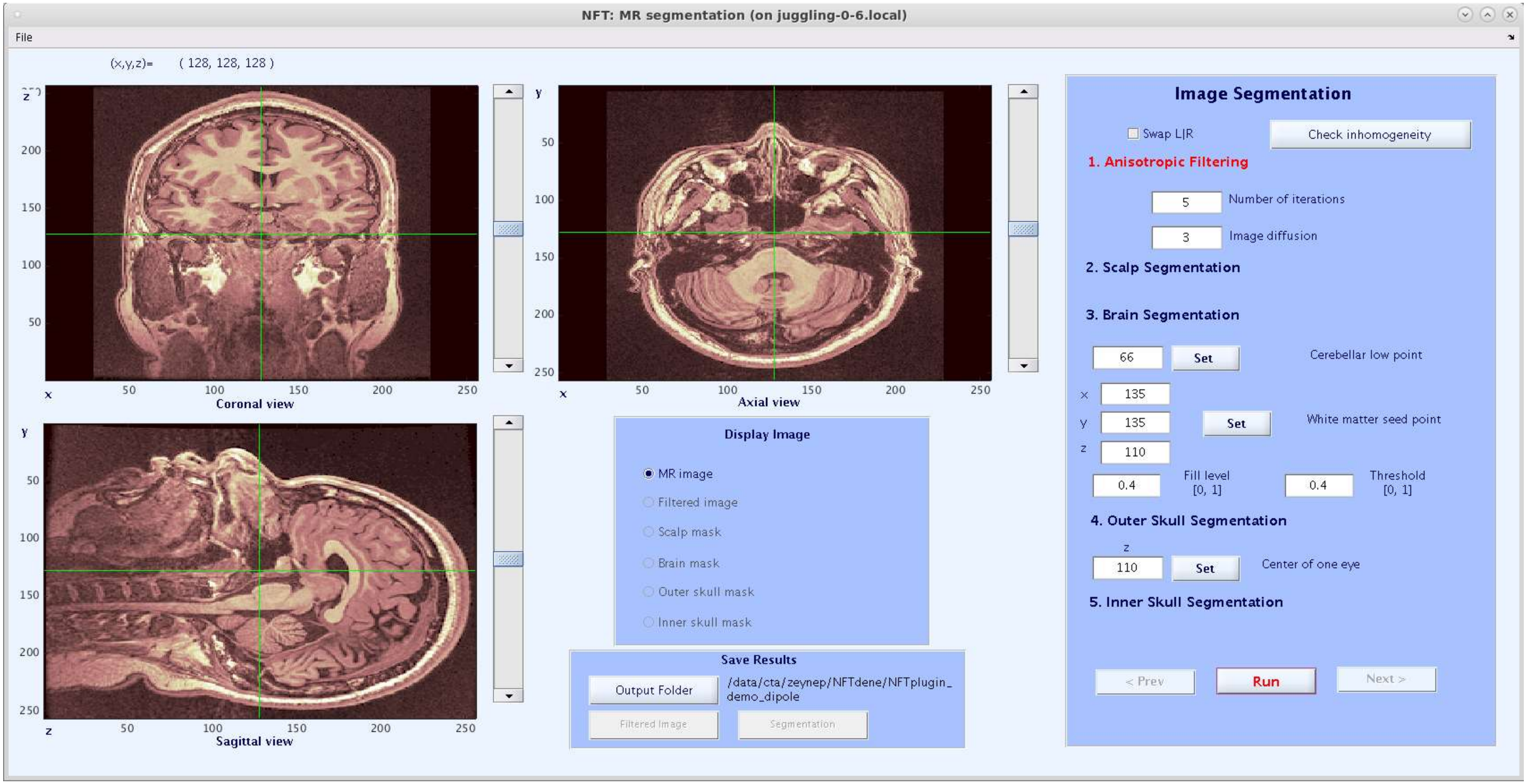
Segmentation

Load image test2c.img

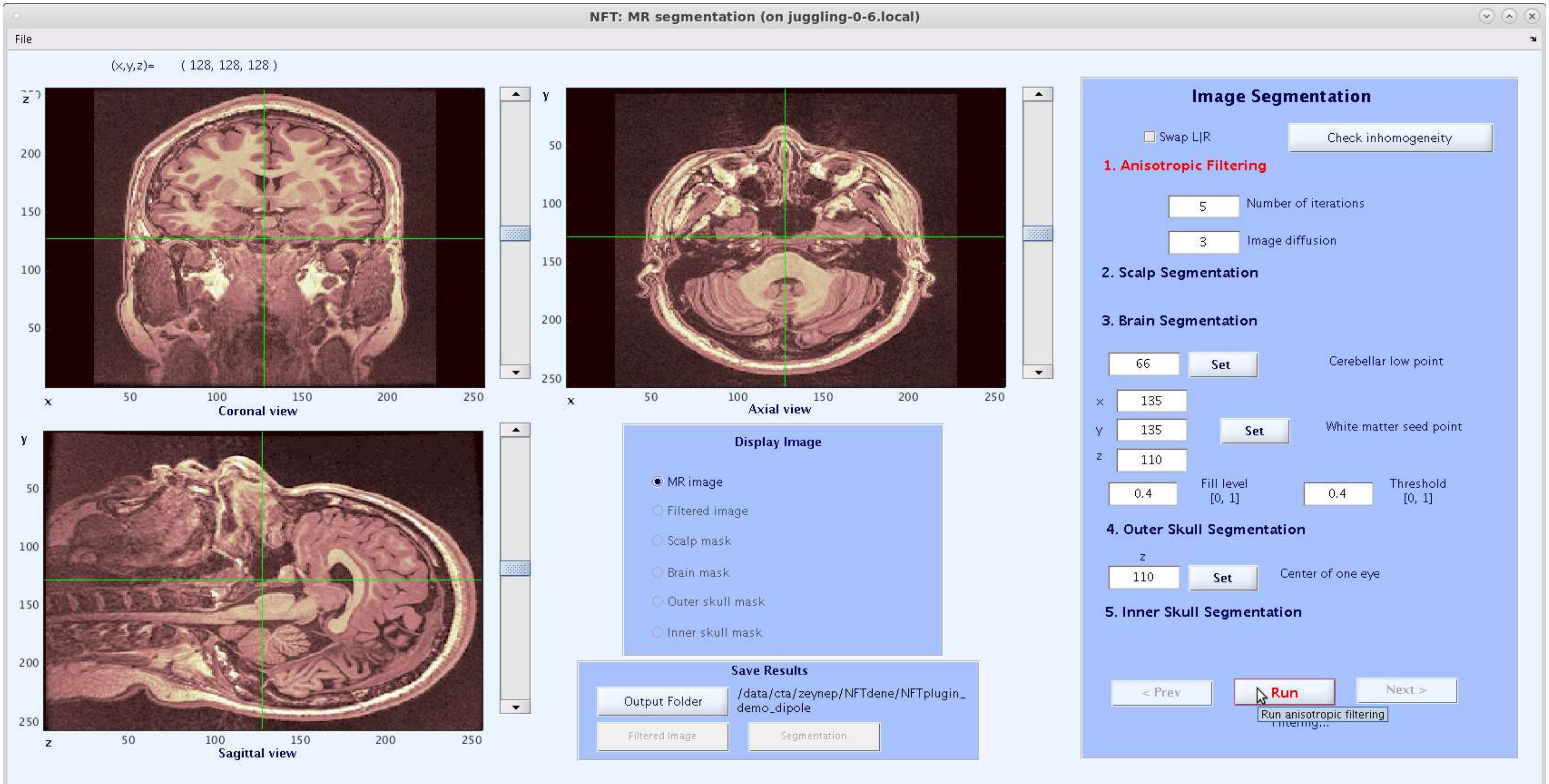


Segmentation

Run filtering

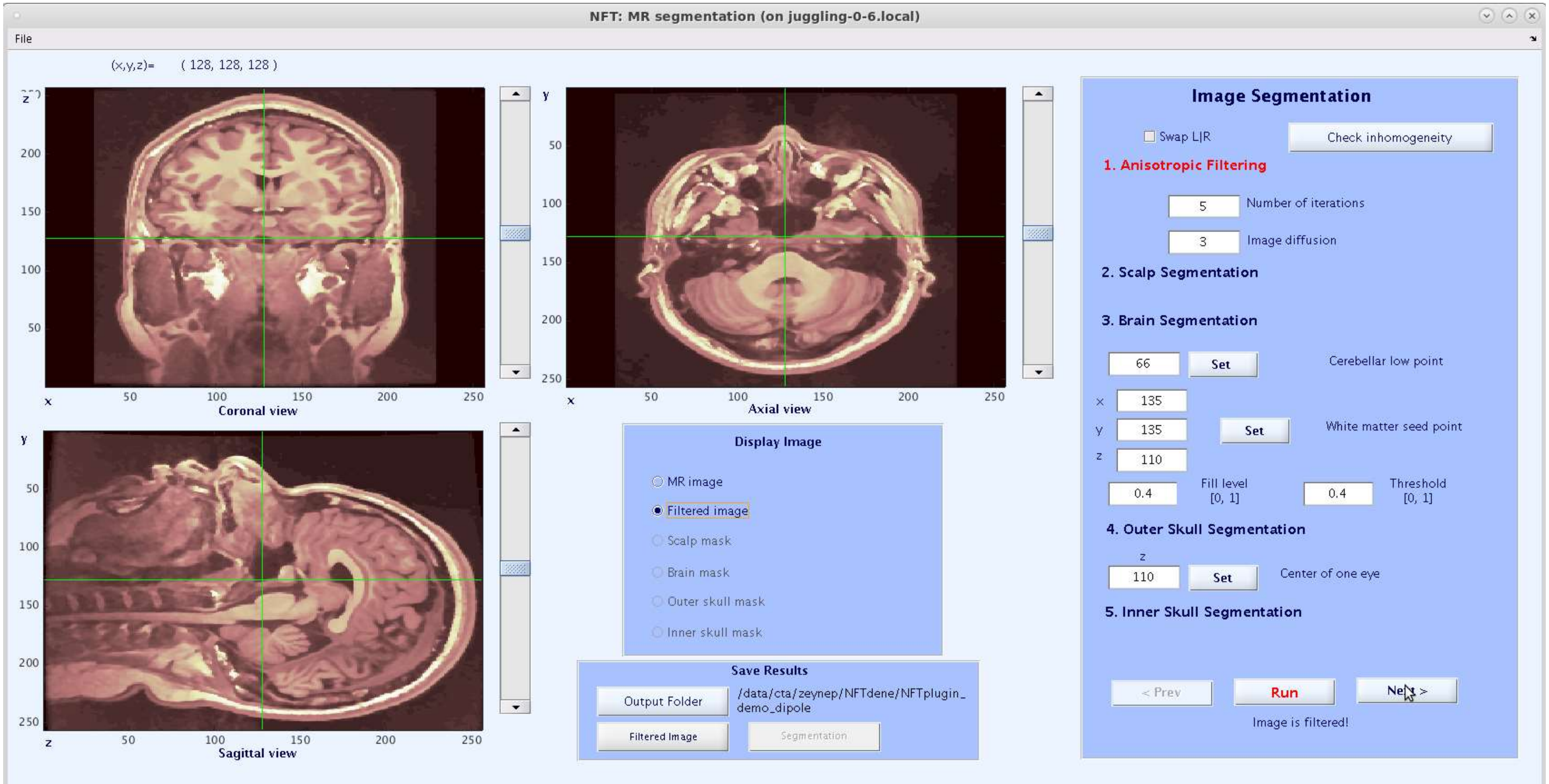


Segmentation



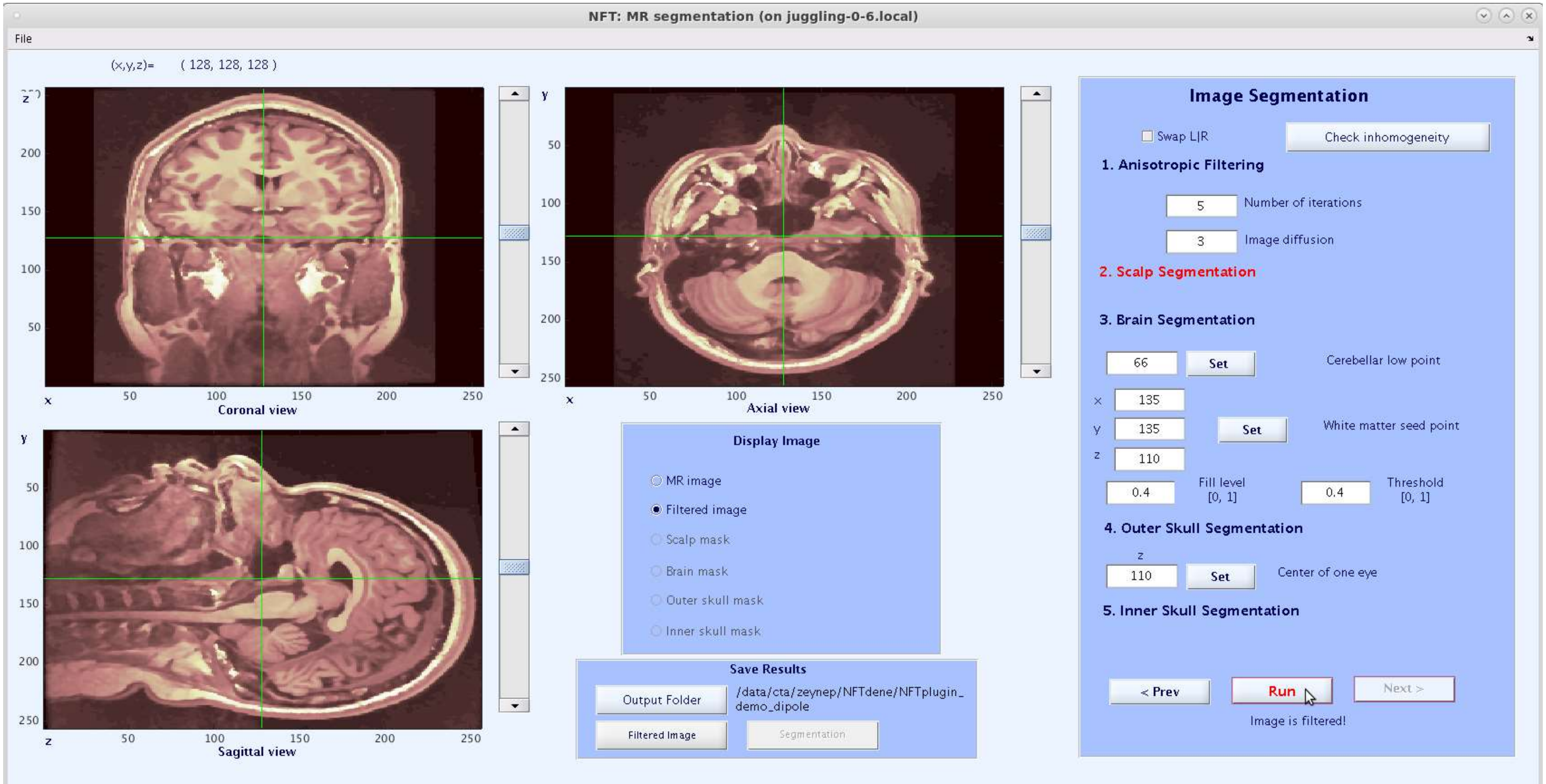
Segmentation

View filtered image



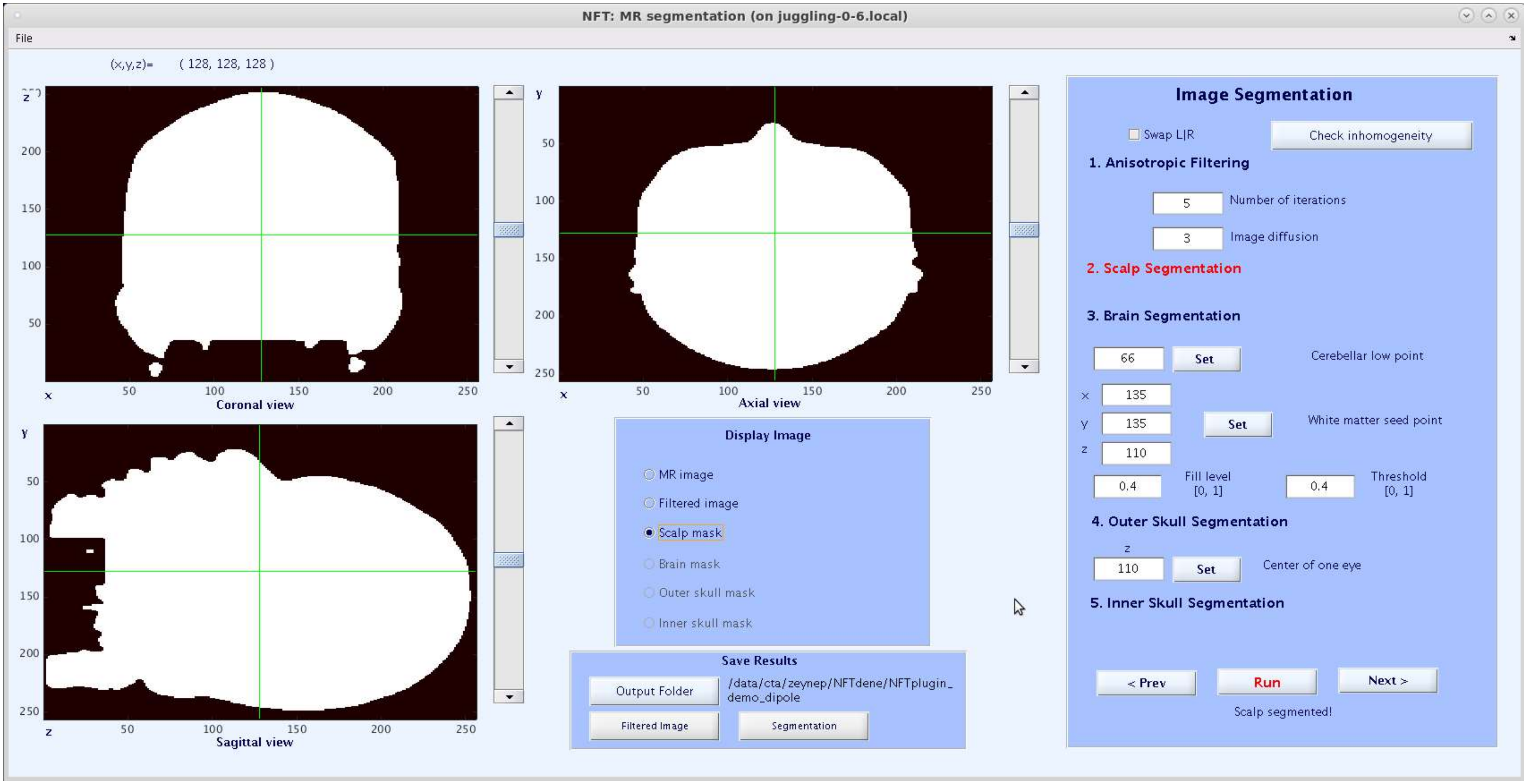
Segmentation

Click ‘Next’ for scalp segmentation and run scalp segmentation



Segmentation

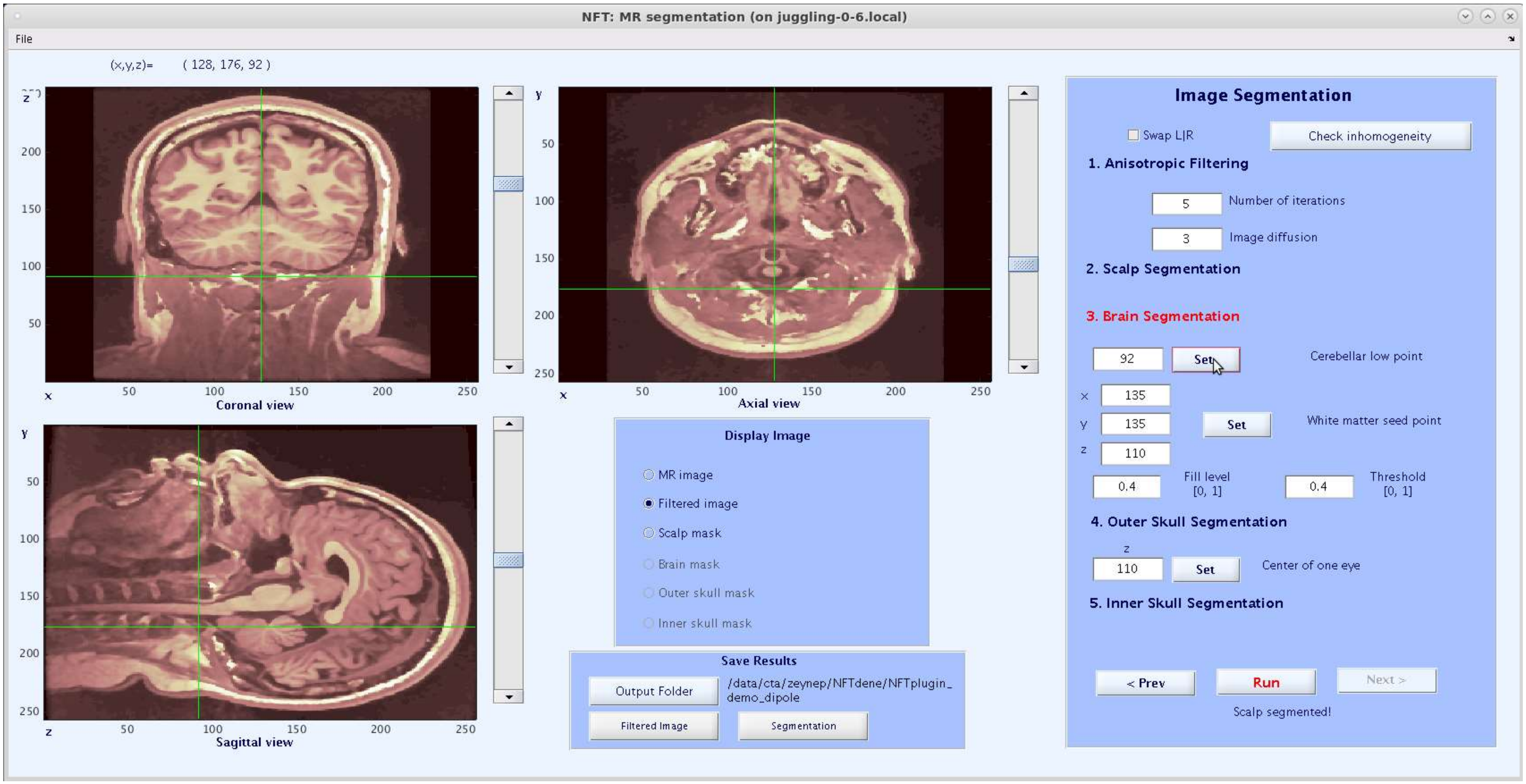
View scalp mask



Segmentation

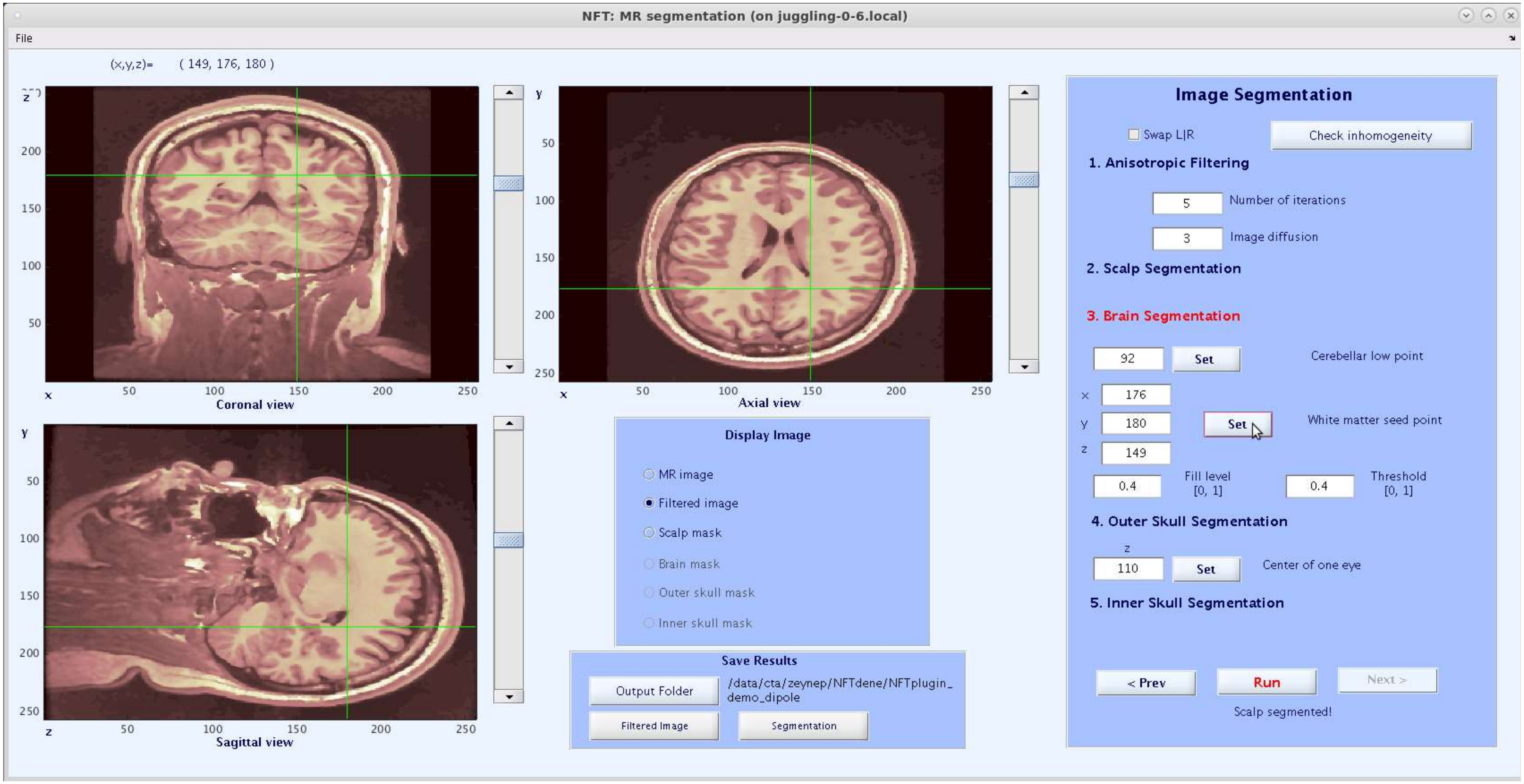
Click ‘Next’ for brain segmentation

Selection of cerebellar low point



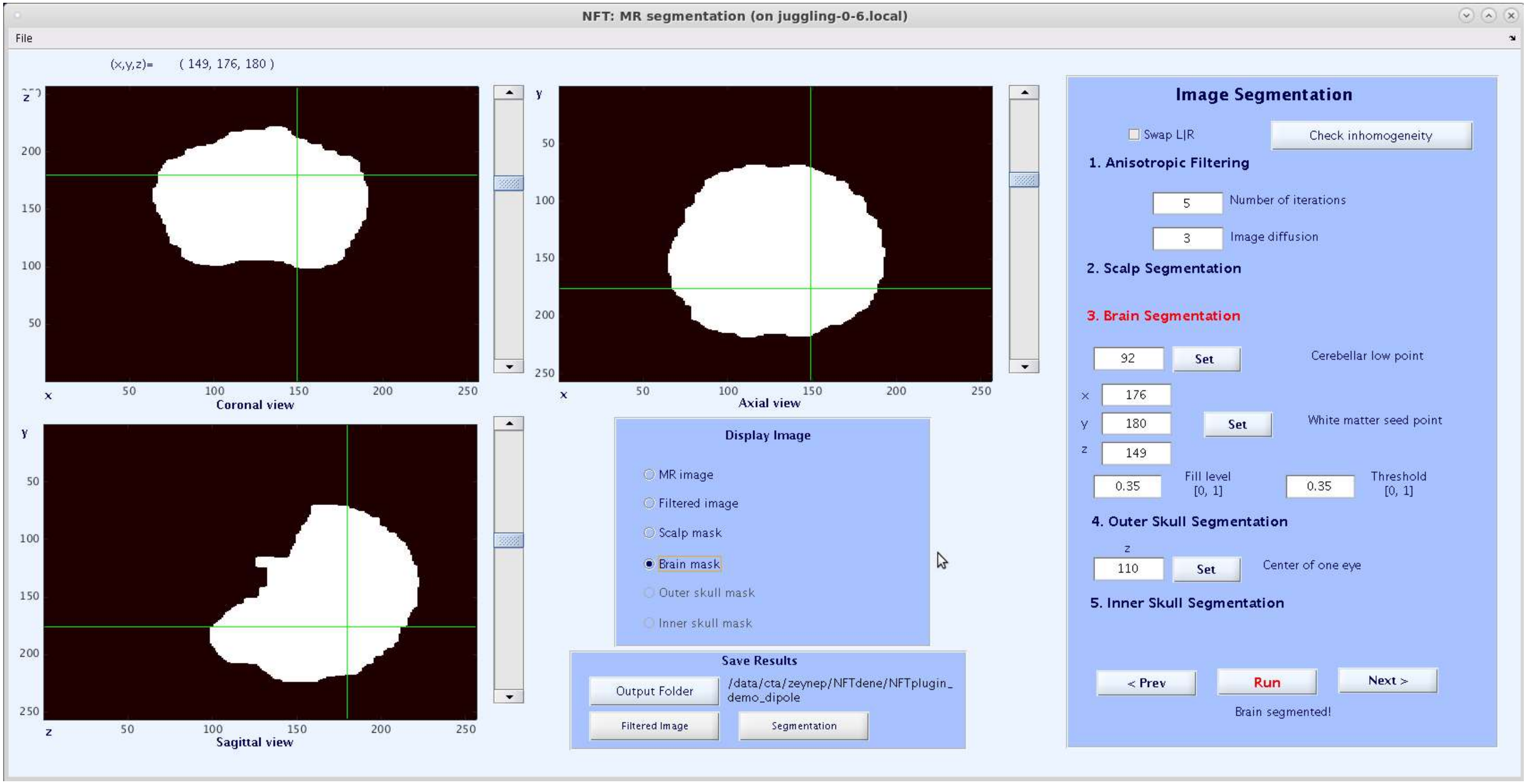
Segmentation

Selection of a white matter point



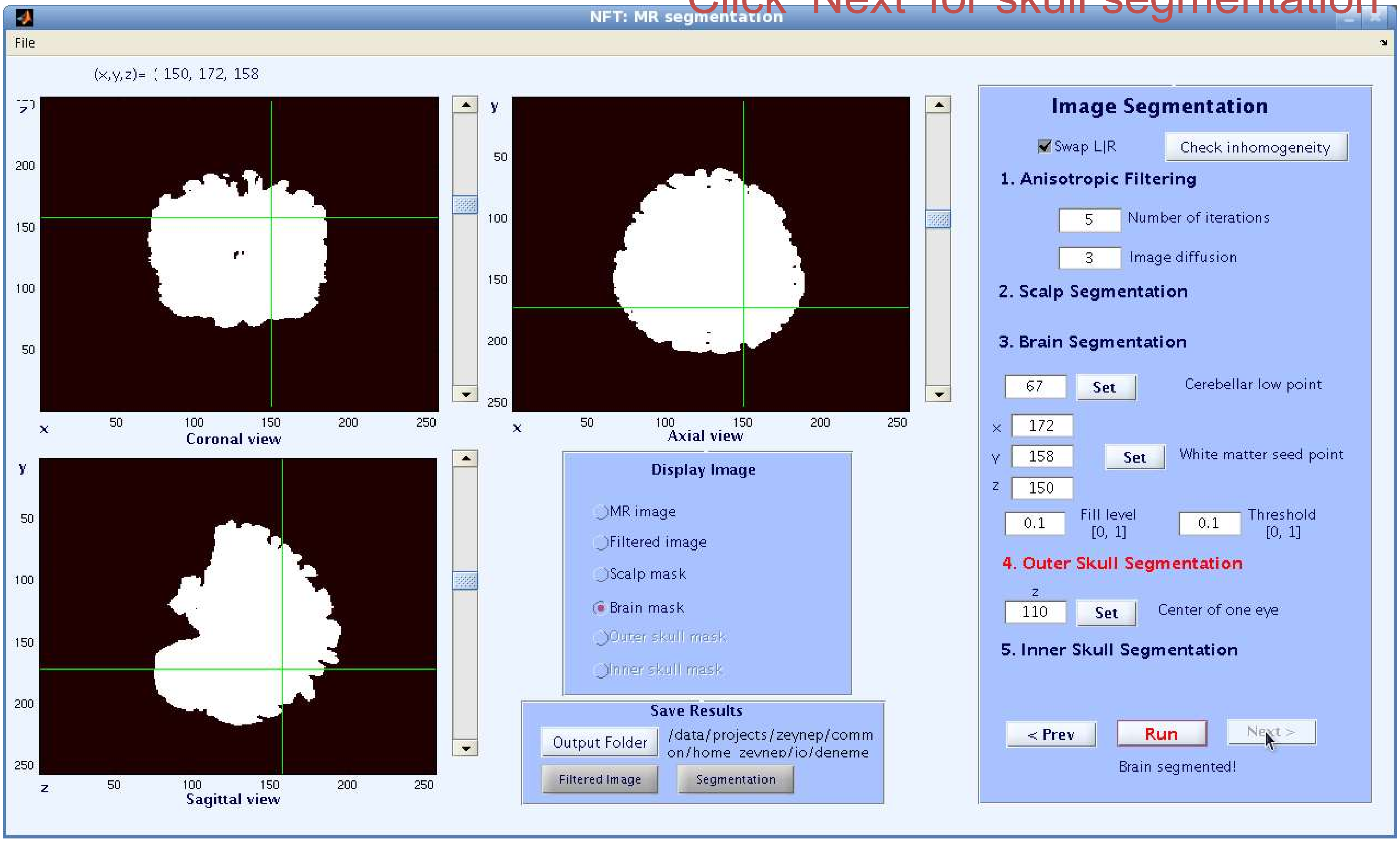
Segmentation

View brain mask



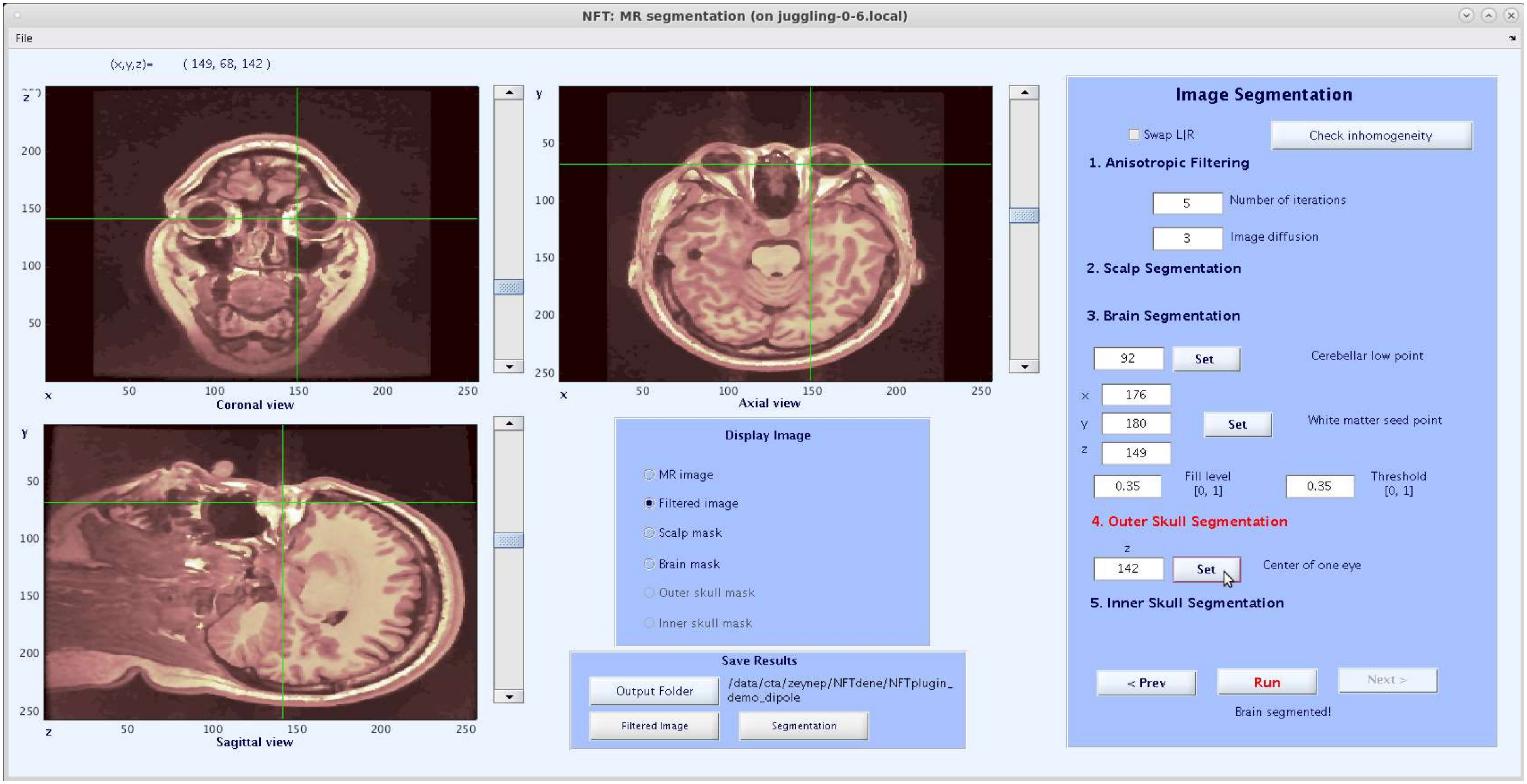
Segmentation

Click 'Next' for skull segmentation



Segmentation

Select a slice for eyes and click 'set'

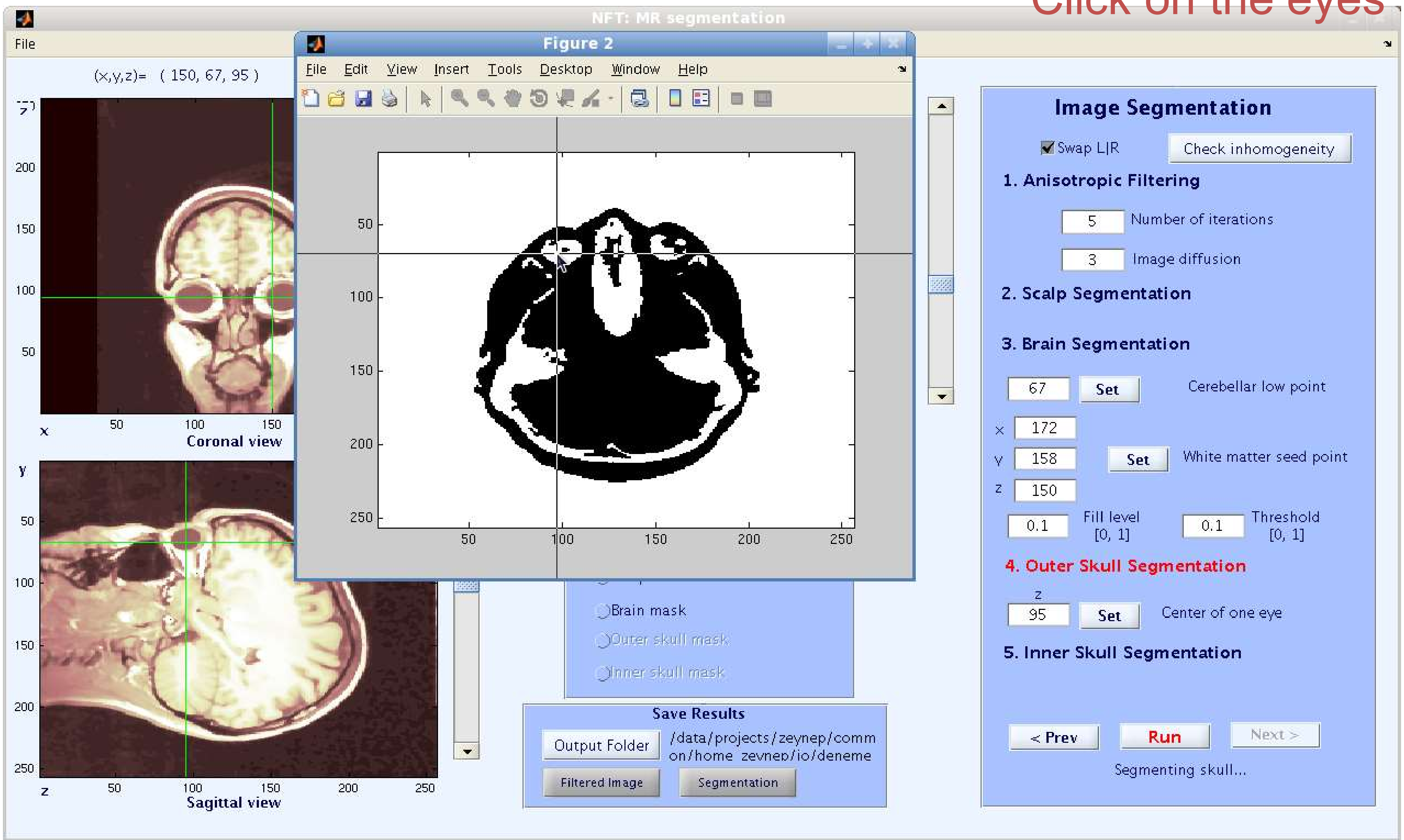


Segmentation

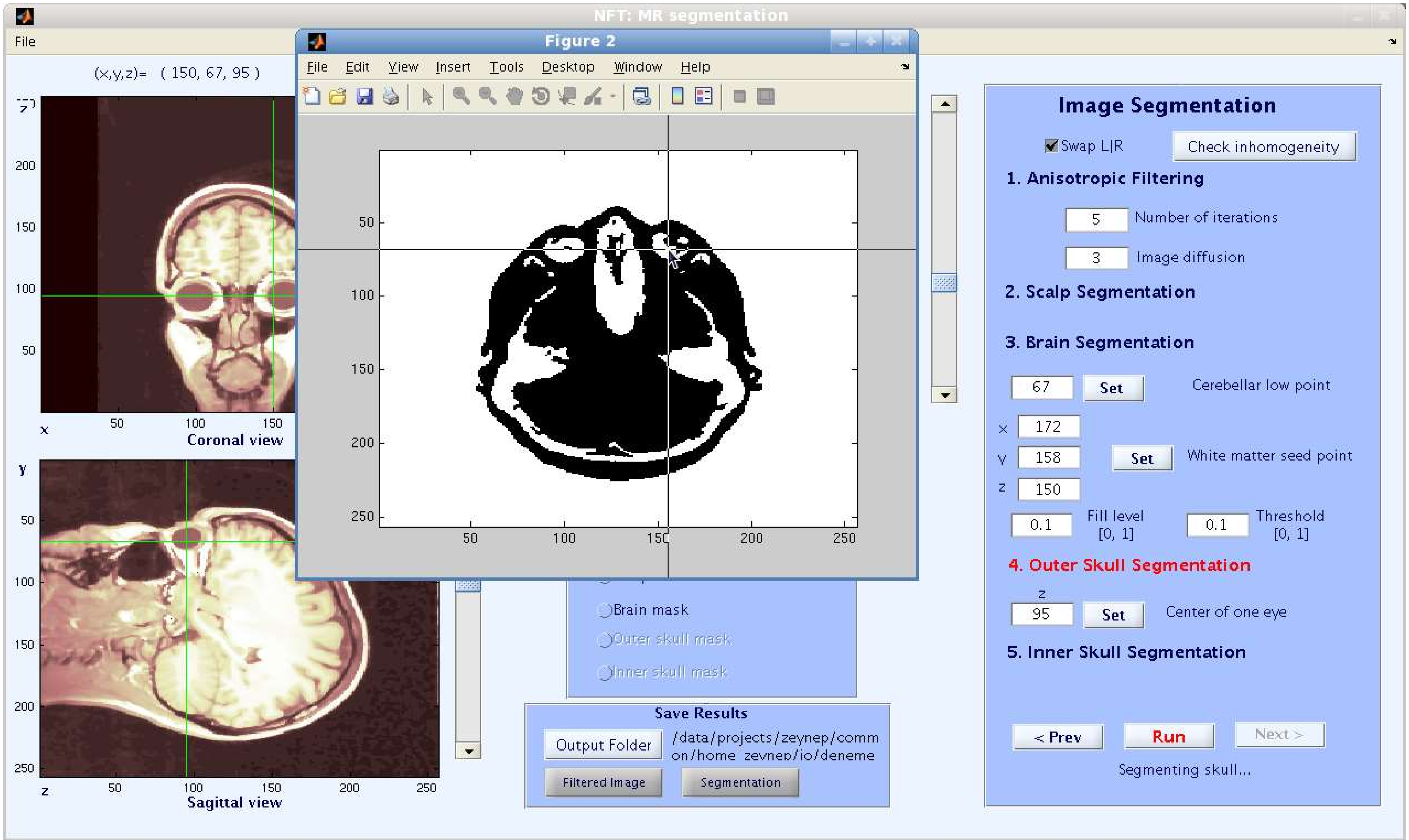
Click ‘Run’ for skull segmentation

Segmentation

Click on the eyes

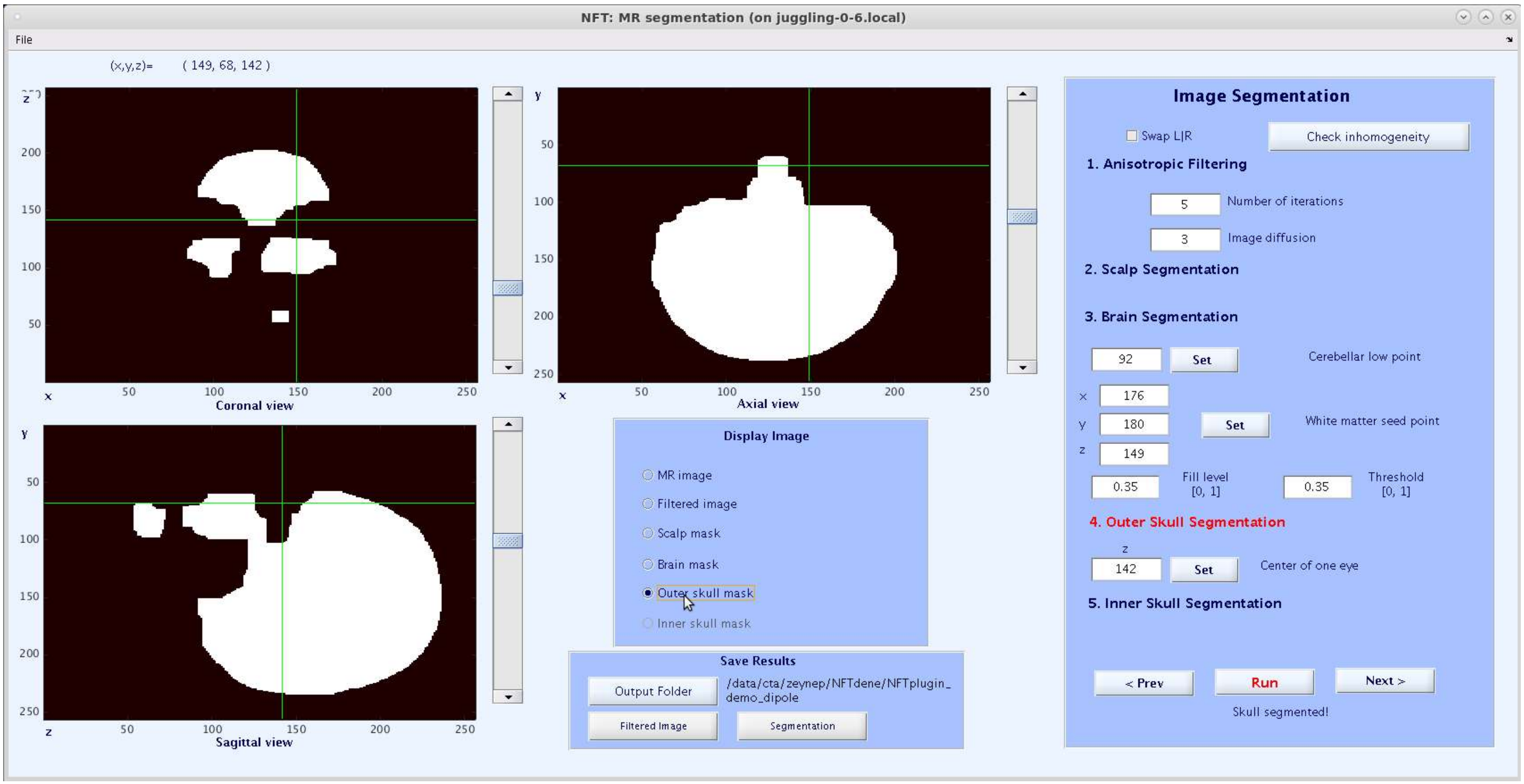


Segmentation



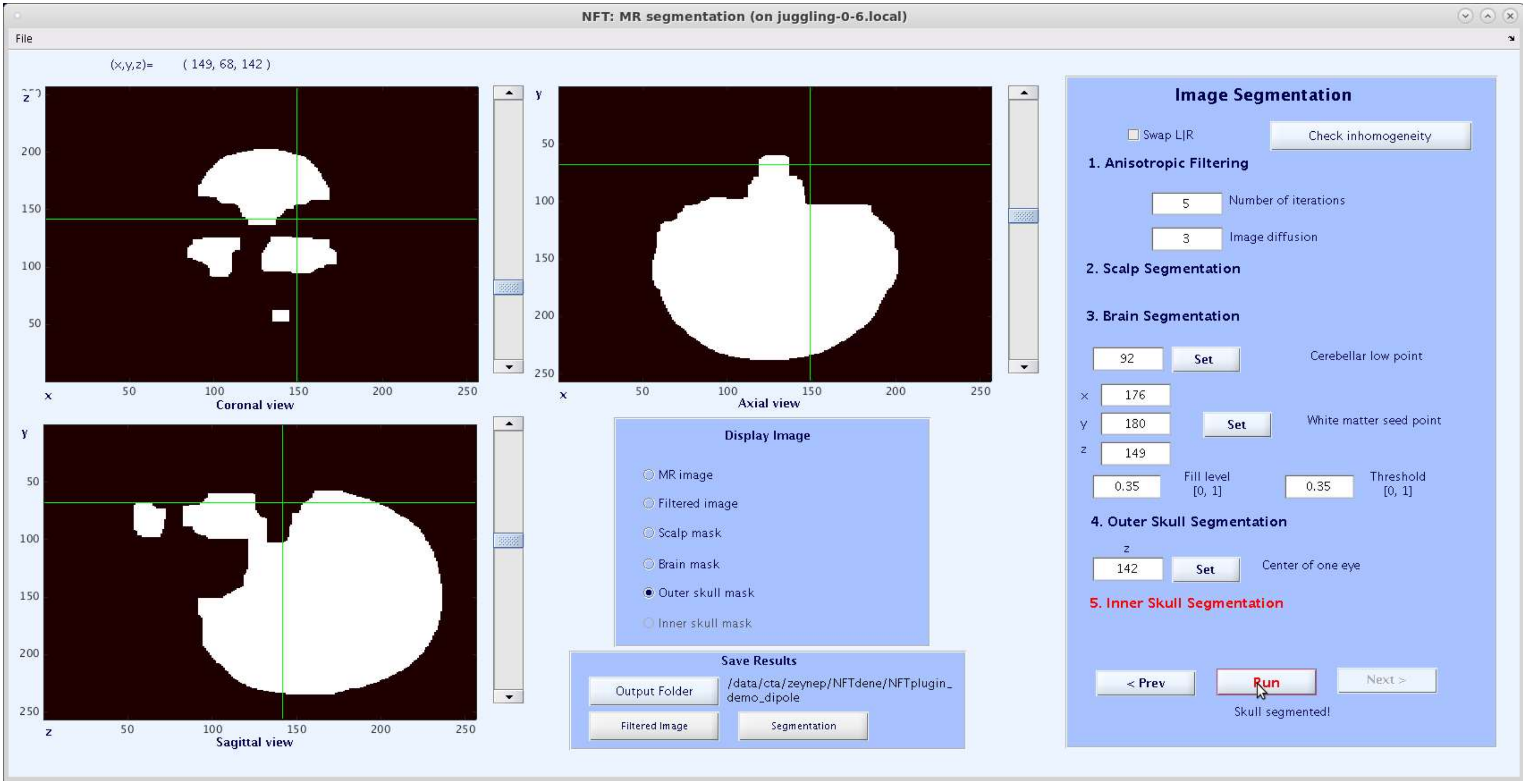
Segmentation

View skull segmentation



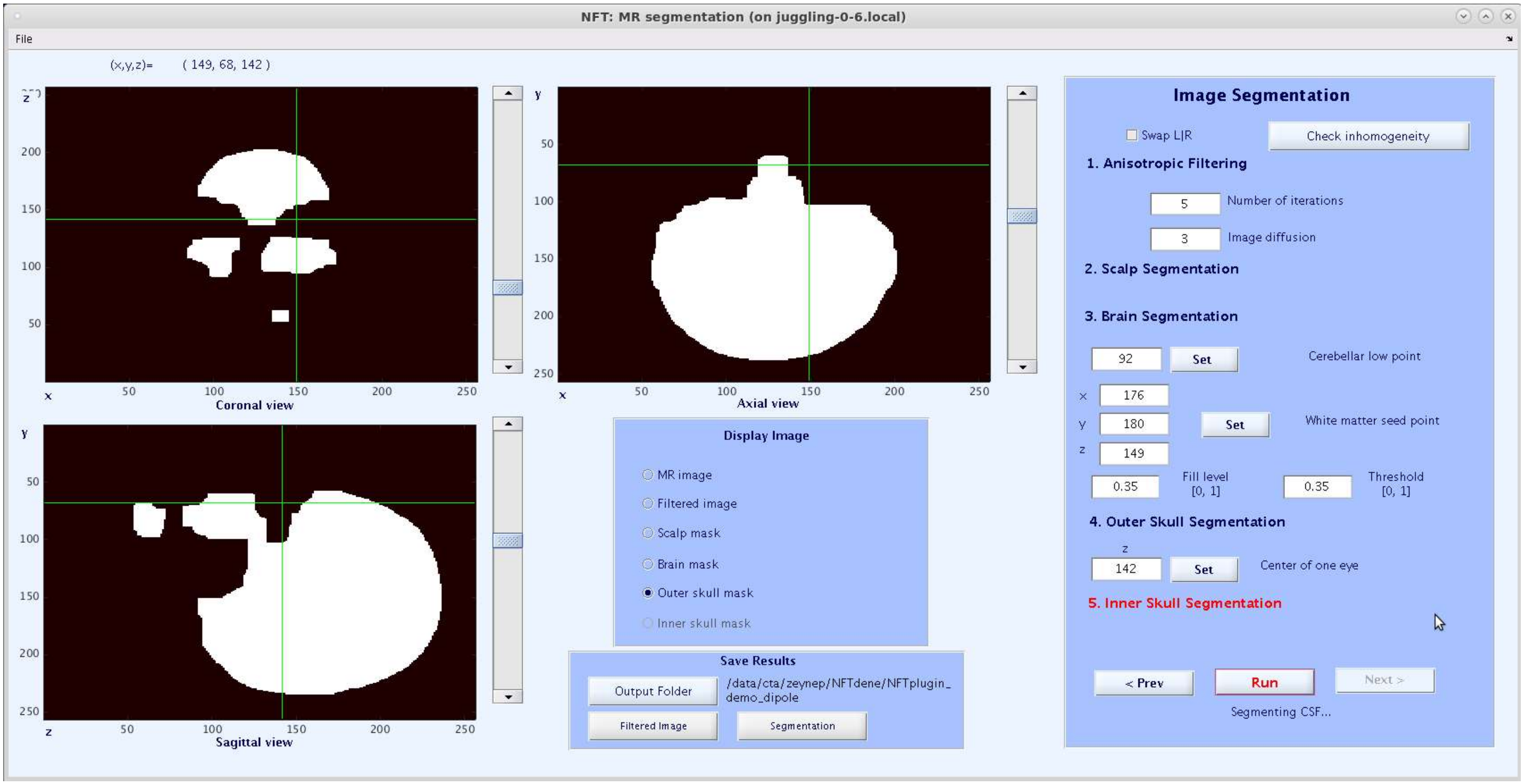
Segmentation

Click ‘Next’ for CSF segmentation

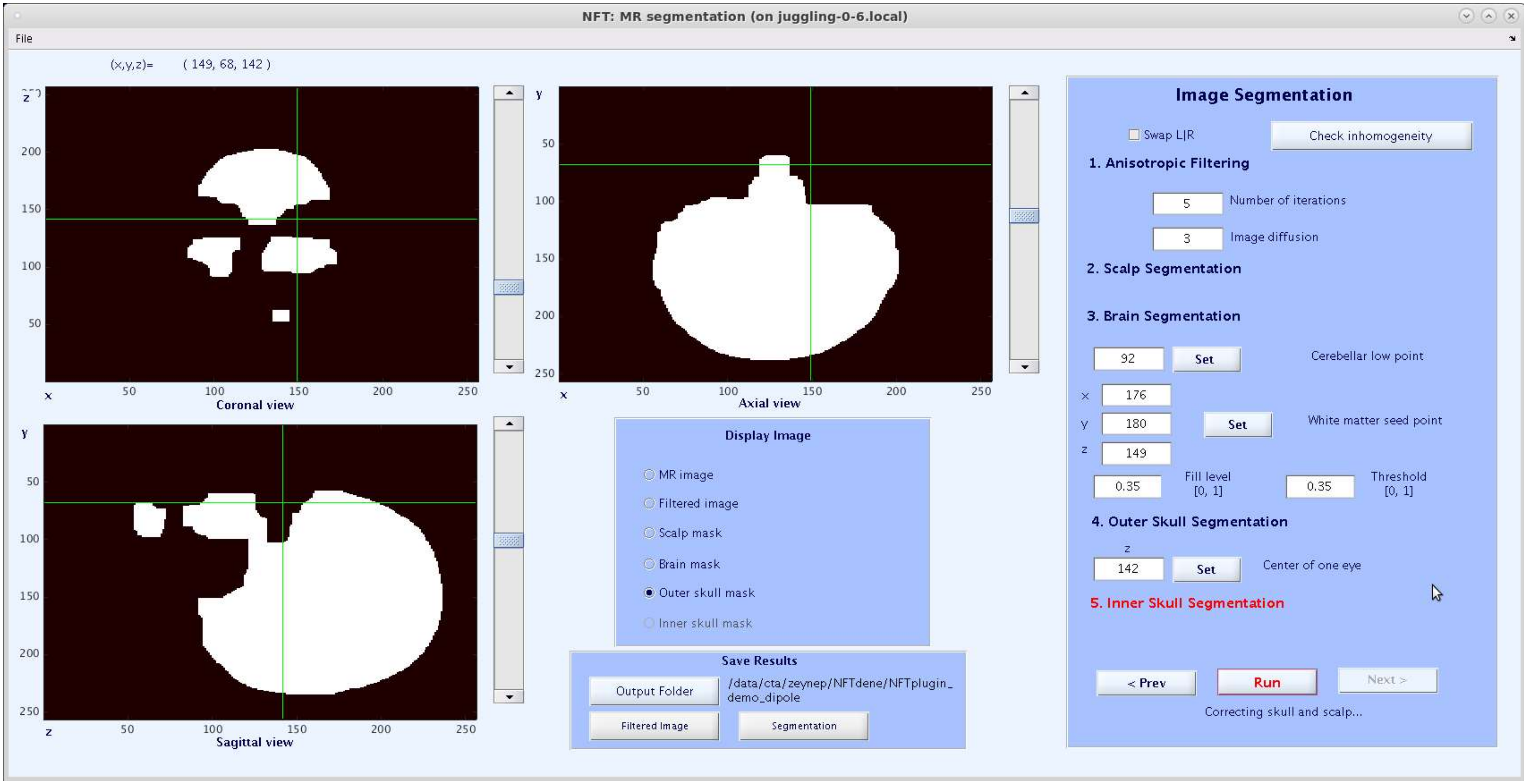


Segmentation

Click ‘Run’ for CSF segmentation

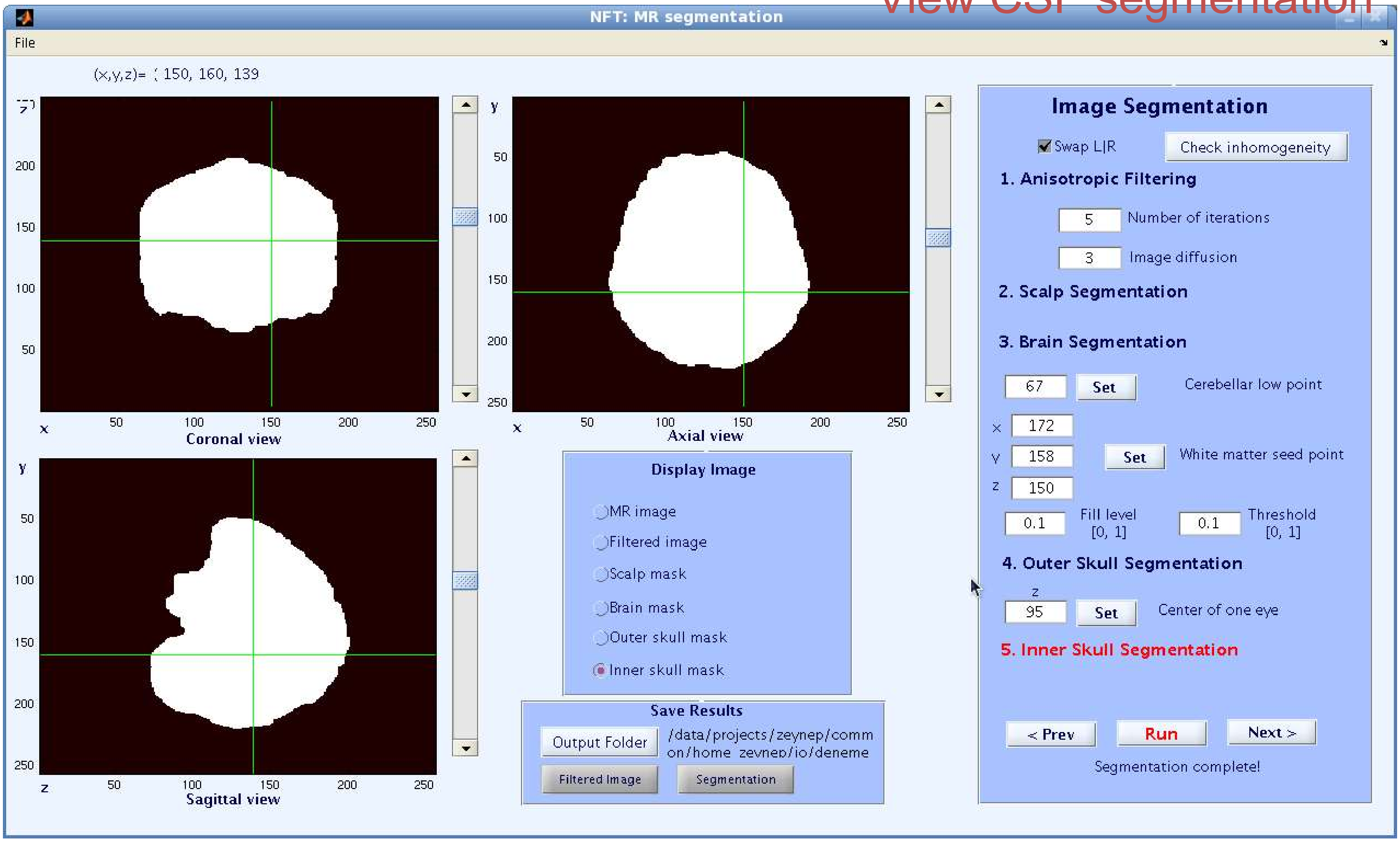


Segmentation



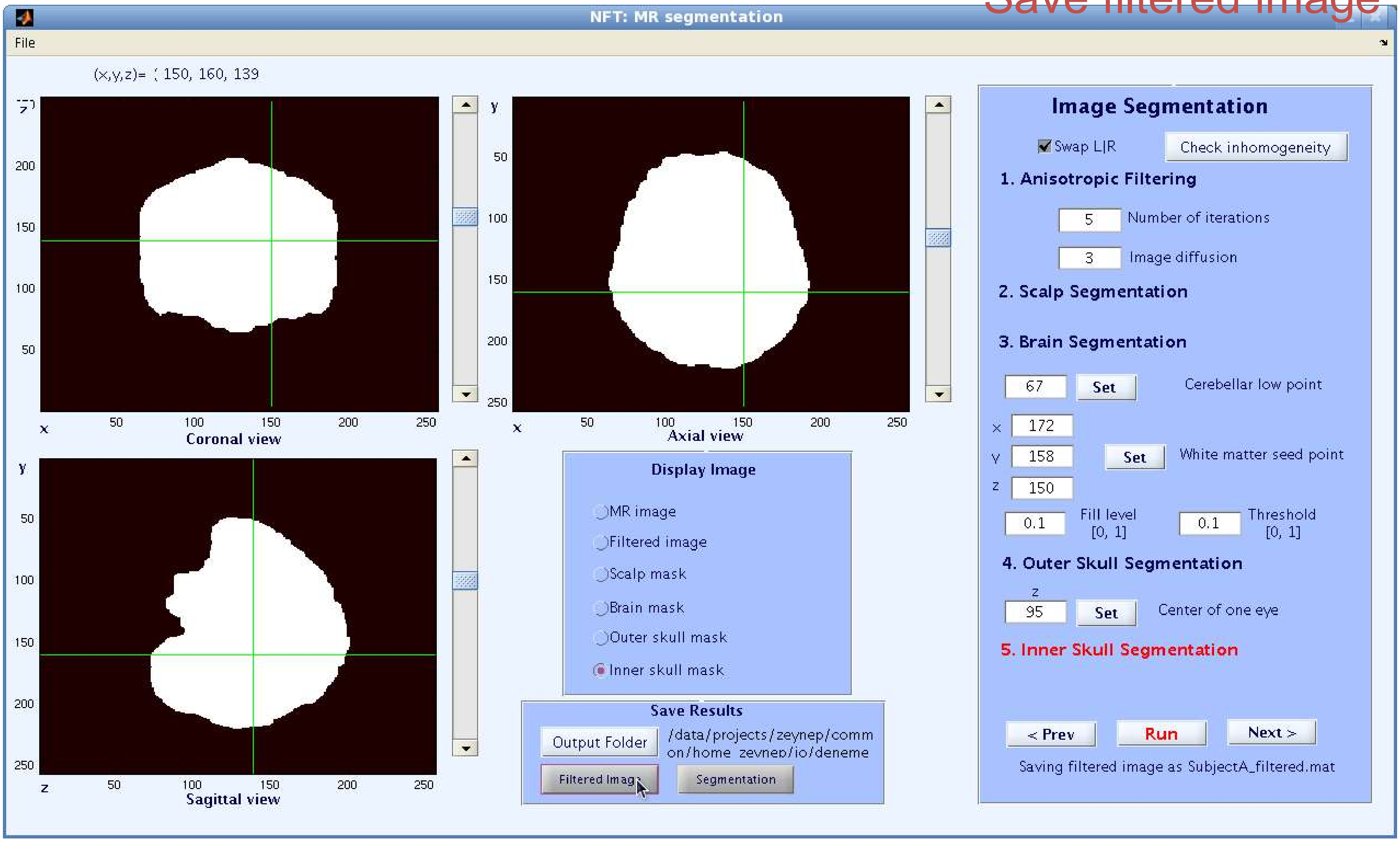
Segmentation

View CSF segmentation

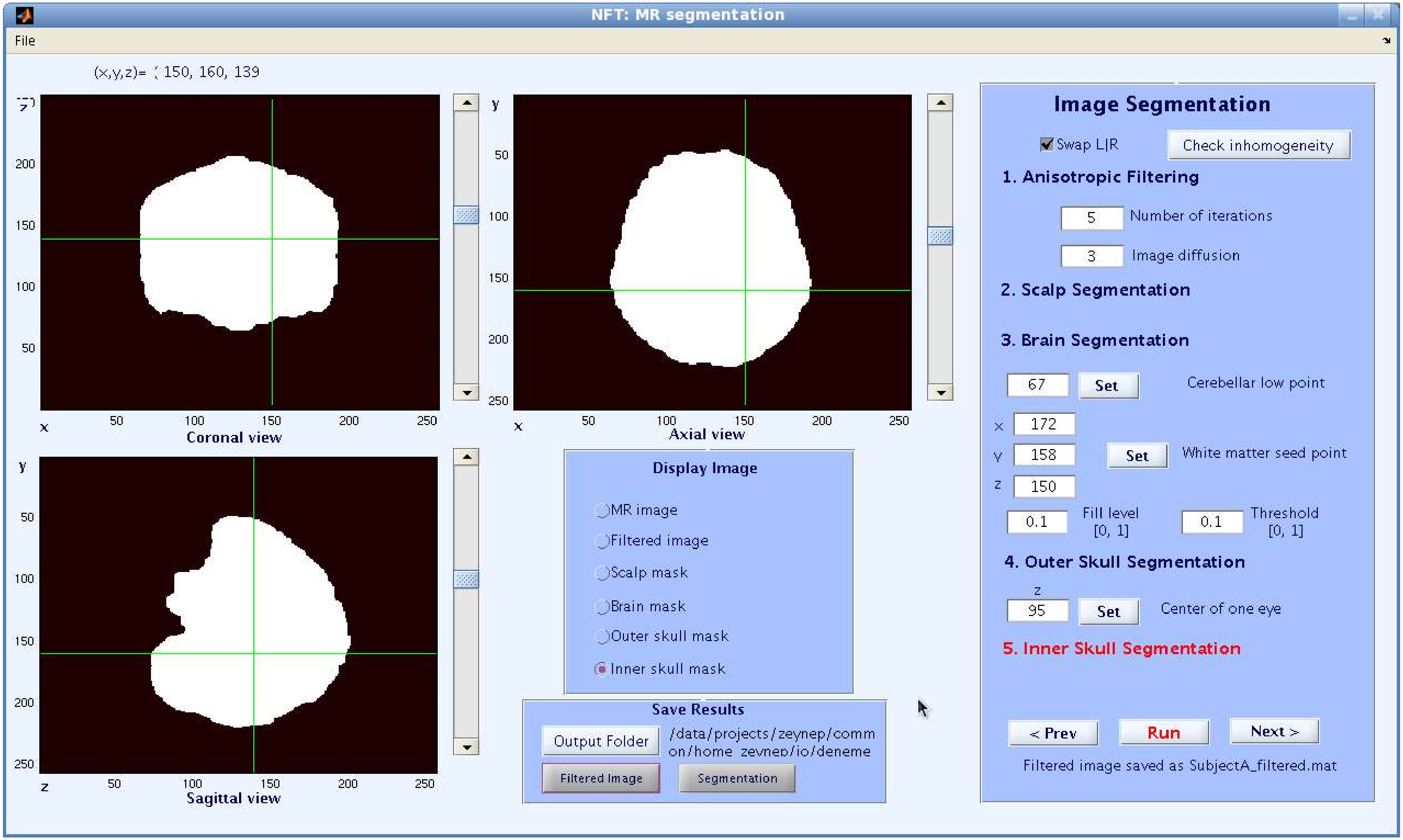


Segmentation

Save filtered image

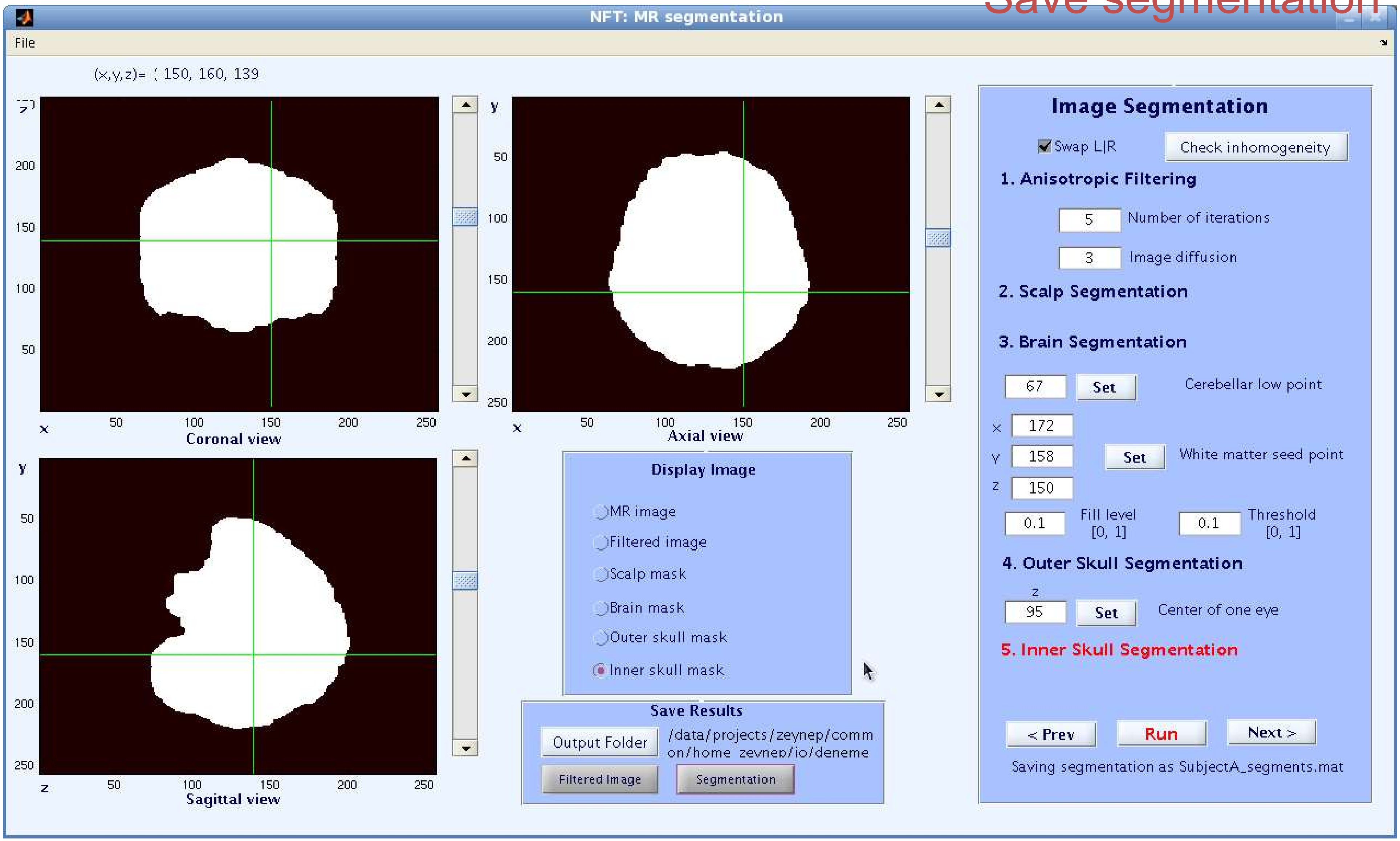


Segmentation



Segmentation

Save segmentation



Segmentation

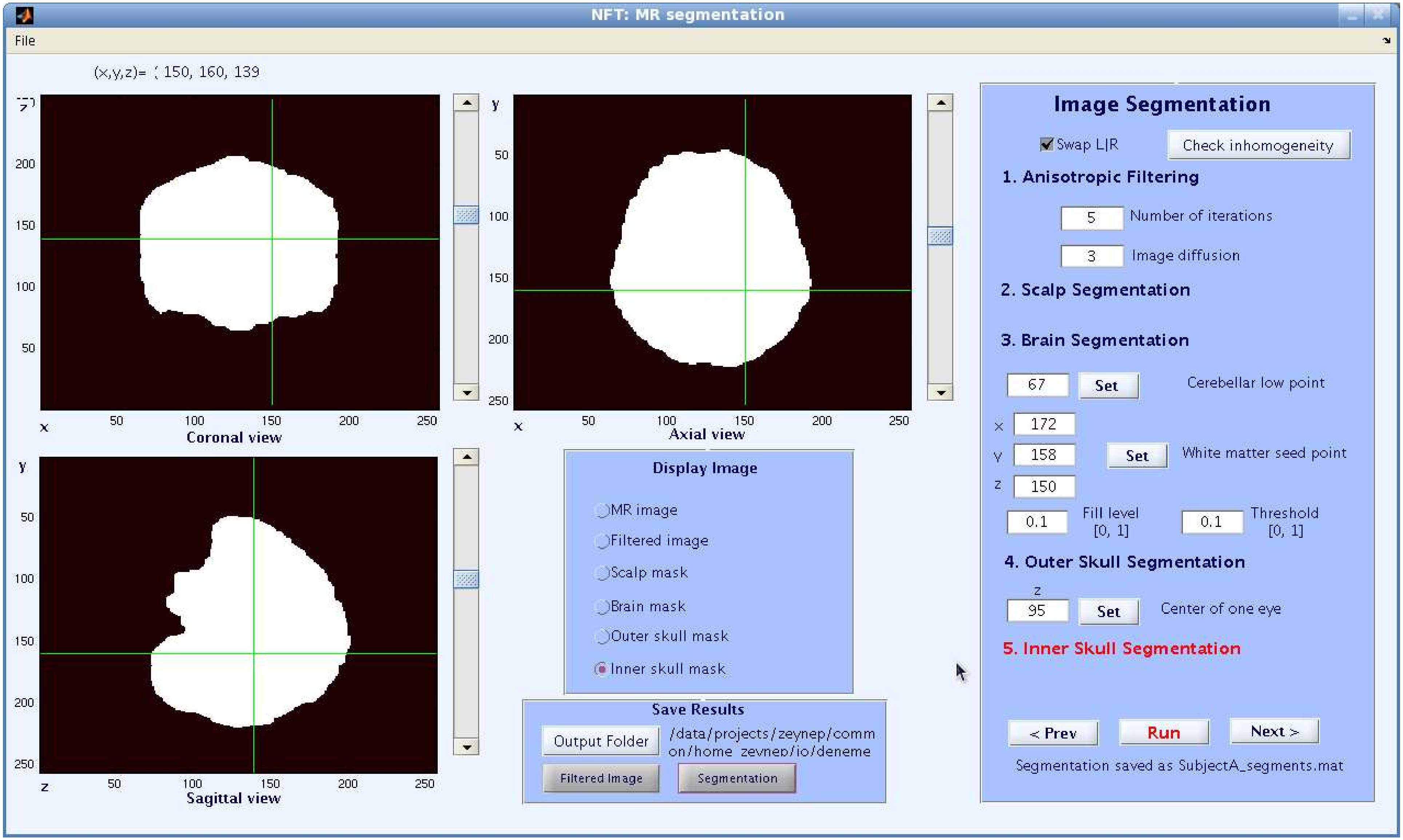
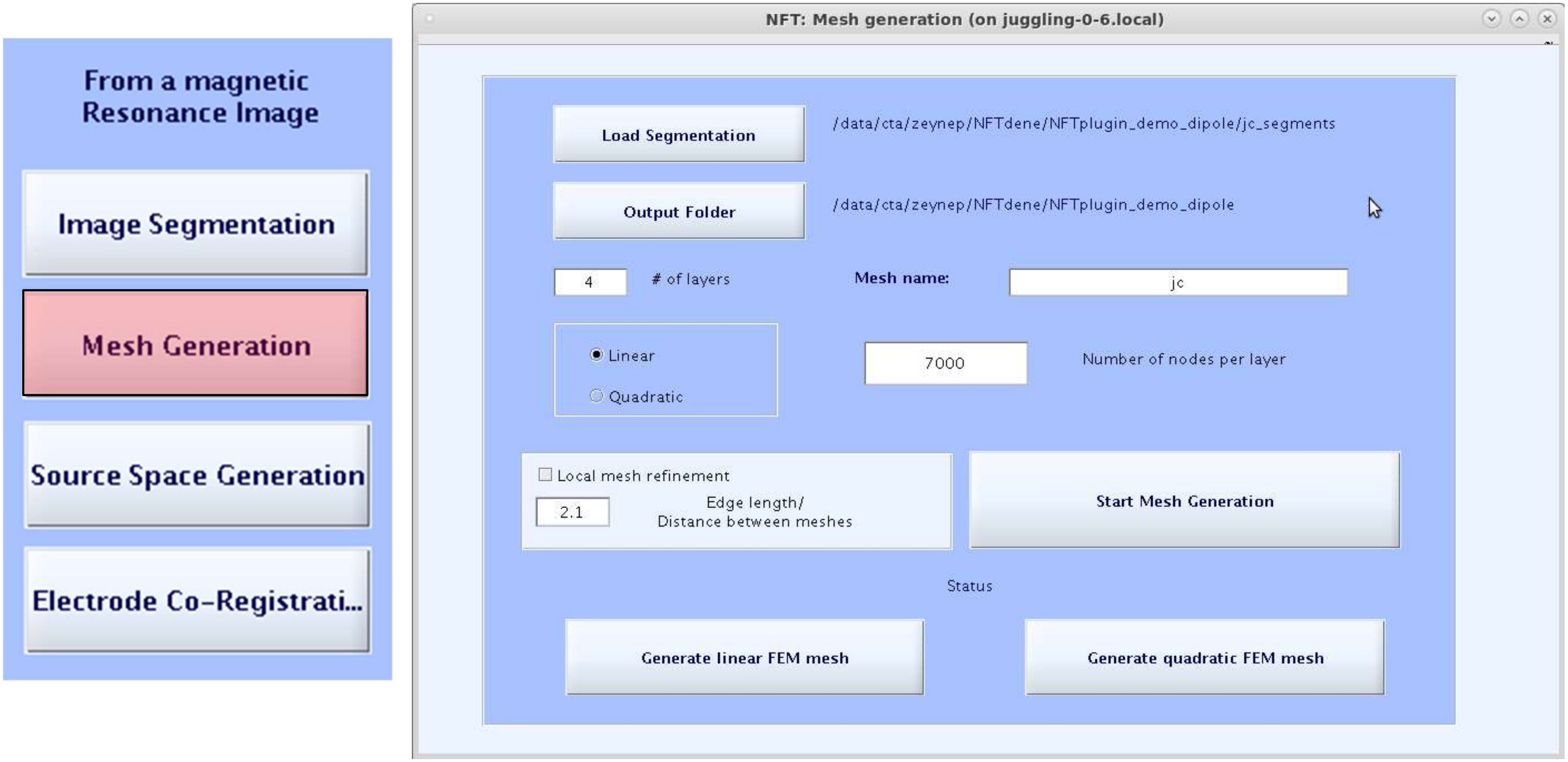


Image Segmentation

```
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

>> dir SubjectA*

SubjectA_mri.mat       SubjectA_segments.mat

>> load SubjectA_mri
>> mri

mri =

          dim: [256 256 256]
        xgrid: [1x256 double]
        ygrid: [1x256 double]
        zgrid: [1x256 double]
      anatomy: [256x256x256 double]
    transform: [4x4 double]
          hdr: []

>> load SubjectA_segments
>> Segm

Segm =

        scalpmask: [256x256x256 logical]
        brainmask: [256x256x256 logical]
    outerskullmask: [256x256x256 logical]
    innerskullmask: [256x256x256 logical]

>> 
```

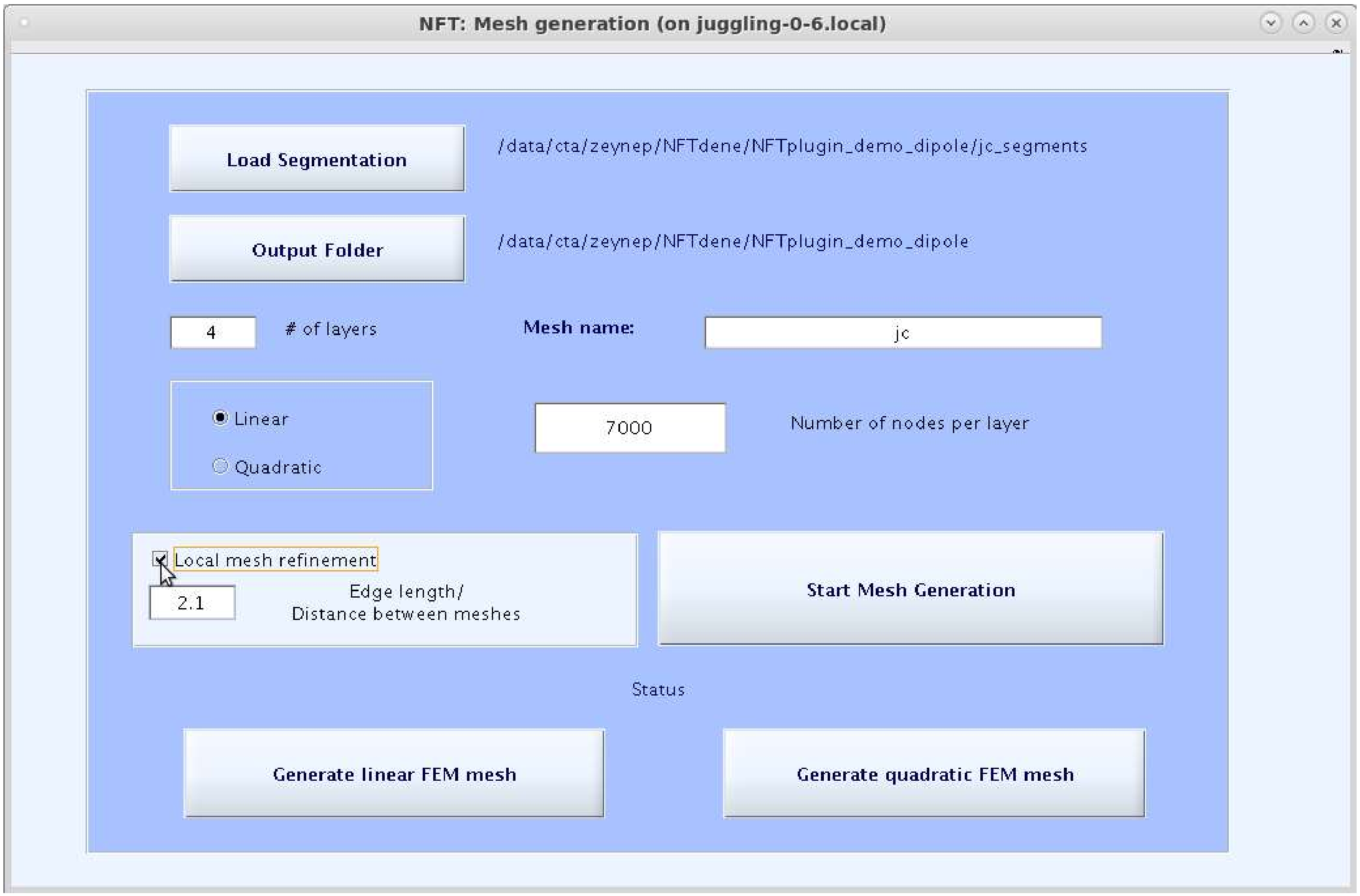
Mesh Generation



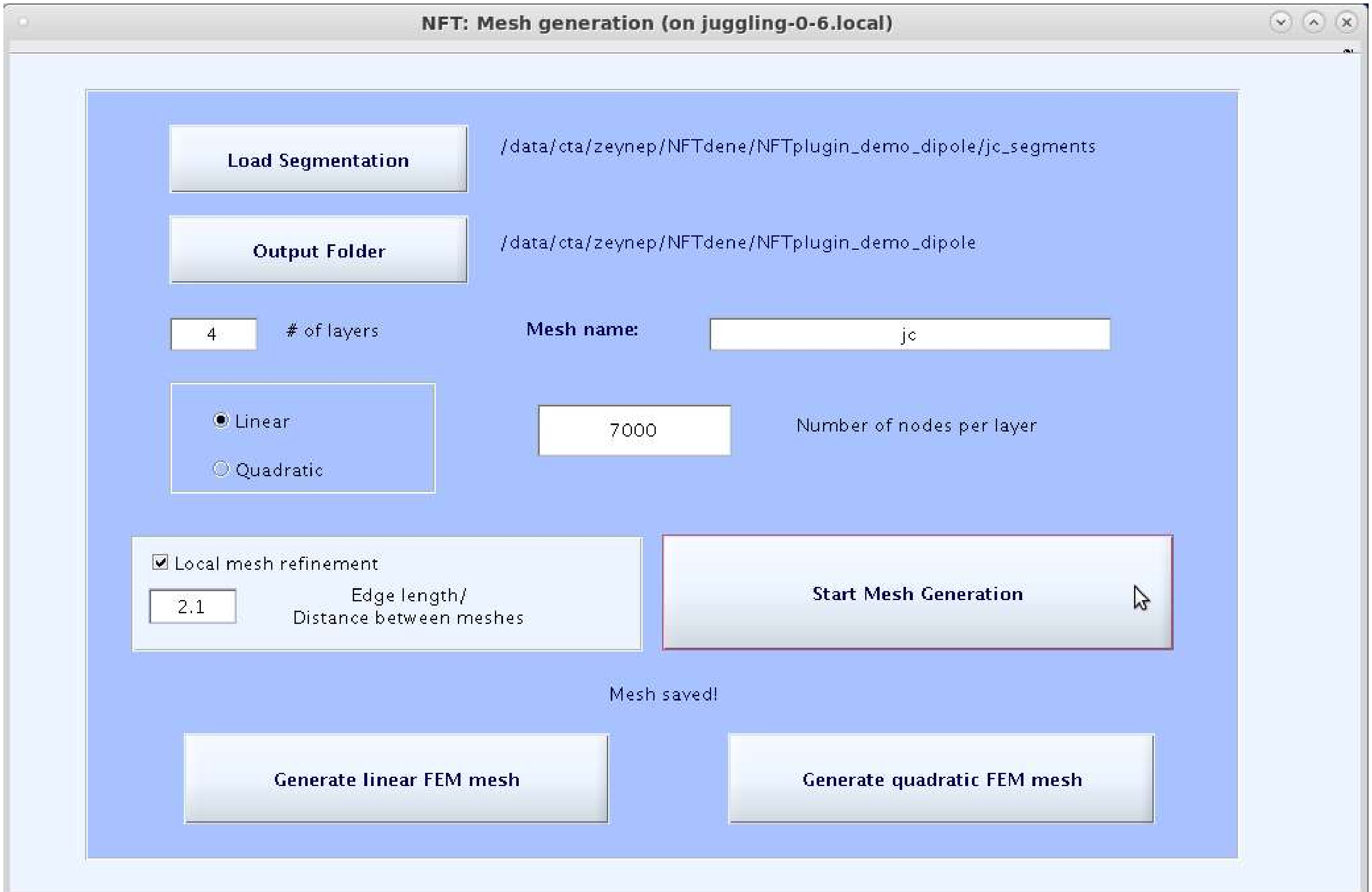
Generate Mesh for a 3 or 4 layer head model

Mesh generation

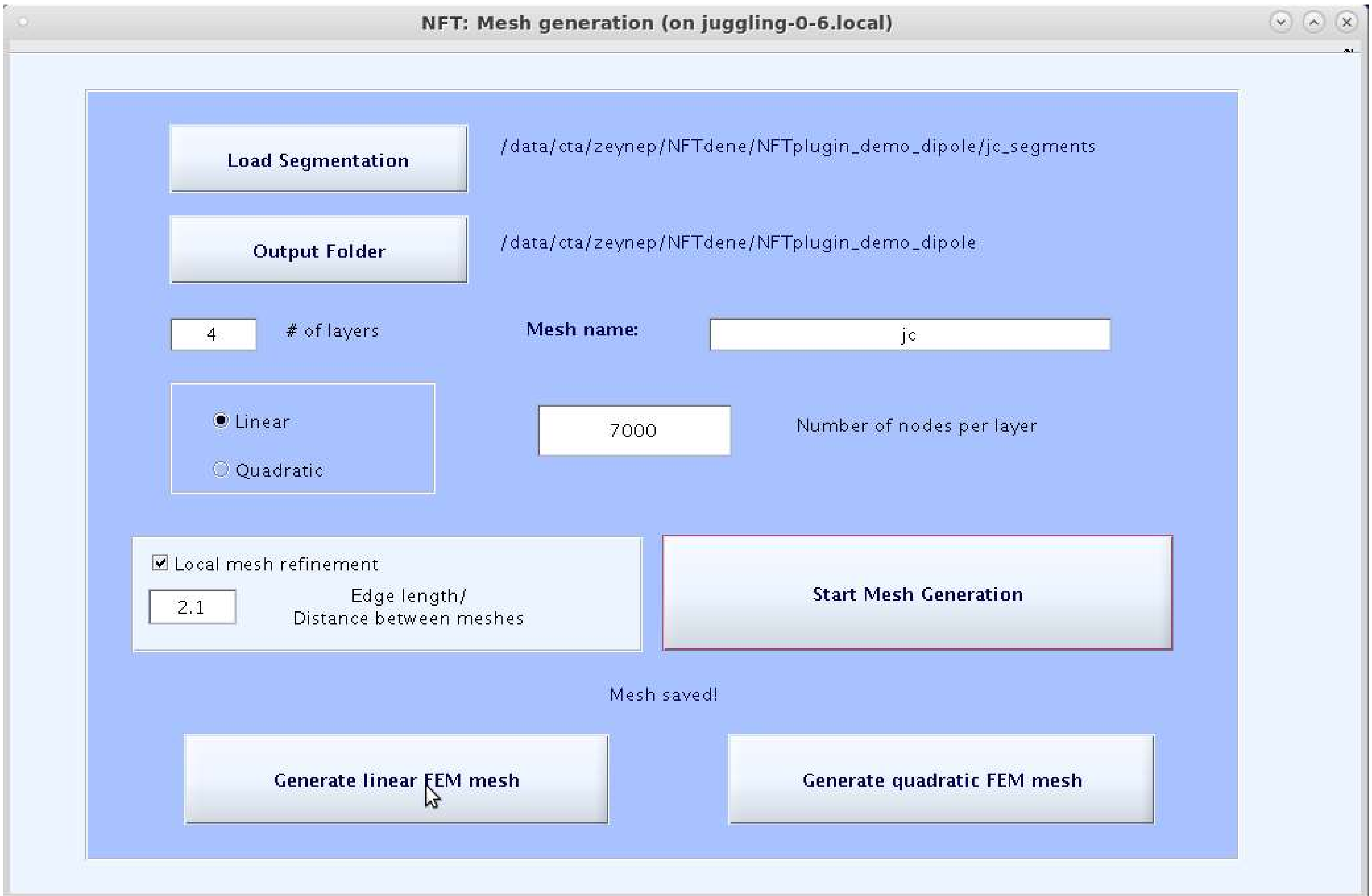
Click local mesh refinement



Mesh generation

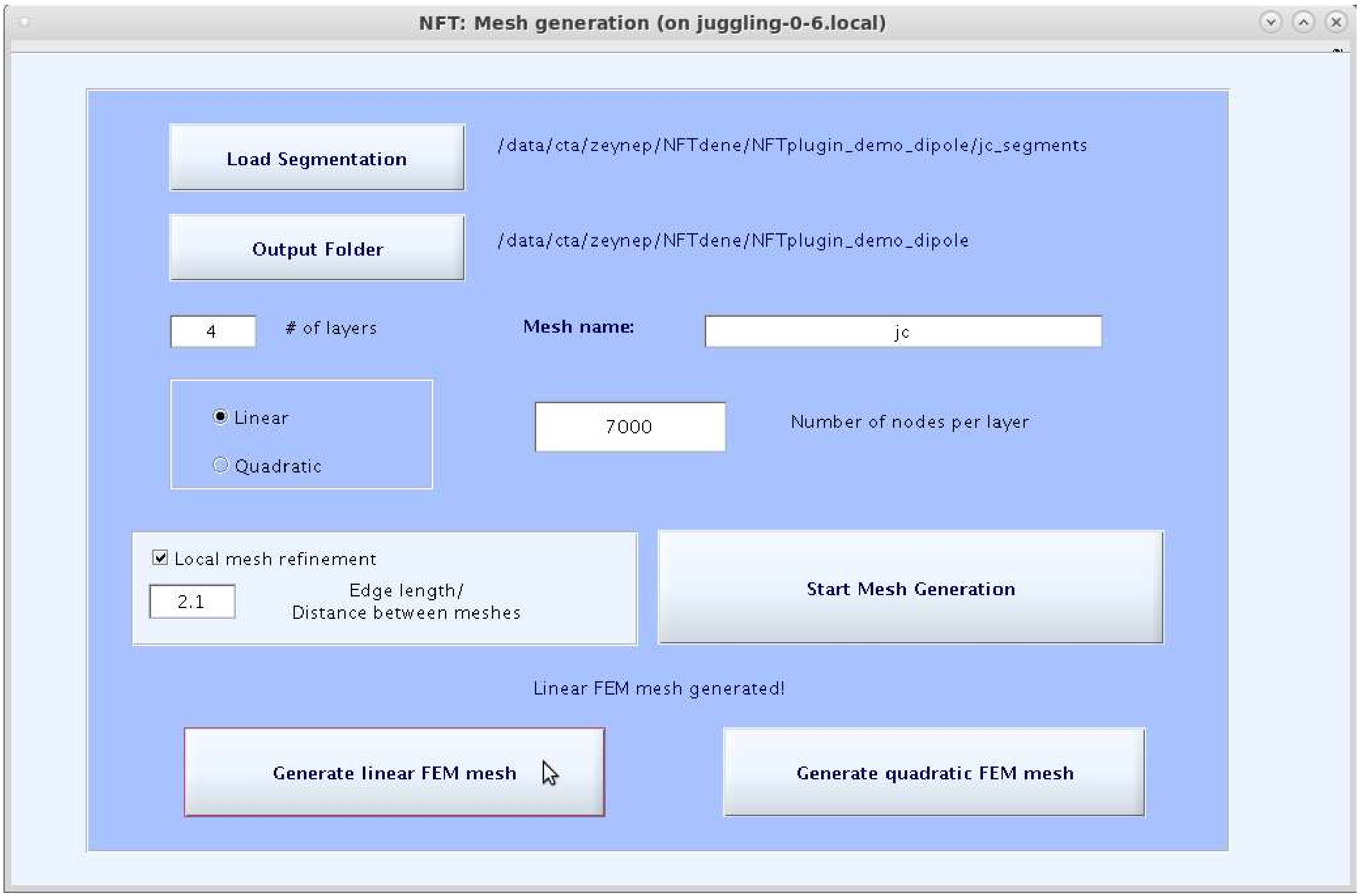


Mesh generation

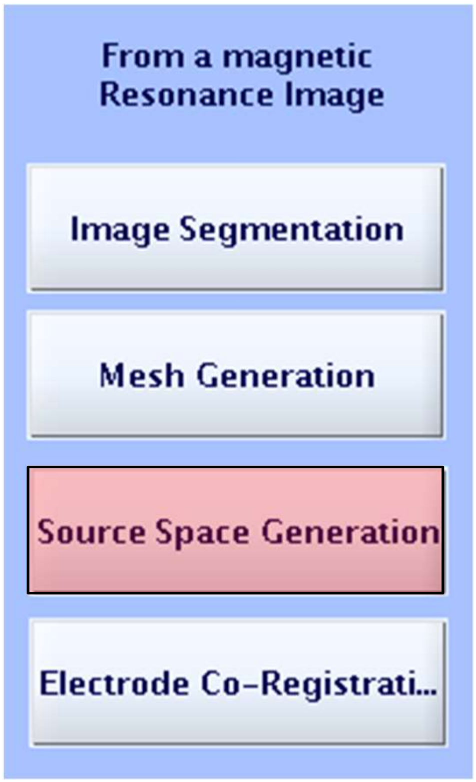


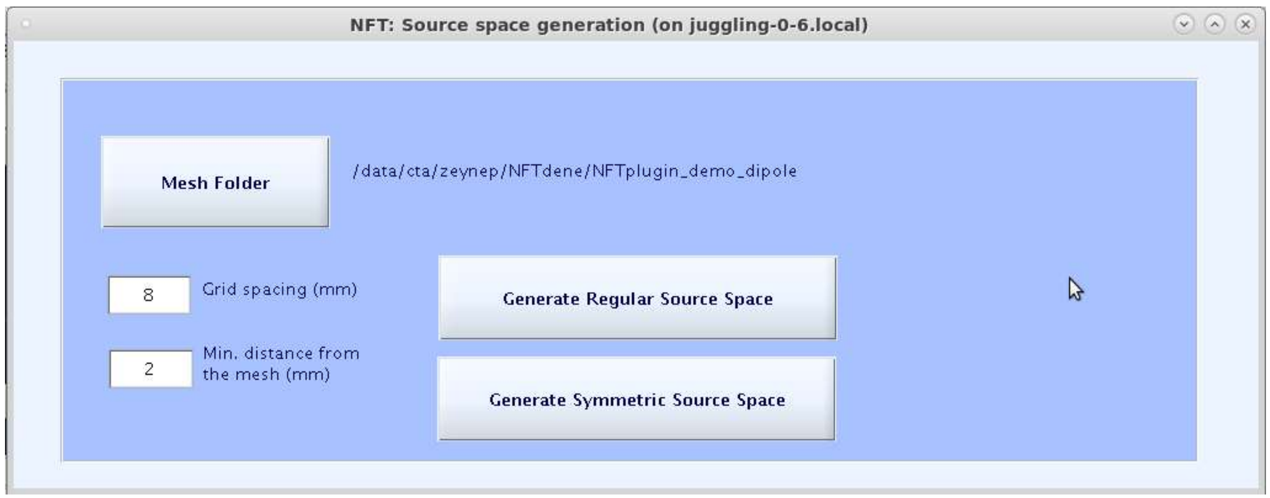
- [C,E] = ReadSMF('Scalp.smf',0,0,0,1);
- Plotmesh(E(:,2:4),C(:,2:4))

Mesh generation



Source Space Generation

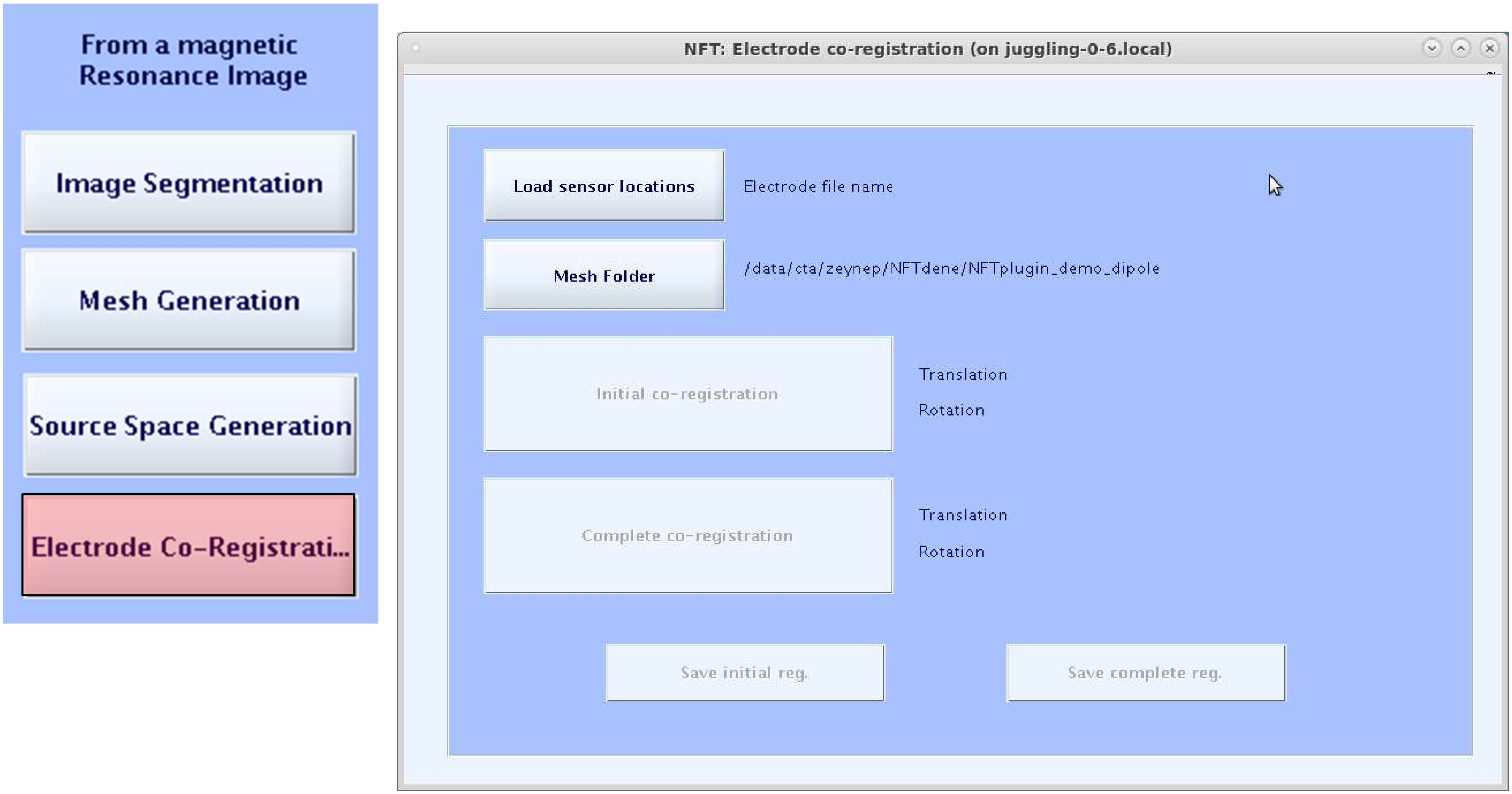




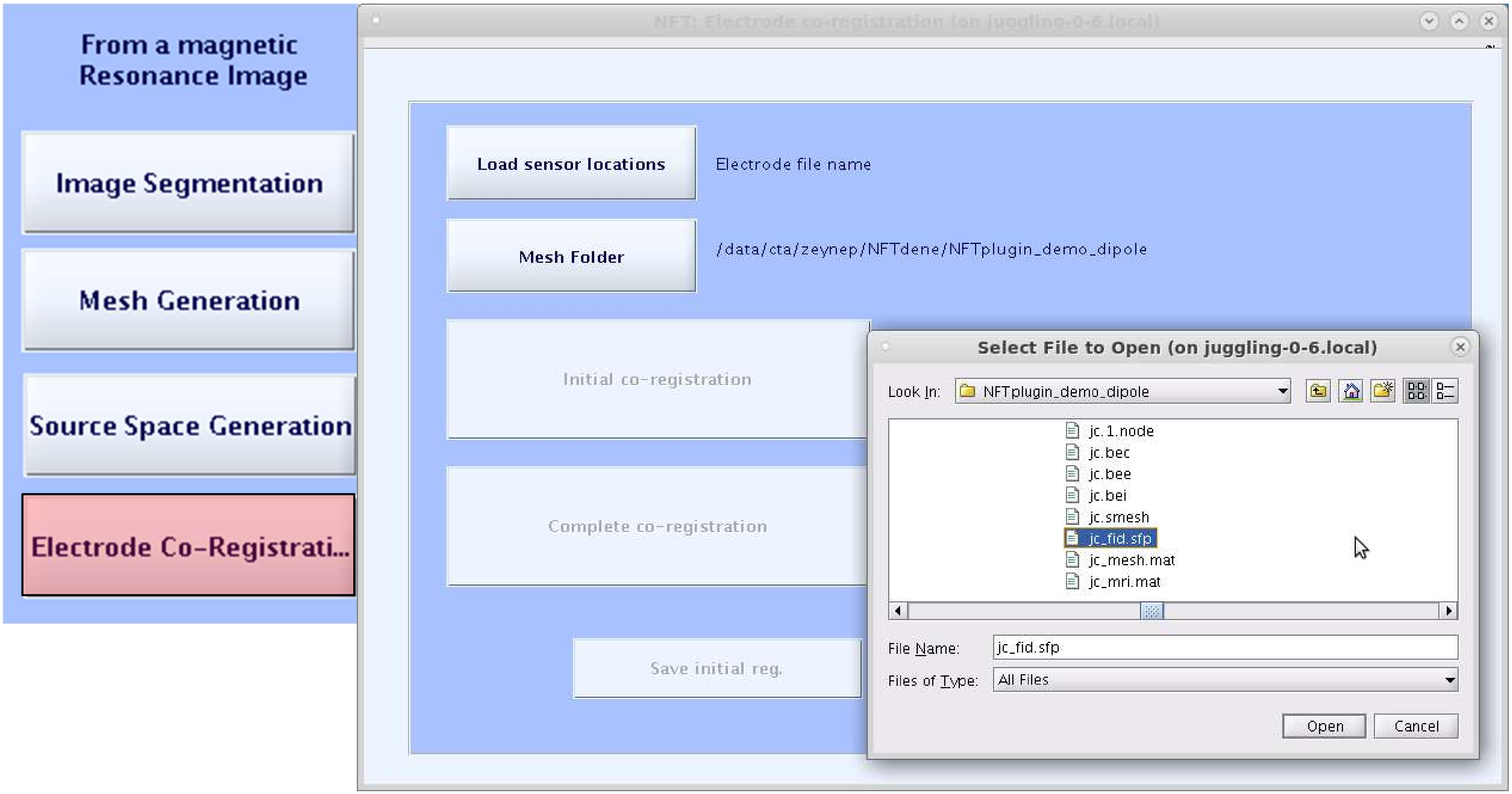
Generates a simple source space:
Regular Grid inside the brain
With a given spacing and distance to the mesh

Source Space Generation

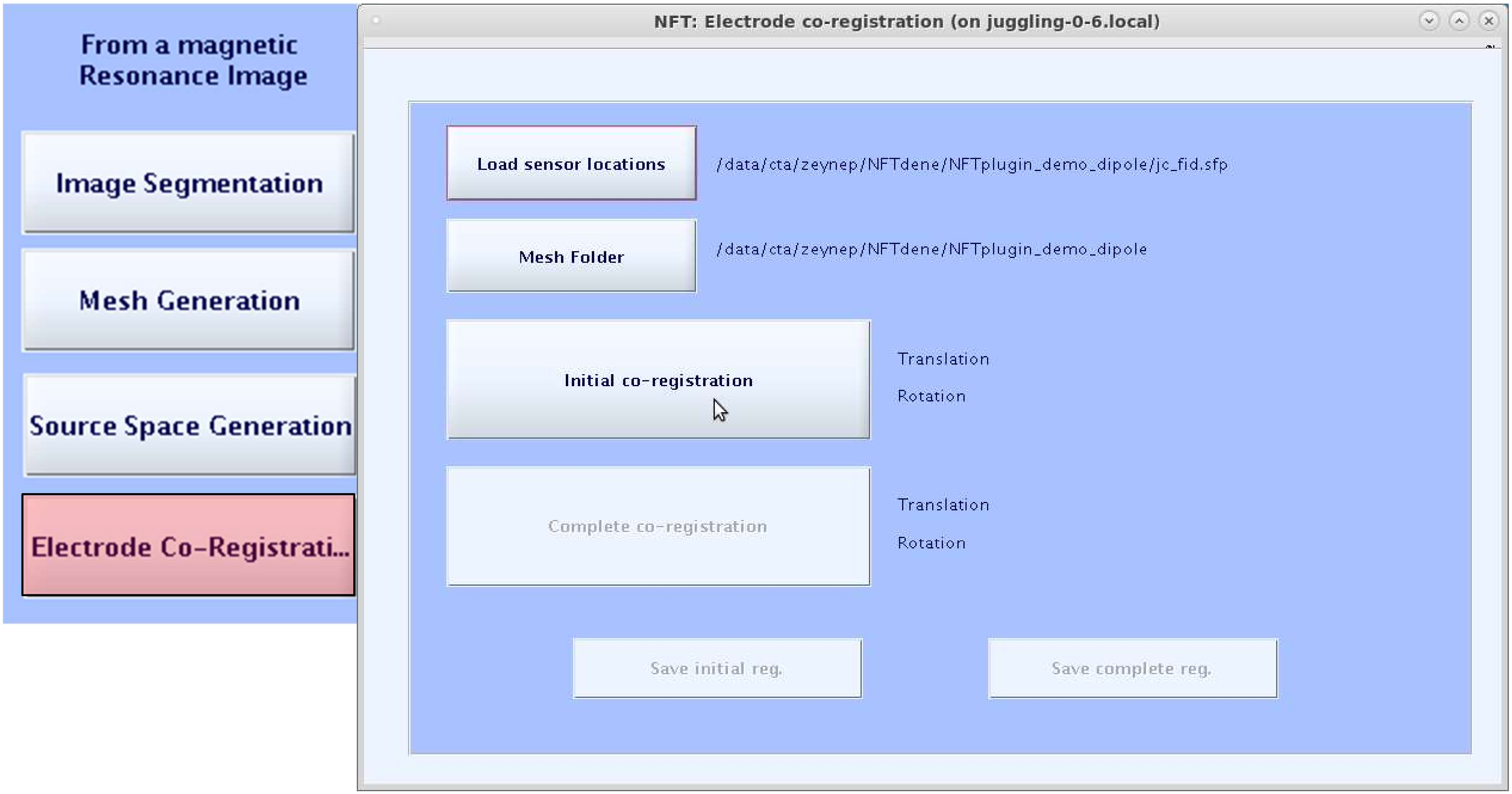
Electrode Co-registration



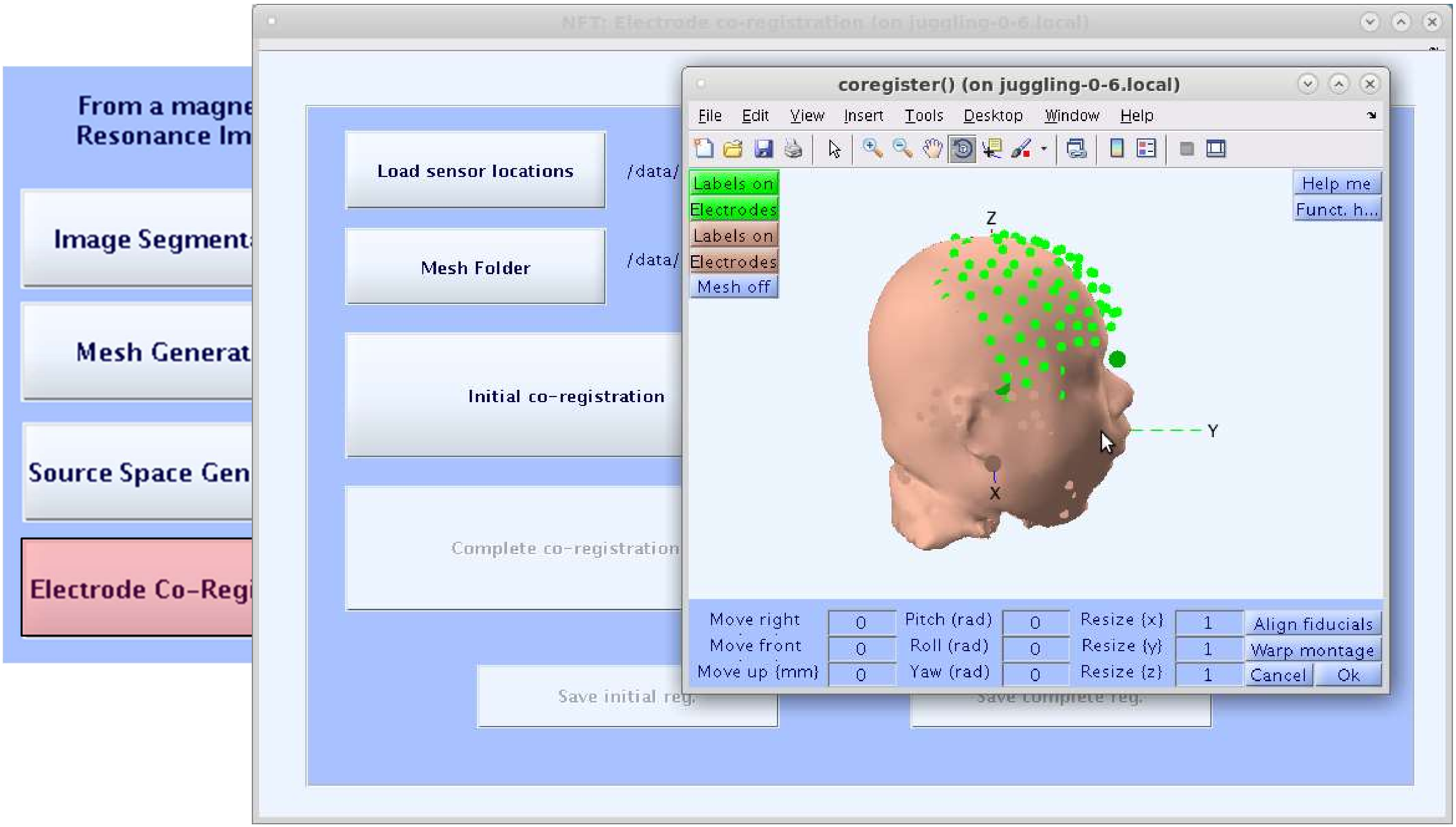
Electrode Co-registration



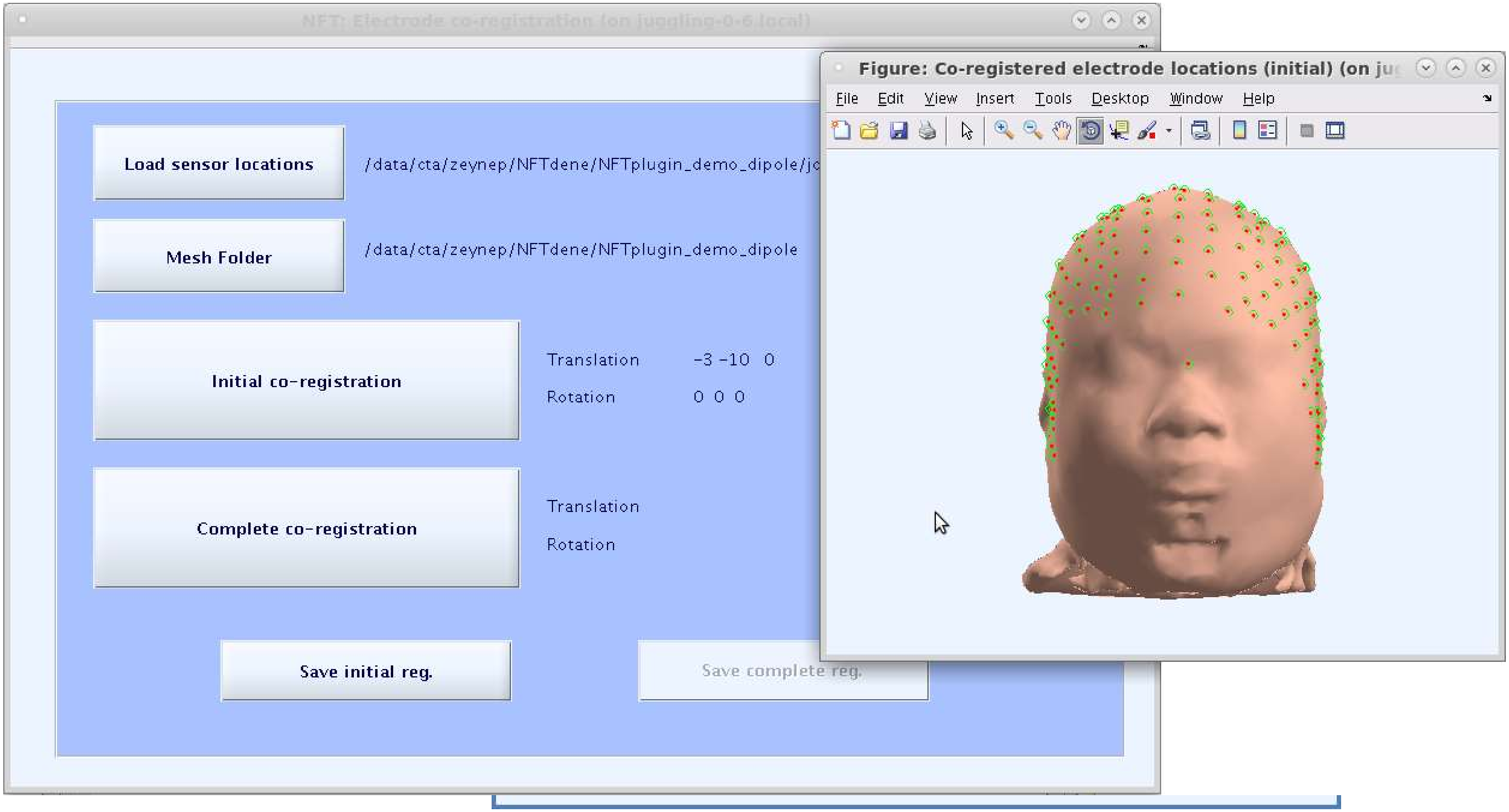
Electrode Co-registration



Electrode Co-registration

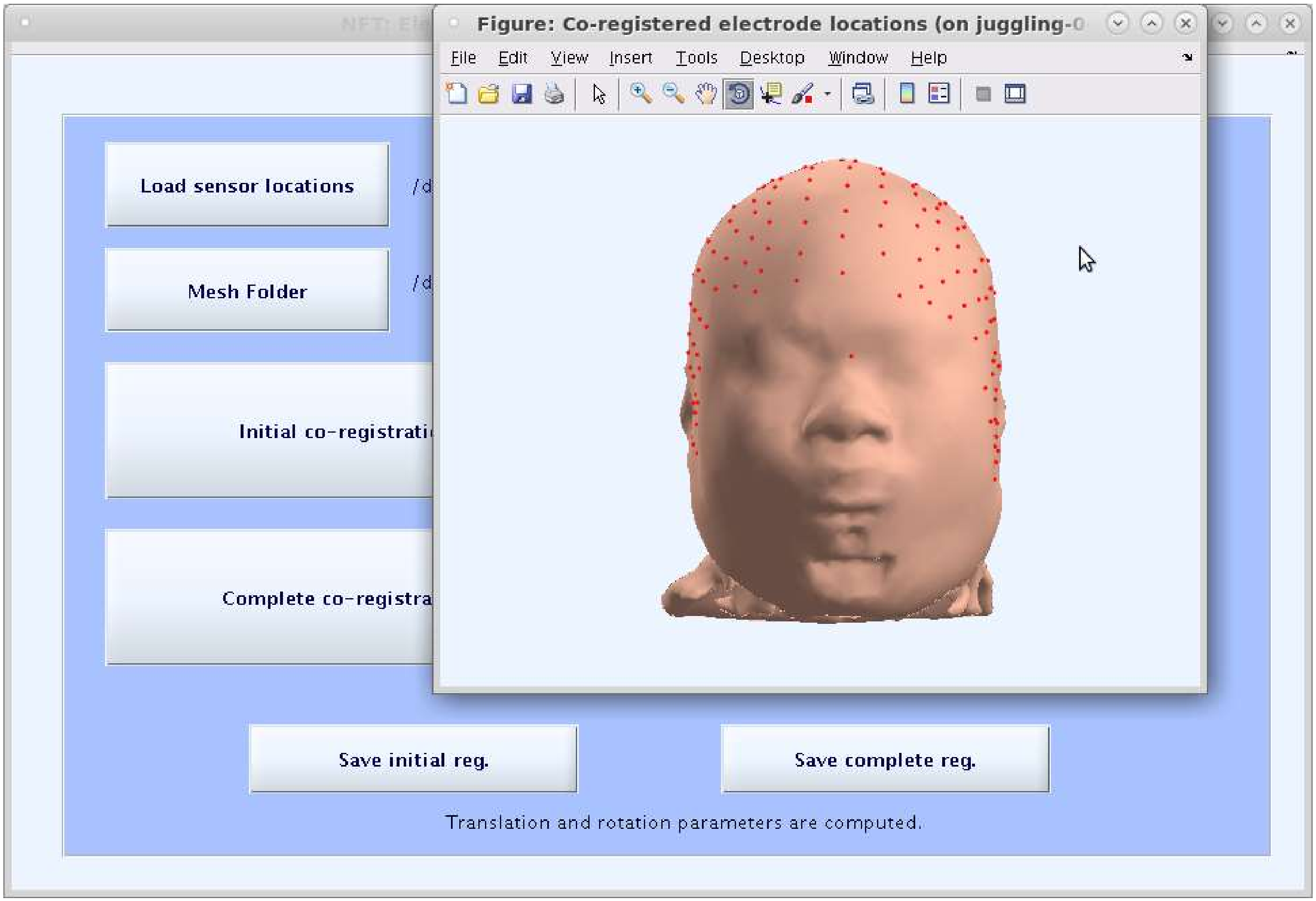


Electrode Co-registration

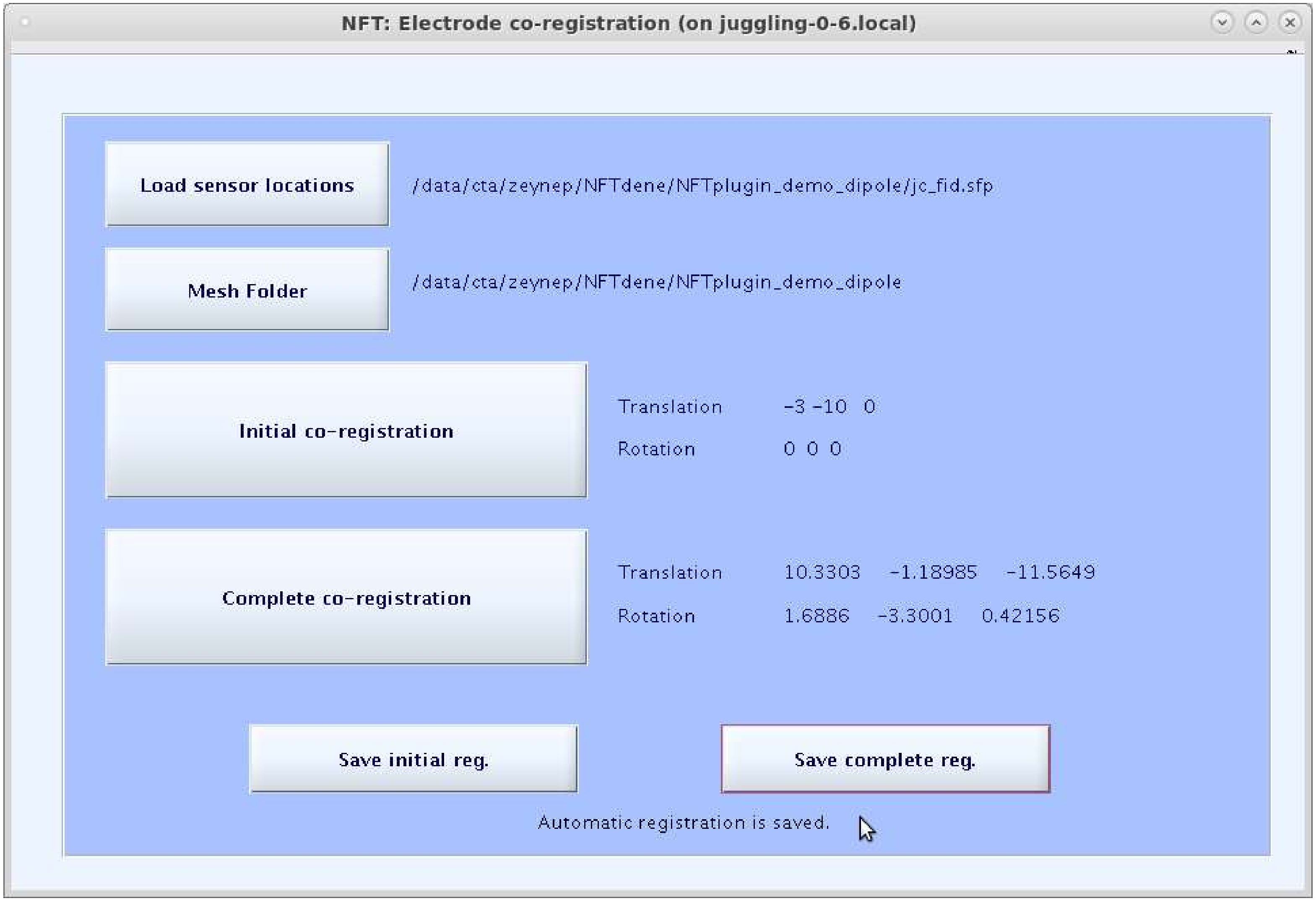


Electrode co-registration

Electrode co-registration

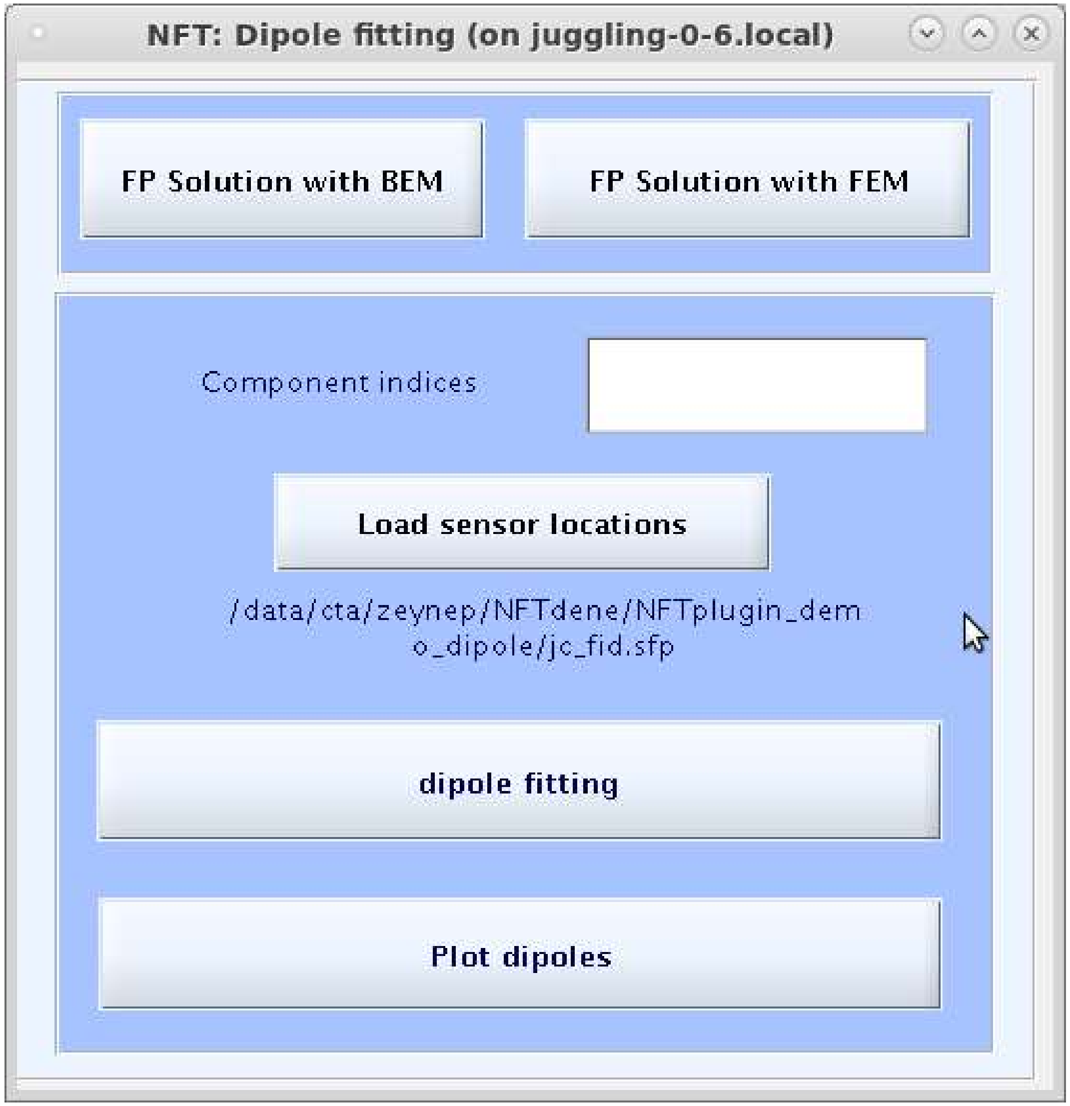


Mesh generation

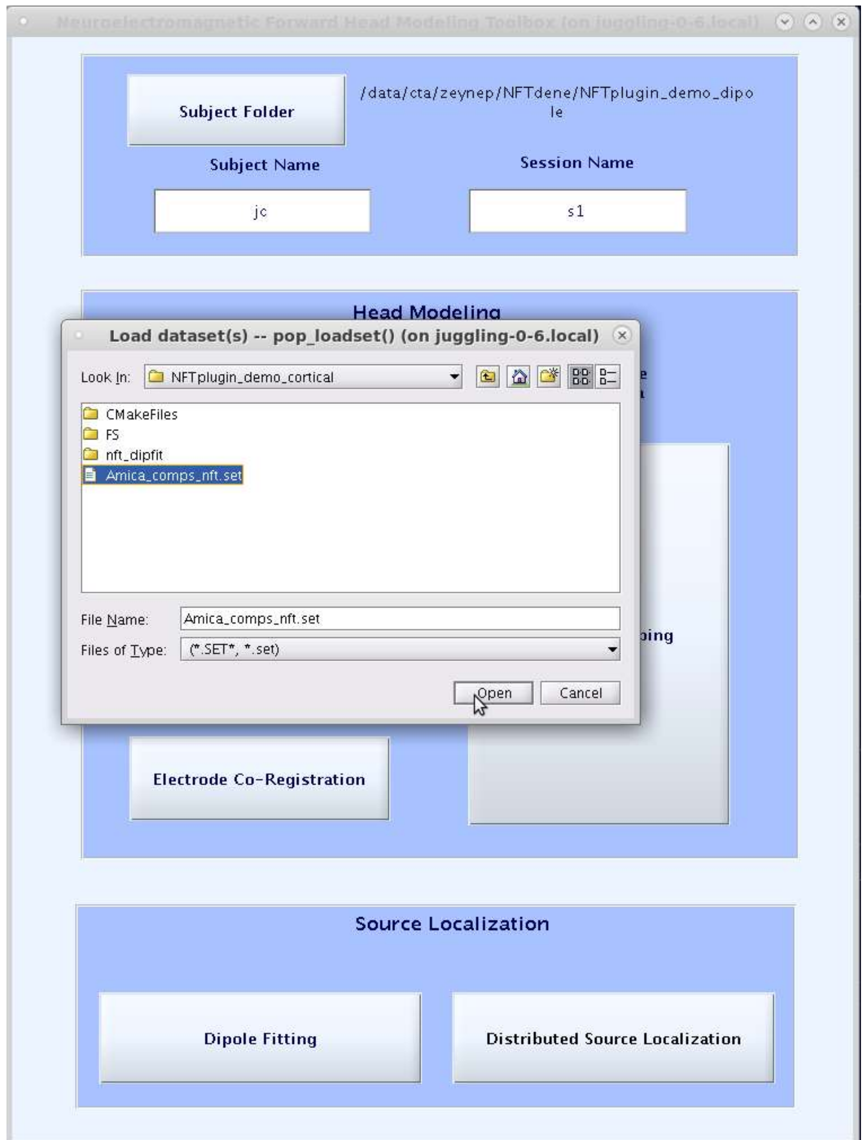


```
>> sens=load('jc_s1.sensors','-mat')

sens =

       fn: '/data/cta/zeynep/NFTdene/JC/jc_fid.sfp'
     eloc: [1x208 struct]
      pnt: [208x3 double]
      ind: [1x208 double]
    param: [1x1 struct]

>> sens.param

ans =

    init: [-3 -10 0 0 0 0 1 1 1]
    auto: [10.3303 -1.1899 -11.5649 1.6886 -3.3001 0.4216]

>> |
```

Dipole source localization



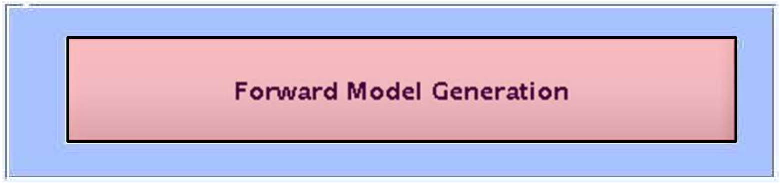
Select EEG data

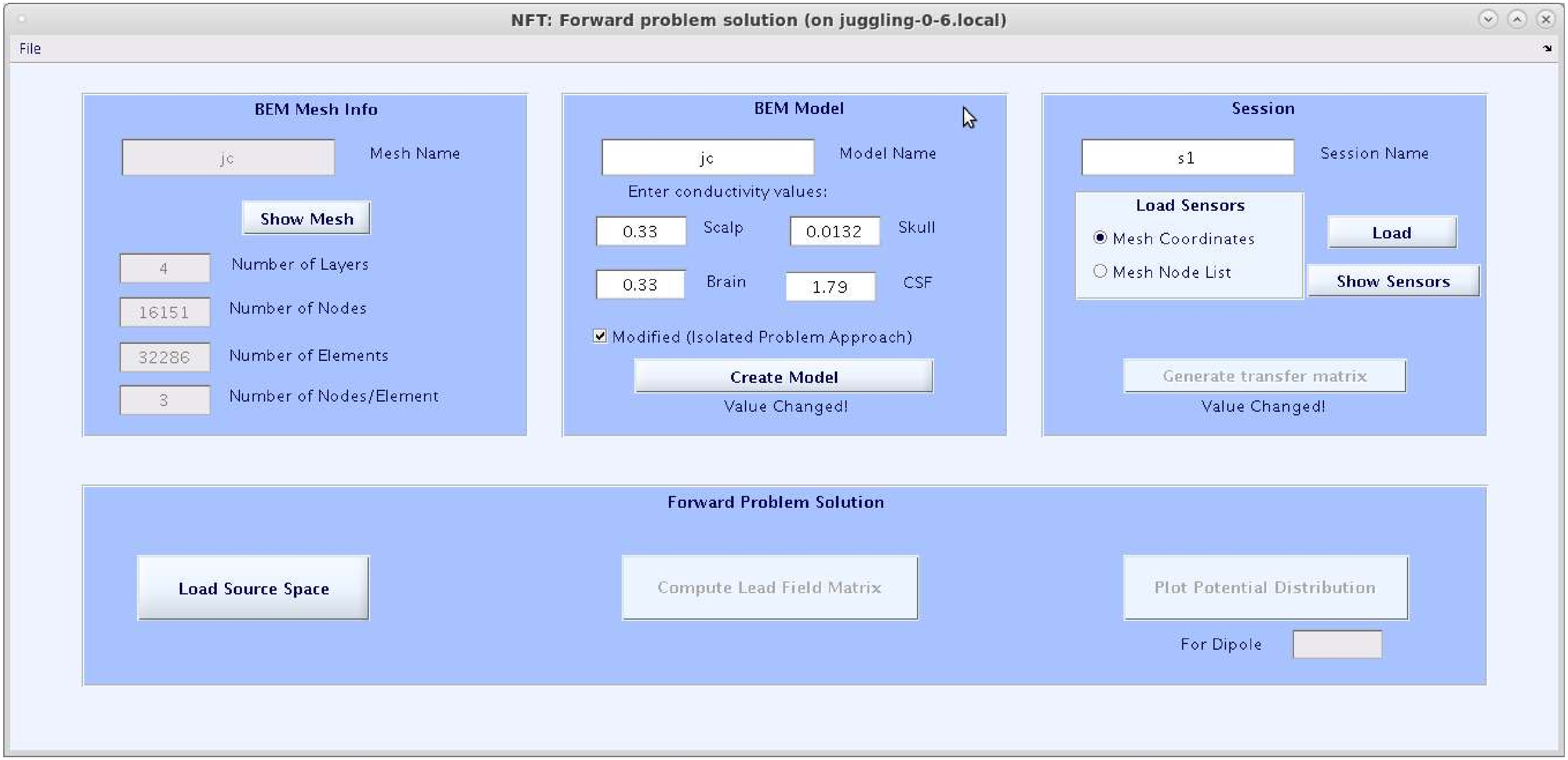


Forward Problem Solver

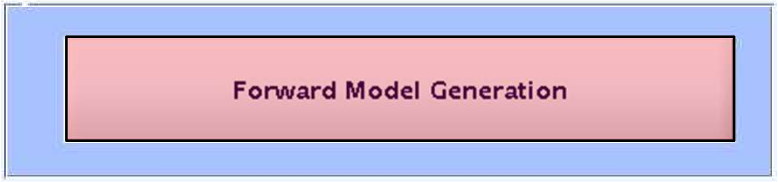
- MATLAB interface to numerical solvers
- Boundary Element Method or Finite Element Method
  - EEG Only (for now)
  - Interfaces to the Matrix generator executable written in C++
- Other computation done in MATLAB
- Generated matrices are stored on disk for future use.

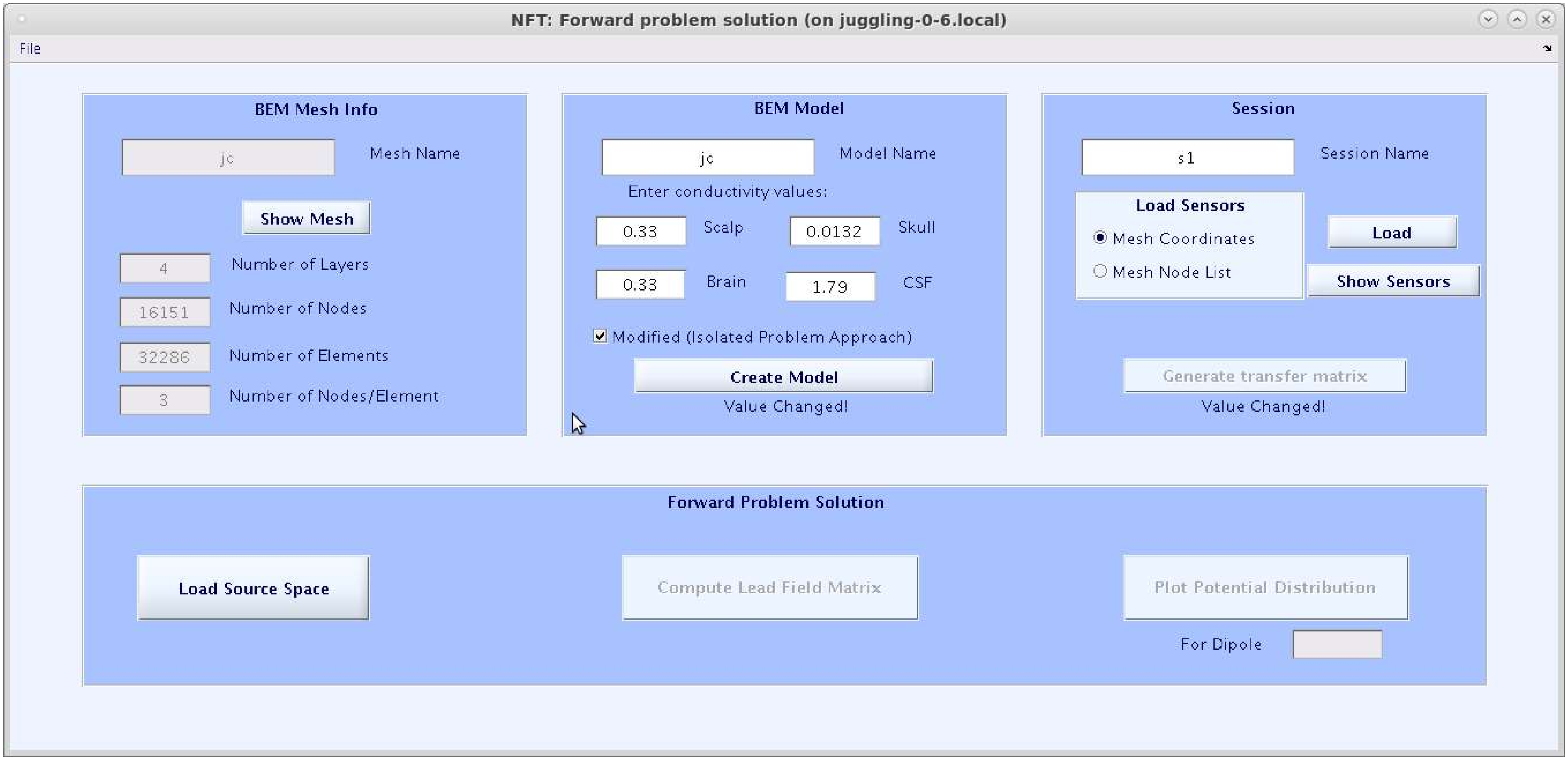
Forward Problem Solution with BEM





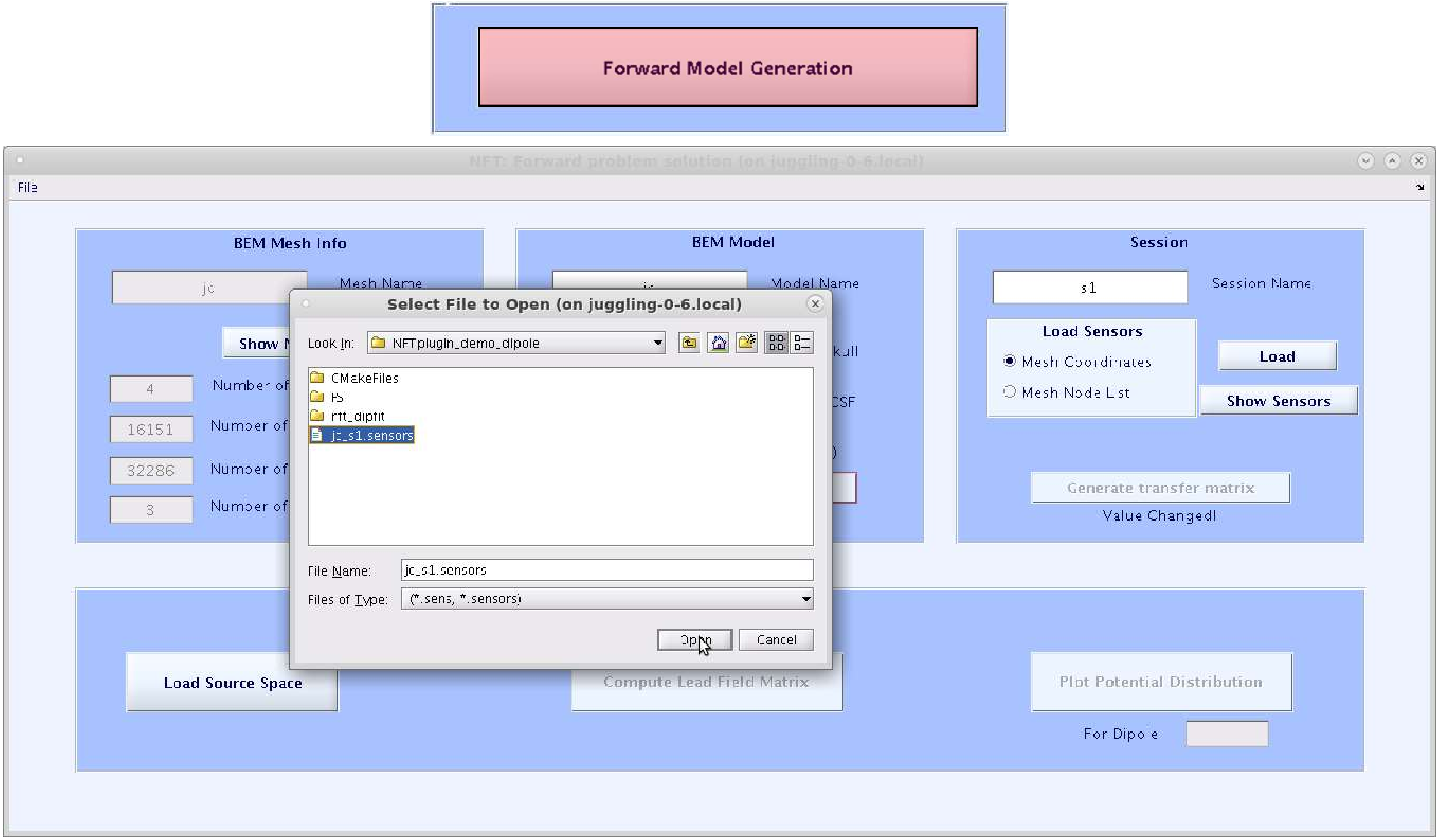
Forward Problem Solution with BEM



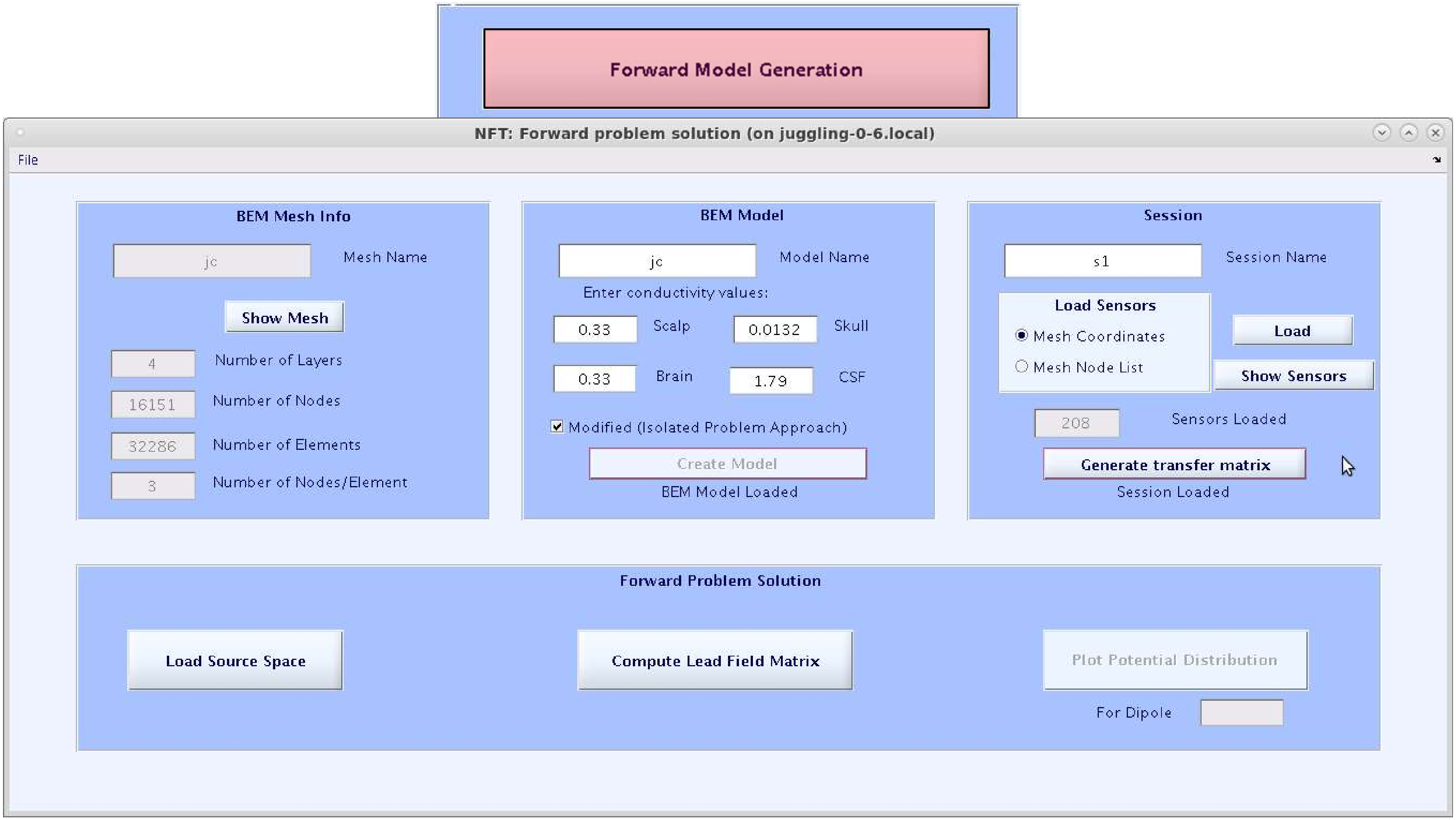


Forward Problem Solution with BEM

Forward Problem Solution with BEM

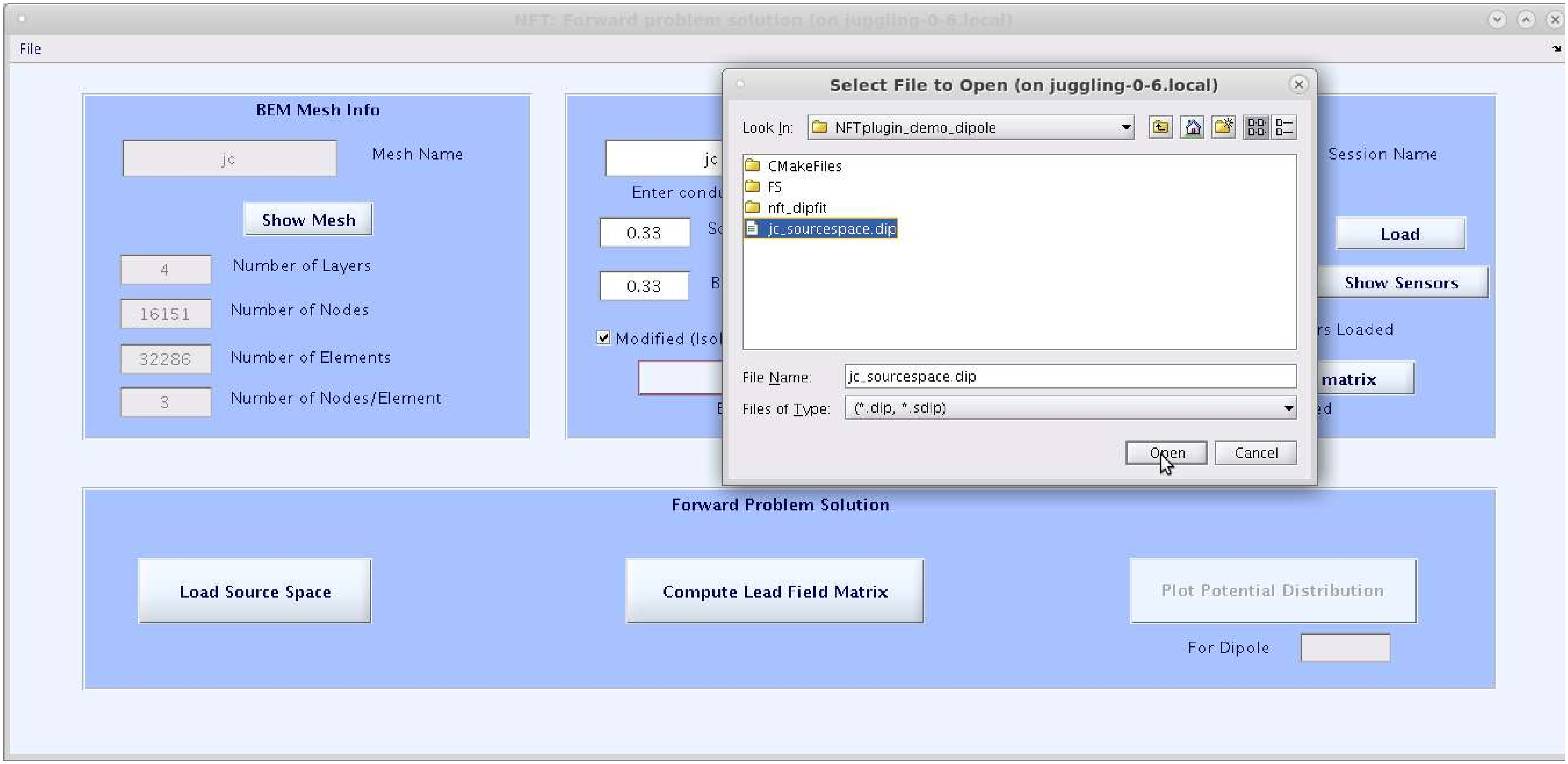


Forward Problem Solution with BEM

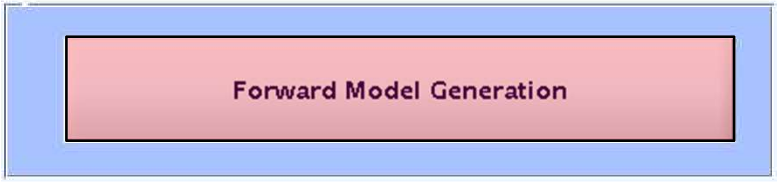


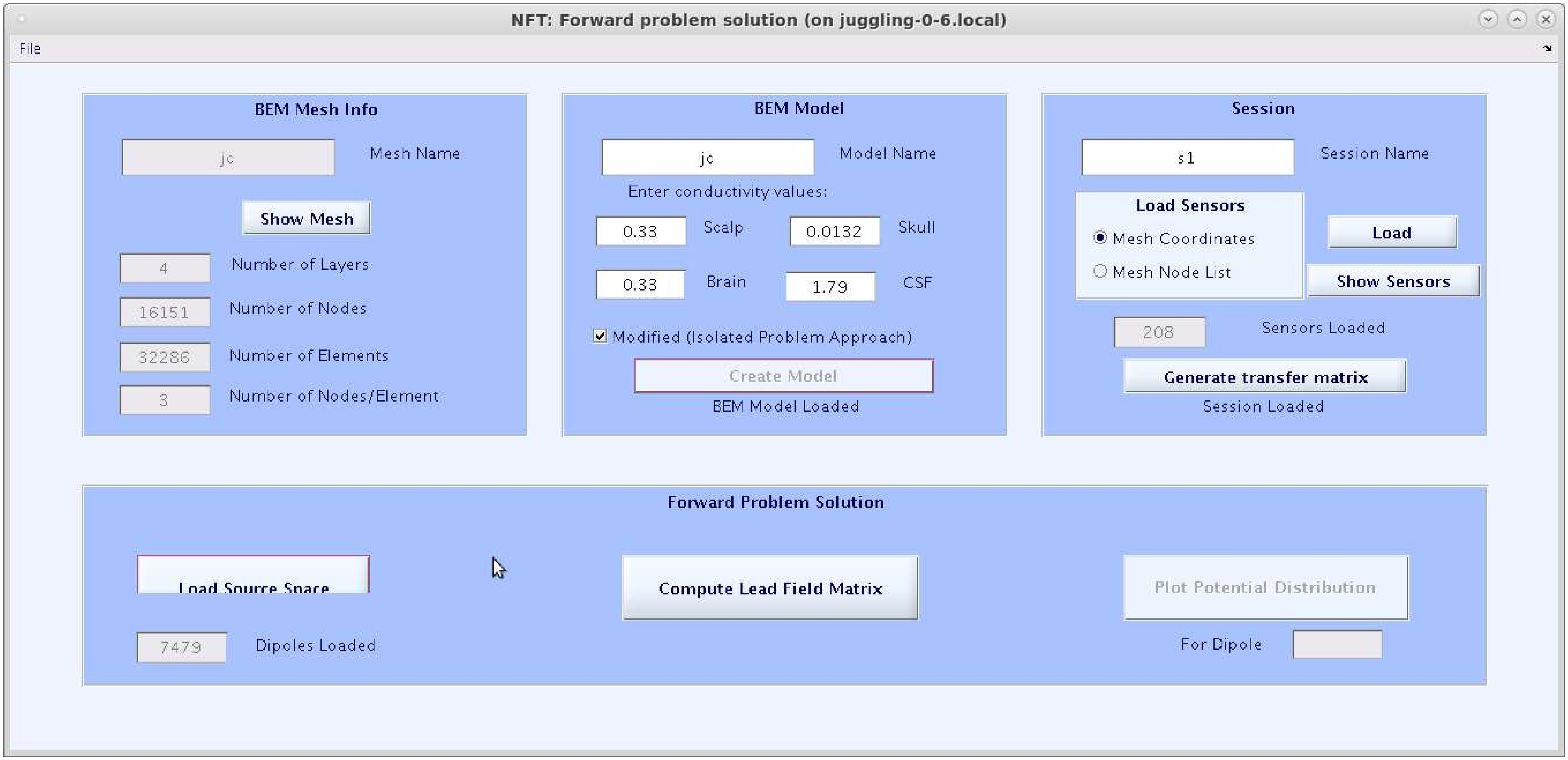
Forward Problem Solution with BEM

Forward Model Generation

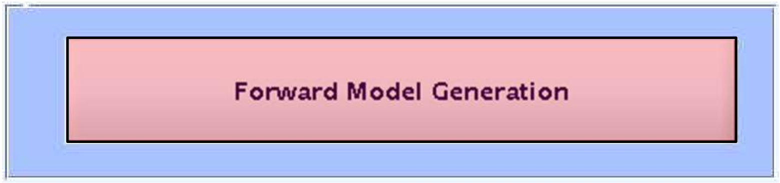


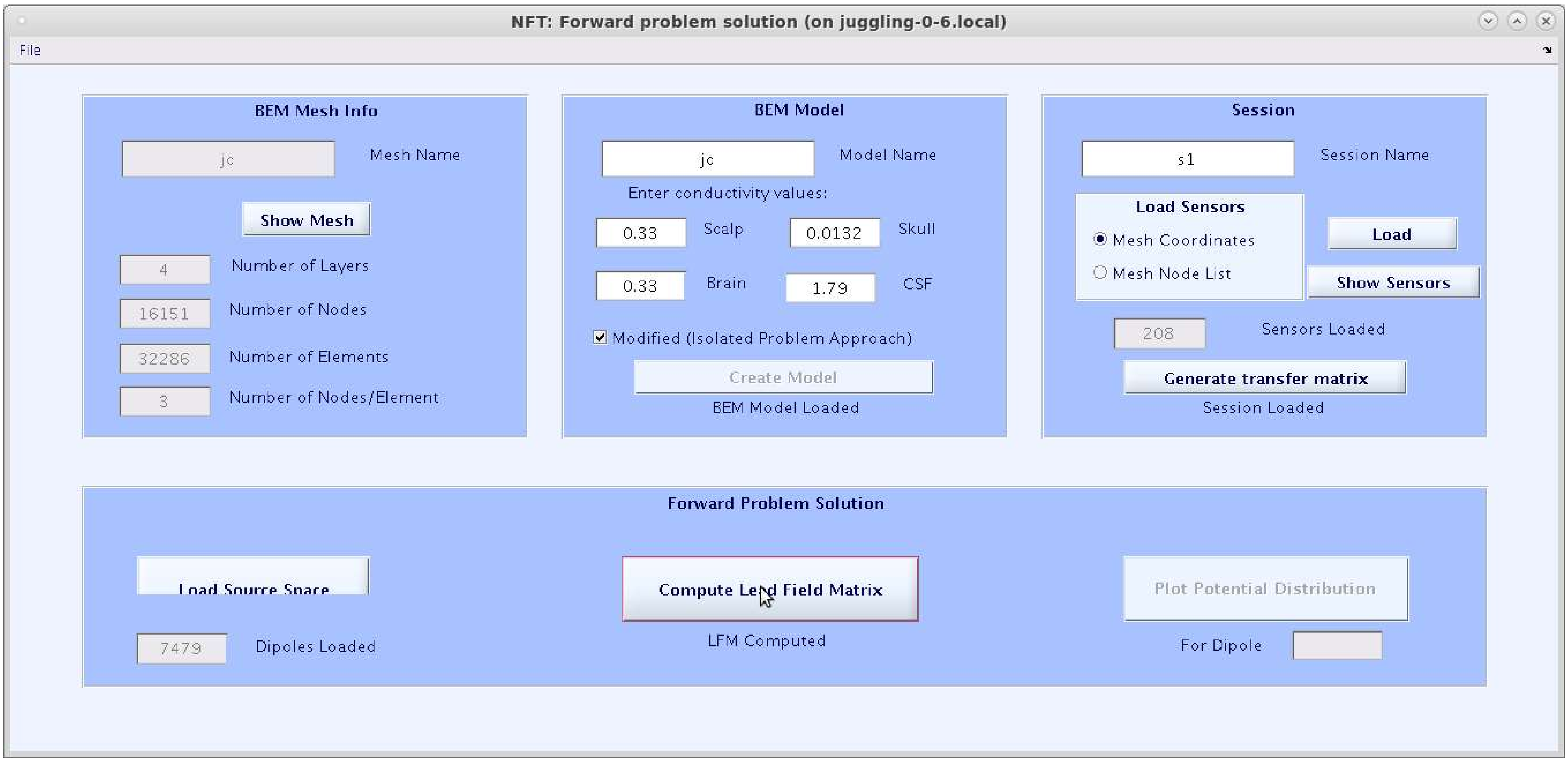
Forward Problem Solution with BEM



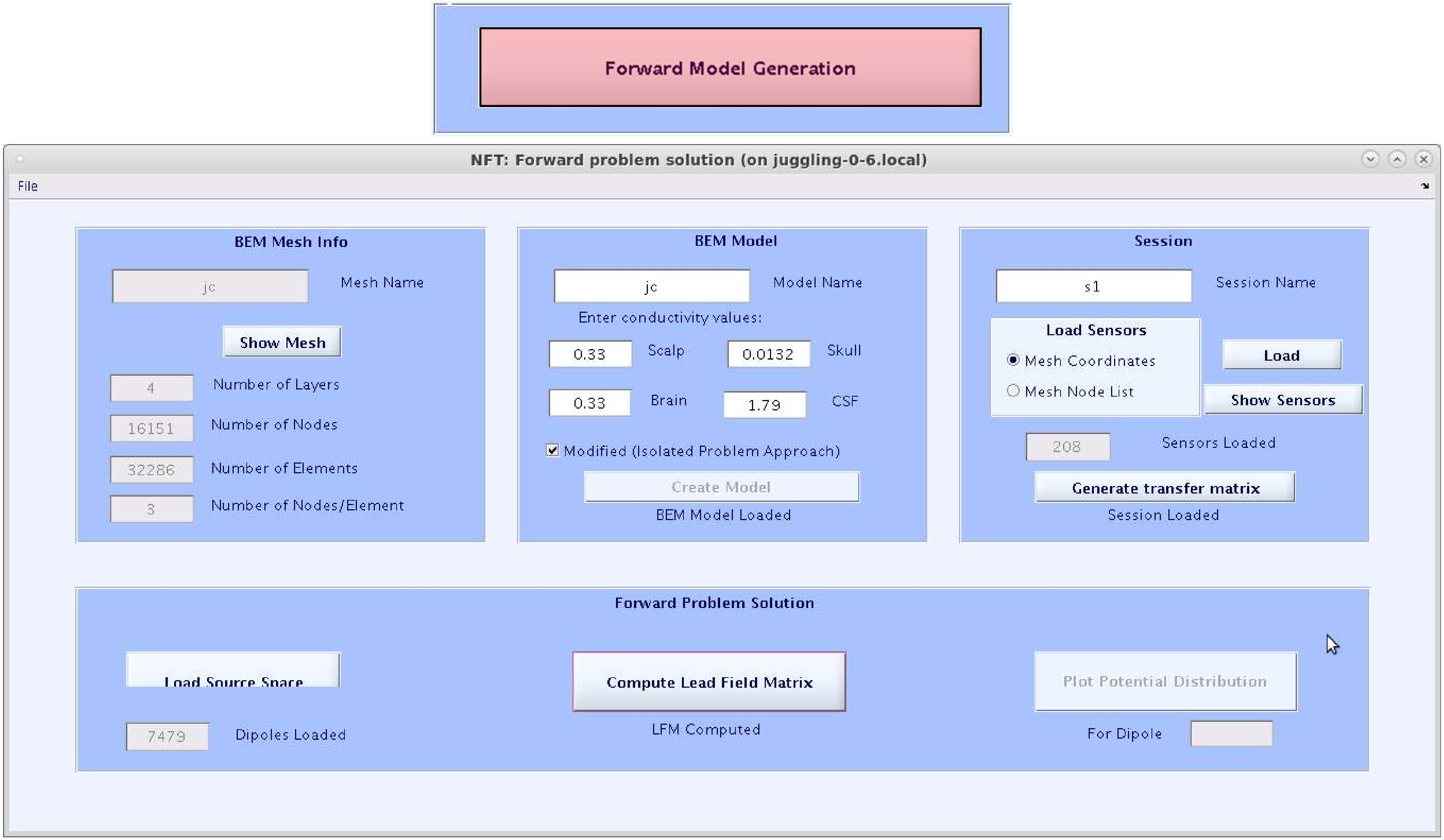


Forward Problem Solution with BEM

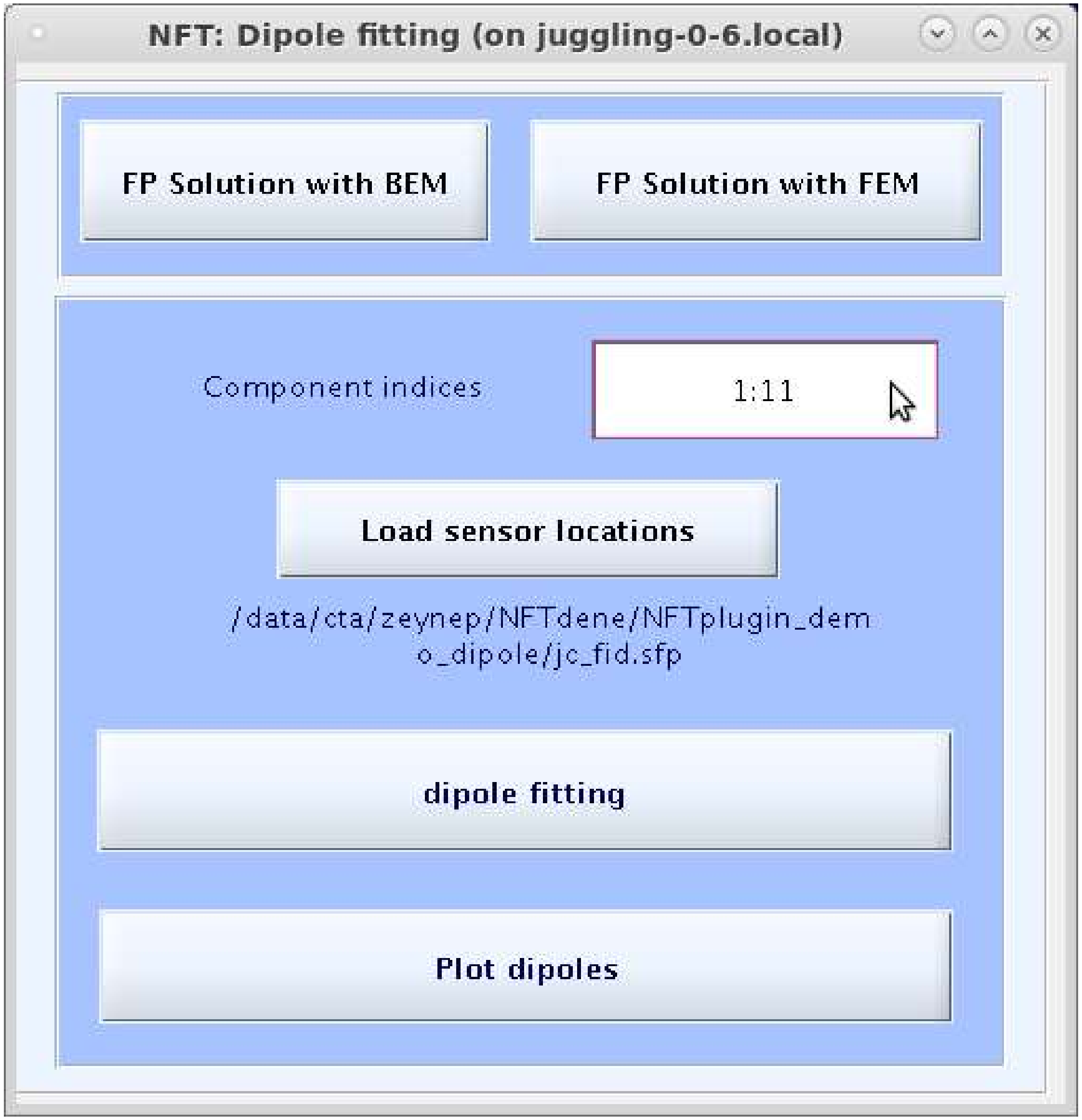




Forward Problem Solution with BEM



Inverse Problem Solution with BEM



Output

- Dipole source localization is saved in EEG structure, under EEG.etc.nft.
- After source localization with NFT, you can continue using EEGLAB;

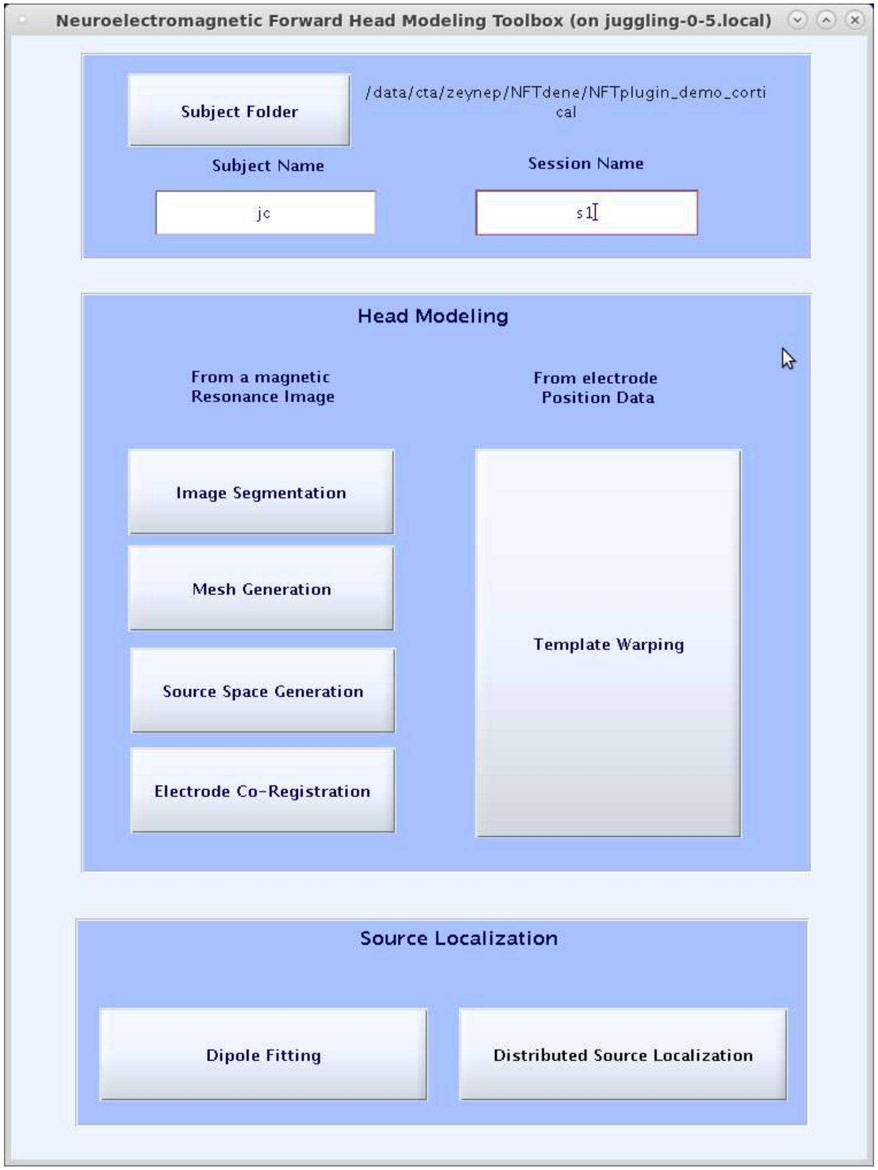
EEG.dipfit.model = EEG.etc.nft.model;

Distributed Source localization

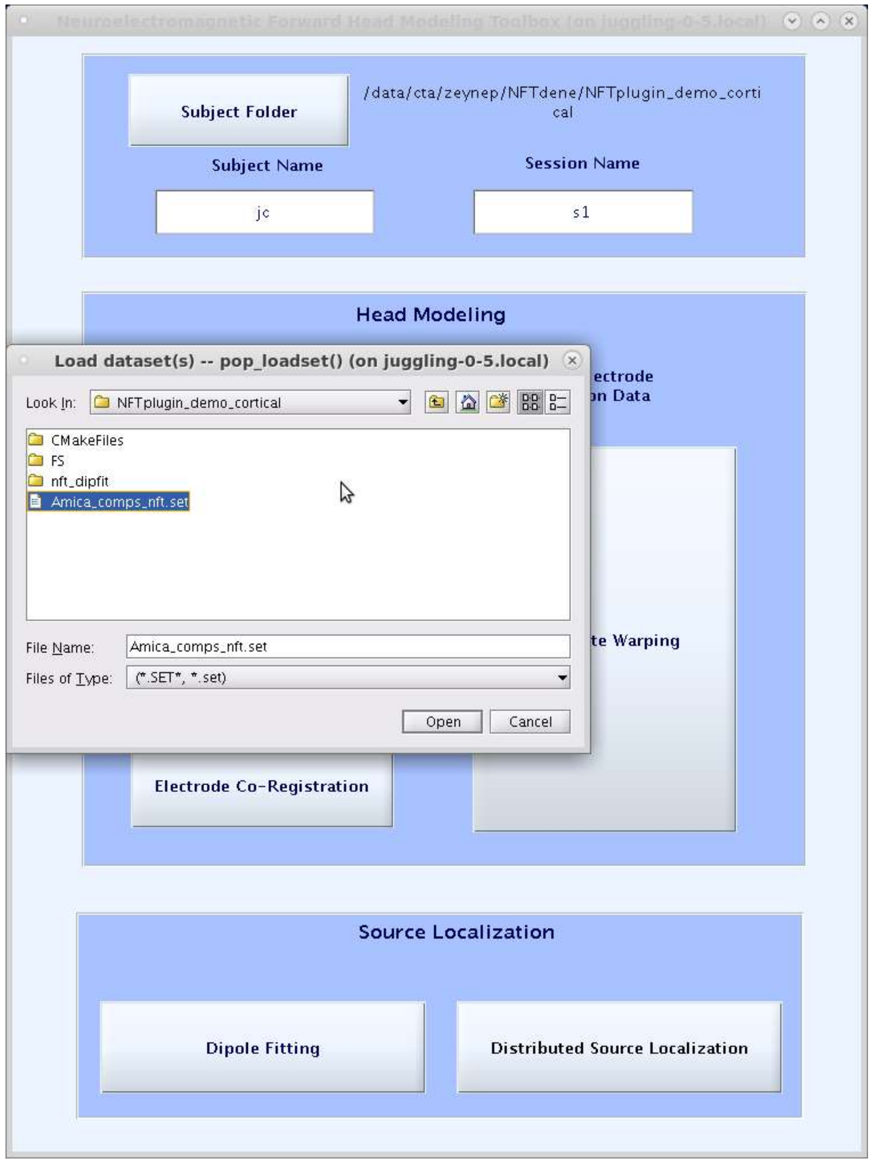
Go to the folder NFTplugin_demo_cortical

addpath

Distributed Source Localization



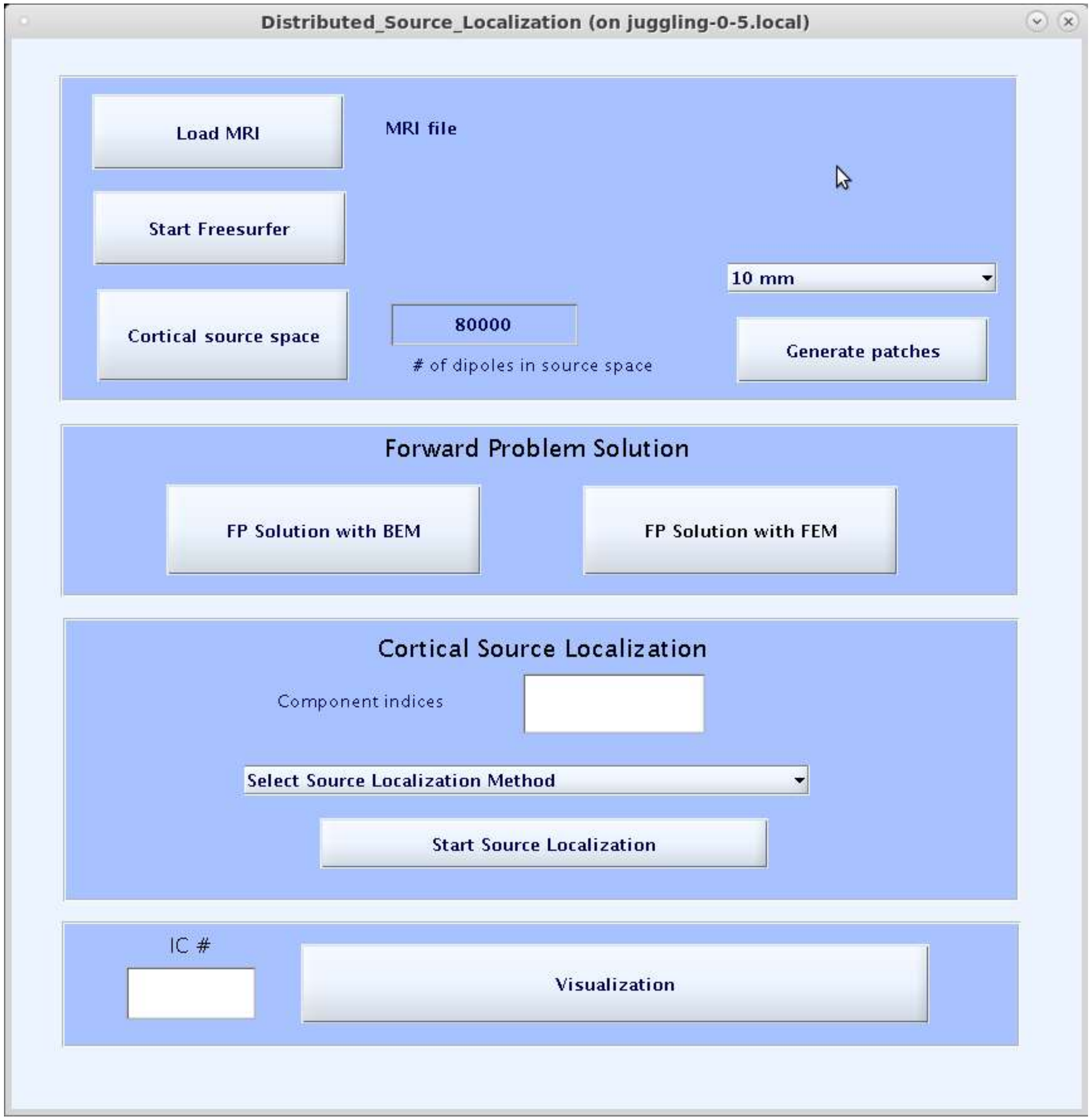
Select EEG data



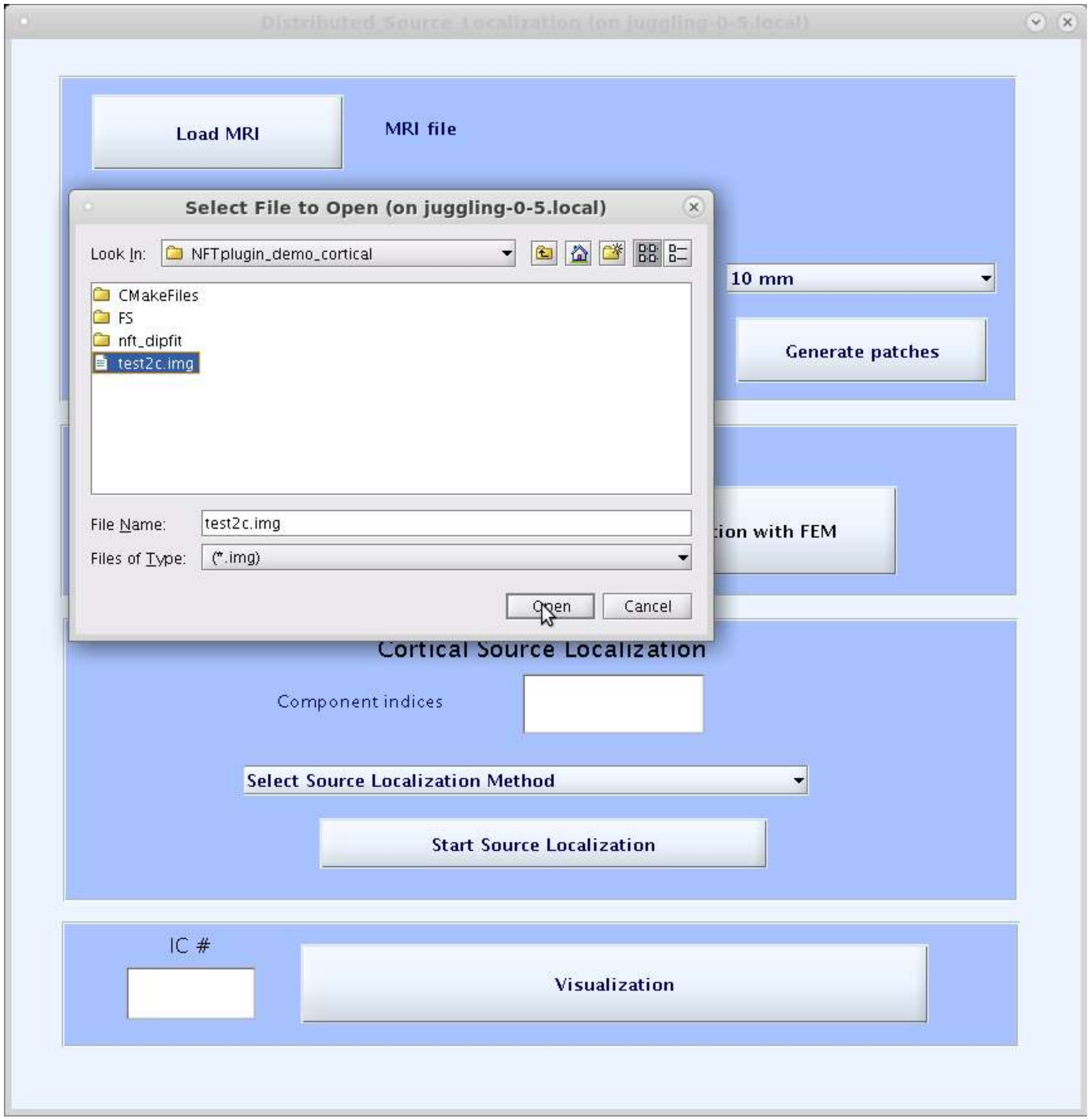
NIST – Generation of a cortical source space

- Load Freesurface cortical surface
- Downsample to 80,000 vertices
- Co-register with the NFT brain surface
- Re-generate NFT head model
- Calculate normals for each vertex on the cortical surface
- Save the cortical source space as: Subject_name FS_ss.dip
- Calculate node area of each vertex for source localization, save as Node_area
- Check if the sensor locations need to be updated according to the new NFT model.

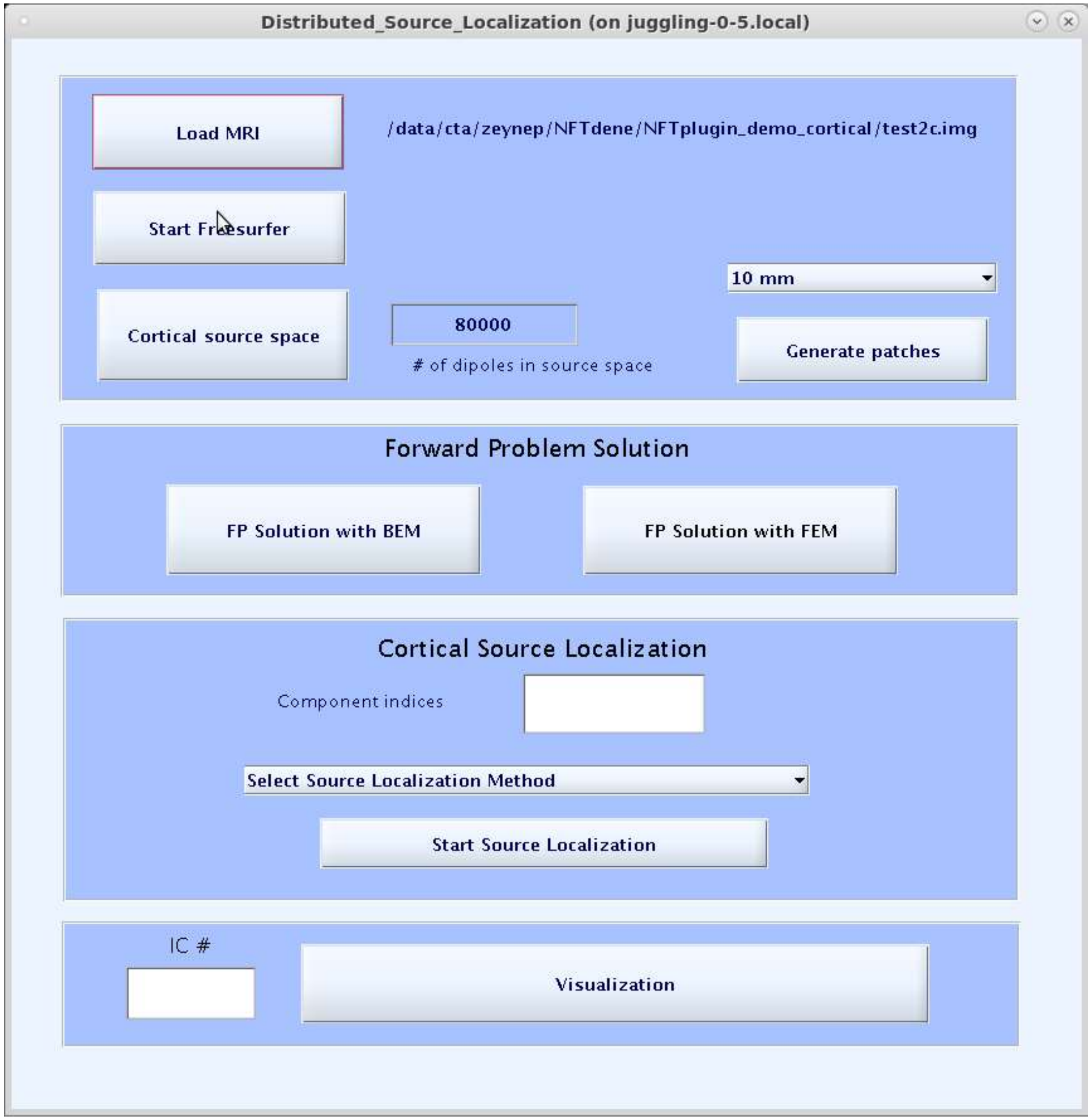
Distributed Source Localization

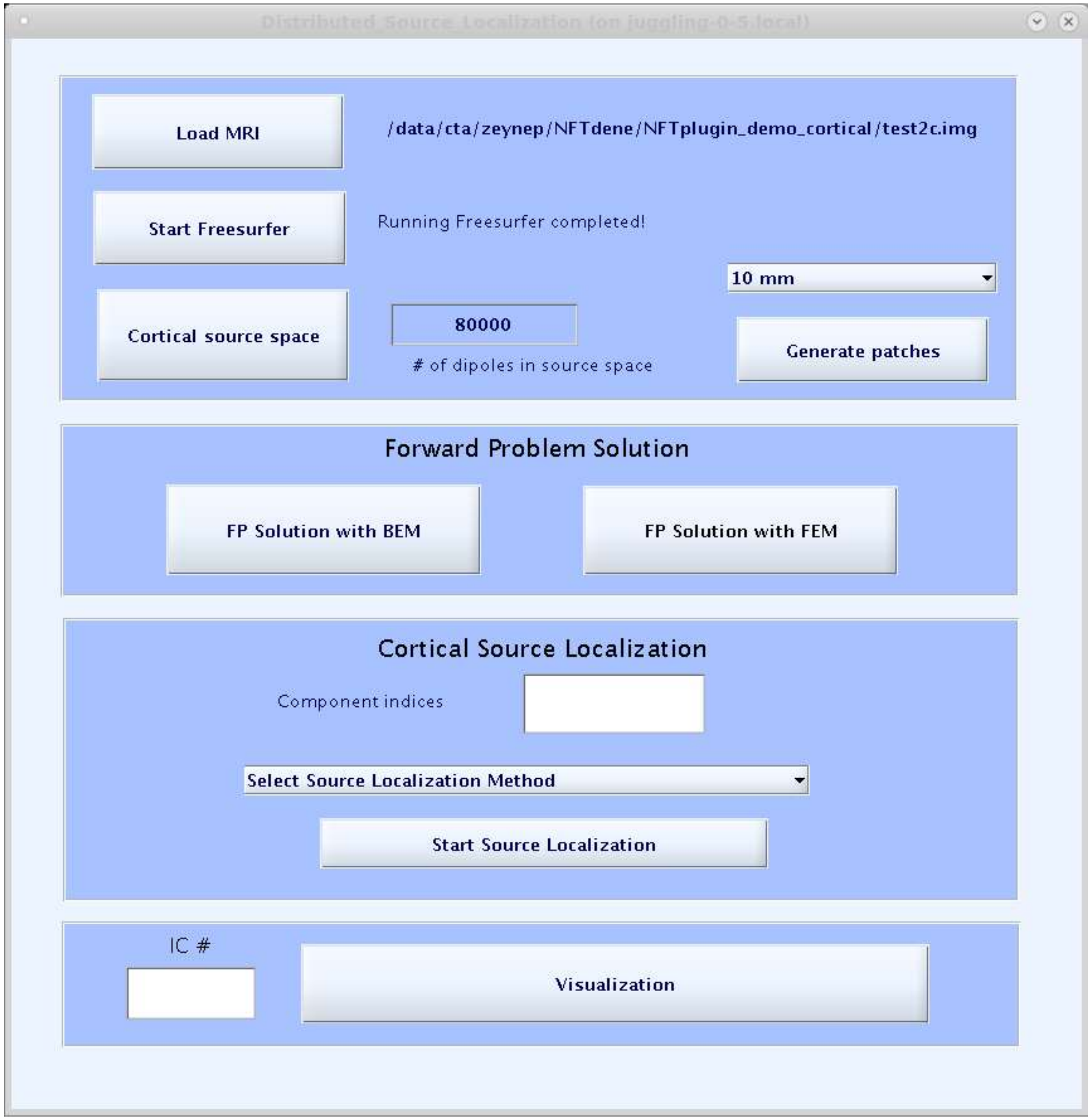


Load MRI

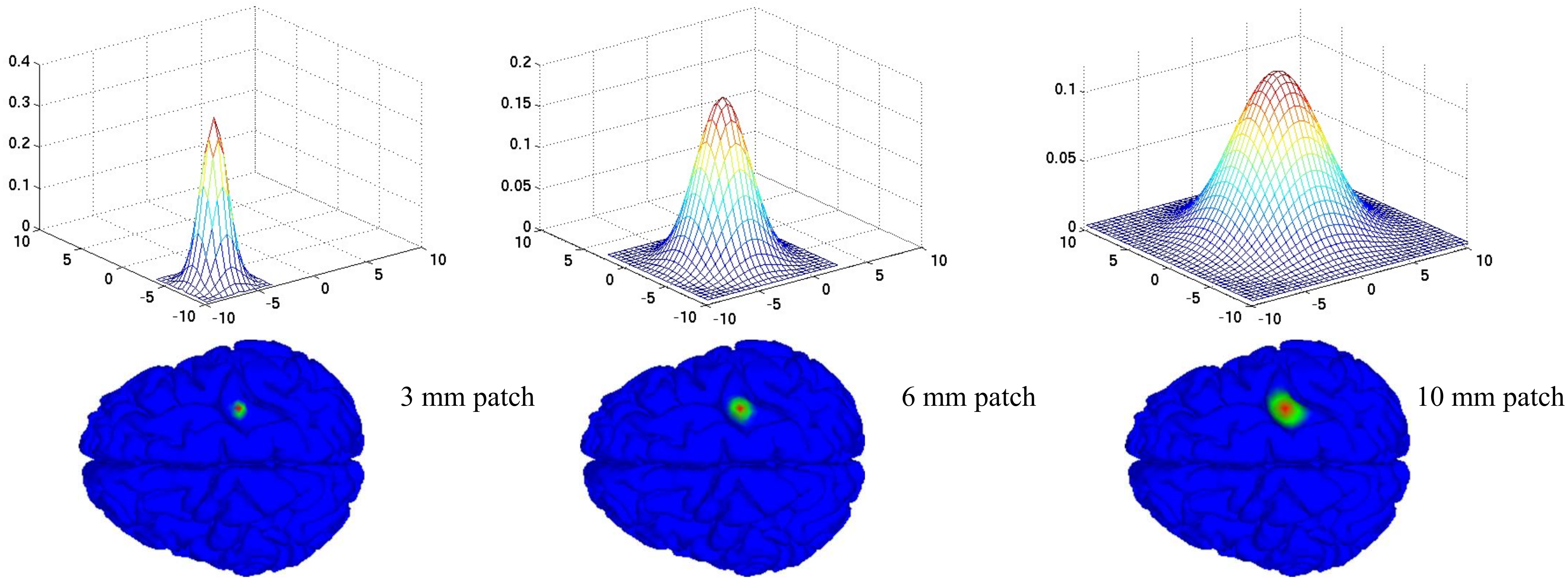

Distributed Source Localization (on juggling-0-5.local)
Load MRI
MRI file
Select File to Open (on juggling-0-5.local)
Look In:
NFTplugin_demo_cortical
CMakeFiles
FS
nft_dipfit
test2c.img
File Name:
test2c.img
Files of Type:
(*.img)
Open
Cancel
10 mm
Generate patches
tion with FEM
Cortical Source Localization
Component indices
Select Source Localization Method
Start Source Localization
IC #
Visualization

Run Freesurfer

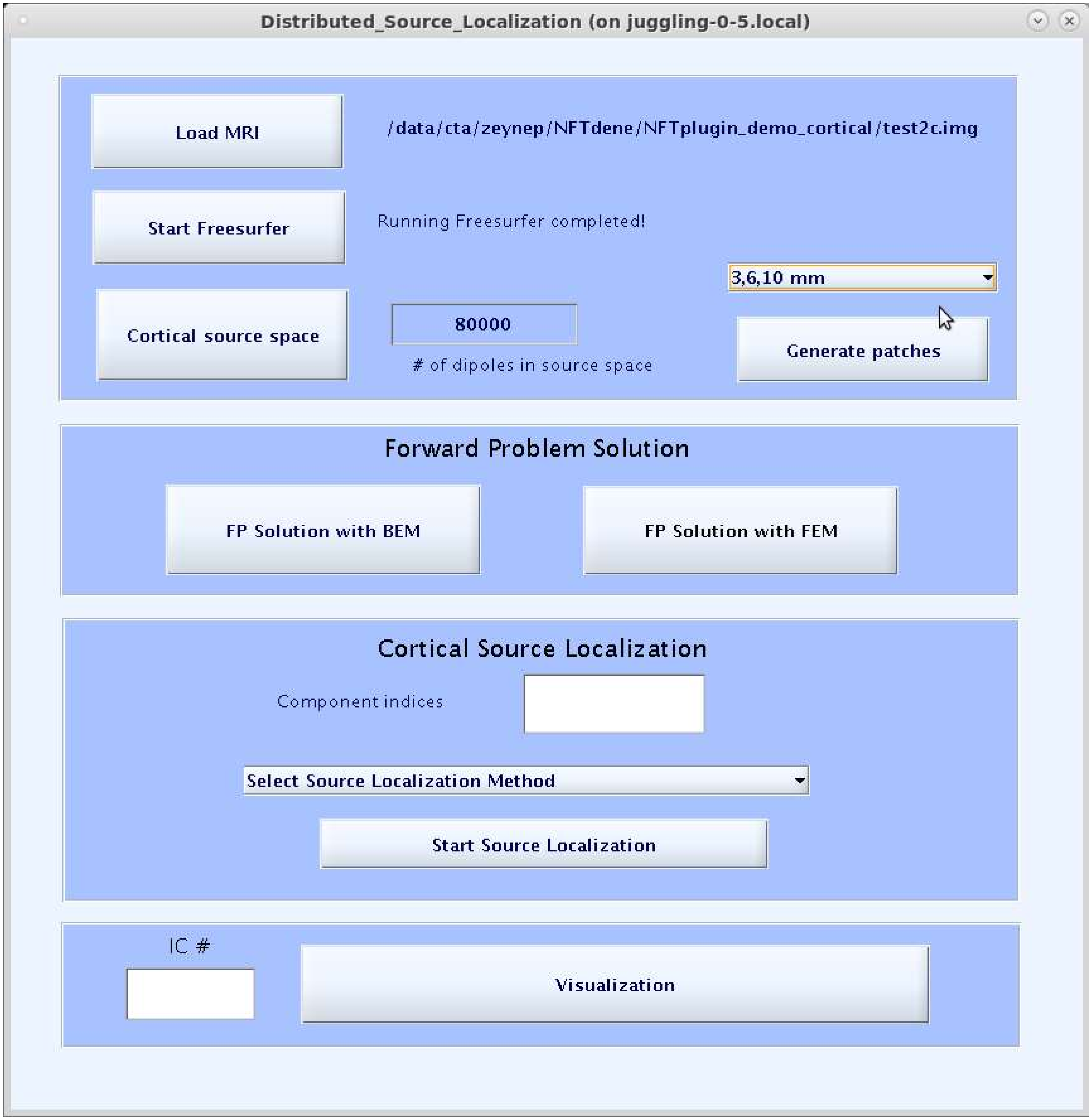



Distributed_Source_Localization (on juggling-0-5.local)
Load MRI
/data/cta/zeynep/NFTdene/NFTplugin_demo_cortical/test2c.img
Start Freesurfer
Running Freesurfer completed!
10 mm
Cortical source space
80000
# of dipoles in source space
Generate patches
Forward Problem Solution
FP Solution with BEM
FP Solution with FEM
Cortical Source Localization
Component indices
Select Source Localization Method
Start Source Localization
IC #
Visualization

NIST – Patch generation

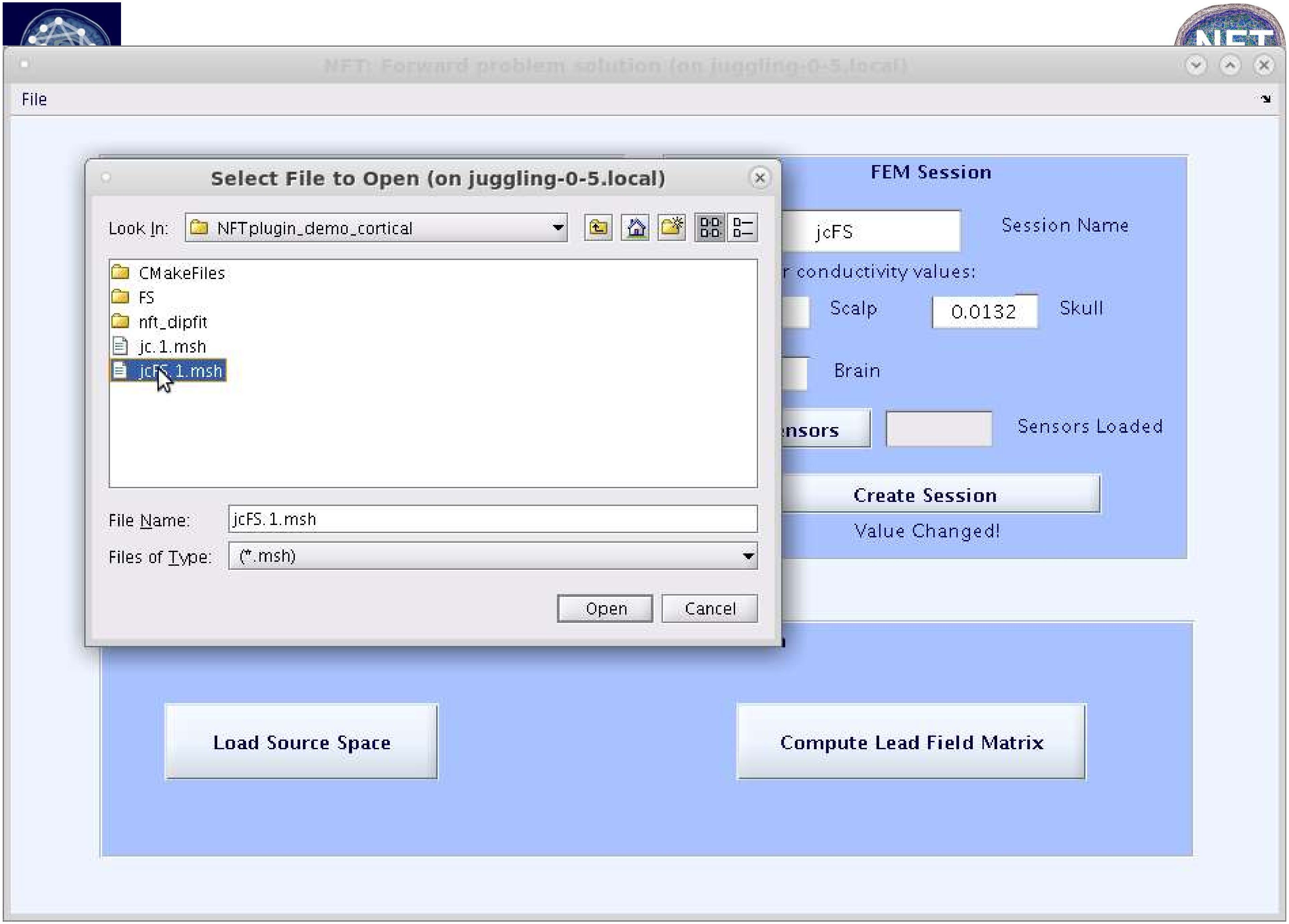


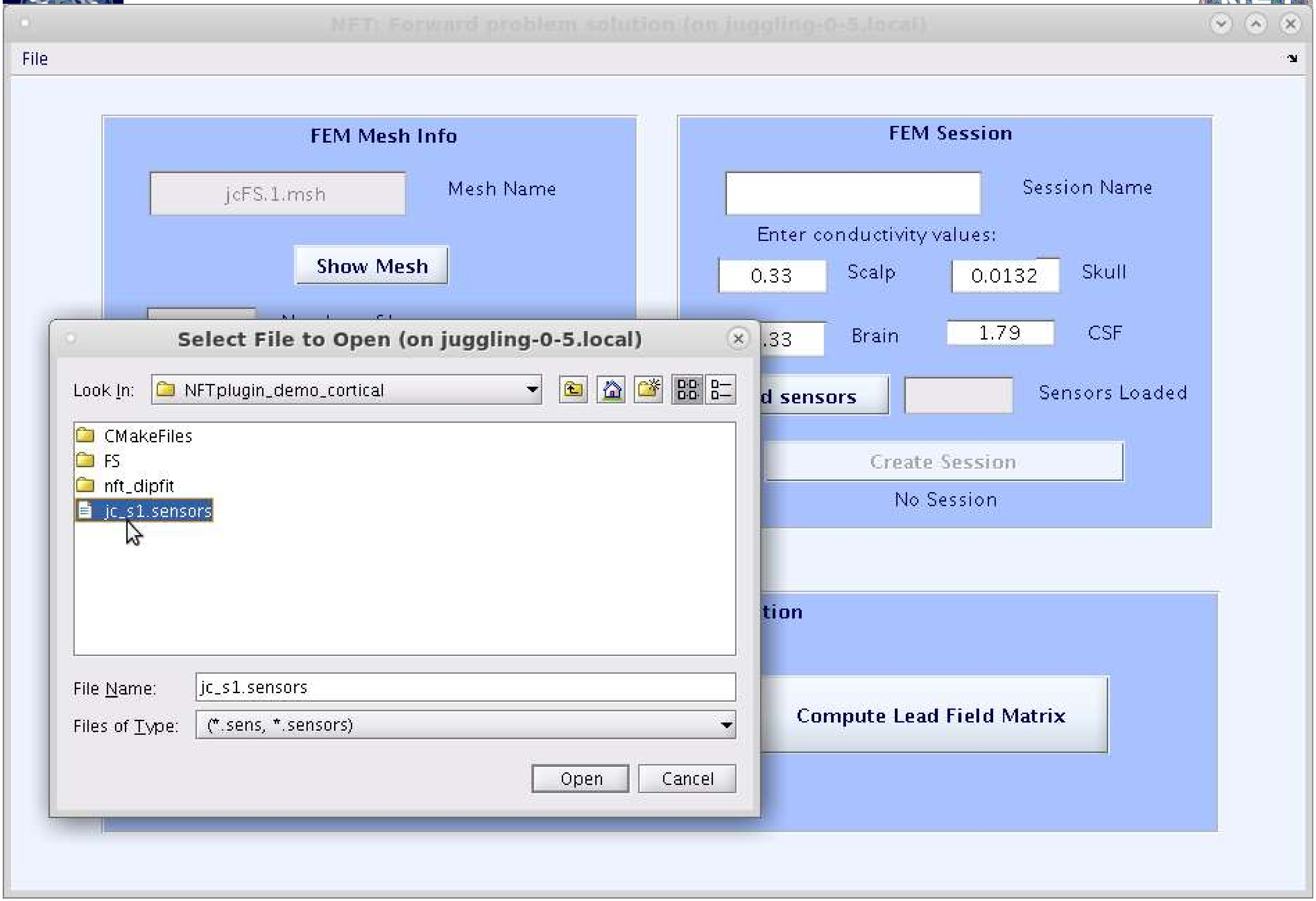
NIST has options to generate Gaussian patches with 10 mm, 6 mm, 3 mm in radius.

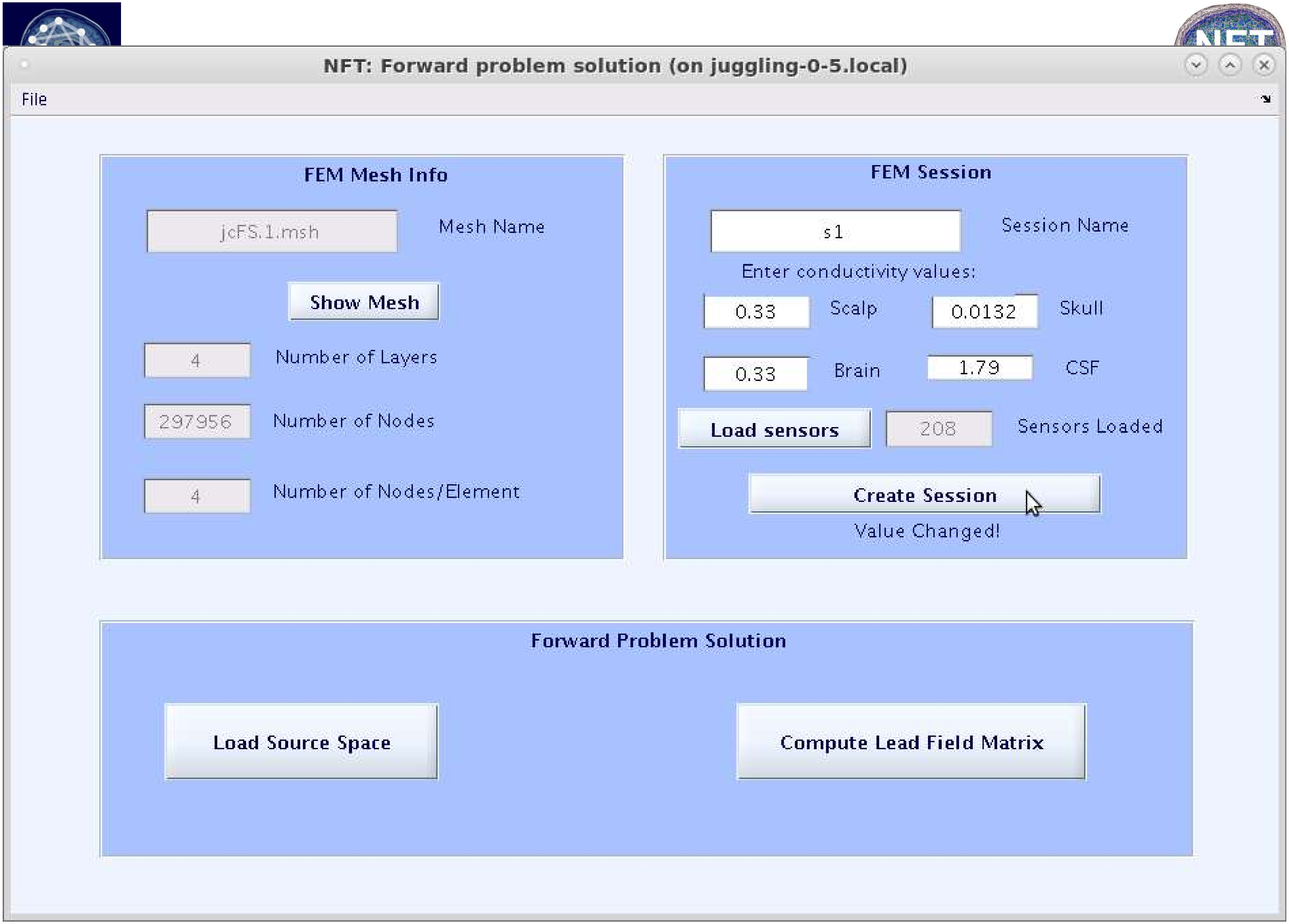

Distributed_Source_Localization (on juggling-0-5.local)
Load MRI
/data/cta/zeynep/NFTdene/NFTplugin_demo_cortical/test2c.img
Start Freesurfer
Running Freesurfer completed!
3,6,10 mm
Cortical source space
80000
# of dipoles in source space
Generate patches
Forward Problem Solution
FP Solution with BEM
FP Solution with FEM
Cortical Source Localization
Component indices
Select Source Localization Method
Start Source Localization
IC #
Visualization

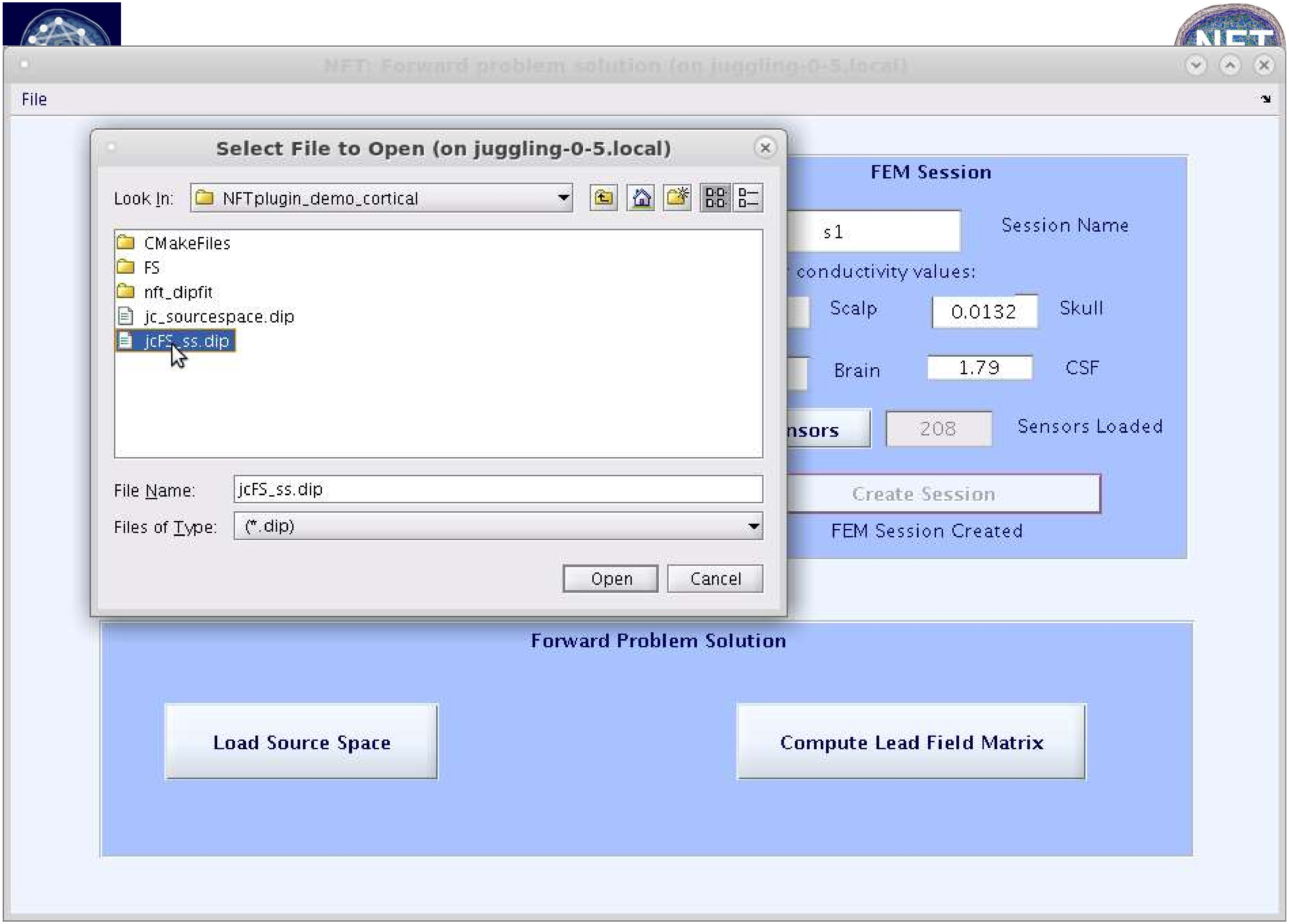
Forward Problem Solution with FEM

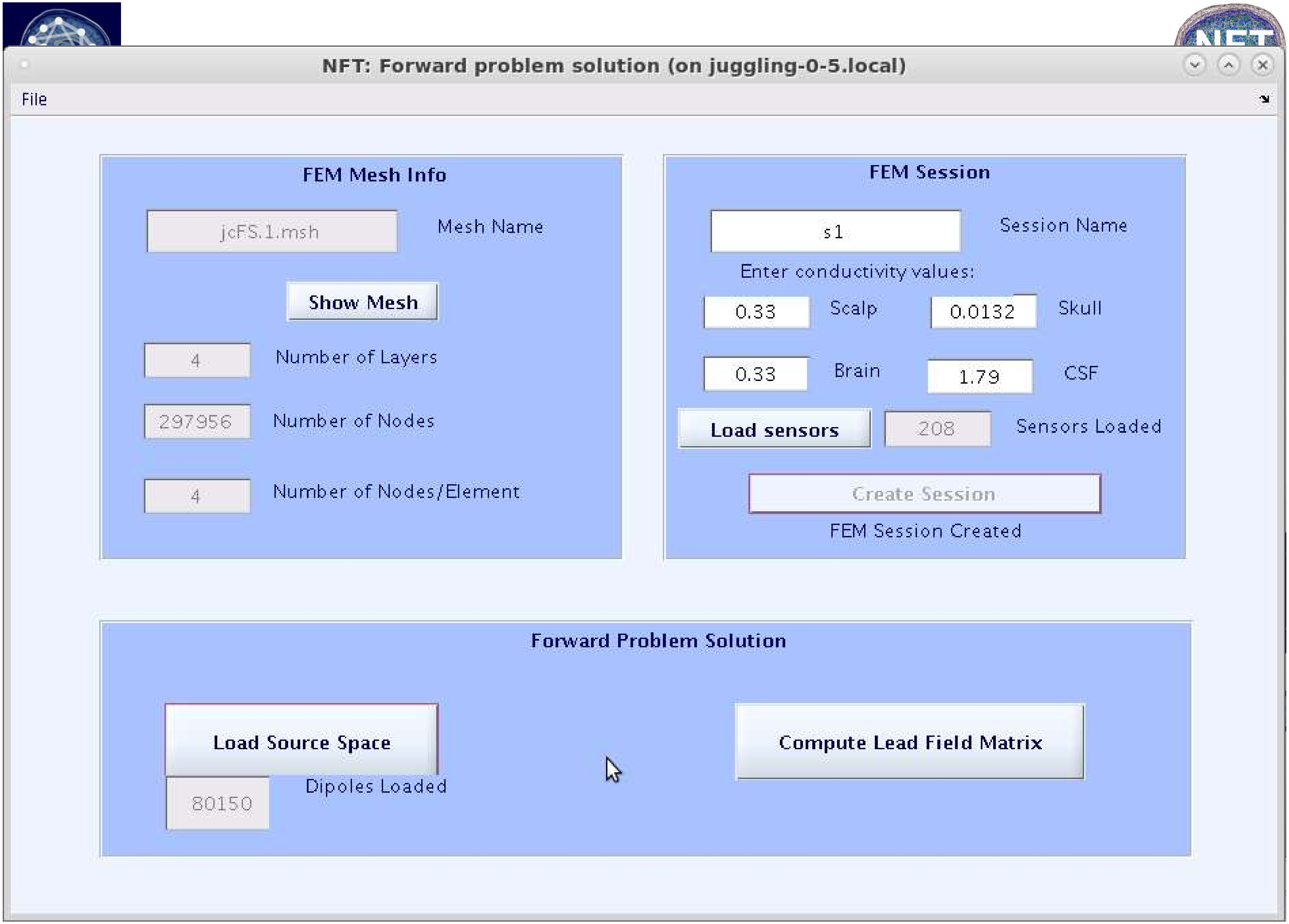
- Tetgen for mesh generation
  - Uses BEM meshes as boundaries
- METU-FEM to generate transfer matrix
  - Compiled from source
  - Requires PETSc for matrix operations
- metufem .mex file for forward solutions in MATLAB
- Instructions available under README.FEM file.

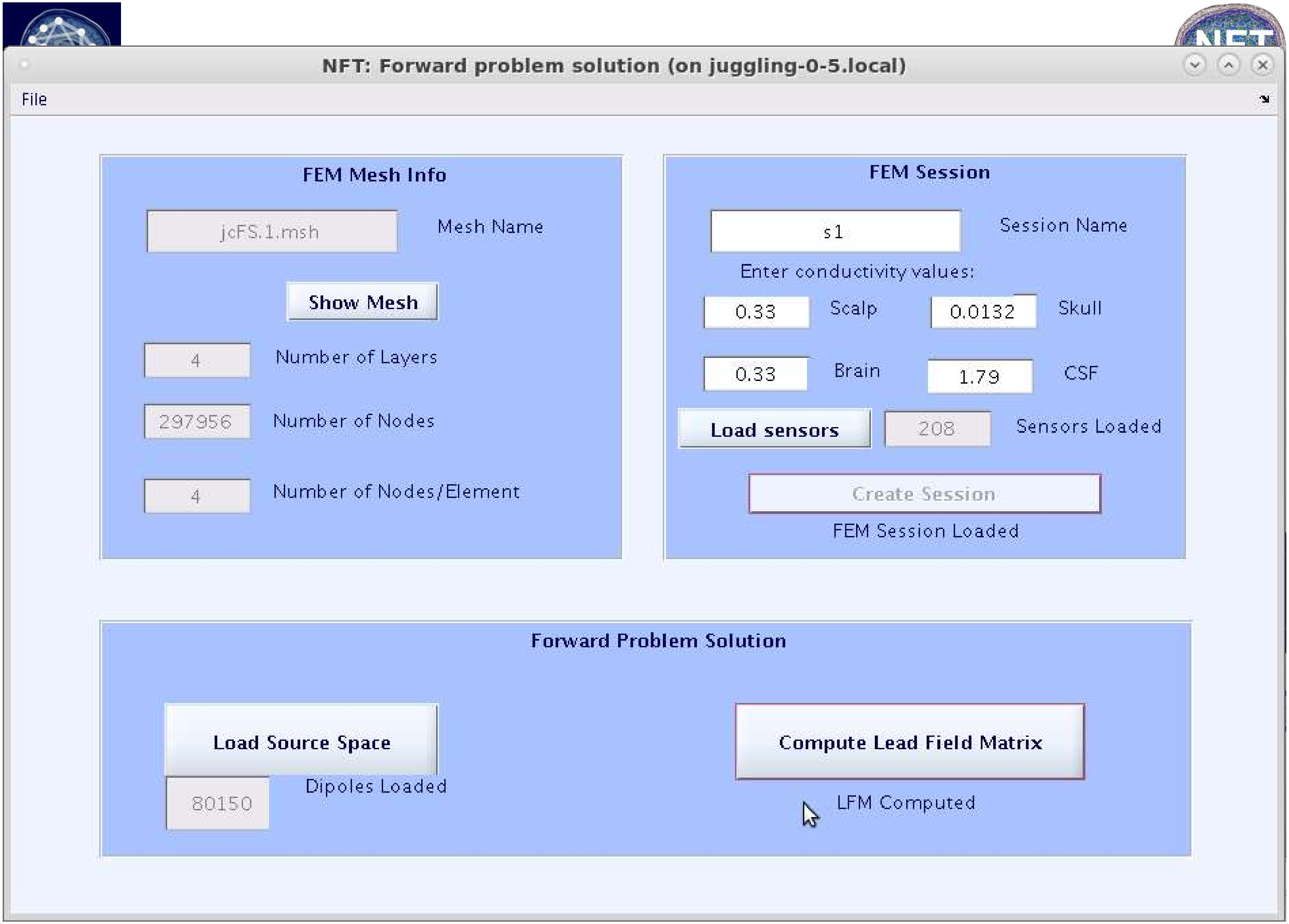

NFT: Forward problem solution (on juggling-0-5.local)
File
Select File to Open (on juggling-0-5.local)
Look In:
NFTplugin_demo_cortical
CMakeFiles
FS
nft_dipfit
jc.1.msh
jcFS.1.msh
File Name:
jcFS.1.msh
Files of Type:
(*.msh)
Open
Cancel
FEM Session
jcFS
Session Name
r conductivity values:
Scalp
0.0132
Skull
Brain
nsors
Sensors Loaded
Create Session
Value Changed!
Load Source Space
Compute Lead Field Matrix

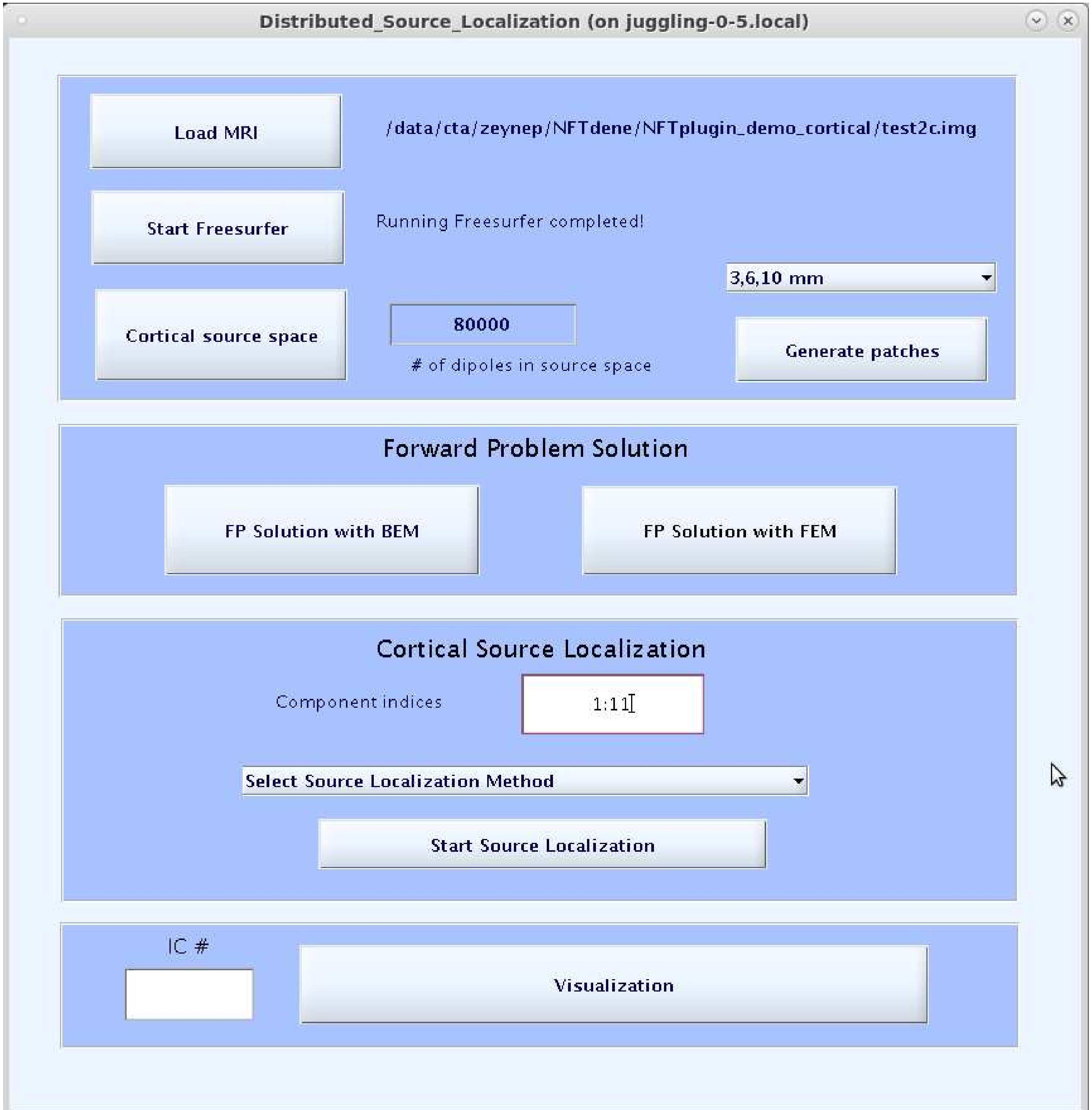

NFT: Forward problem solution (on juggling-0-5.local)
File
FEM Mesh Info
jcFS.1.msh
Mesh Name
Show Mesh
FEM Session
Session Name
Enter conductivity values:
0.33
Scalp
0.0132
Skull
.33
Brain
1.79
CSF
d sensors
Sensors Loaded
Create Session
No Session
tion
Compute Lead Field Matrix
Select File to Open (on juggling-0-5.local)
Look In:
NFTplugin_demo_cortical
CMakeFiles
FS
nft_dipfit
jc_s1.sensors
File Name:
jc_s1.sensors
Files of Type:
(*.sens, *.sensors)
Open
Cancel

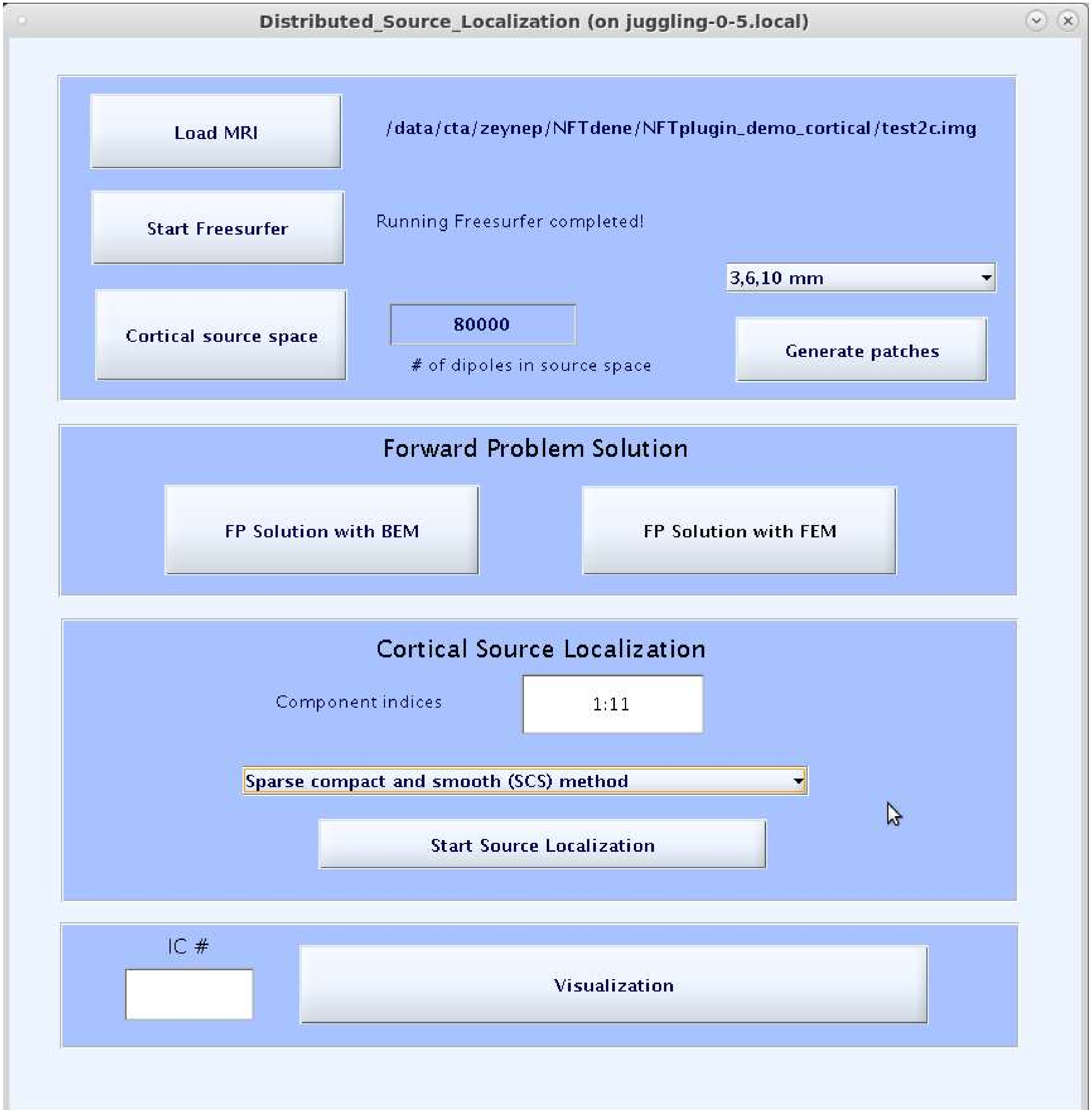

NFT: Forward problem solution (on juggling-0-5.local)
File
FEM Mesh Info
jcFS.1.msh
Mesh Name
Show Mesh
4
Number of Layers
297956
Number of Nodes
4
Number of Nodes/Element
FEM Session
s1
Session Name
Enter conductivity values:
0.33
Scalp
0.0132
Skull
0.33
Brain
1.79
CSF
Load sensors
208
Sensors Loaded
Create Session
Value Changed!
Forward Problem Solution
Load Source Space
Compute Lead Field Matrix

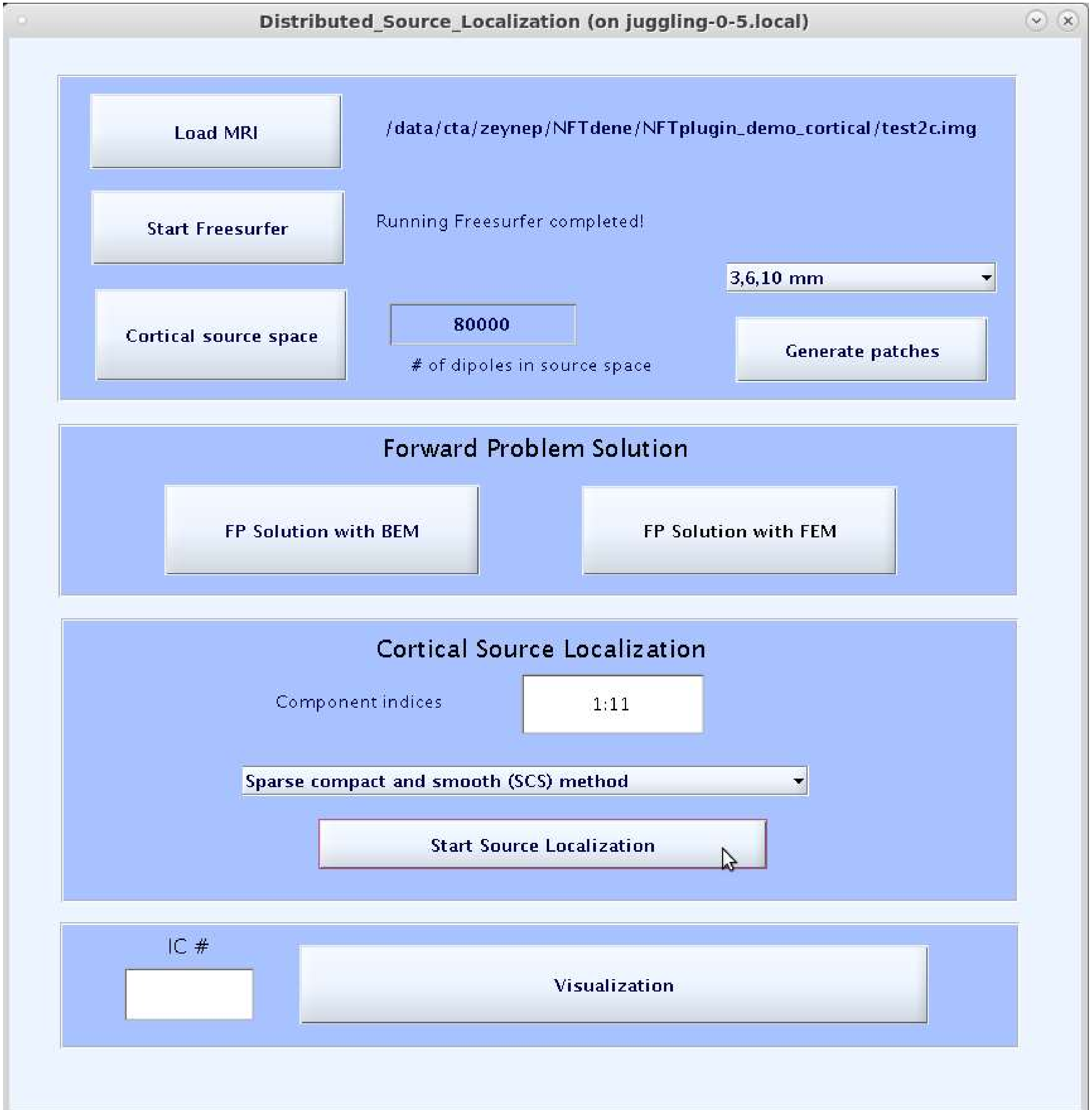

NFT: Forward problem solution (on juggling-0-5.local)
File
Select File to Open (on juggling-0-5.local)
Look In: NFTplugin_demo_cortical
CMakeFiles
FS
nft_dipfit
jc_sourcespace.dip
jcFS_ss.dip
File Name: jcFS_ss.dip
Files of Type: (*.dip)
Open
Cancel
FEM Session
s1
Session Name
conductivity values:
Scalp
0.0132
Skull
Brain
1.79
CSF
nsors
208
Sensors Loaded
Create Session
FEM Session Created
Forward Problem Solution
Load Source Space
Compute Lead Field Matrix


NFT: Forward problem solution (on juggling-0-5.local)
File
FEM Mesh Info
jcFS.1.msh
Mesh Name
Show Mesh
4
Number of Layers
297956
Number of Nodes
4
Number of Nodes/Element
FEM Session
s1
Session Name
Enter conductivity values:
0.33
Scalp
0.0132
Skull
0.33
Brain
1.79
CSF
Load sensors
208
Sensors Loaded
Create Session
FEM Session Created
Forward Problem Solution
Load Source Space
80150
Dipoles Loaded
Compute Lead Field Matrix

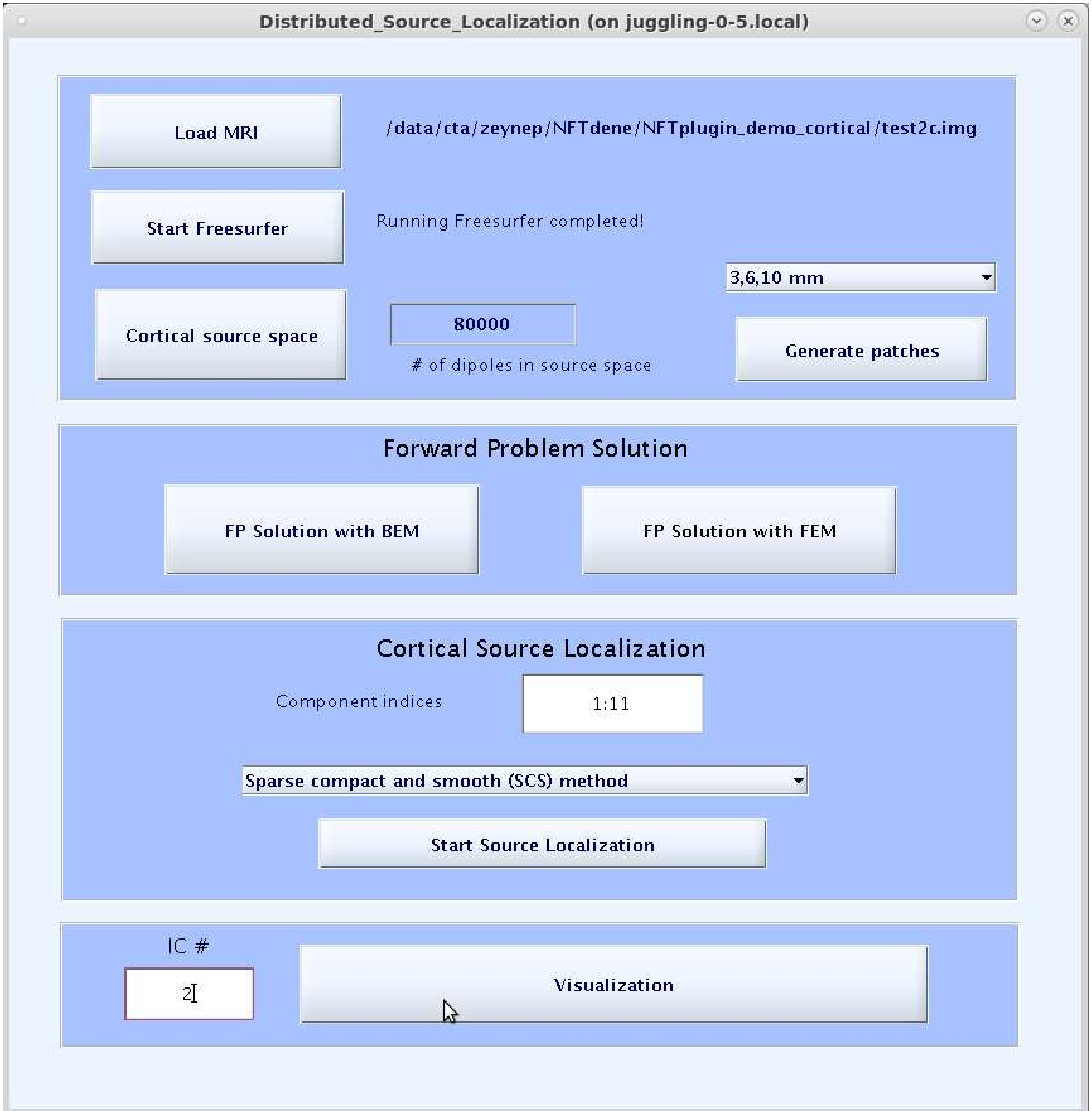

NFT: Forward problem solution (on juggling-0-5.local)
File
FEM Mesh Info
jcFS.1.msh
Mesh Name
Show Mesh
4
Number of Layers
297956
Number of Nodes
4
Number of Nodes/Element
FEM Session
s1
Session Name
Enter conductivity values:
0.33
Scalp
0.0132
Skull
0.33
Brain
1.79
CSF
Load sensors
208
Sensors Loaded
Create Session
FEM Session Loaded
Forward Problem Solution
Load Source Space
80150
Dipoles Loaded
Compute Lead Field Matrix
LFM Computed


Distributed_Source_Localization (on juggling-0-5.local)
Load MRI
/data/cta/zeynep/NFTdene/NFTplugin_demo_cortical/test2c.img
Start Freesurfer
Running Freesurfer completed!
3,6,10 mm
Cortical source space
80000
# of dipoles in source space
Generate patches
Forward Problem Solution
FP Solution with BEM
FP Solution with FEM
Cortical Source Localization
Component indices
1:11
Select Source Localization Method
Start Source Localization
IC #
Visualization


Distributed_Source_Localization (on juggling-0-5.local)
Load MRI
/data/cta/zeynep/NFTdene/NFTplugin_demo_cortical/test2c.img
Start Freesurfer
Running Freesurfer completed!
3,6,10 mm
Cortical source space
80000
# of dipoles in source space
Generate patches
Forward Problem Solution
FP Solution with BEM
FP Solution with FEM
Cortical Source Localization
Component indices
1:11
Sparse compact and smooth (SCS) method
Start Source Localization
IC #
Visualization



Distributed_Source_Localization (on juggling-0-5.local)
Load MRI
/data/cta/zeynep/NFTdene/NFTplugin_demo_cortical/test2c.img
Start Freesurfer
Running Freesurfer completed!
3,6,10 mm
Cortical source space
80000
# of dipoles in source space
Generate patches
Forward Problem Solution
FP Solution with BEM
FP Solution with FEM
Cortical Source Localization
Component indices
1:11
Sparse compact and smooth (SCS) method
Start Source Localization
IC #
Visualization


Distributed_Source_Localization (on juggling-0-5.local)
Load MRI
/data/cta/zeynep/NFTdene/NFTplugin_demo_cortical/test2c.img
Start Freesurfer
Running Freesurfer completed!
3,6,10 mm
Cortical source space
80000
# of dipoles in source space
Generate patches
Forward Problem Solution
FP Solution with BEM
FP Solution with FEM
Cortical Source Localization
Component indices
1:11
Sparse compact and smooth (SCS) method
Start Source Localization
IC #
2
Visualization

Output

- Source estimates are saved as cortex_source_scs and/or cortex_source_sbl

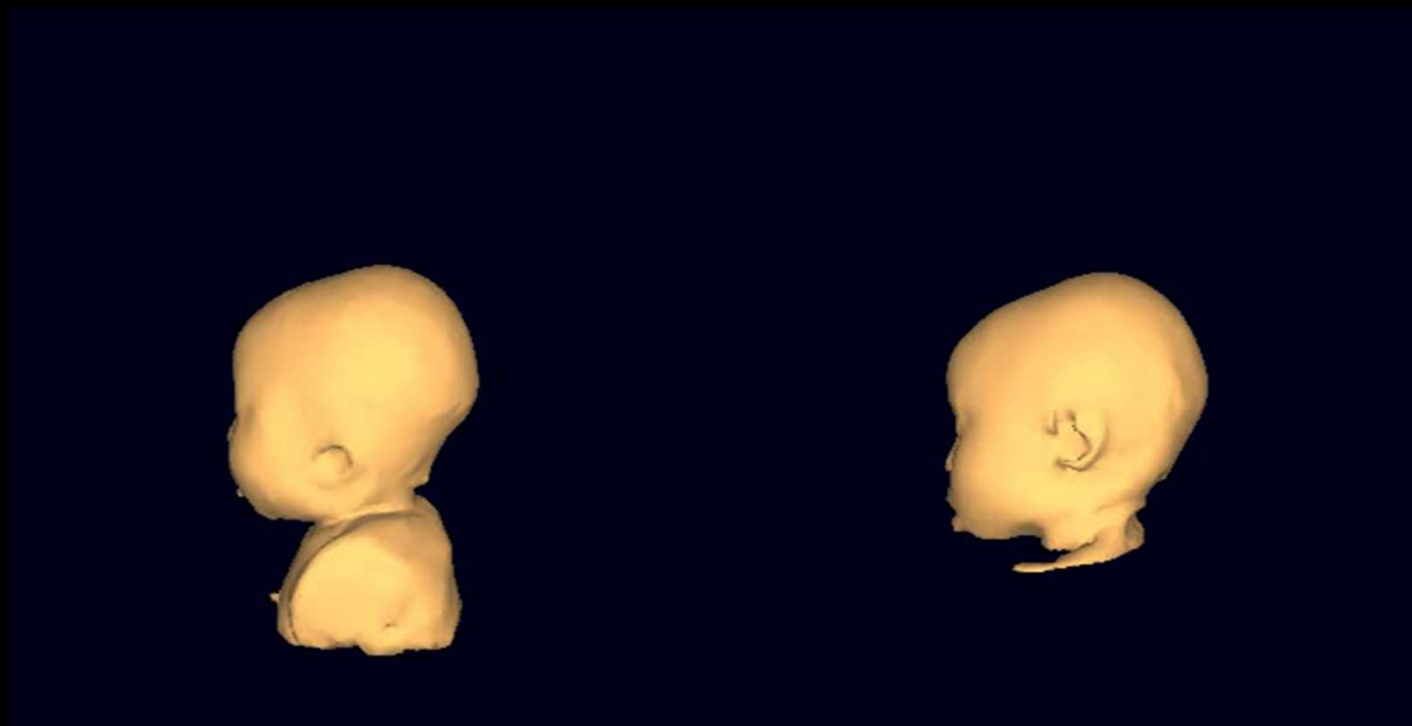
Child Head modeling

- Segmentation of infant/child head into scalp, skull, CSF, and brain tissues.
- Electrical head mesh generation using NFT, making possible
- Non-invasive conductivity estimation of major head tissues, and also
- Making available accurate, age-specific developmental template head models

Generation of individual head models

NFT (sccn.ucsd.edu/nft/) was used to generate four-layer Finite Element (FEM) head models.

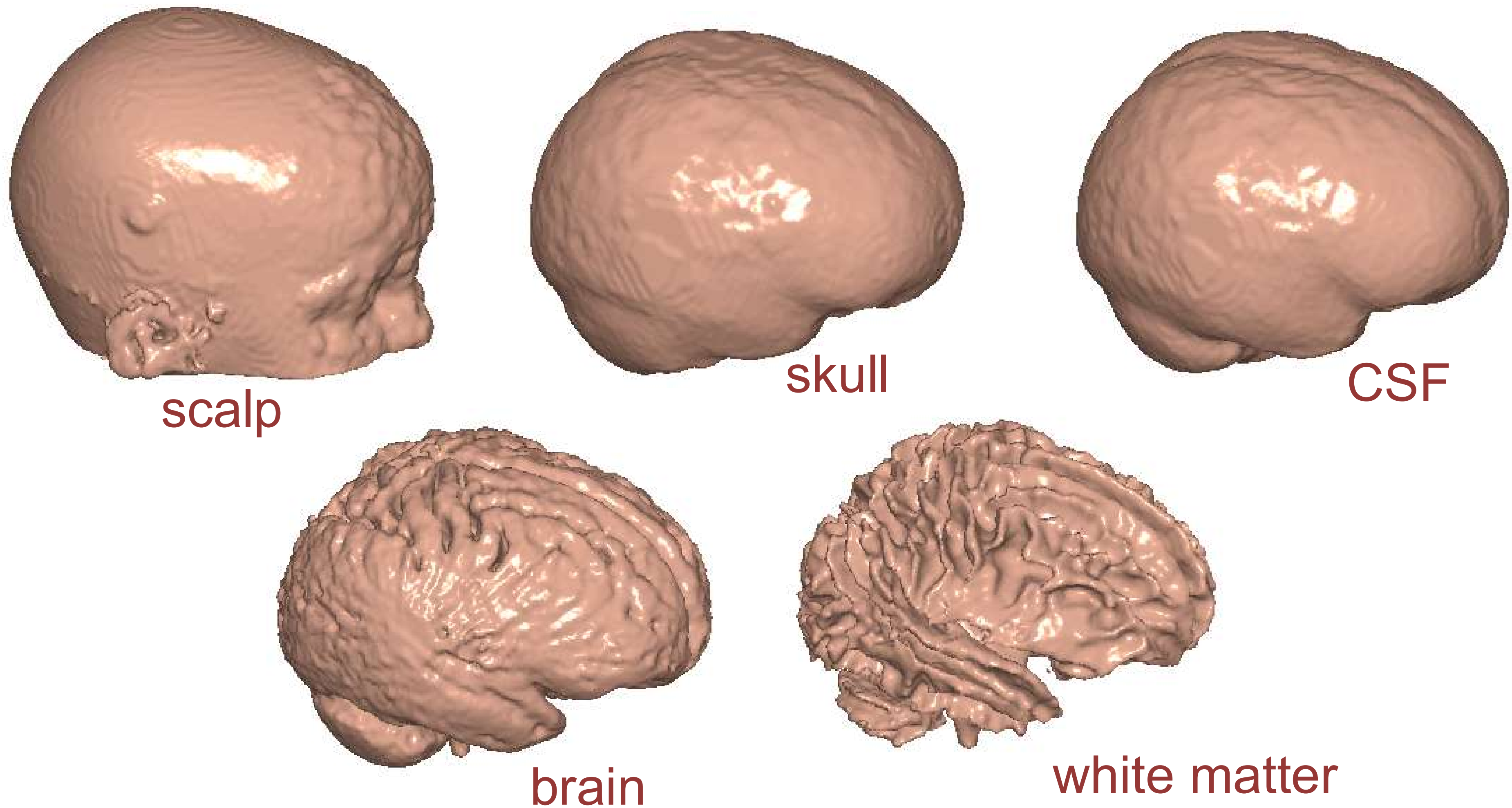
6 months | 12 months



1,441,777 tetrahedral volume elements

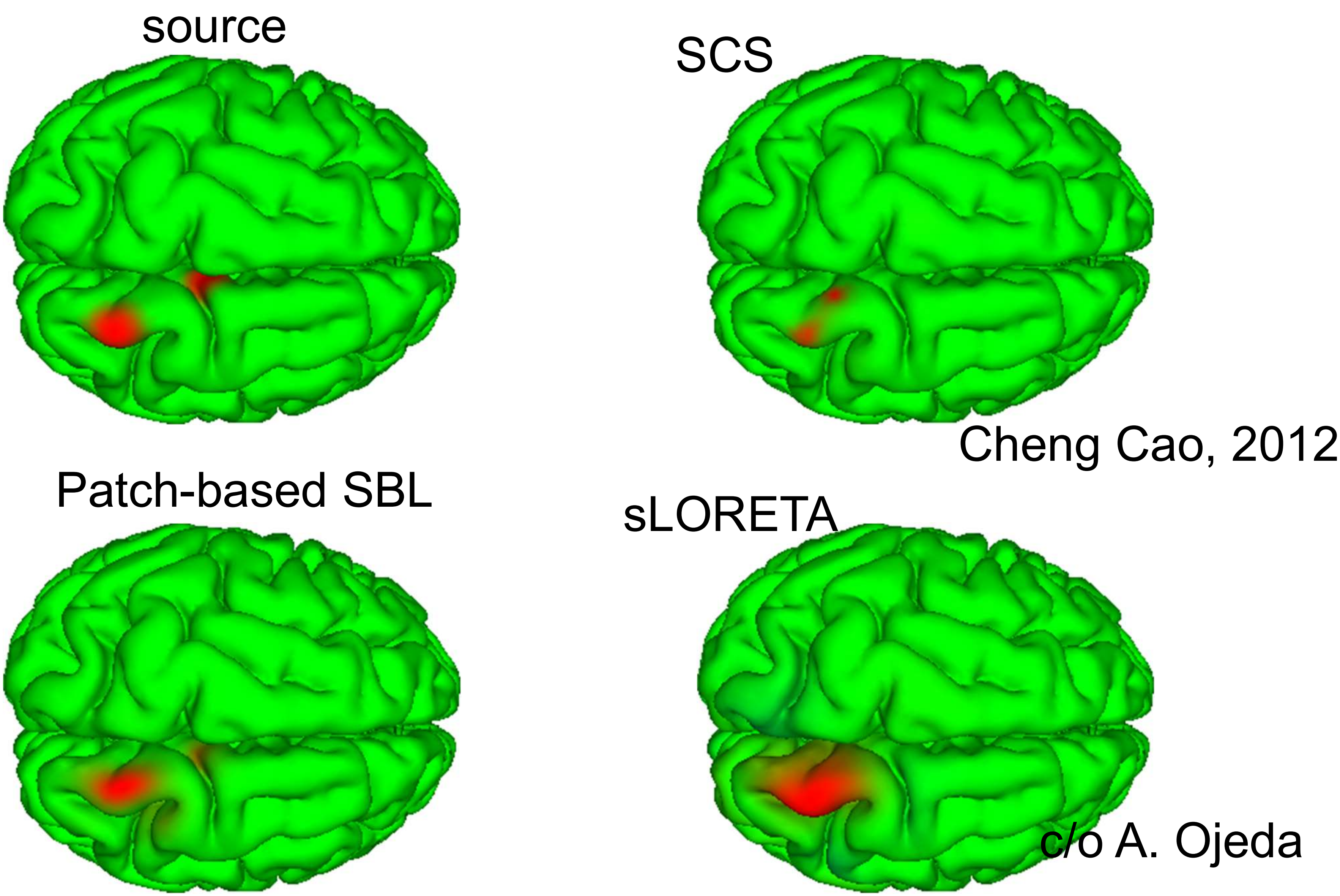
1,025,643 tetrahedral volume elements

Richards Database for child head modeling

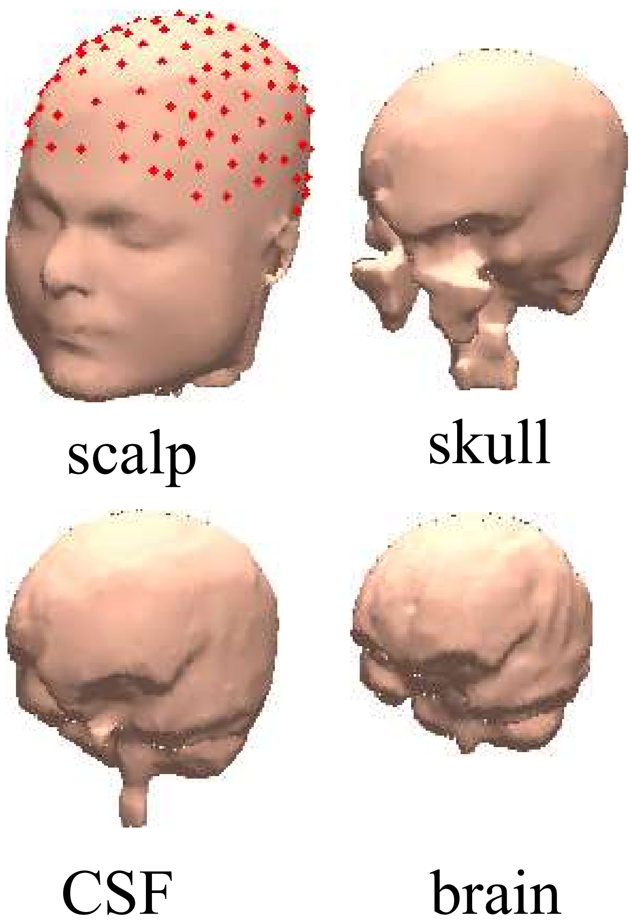


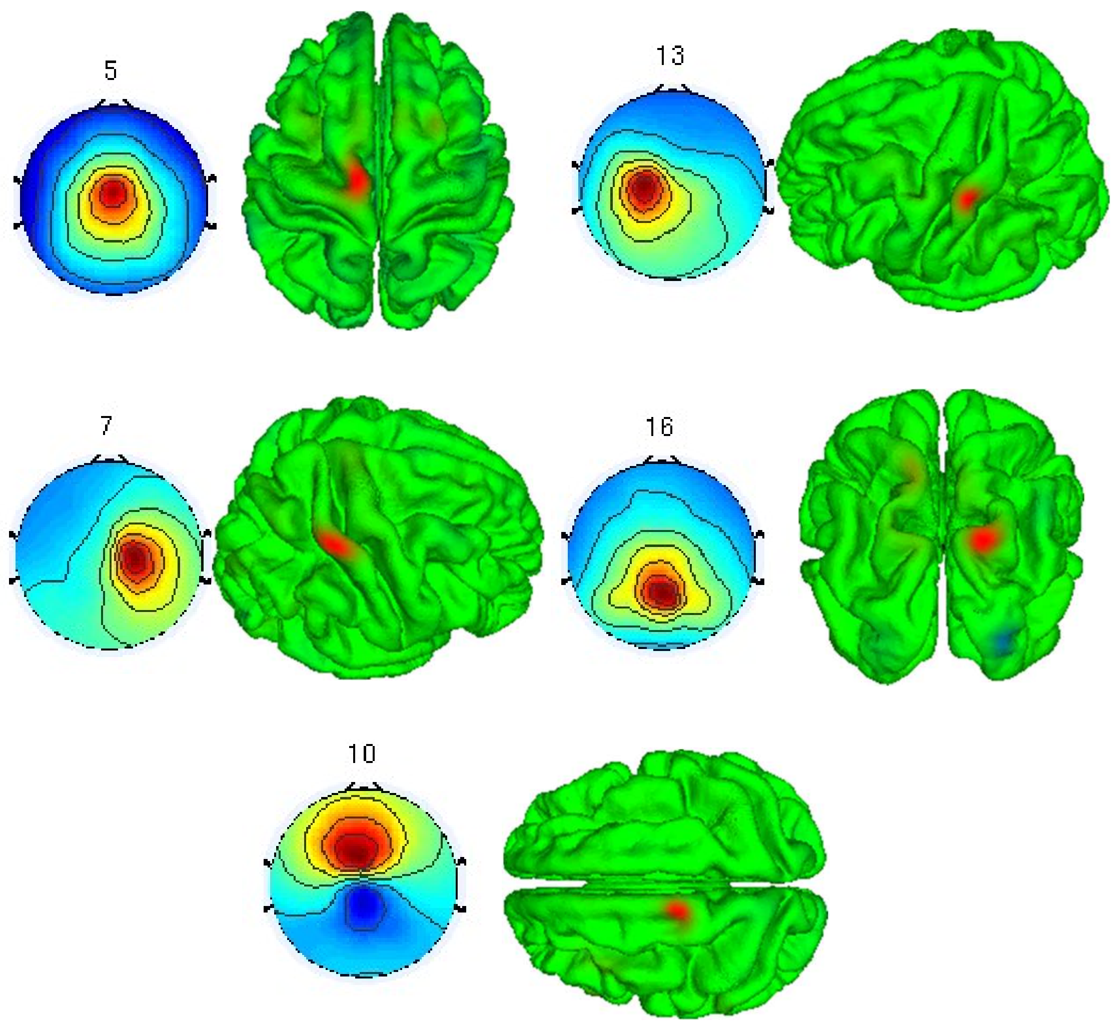
5-layer template BEM head model for three-year olds.

Source localization results

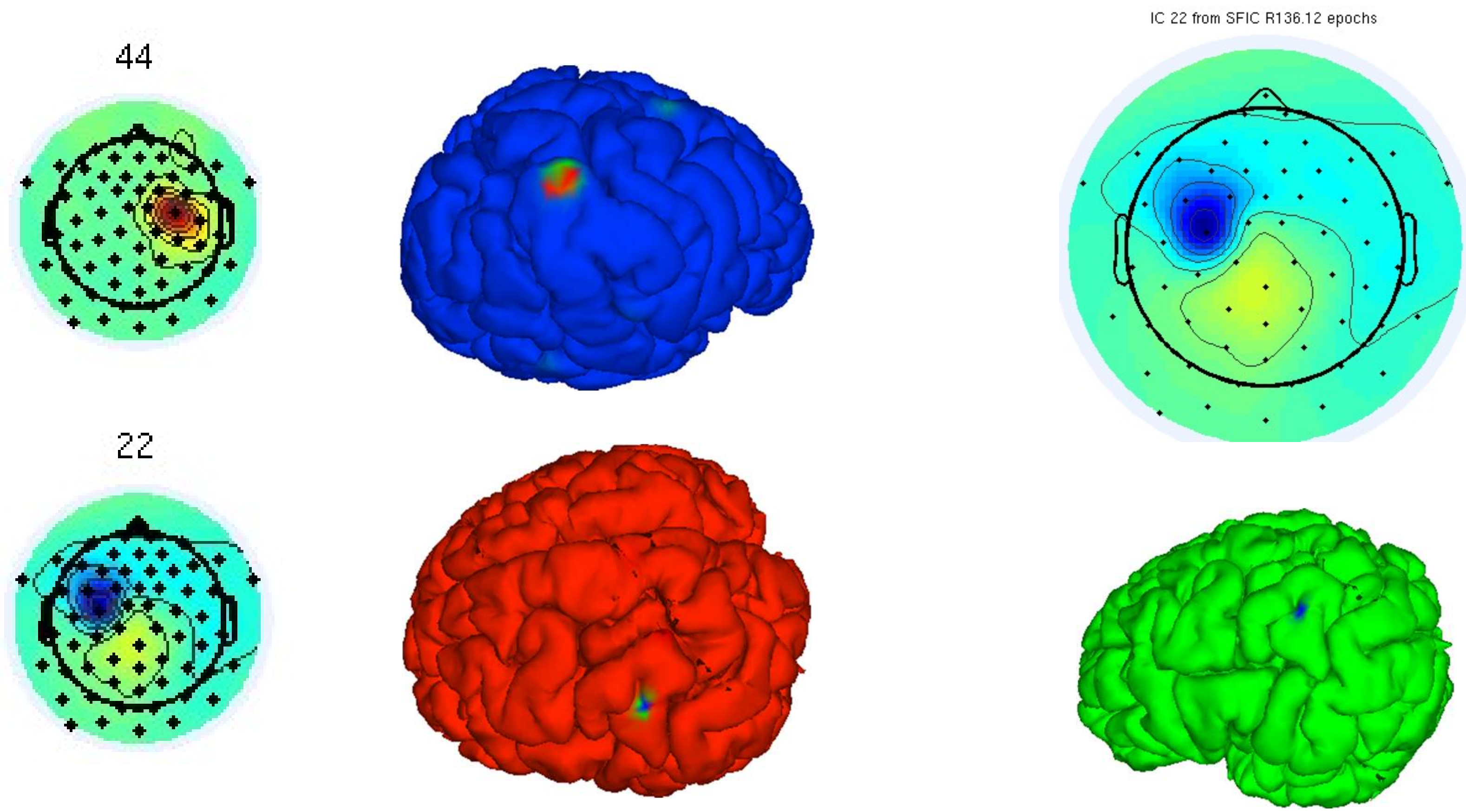


EEG source localization





Source localization for 1-year old



Visualization

- NIST uses Showmesh for visualization of potentials on the cortical surface.
- Showmesh loads a mesh in .smf format,
- Loads potential distribution.
- There are options to load a point set, zoom in, out, rotate, take snapshots.

SHOWMESH TUTORIAL

NFT Matlab Scripts

- Start EEGLAB and set your parameters:

```
eeglab
EEG = pop_loadset('filename',eeg_file,'filepath',eeg_path);
[ALLEEG, EEG, CURRENTSET] = eeg_store( ALLEEG, EEG, 0 );
% set 'of' (output folder), subject_name,
%   session_name, and elec_file
subject_name = 'SubjectA';
session_name = 's1';
nl = 4;   % number of layers
plotting = 1;
comp_index = 1:20; % component index for source
   localization
```

NFT Matlab Scripts

- Realistic modeling from MRI

```
% Do segmentation using the GUI
nft_mesh_generation(subject_name, of, nl)
nft_source_space_generation(subject_name, of)
% Do co-registration using the GUI
nft_forward_problem_solution(subject_name,
   session_name, of);
dip1 = nft_inverse_problem_solution(subject_name,
   session_name, of, EEG, comp_index, plotting,
   elec_file)
```

NFT Matlab Scripts

- BEM warping mesh

  nft_warping_mesh(subject_name, session_name, elec_file, nl, of,0,0);

  nft_forward_problem_solution(subject_name, session_name, of);

  dip1 = nft_inverse_problem_solution(subject_name, session_name, of, EEG, comp_index, plotting, elec_file)

NFT Matlab Scripts

- FEM warping mesh

  session_name='s1_fem';

  nft_warping_mesh(subject_name, session_name, elec_file, nl, of,0,1);

  nft_fem_forward_problem_solution(subject_name, session_name, of);

  dip1 = nft_inverse_problem_solution(subject_name, session_name, of, EEG, comp_index, plotting, elec_file)

NFT Matlab Scripts

- Set NFT dipole structure to EEGLAB dipole structure

```
eeglab_folder = dirname(which('eeglab'));
mri_file = [eeglab_folder
    '/plugins/dipfit2.2/standard_BEM/standard_mri.mat'];
EEG.dipfit.mrifile =  mri_file;
EEG.dipfit.model = EEG.etc.nft.model;
```

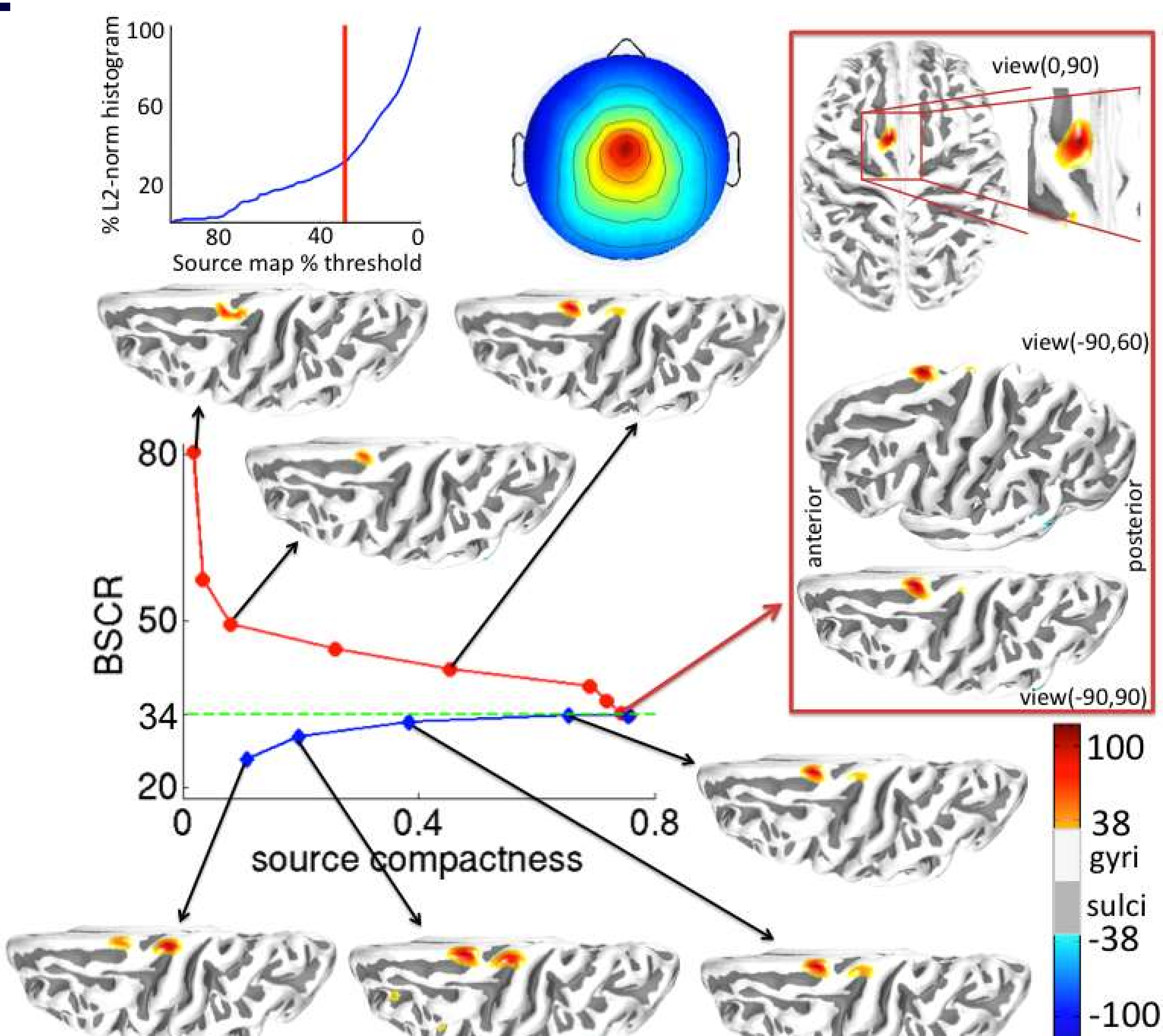
NFT download and reference

- http://www.sccn.ucsd.edu/nft

- Akalin Acar Z, Makeig S, Neuroelectromagnetic Forward Head Modeling Toolbox, J. of Neuroscience Methods, vol 190(2), 258-270, 2010.

SCALE

- Simultaneous skull conductivity and source localization estimation.
- Inputs: EEG and MRI
- Outputs: Source estimates and skull conductivity estimate



Akalin Acar and Makeig, 2016

BSCR values for 9 subjects

SCALE applied to 9 subjects between 18-25 years old.
4-layer head model is used: scalp, skull, CSF, and brain
Layer conductivities: Scalp: 0.33S/m, CSF:1.79S/m, Brain: 0.33S/m
Skull conductivity and brain sources are searched.

	BD	FR	AV	RB	LH	GV	AS	JH	SE
STRUM	28 (10)	**43** (14)	30 (13)	68 (15)		31 (11)	63 (12)	**48** (15)	**31.5** (15)
Darts		**45** (14)			20 (16)				
Arrow Flanker								**54** (13)	**34** (13)

The number of IC's to run SCALE is shown in parenthesis.

Computation times for 1 SCALE iteration

Computation times for:
13 ICs, 127 electrodes,
80,000 cortical sources, 9,000 grid sources

	Juggling	Juggling with Opt	Comet (24)
FP - cortical source	239	125	5.6
FP – grid sources	25	13	0.7
IP - SCS	43	38	4.7
Compactness	0.8	0.7	0.06
Sensitivity	396	391	4.5
Dipole Soln	91	52	16
TOTAL	794 min = 13.2 hr	619 min = 10.3 hr	31 min

Neuroscience Gateway Portal