

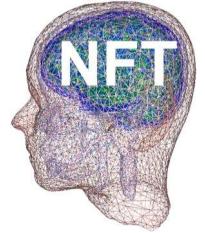
NFT & NIST

Neuroelectromagnetic Forward and Inverse Head Modeling Toolbox

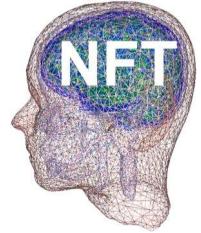
Zeynep AKALIN ACAR
EEGLAB Workshop, San Diego
November, 2018



NFT

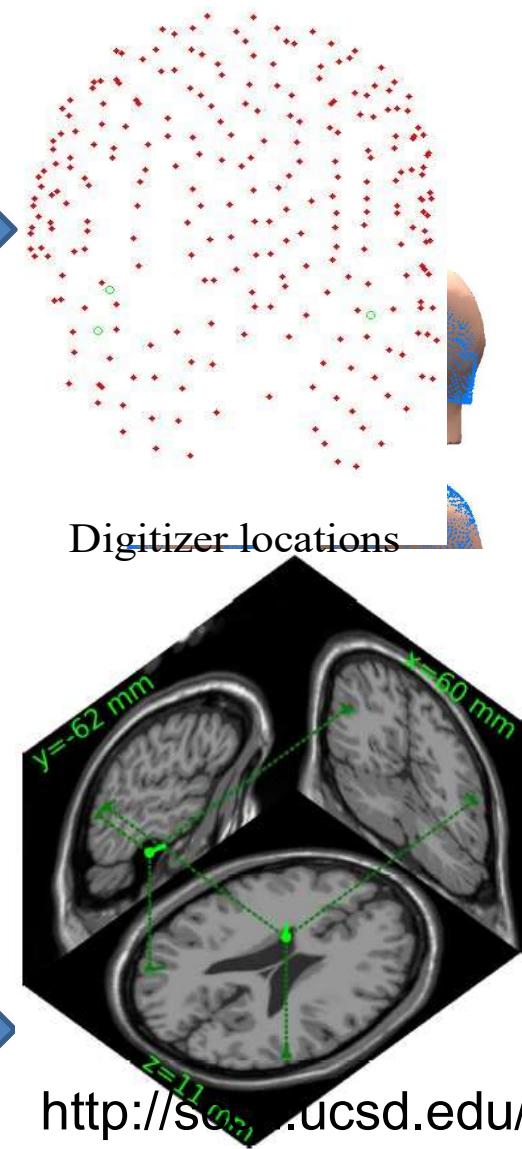
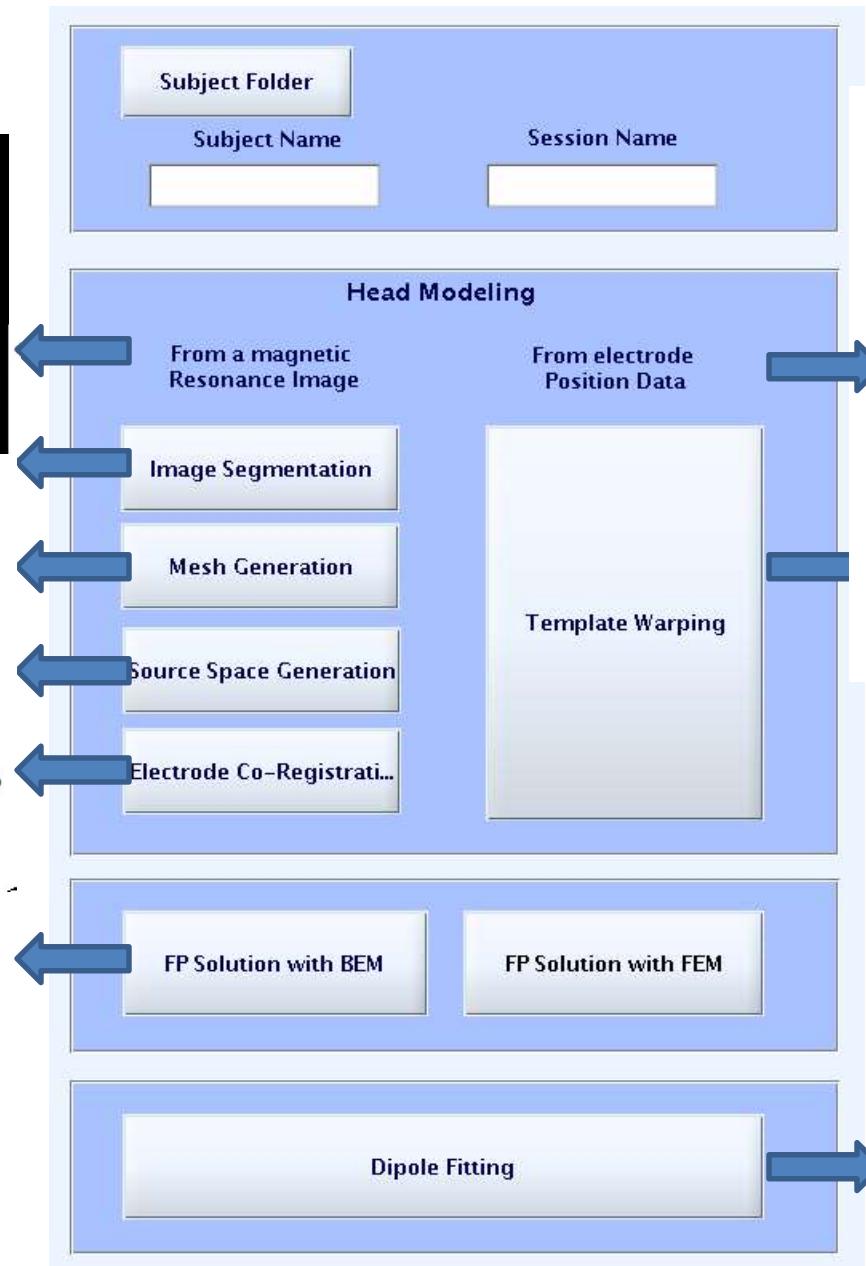
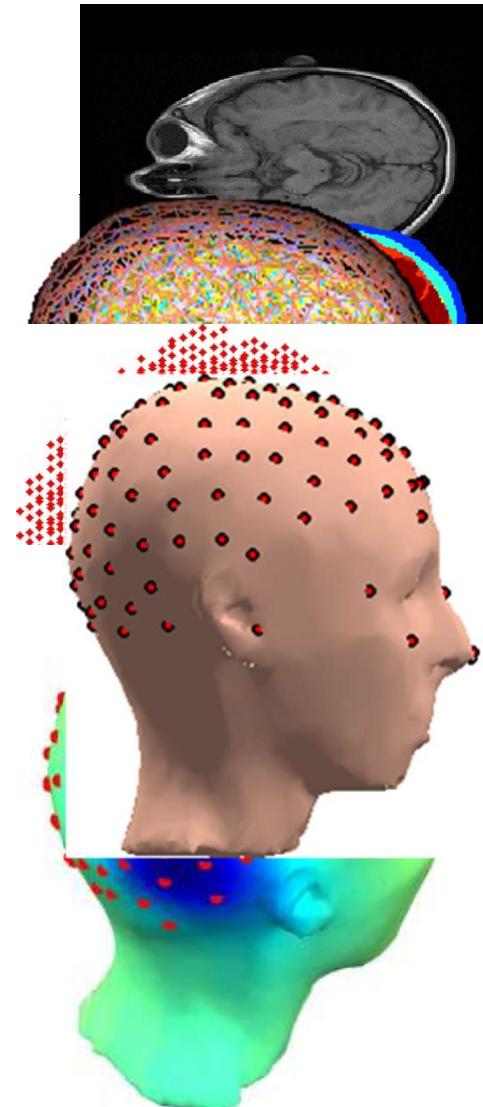
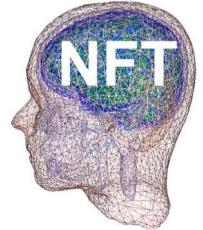


- ◆ A complete framework for accurate forward problem solution.
- ◆ Easy-to-use MATLAB environment with GUI and command-line functions.
- ◆ Ability to use available subject information
 - T1-weighted 3D MR images
 - Digitized sensor (electrode) locations

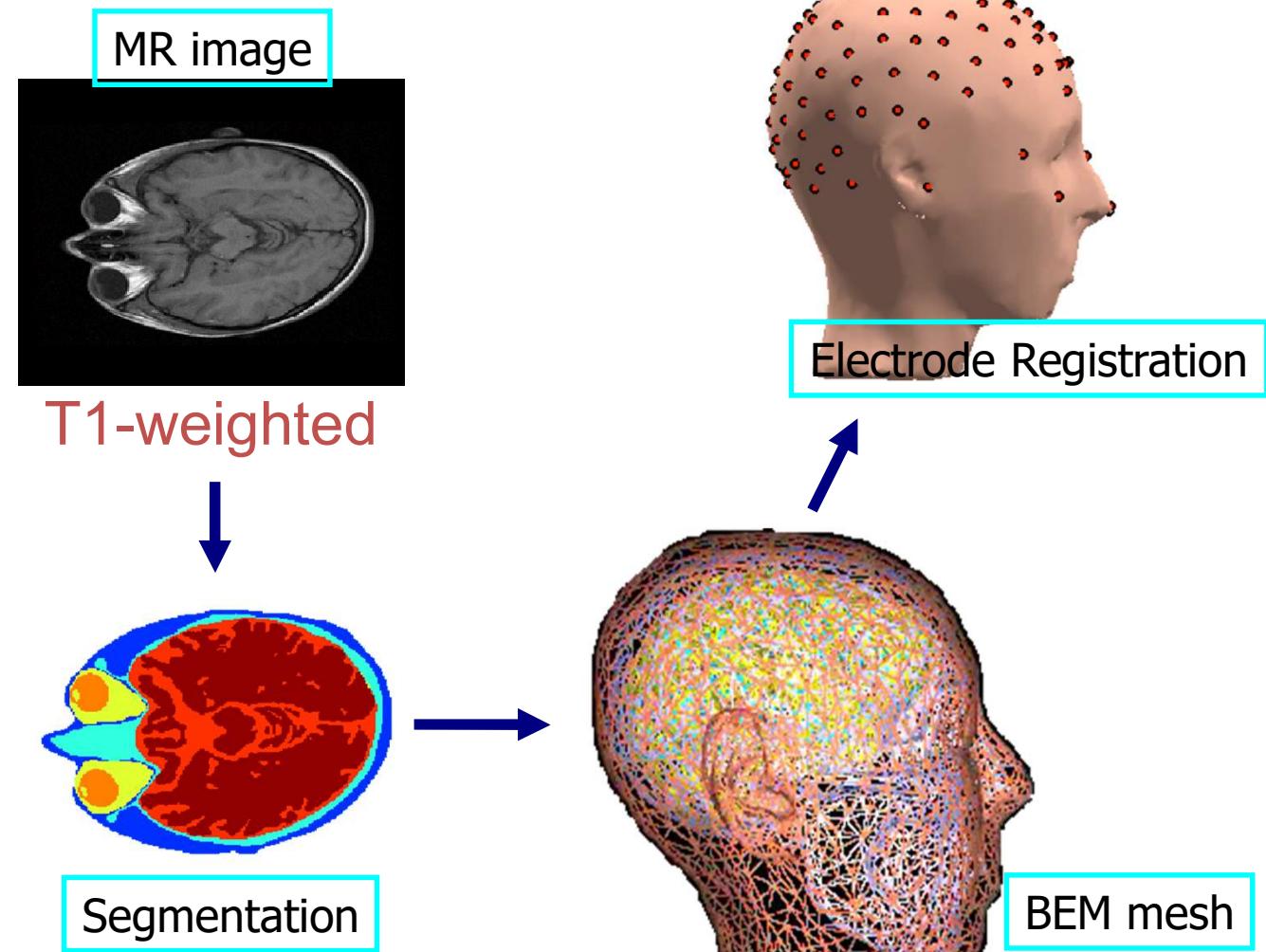
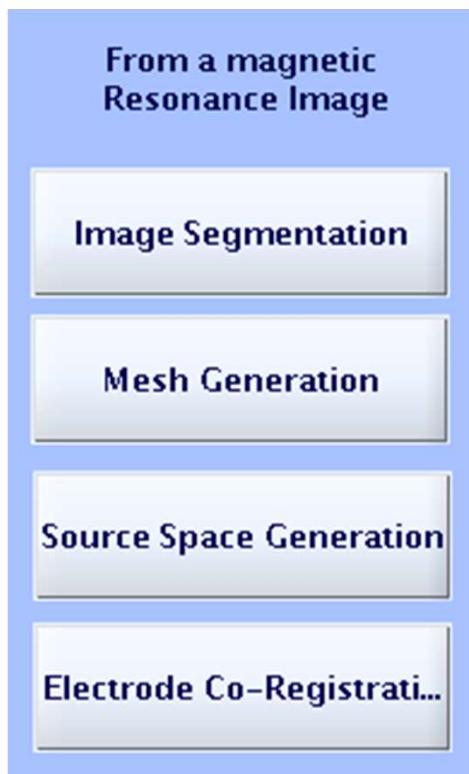


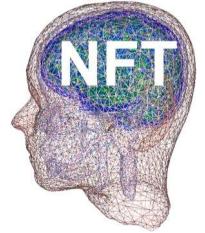
Comparison with Dipfit

- ◆ 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.



Head modeling from MR images

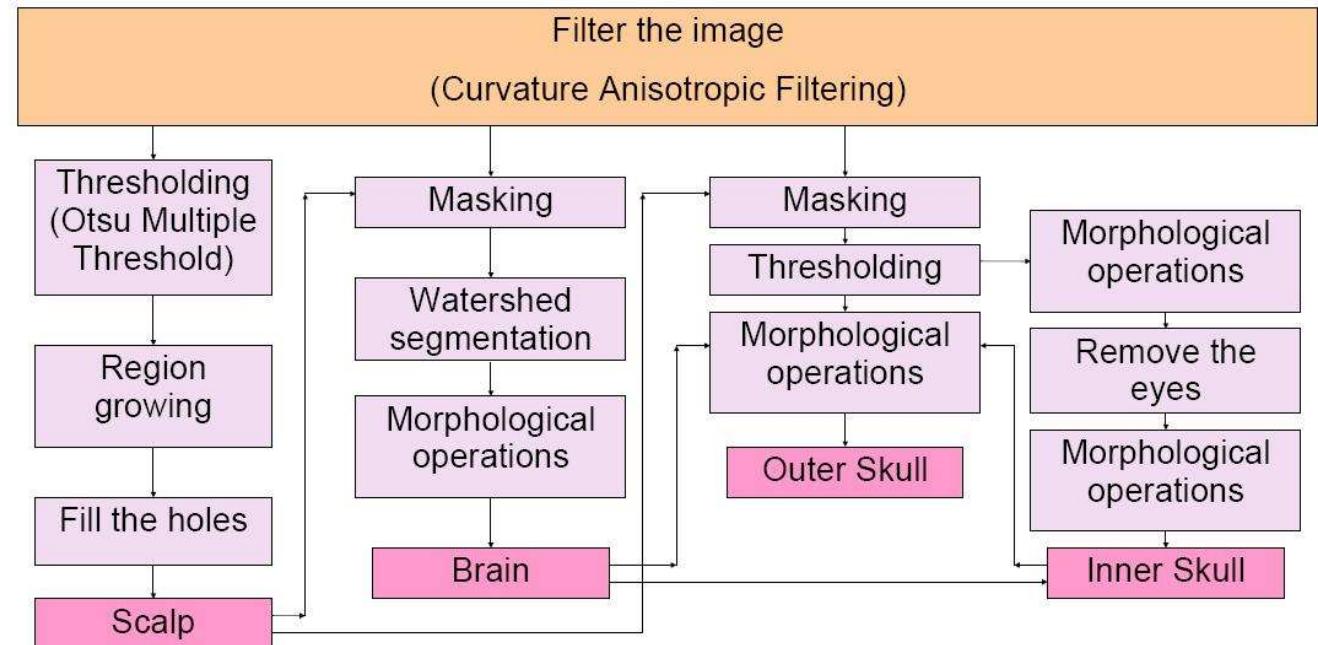
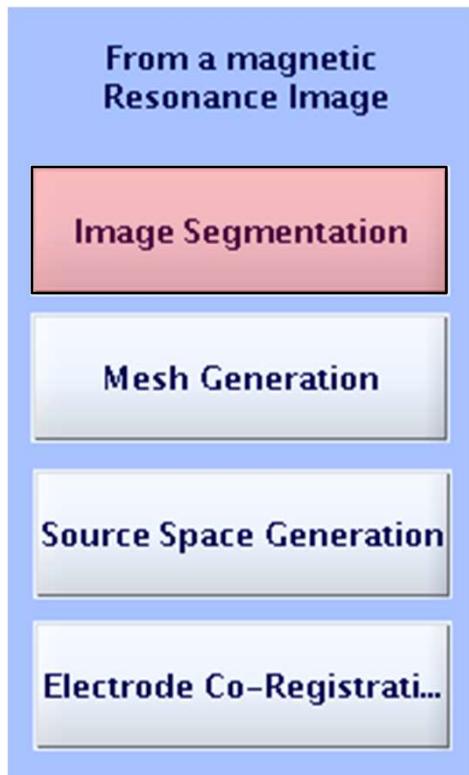




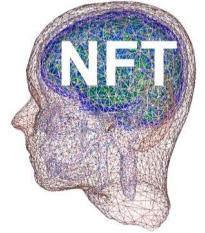
Preparing the MR Image

- ◆ 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

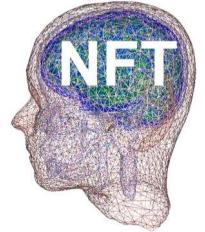


Starting NFT

- ◆ To start from EEGLAB
EEGLAB -> Tools -> NFT

- ◆ To start as a standalone toolbox
addpath NFT directory
Type 'NFT' in Matlab

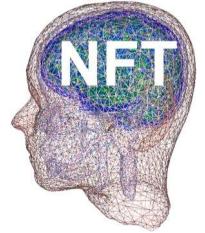
For demo: go to NFT-2.4_demo folder



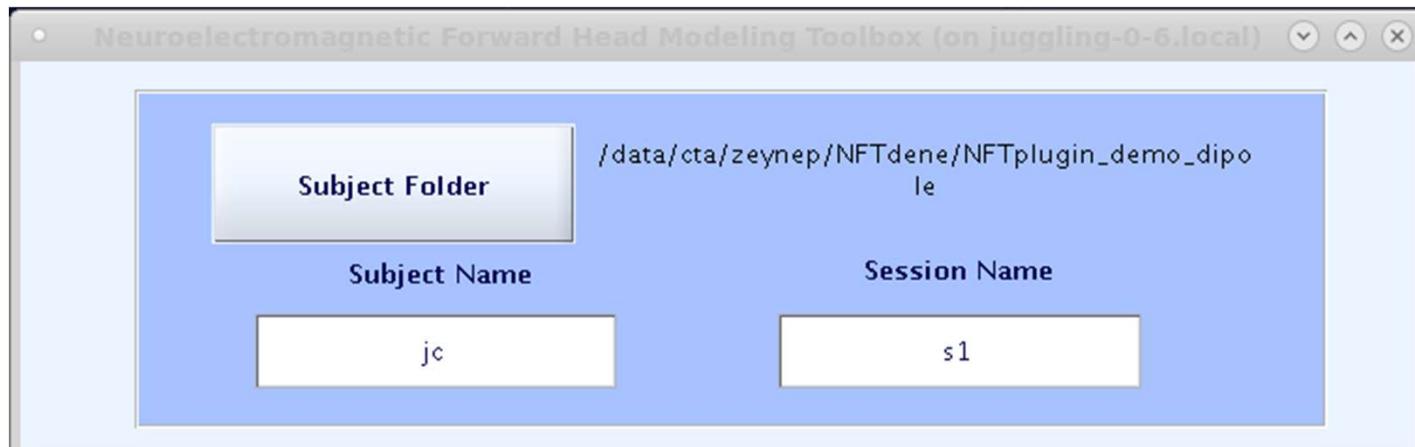
Subject Selection

A screenshot of a software interface titled "Subject Selection". It has a light blue background. On the left, there is a button labeled "Subject Folder". To the right of the button are two input fields: one labeled "Subject Name" and another labeled "Session Name", both enclosed in thin grey boxes.

- ◆ 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

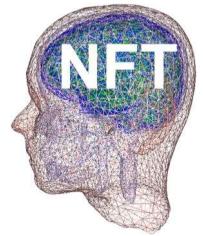


Image Segmentation

NFT: MR segmentation

File

(x,y,z)=

Coronal view

Axial view

Sagittal view

Display Image

- MR image
- Filtered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Save Results

Output Folder: /data/projects/zeynep/comm on/home zeynep/io/deneme

Filtered Image Segmentation

Image Segmentation

Swap LJR Check inhomogeneity

1. Anisotropic Filtering

5 Number of iterations

3 Image diffusion

2. Scalp Segmentation

3. Brain Segmentation

66	Set	Cerebellar low point
x: 135		
y: 135	Set	White matter seed point
z: 110		
0.4 Fill level [0, 1]	0.4 Threshold [0, 1]	

4. Outer Skull Segmentation

z: 110 Set Center of one eye

5. Inner Skull Segmentation

< Prev Run Next >

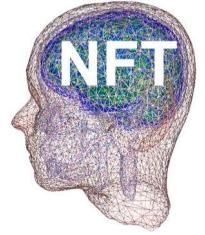


Image Segmentation

Load image

NFT: MR segmentation

File

Open ... **Close**

Coronal view

Axial view

Sagittal view

Display Image

- MR image
- Filtered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Save Results

Output Folder: /data/projects/zeynep/comm on/home zeynep/io/deneme

Segmentation

Image Segmentation

Swap LJR Check inhomogeneity

1. Anisotropic Filtering

5 Number of iterations

3 Image diffusion

2. Scalp Segmentation

3. Brain Segmentation

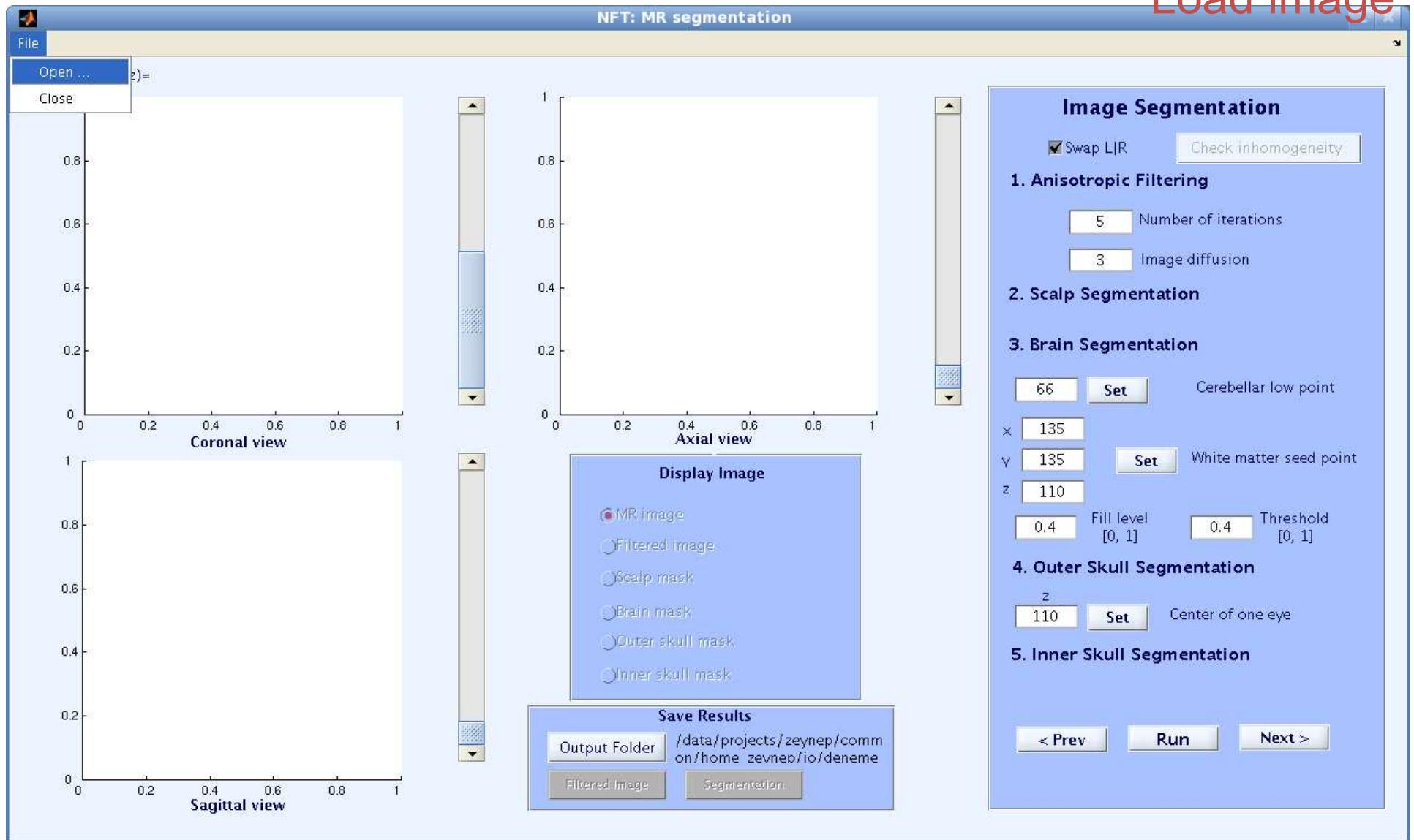
66	Set	Cerebellar low point
x: 135		
y: 135	Set	White matter seed point
z: 110		
0.4 Fill level [0, 1]		0.4 Threshold [0, 1]

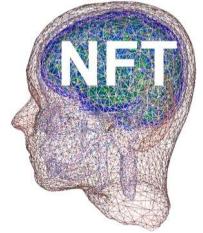
4. Outer Skull Segmentation

z: 110 **Set** Center of one eye

5. Inner Skull Segmentation

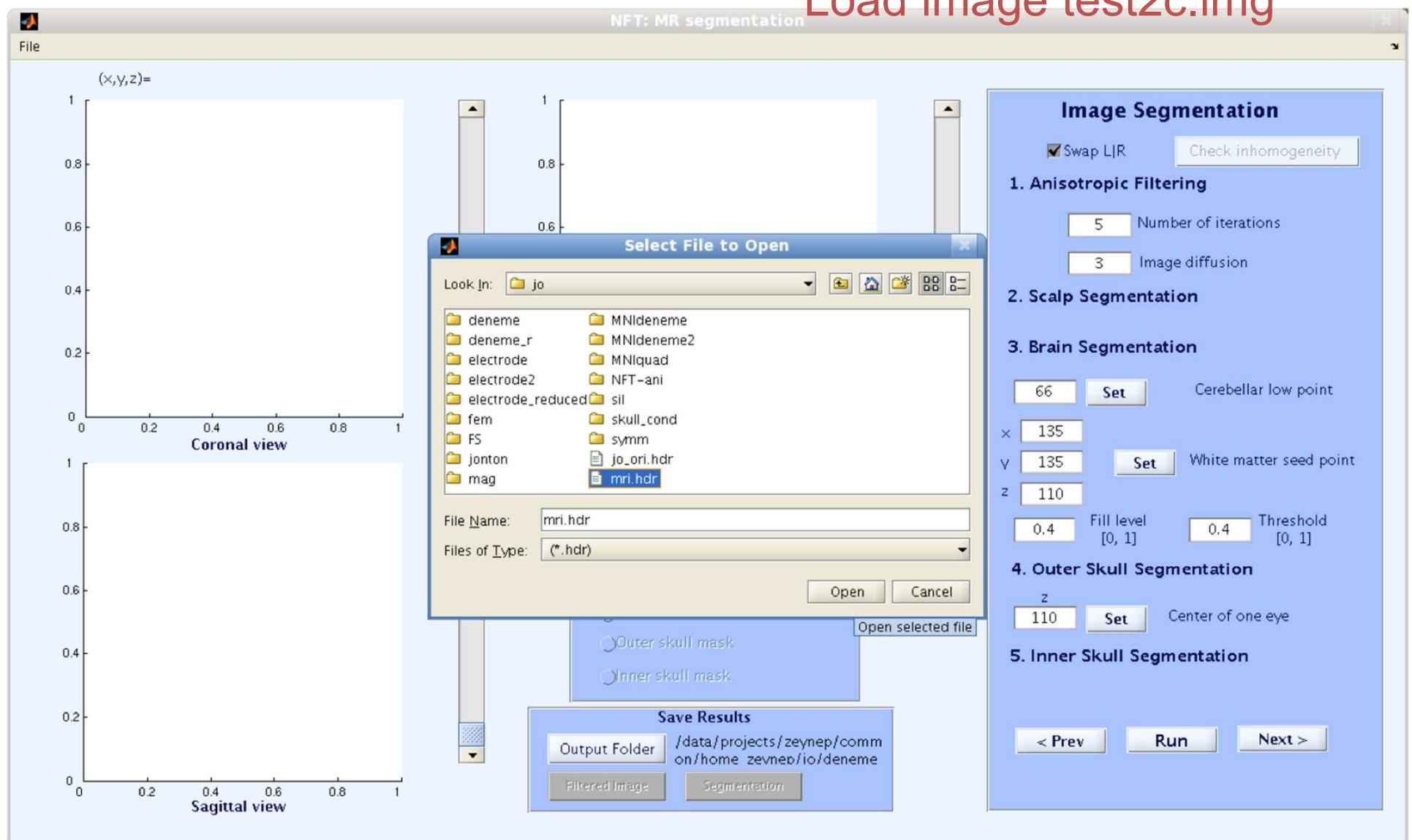
< Prev Run Next >

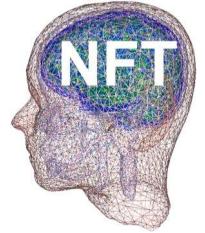




Segmentation

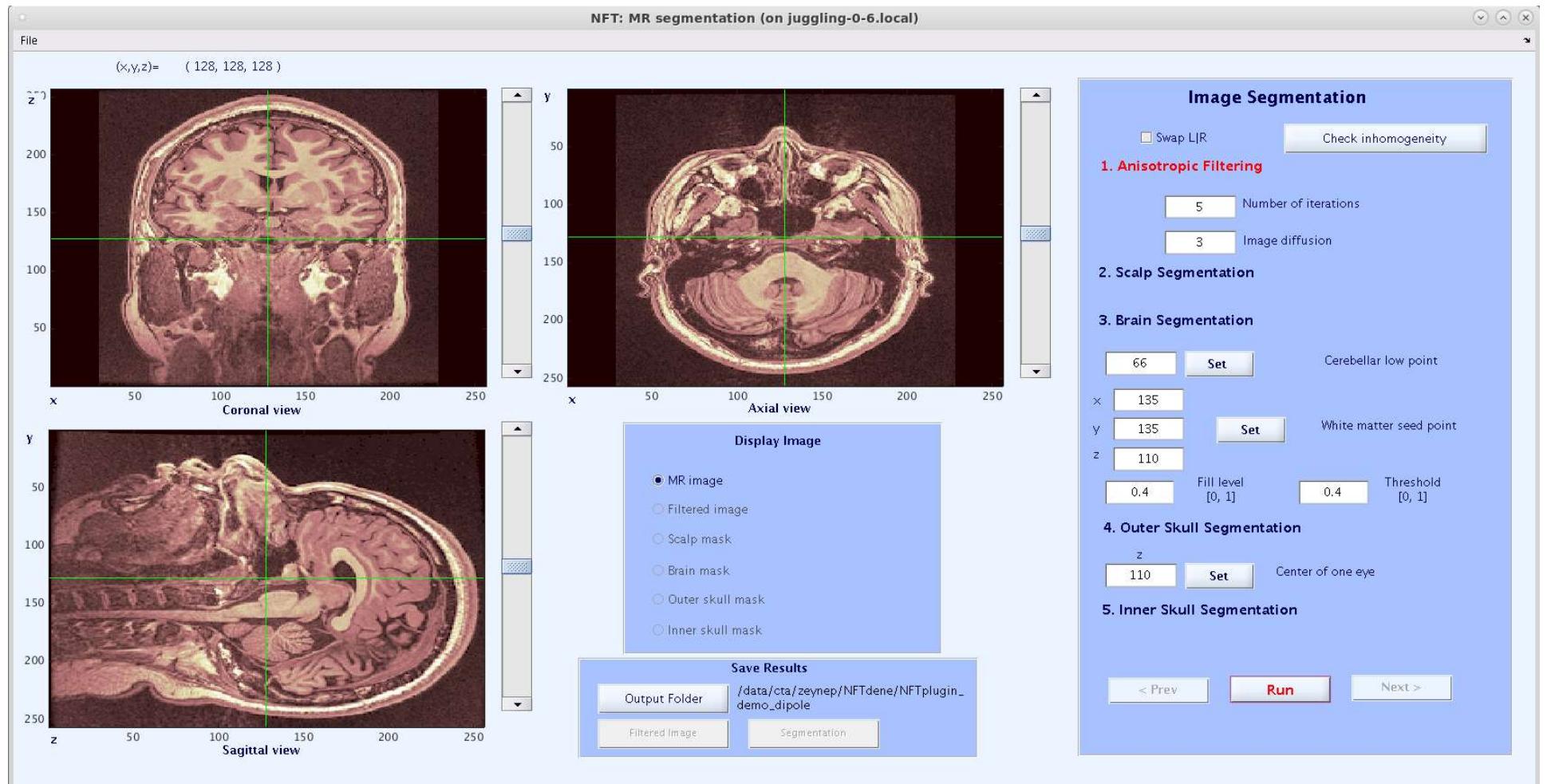
Load image test2c.img

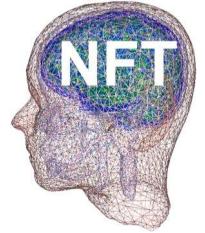




Segmentation

Run filtering





Segmentation

NFT: MR segmentation (on juggling-0-6.local)

File

(x,y,z)= (128, 128, 128)

Coronal view

Axial view

Sagittal view

Display Image

- MR image
- Filtered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Save Results

Output Folder: /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole

Filtered Image Segmentation

Image Segmentation

Swap L/R Check inhomogeneity

1. Anisotropic Filtering

Number of iterations: 5

Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation

Cerebellar low point: 66

White matter seed point: x: 135, y: 135, z: 110

Fill level: 0.4

Threshold: 0.4

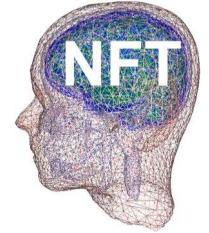
4. Outer Skull Segmentation

Center of one eye: z: 110

5. Inner Skull Segmentation

< Prev Run Next >

Run anisotropic filtering
Filtering...



Segmentation

View filtered image

NFT: MR segmentation (on juggling-0-6.local)

File

(x,y,z)= (128, 128, 128)

Coronal view

Axial view

Sagittal view

Display Image

- MR image
- Filtered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Save Results

Output Folder: /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole

Filtered Image Segmentation

Image Segmentation

Swap L/R Check inhomogeneity

1. Anisotropic Filtering

Number of iterations: 5 Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation

Cerebellar low point: 66 Set

White matter seed point: x: 135, y: 135, z: 110

Fill level: 0.4 Threshold: 0.4

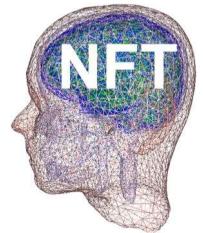
4. Outer Skull Segmentation

Center of one eye: z: 110 Set

5. Inner Skull Segmentation

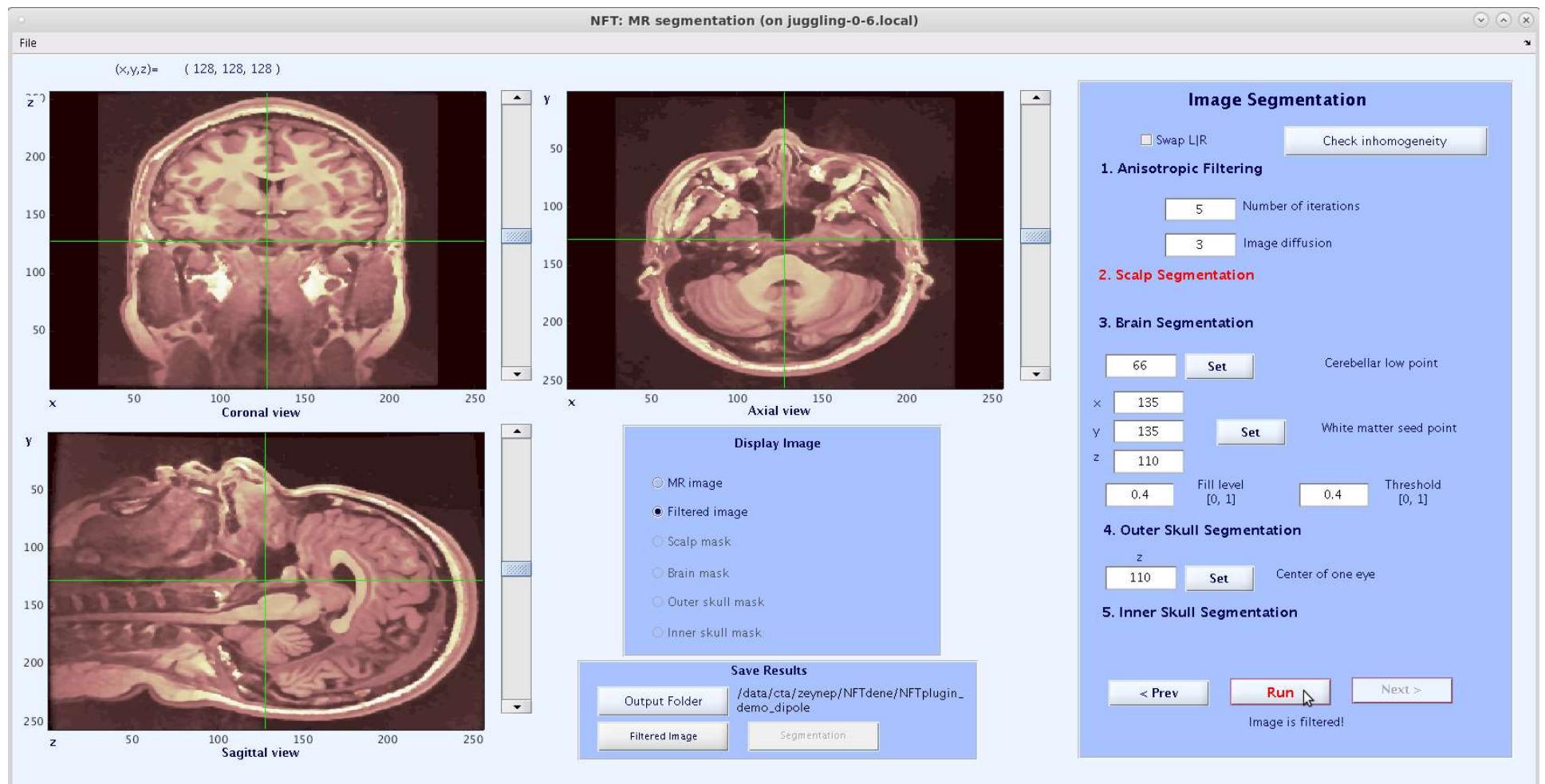
< Prev Run Next >

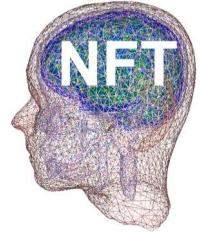
Image is filtered!



Segmentation

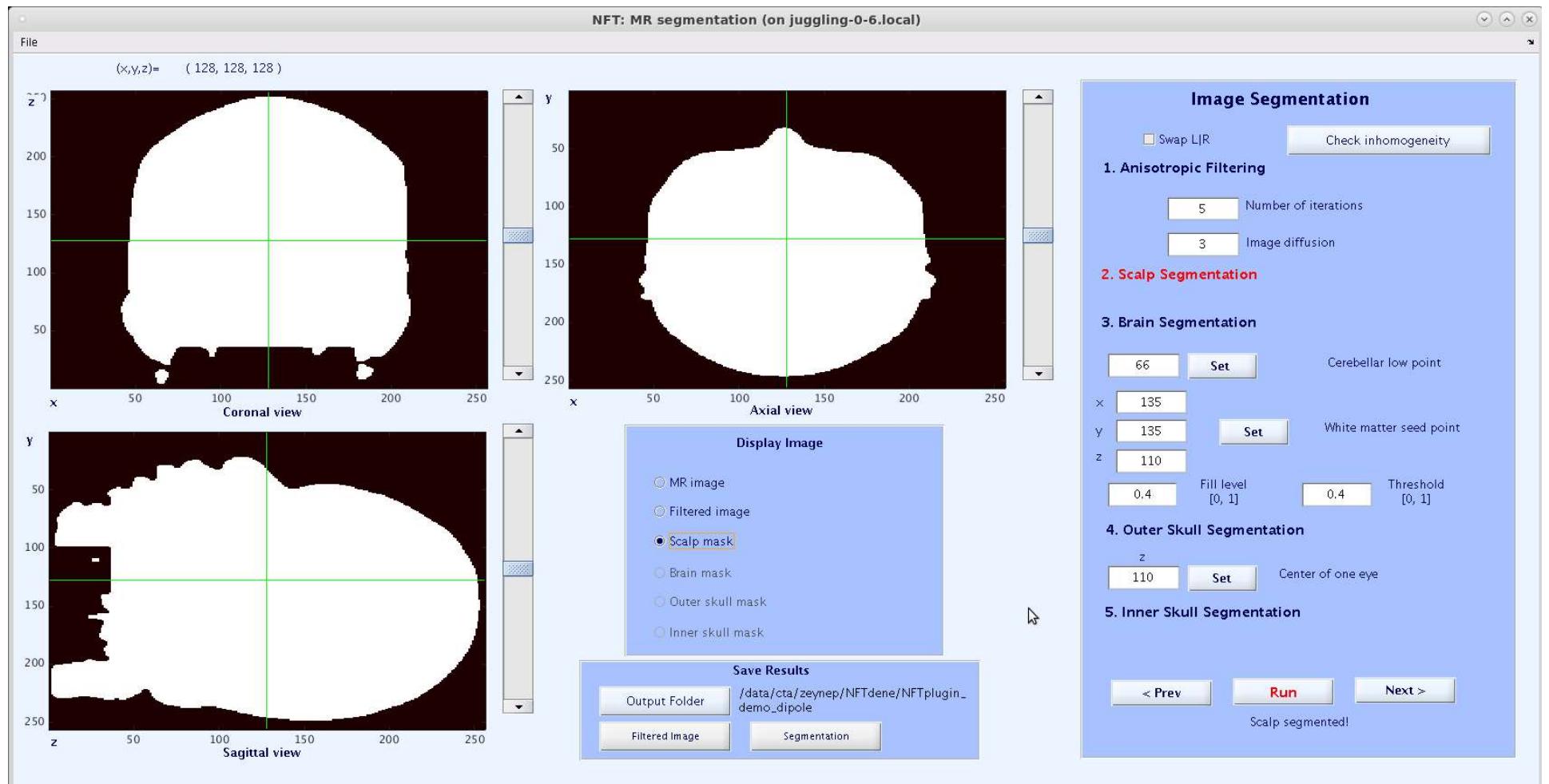
Click 'Next' for scalp segmentation and run scalp segmentation

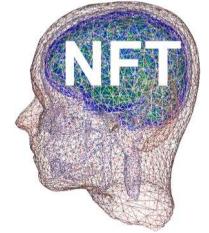




Segmentation

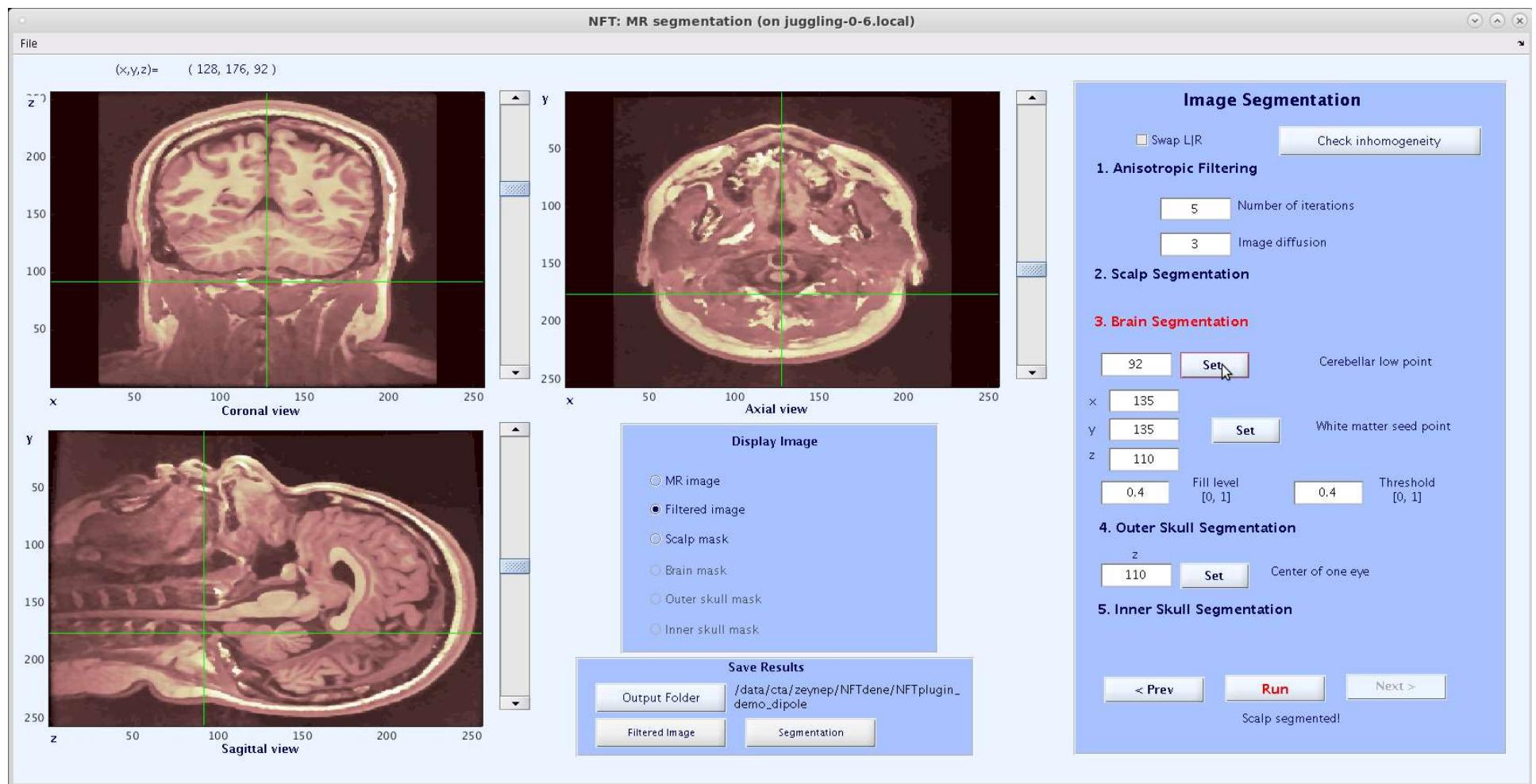
View scalp mask

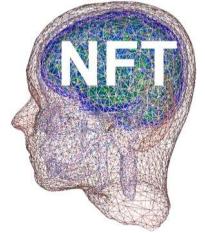




Segmentation

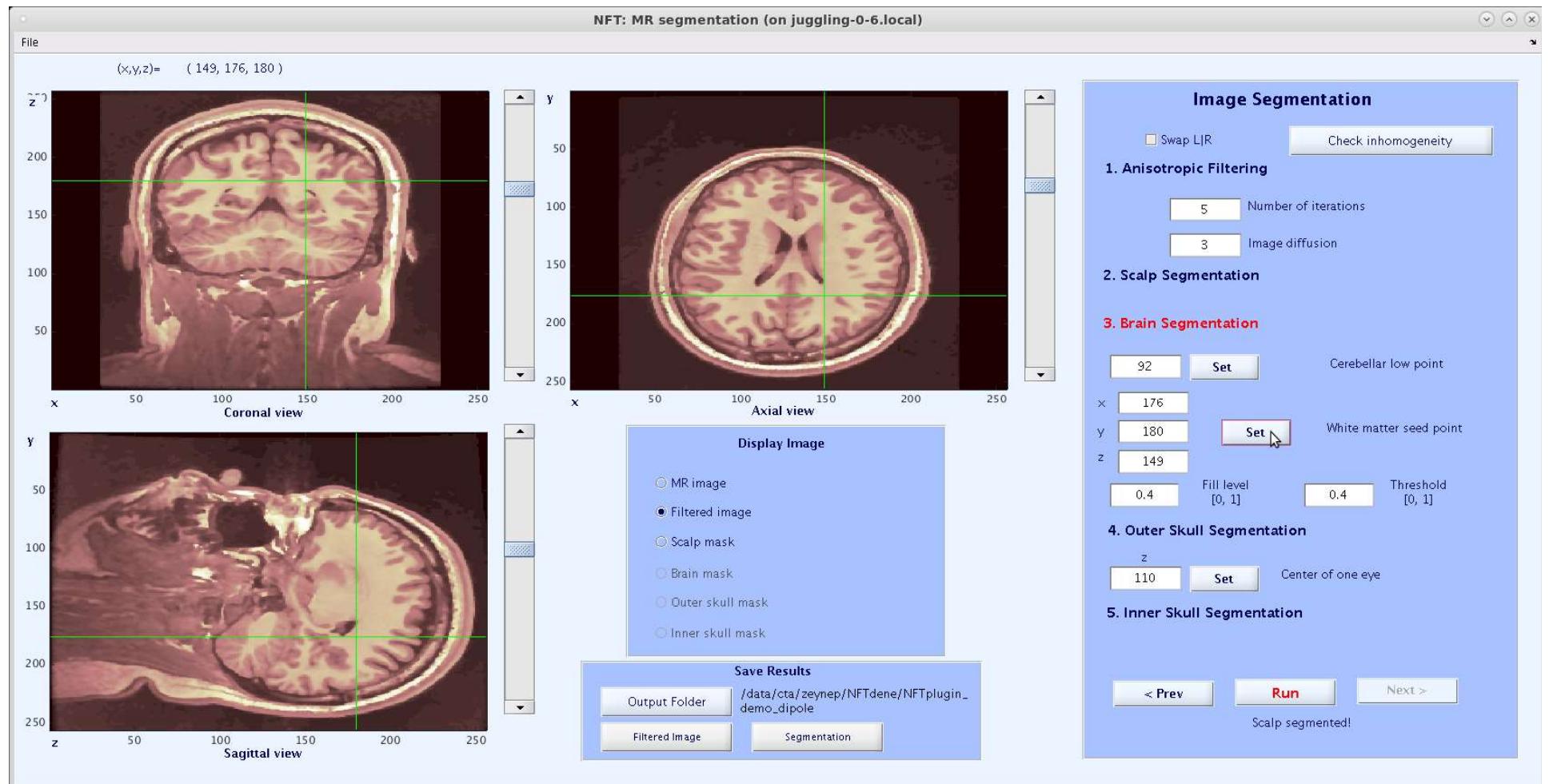
Click 'Next' for brain segmentation
Selection of cerebellar low point

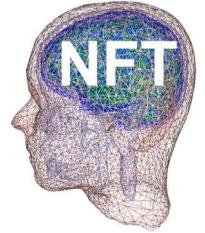




Segmentation

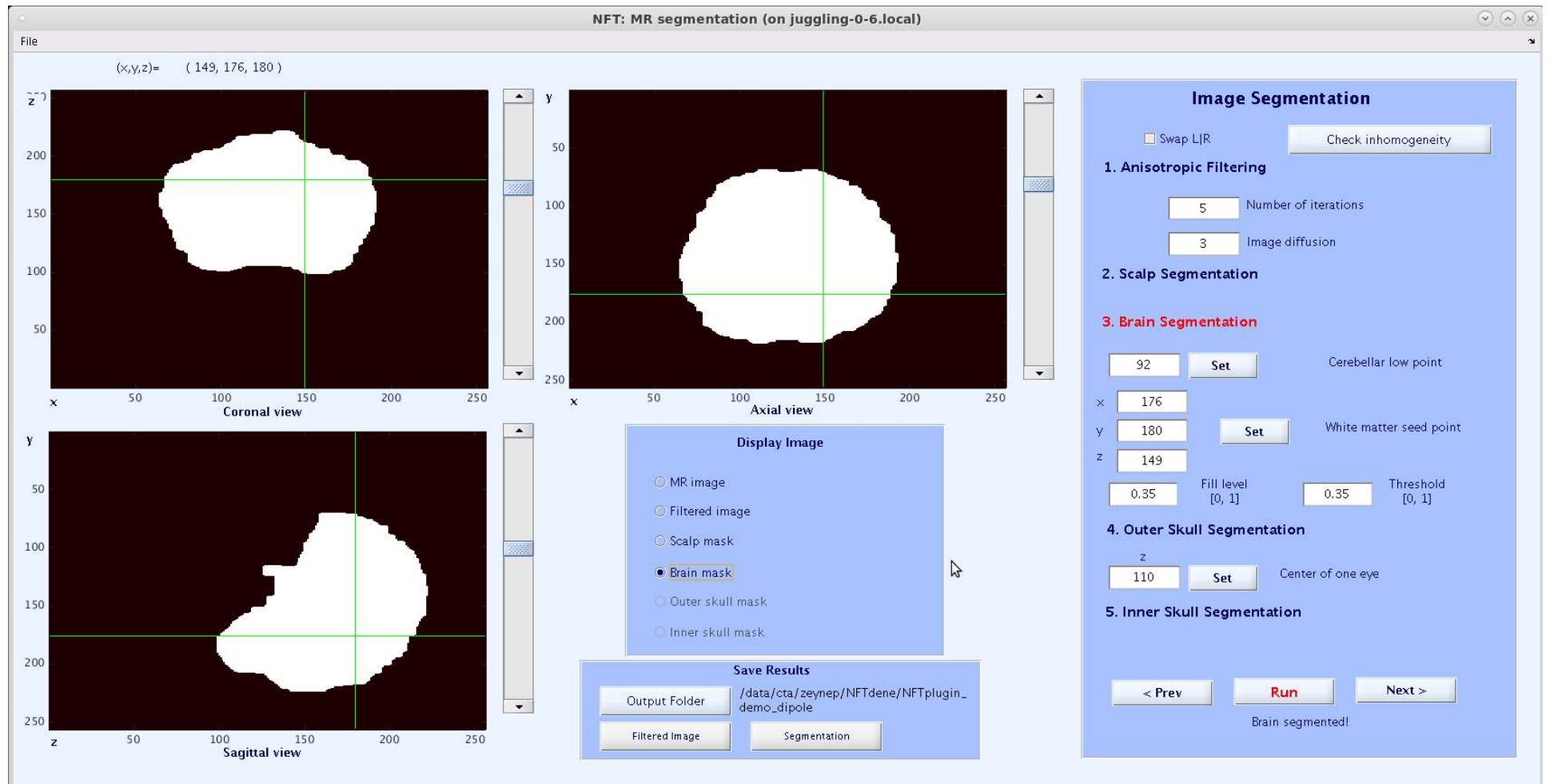
Selection of a white matter point

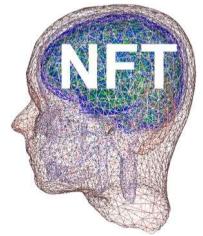




Segmentation

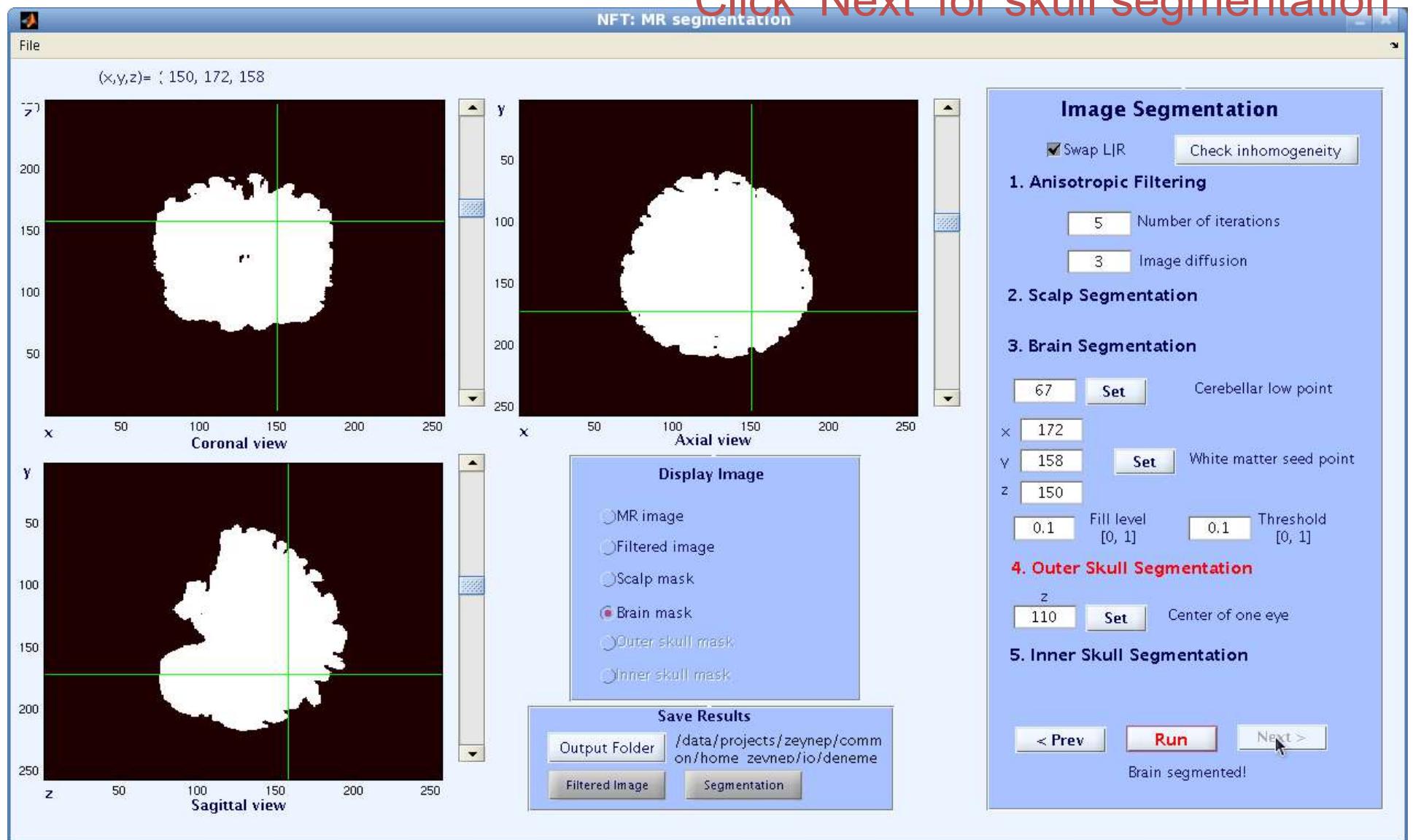
View brain mask

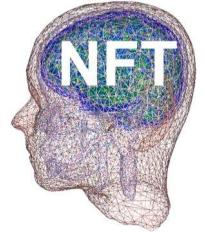




Segmentation

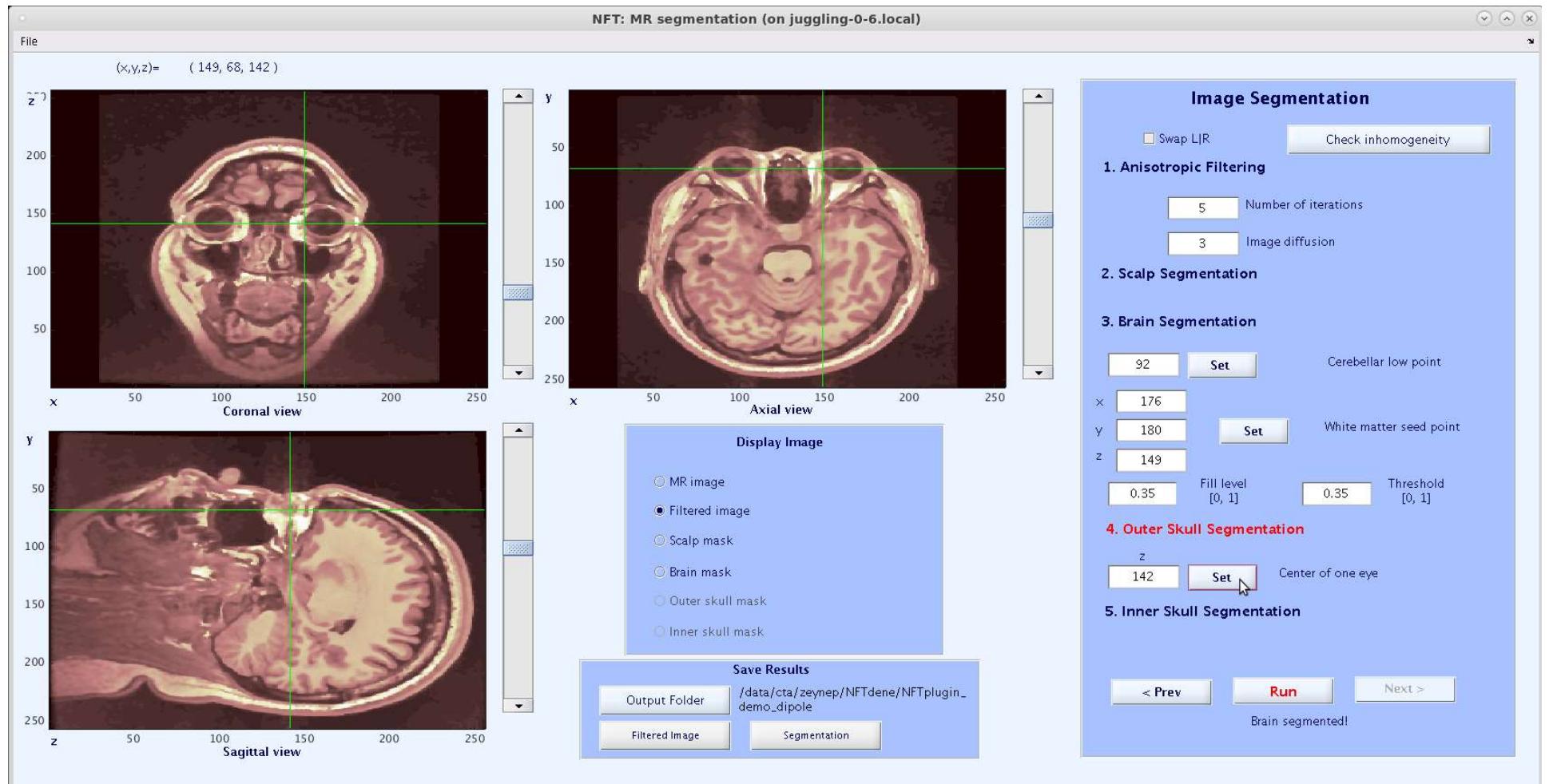
Click 'Next' for skull segmentation

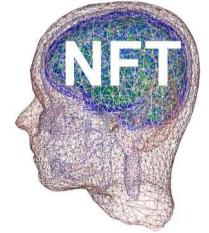




Segmentation

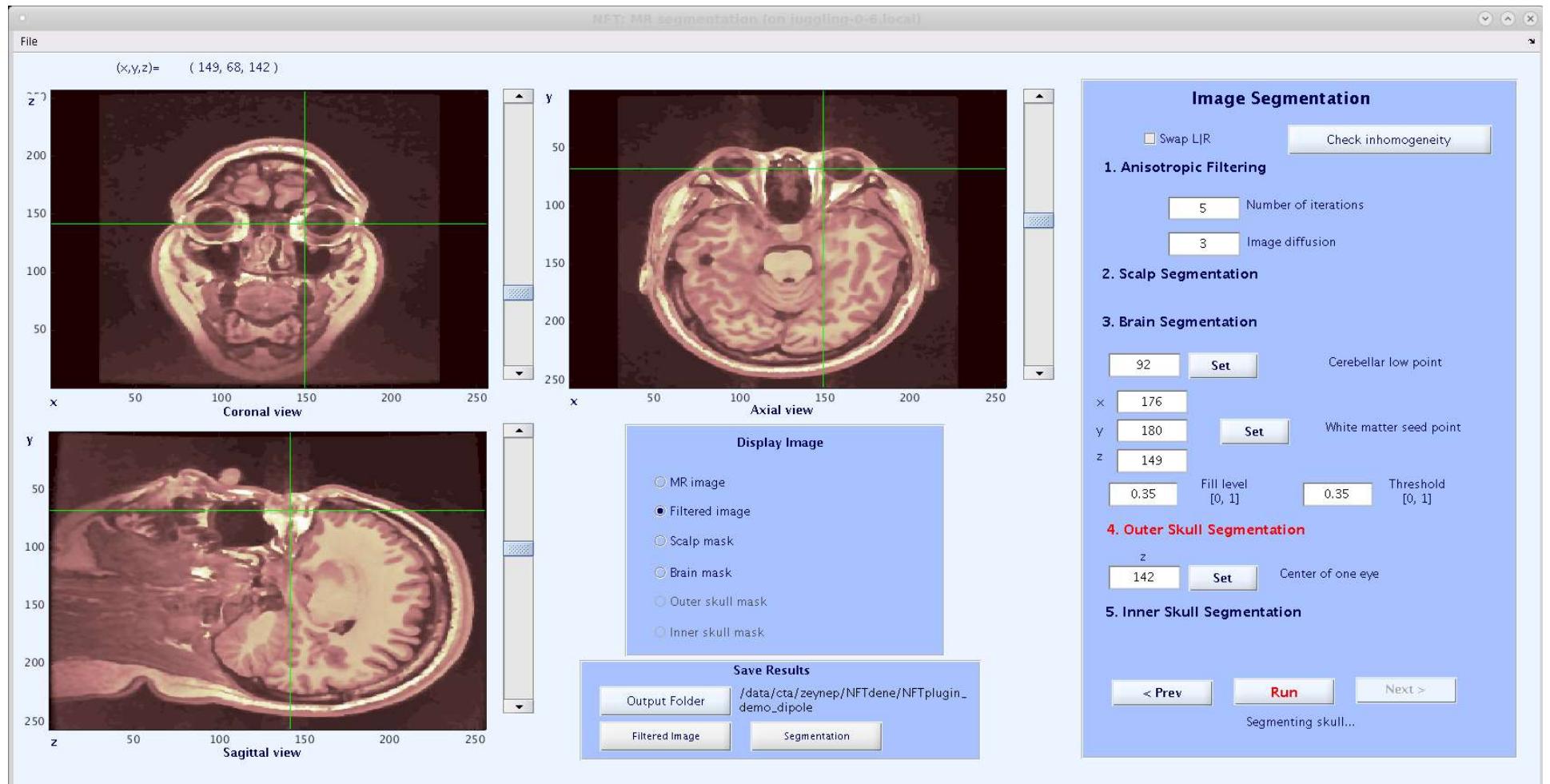
Select a slice for eyes and click 'set'

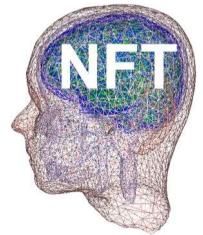




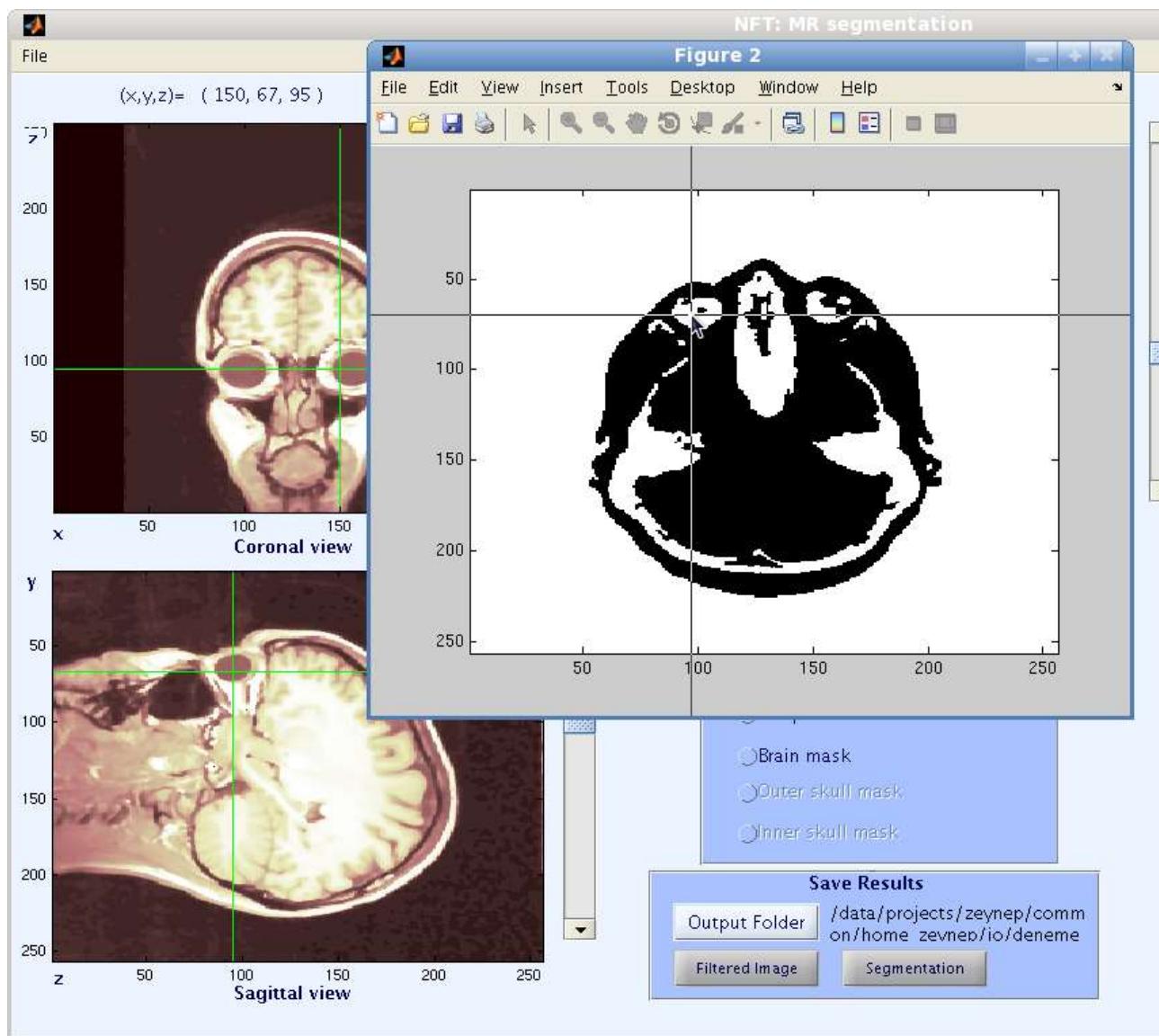
Segmentation

Click 'Run' for skull segmentation





Segmentation



Click on the eyes

Image Segmentation

Swap L/R Check inhomogeneity

1. Anisotropic Filtering

5 Number of iterations

3 Image diffusion

2. Scalp Segmentation

3. Brain Segmentation

67	Set	Cerebellar low point
x 172		
y 158	Set	White matter seed point
z 150		
0.1 Fill level [0, 1]		0.1 Threshold [0, 1]

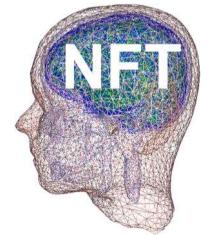
4. Outer Skull Segmentation

z 95	Set	Center of one eye
------	-----	-------------------

5. Inner Skull Segmentation

< Prev Run Next >

Segmenting skull...



Segmentation

NFT: MR segmentation

Figure 2

File Edit View Insert Tools Desktop Window Help

(x,y,z)= (150, 67, 95)

Coronal view

Sagittal view

Image Segmentation

Swap L/R Check inhomogeneity

1. Anisotropic Filtering

5 Number of iterations

3 Image diffusion

2. Scalp Segmentation

3. Brain Segmentation

67 Set Cerebellar low point

x 172 Set White matter seed point

y 158 Set

z 150

0.1 Fill level [0, 1] 0.1 Threshold [0, 1]

4. Outer Skull Segmentation

z 95 Set Center of one eye

5. Inner Skull Segmentation

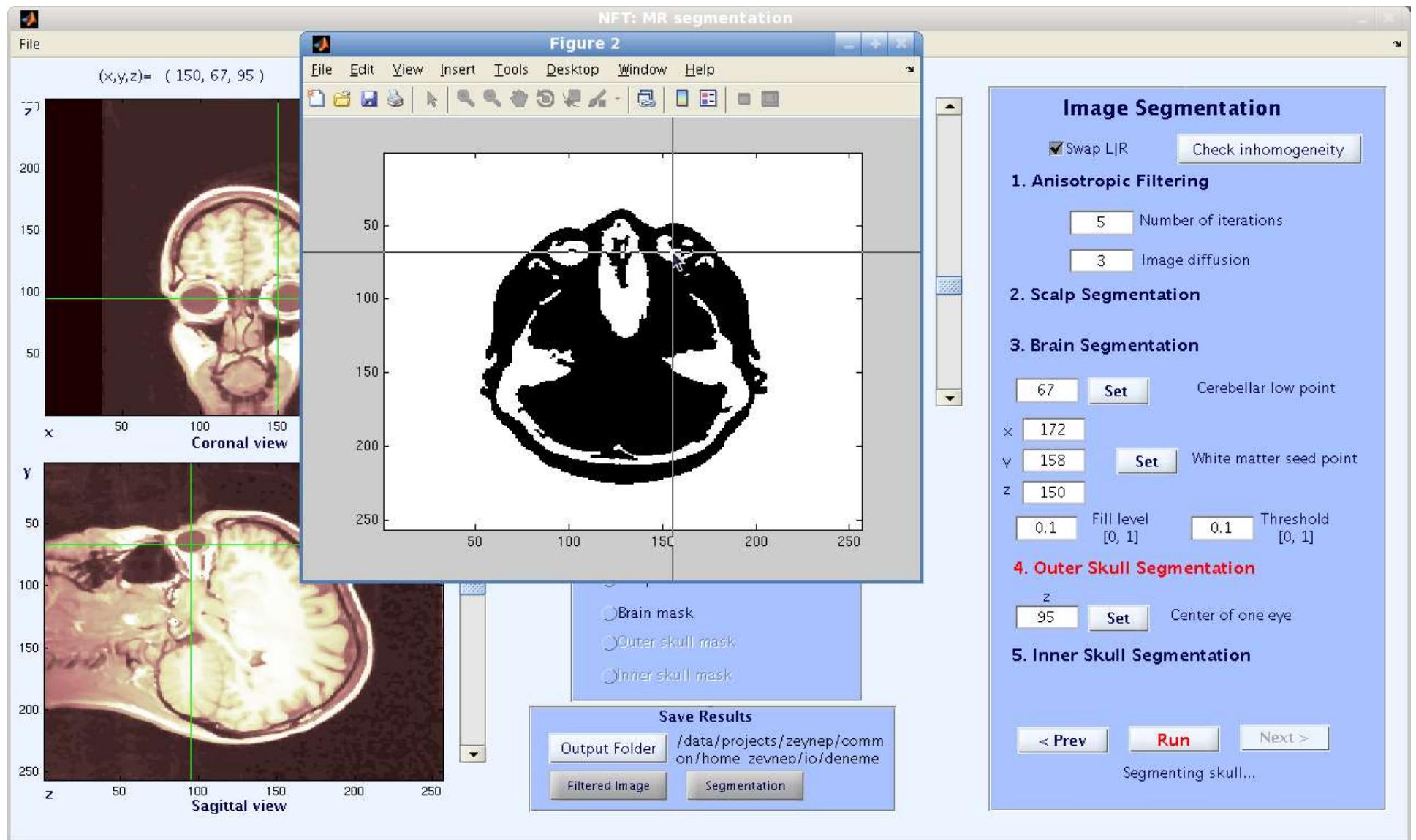
< Prev Run Next >

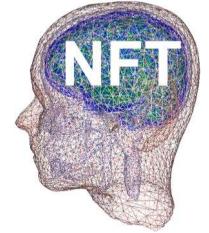
Save Results

Output Folder /data/projects/zeynep/comm on/home zeynep/io/deneme

Filtered Image Segmentation

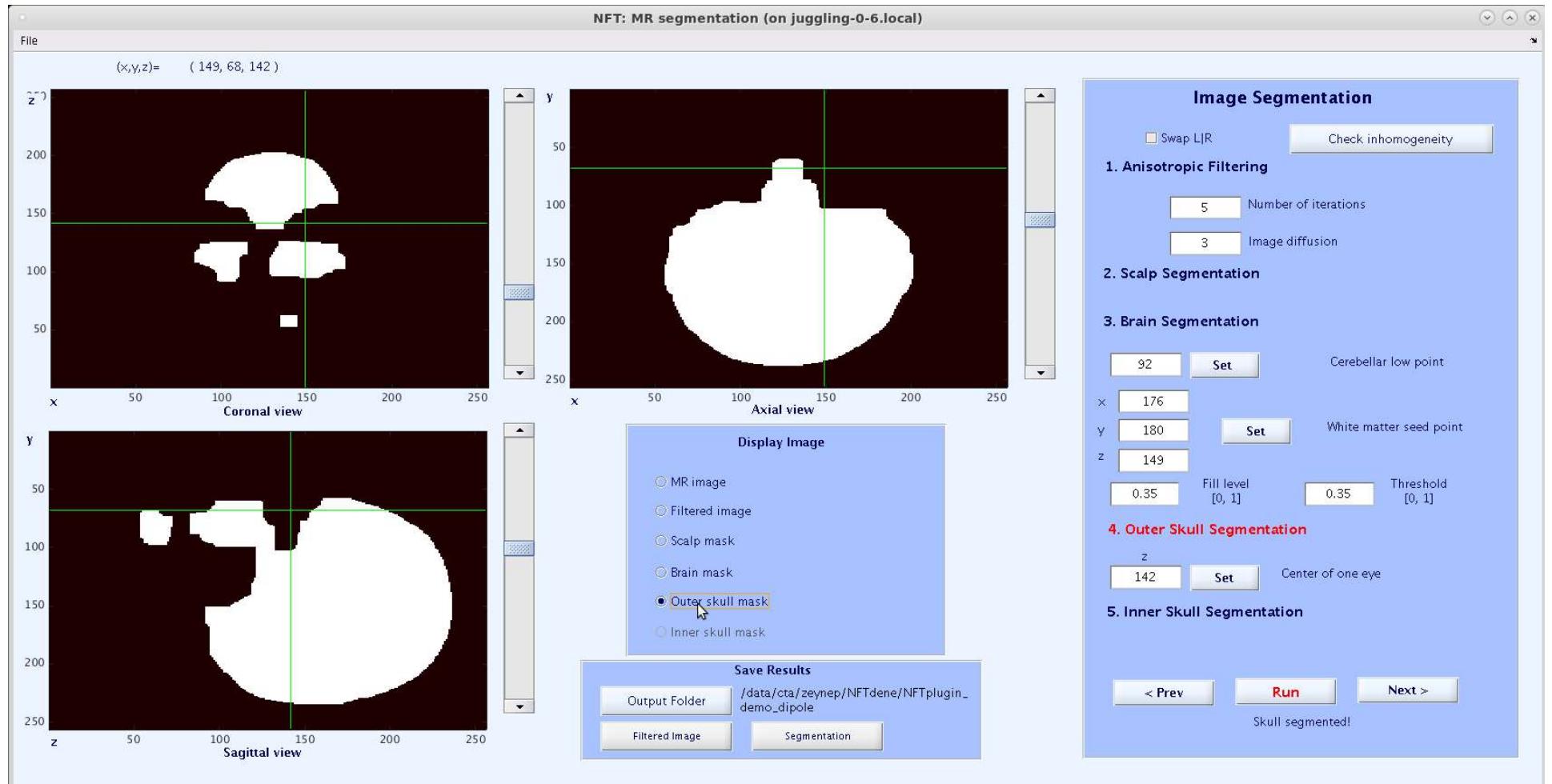
Segmenting skull...

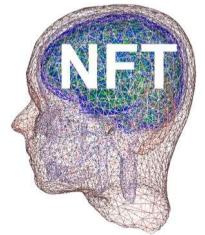




Segmentation

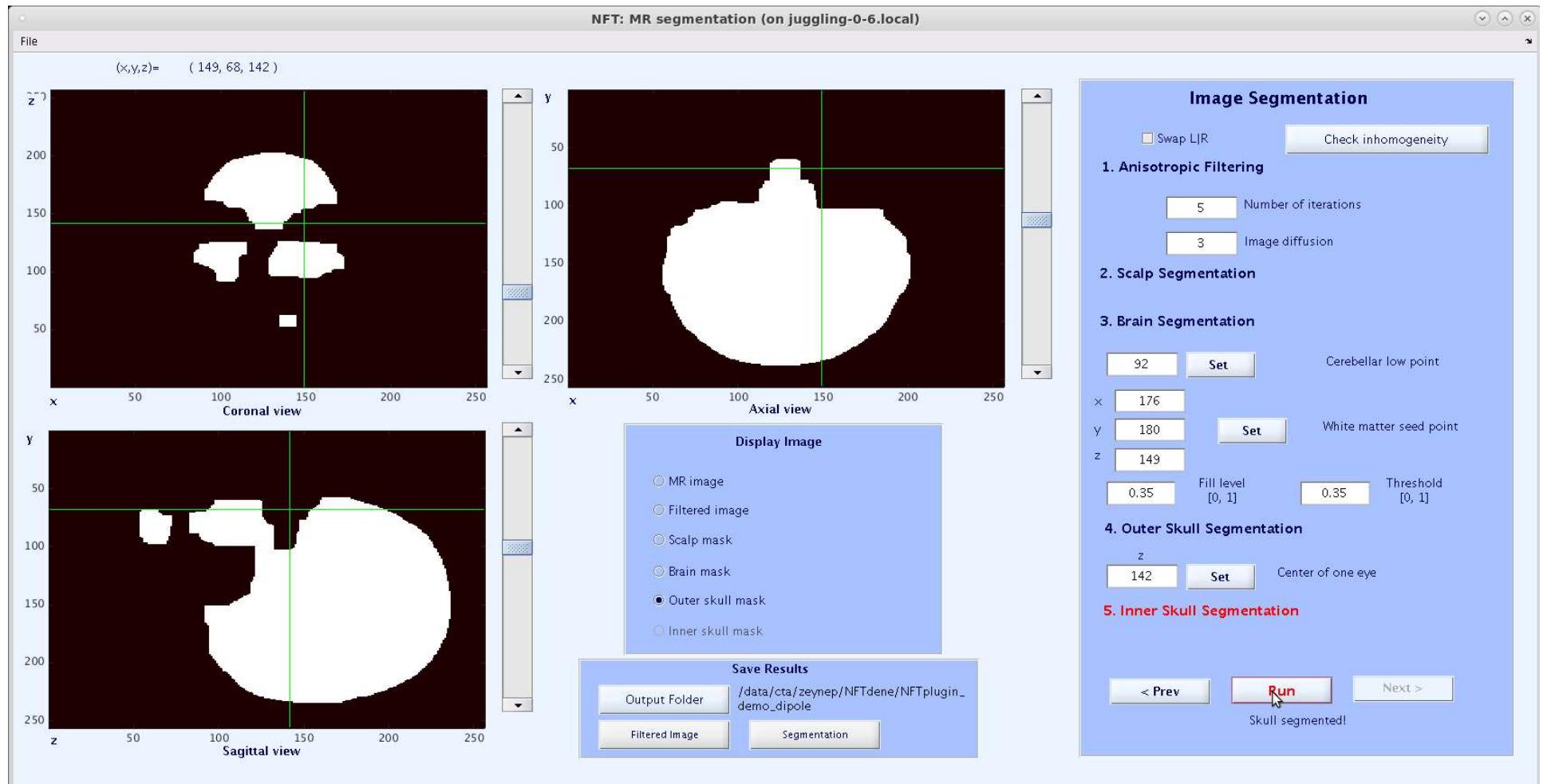
View skull segmentation

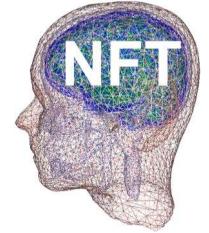




Segmentation

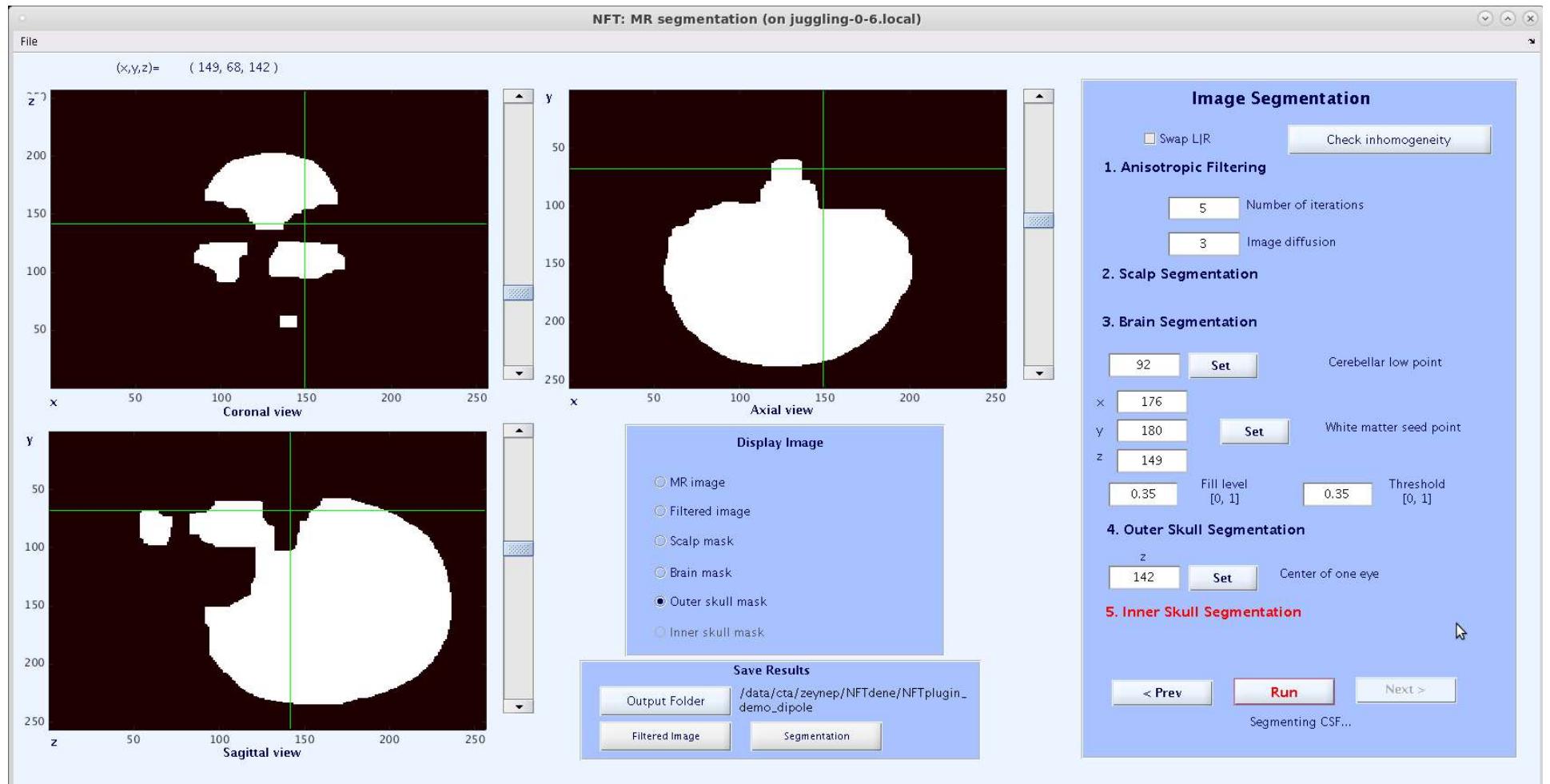
Click 'Next' for CSF segmentation

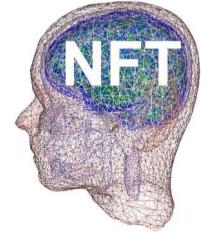




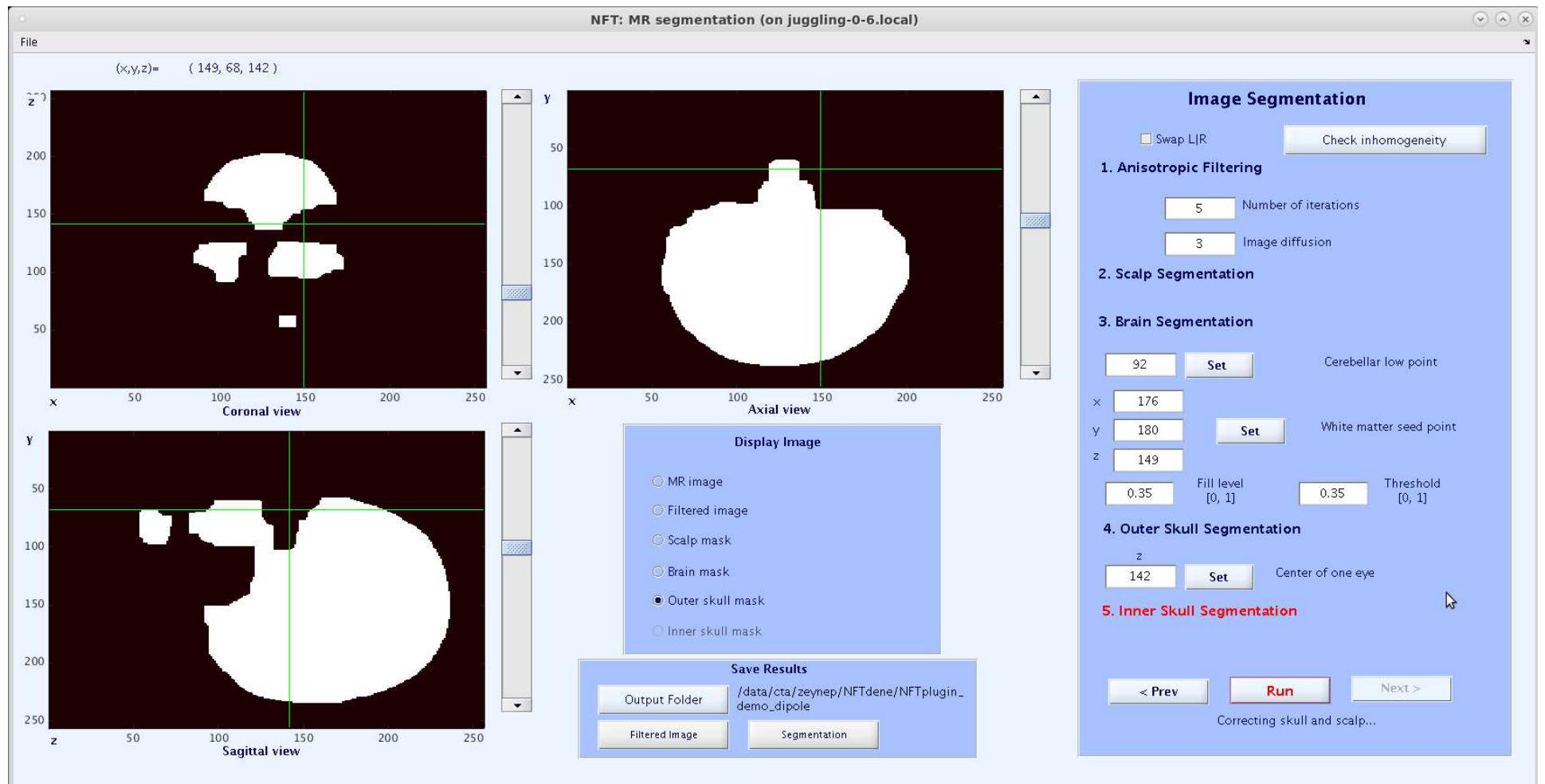
Segmentation

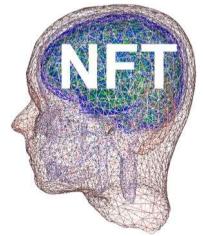
Click 'Run' for CSF segmentation





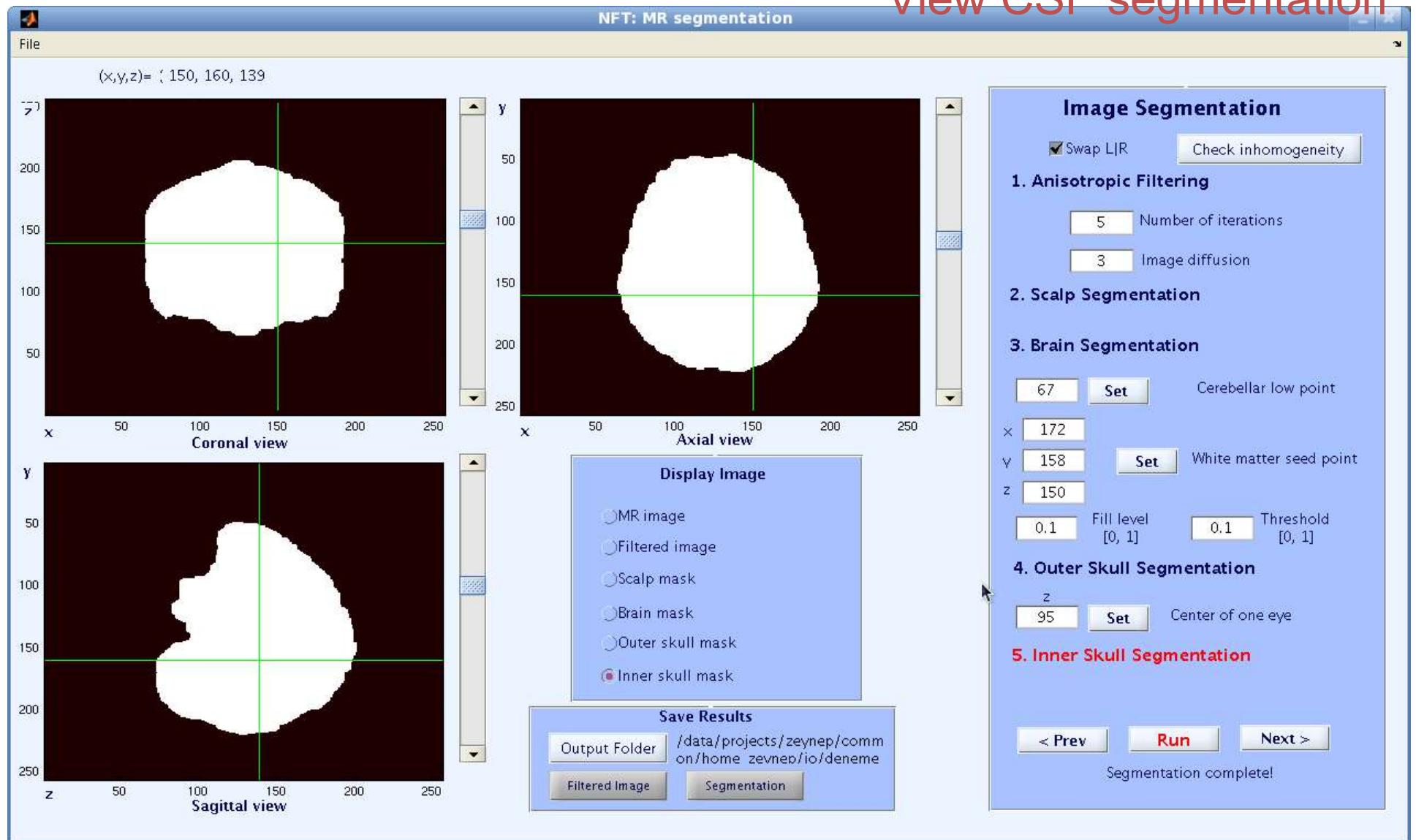
Segmentation

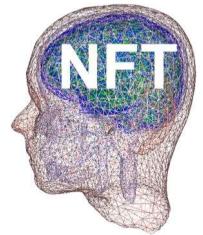




Segmentation

View CSF segmentation





Segmentation

NFT: MR segmentation

(x,y,z)= (150, 160, 139)

Coronal view Axial view Sagittal view

Image Segmentation

Swap L/R Check inhomogeneity

1. Anisotropic Filtering

Number of iterations: 5 Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation

x: 172	Set	Cerebellar low point
y: 158	Set	White matter seed point
z: 150		
0.1	Fill level [0, 1]	0.1 Threshold [0, 1]

4. Outer Skull Segmentation

z: 95 Set Center of one eye

5. Inner Skull Segmentation

Display Image

- MR image
- Filtered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Save Results

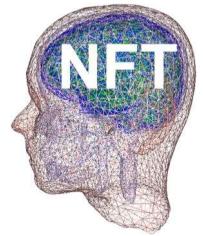
Output Folder: /data/projects/zeynep/comm on/home zeynep/io/deneme

Filtered Image Segmentation

Save filtered image

< Prev Run Next >

Saving filtered image as SubjectA_filtered.mat



Segmentation

NFT: MR segmentation

(x,y,z)= (150, 160, 139)

Coronal view Axial view Sagittal view

Image Segmentation

Swap L/R Check inhomogeneity

1. Anisotropic Filtering

5 Number of iterations
3 Image diffusion

2. Scalp Segmentation

3. Brain Segmentation

67	Set	Cerebellar low point
x 172		
y 158	Set	White matter seed point
z 150		
0.1 Fill level [0, 1]		0.1 Threshold [0, 1]

4. Outer Skull Segmentation

z 95 Set Center of one eye

5. Inner Skull Segmentation

< Prev Run Next >

Filtered image saved as SubjectA_filtered.mat

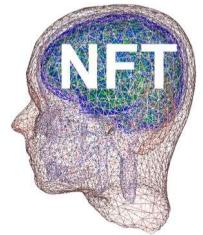
Display Image

- MR image
- Filtered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Save Results

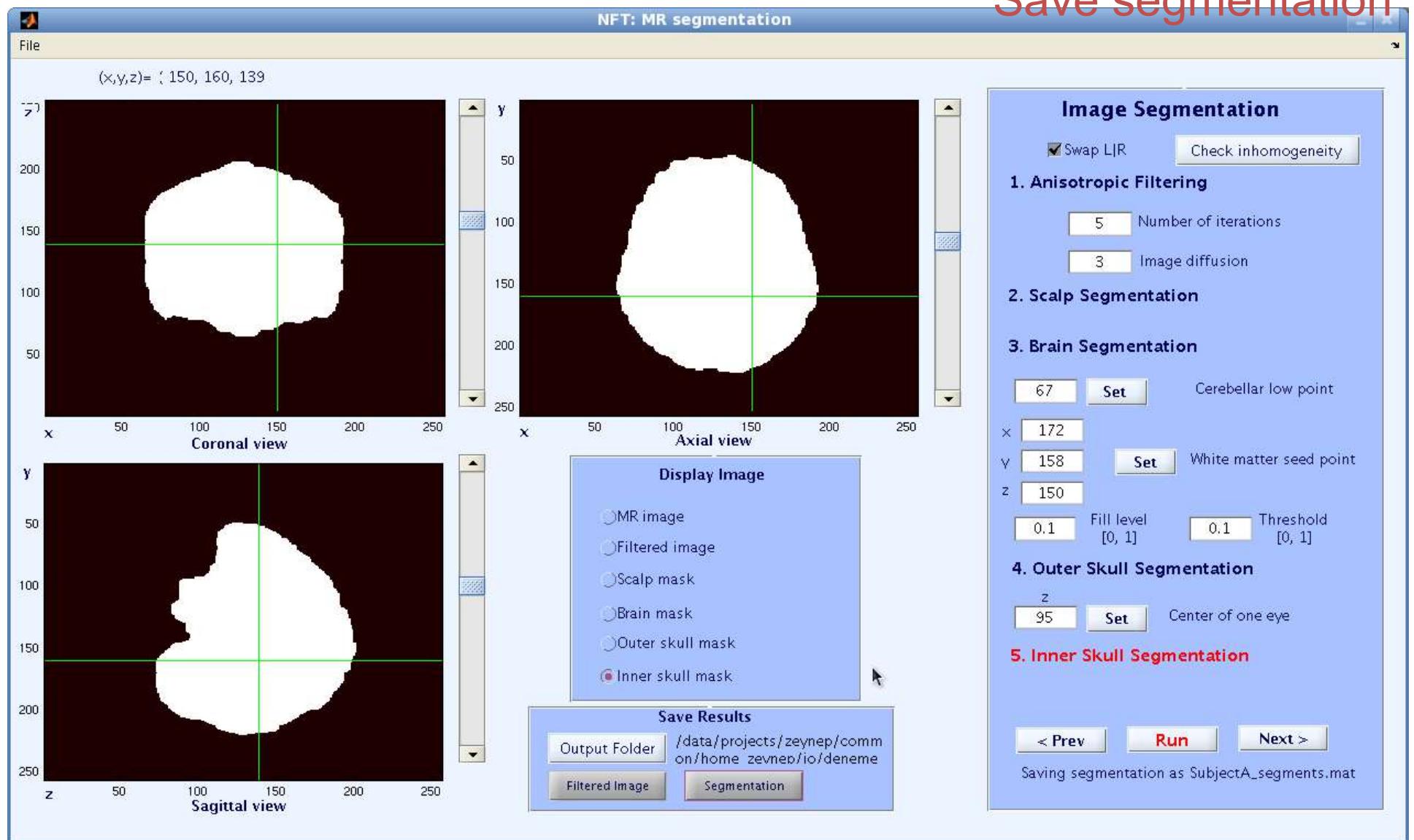
Output Folder: /data/projects/zeynep/comm on/home zeynep/io/deneme

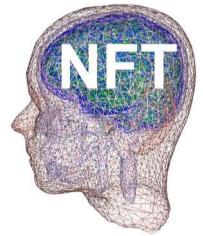
Filtered Image Segmentation



Segmentation

Save segmentation





Segmentation

NFT: MR segmentation

(x,y,z)= (150, 160, 139)

Coronal view Axial view Sagittal view

Image Segmentation

Swap L/R Check inhomogeneity

1. Anisotropic Filtering

5 Number of iterations
3 Image diffusion

2. Scalp Segmentation

3. Brain Segmentation

67	Set	Cerebellar low point
x 172		
y 158	Set	White matter seed point
z 150		
0.1 Fill level [0, 1]		0.1 Threshold [0, 1]

4. Outer Skull Segmentation

z 95 Set Center of one eye

5. Inner Skull Segmentation

< Prev Run Next >

Segmentation saved as SubjectA_segments.mat

Display Image

- MR image
- Filtered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Save Results

Output Folder /data/projects/zeynep/comm on/home zeynep/io/deneme

Filtered Image 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

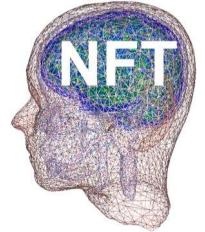
>> load SubjectA_mri
>> mri

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]

fx >> |
```



Mesh Generation

From a magnetic Resonance Image

Image Segmentation

Mesh Generation

Source Space Generation

Electrode Co-Registration...

NFT: Mesh generation (on juggling-0-6.local)

Load Segmentation /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/jc_segments

Output Folder /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole

of layers: 4 Mesh name: jc

Linear Quadratic Number of nodes per layer: 7000

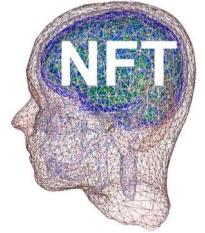
Local mesh refinement Edge length/
Distance between meshes: 2.1

Start Mesh Generation

Status

Generate linear FEM mesh Generate quadratic FEM mesh

Generate Mesh for a 3 or 4 layer head model



Mesh generation

Click local mesh refinement

NFT: Mesh generation (on juggling-0-6.local)

Load Segmentation /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/jc_segments

Output Folder /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/

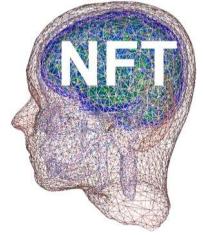
4 # of layers Mesh name: jc

Linear Quadratic 7000 Number of nodes per layer

Local mesh refinement
2.1 Edge length/
Distance between meshes Start Mesh Generation

Status

Generate linear FEM mesh Generate quadratic FEM mesh



Mesh generation

NFT: Mesh generation (on juggling-0-6.local)

Load Segmentation /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/jc_segments

Output Folder /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole

of layers: 4 Mesh name: jc

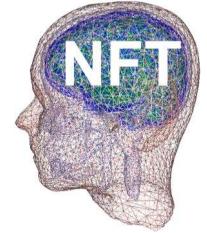
Linear Number of nodes per layer: 7000
 Quadratic

Local mesh refinement Edge length/
Distance between meshes: 2.1

Start Mesh Generation

Mesh saved!

Generate linear FEM mesh **Generate quadratic FEM mesh**



Mesh generation

NFT: Mesh generation (on juggling-0-6.local)

Load Segmentation /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/jc_segments

Output Folder /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole

of layers: 4 Mesh name: jc

Linear Number of nodes per layer: 7000
 Quadratic

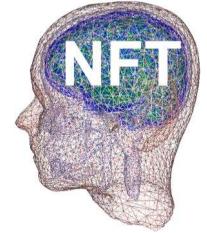
Local mesh refinement
Edge length/
Distance between meshes: 2.1

Start Mesh Generation

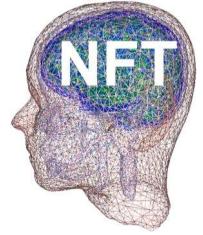
Mesh saved!

Generate linear FEM mesh 

Generate quadratic FEM mesh



- ◆ $[C, E] = \text{ReadSMF}('Scalp.smf', 0, 0, 0, 1);$
- ◆ $\text{Plotmesh}(E(:, 2:4), C(:, 2:4))$



Mesh generation

NFT: Mesh generation (on juggling-0-6.local)

Load Segmentation /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/jc_segments

Output Folder /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole

4 # of layers Mesh name: jc

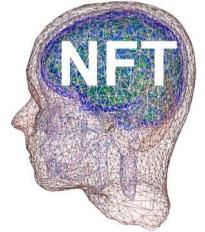
Linear Quadratic 7000 Number of nodes per layer

Local mesh refinement
Edge length/
Distance between meshes 2.1

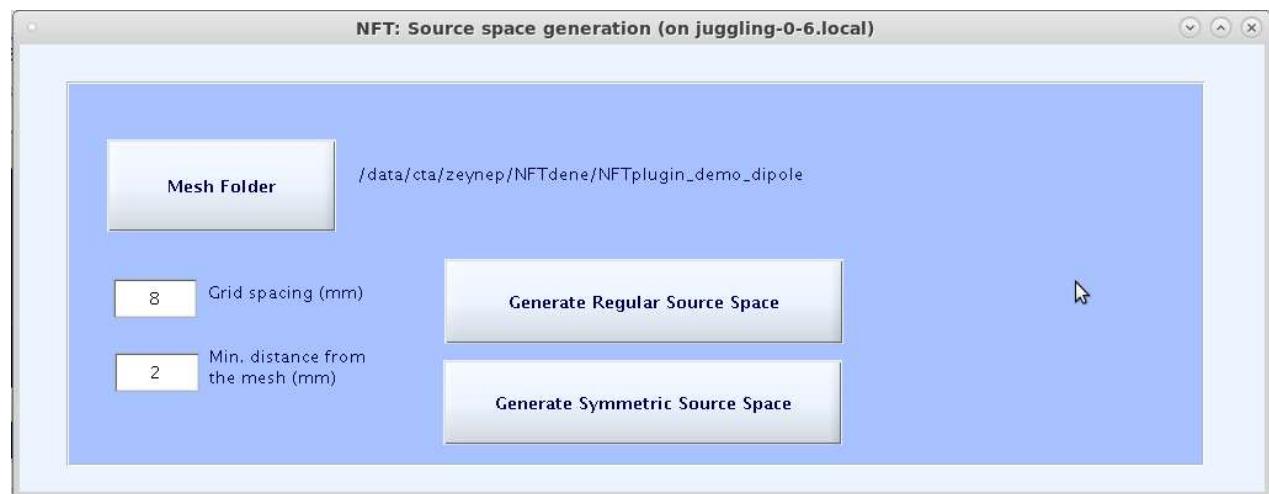
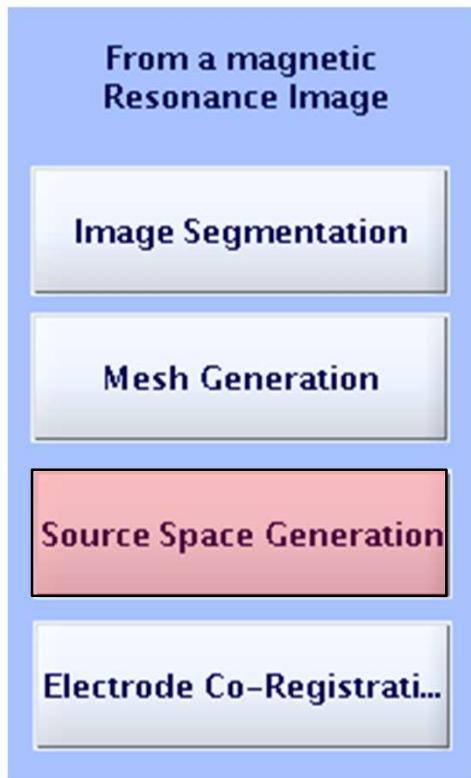
Start Mesh Generation

Linear FEM mesh generated!

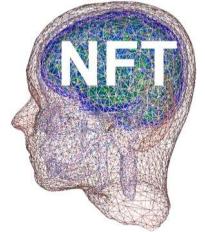
Generate linear FEM mesh → Generate quadratic FEM mesh



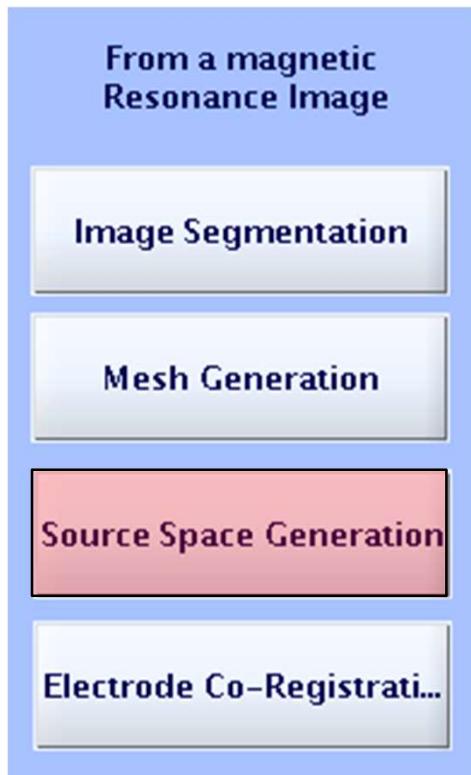
Source Space Generation

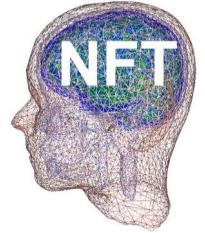


Generates a simple source space:
Regular Grid inside the brain
With a given spacing and distance to the mesh



Source Space Generation





Electrode Co-registration

From a magnetic Resonance Image

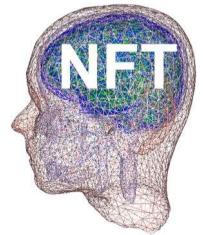
Image Segmentation

Mesh Generation

Source Space Generation

Electrode Co-Registration...

A vertical sidebar menu with four main items: "Image Segmentation", "Mesh Generation", "Source Space Generation", and "Electrode Co-Registration...". The last item is highlighted with a pink background.



Electrode Co-registration

From a magnetic Resonance Image

Image Segmentation

Mesh Generation

Source Space Generation

Electrode Co-Registration...

NFT: Electrode co-registration (on juggling-0-6.local)

Load sensor locations Electrode file name

Mesh Folder /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole

Initial co-registration

Complete co-registration

Save initial reg.

Select File to Open (on juggling-0-6.local)

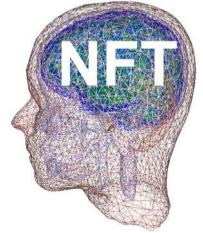
Look In: NFTplugin_demo_dipole

File Name: jc_fid.sfp

Files of Type: All Files

Open Cancel

The screenshot shows a software interface for electrode co-registration. On the left, a vertical menu lists steps: 'From a magnetic Resonance Image', 'Image Segmentation', 'Mesh Generation', 'Source Space Generation', and 'Electrode Co-Registration...'. The 'Electrode Co-Registration...' option is highlighted with a pink background. The main window title is 'NFT: Electrode co-registration (on juggling-0-6.local)'. It contains several input fields and buttons: 'Load sensor locations' (button), 'Electrode file name' (text field containing '/data/cta/zeynep/NFTdene/NFTplugin_demo_dipole'), 'Mesh Folder' (button), and 'Save initial reg.' (button). A 'Select File to Open' dialog box is open in the foreground, showing a file list with 'jc_fid.sfp' selected. The dialog includes 'Look In:' dropdown set to 'NFTplugin_demo_dipole', 'File Name:' field with 'jc_fid.sfp', 'Files of Type:' dropdown set to 'All Files', and 'Open' and 'Cancel' buttons.



Electrode Co-registration

From a magnetic Resonance Image

Image Segmentation

Mesh Generation

Source Space Generation

Electrode Co-Registration

NFT: Electrode co-registration (on juggling-0-6.local)

Load sensor locations /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/jc_fid.sfp

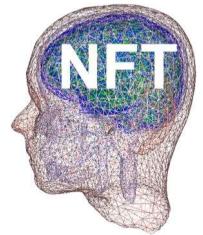
Mesh Folder /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole

Initial co-registration Translation
Rotation

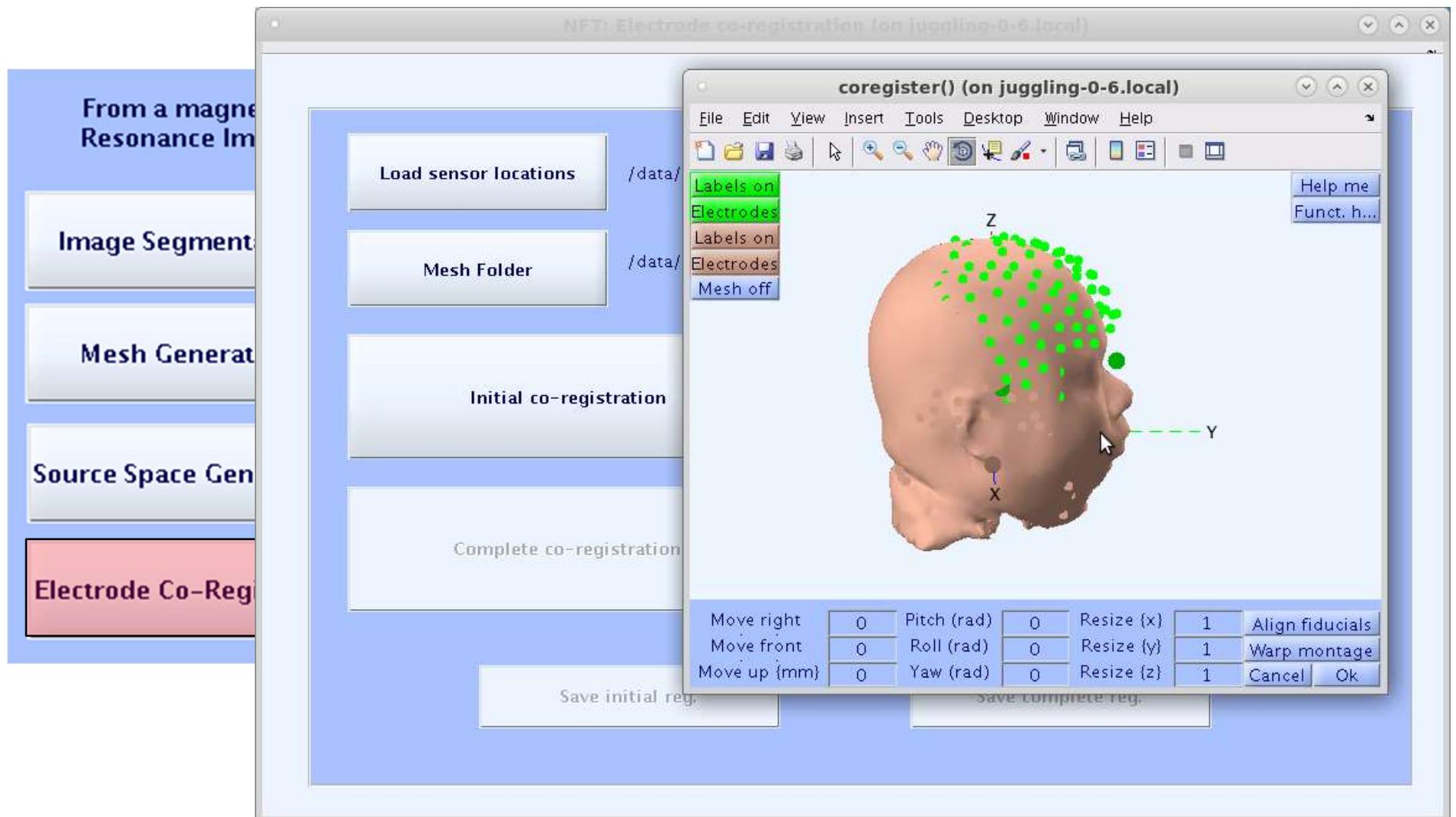
Complete co-registration Translation
Rotation

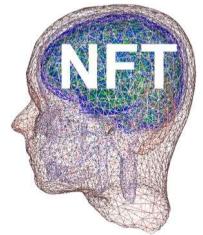
Save initial reg. Save complete reg.

A screenshot of a software application window titled "NFT: Electrode co-registration (on juggling-0-6.local)". The window contains several buttons and input fields. On the left, there's a vertical menu with options: "From a magnetic Resonance Image", "Image Segmentation", "Mesh Generation", "Source Space Generation", and "Electrode Co-Registration". The "Electrode Co-Registration" option is highlighted with a red background. Inside the main window, there are two sets of buttons for loading sensor locations and mesh folders, both pointing to local paths. Below these are two large buttons for "Initial co-registration" and "Complete co-registration", each with "Translation" and "Rotation" sub-options. At the bottom, there are two buttons for saving the registration results: "Save initial reg." and "Save complete reg.".



Electrode Co-registration





Electrode Co-registration

NFT: Electrode co-registration (on juggling-0-6 (oral))

Load sensor locations /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/juggling-0-6/sensor_locations

Mesh Folder /data/cta/zeynep/NFTdene/NFTplugin_demo_dipole/juggling-0-6/meshes

Initial co-registration

Translation -3 -10 0
Rotation 0 0 0

Complete co-registration

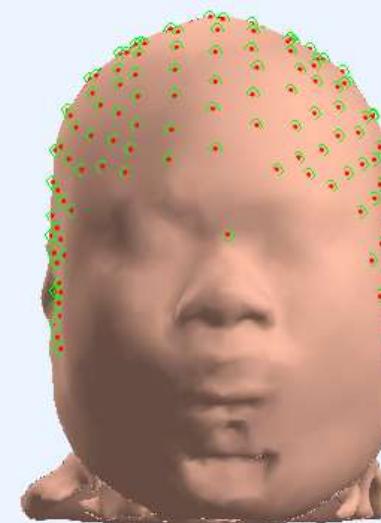
Translation
Rotation

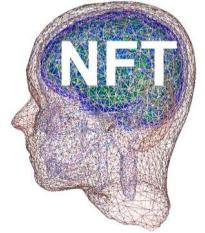
Save initial reg.

Save complete reg.

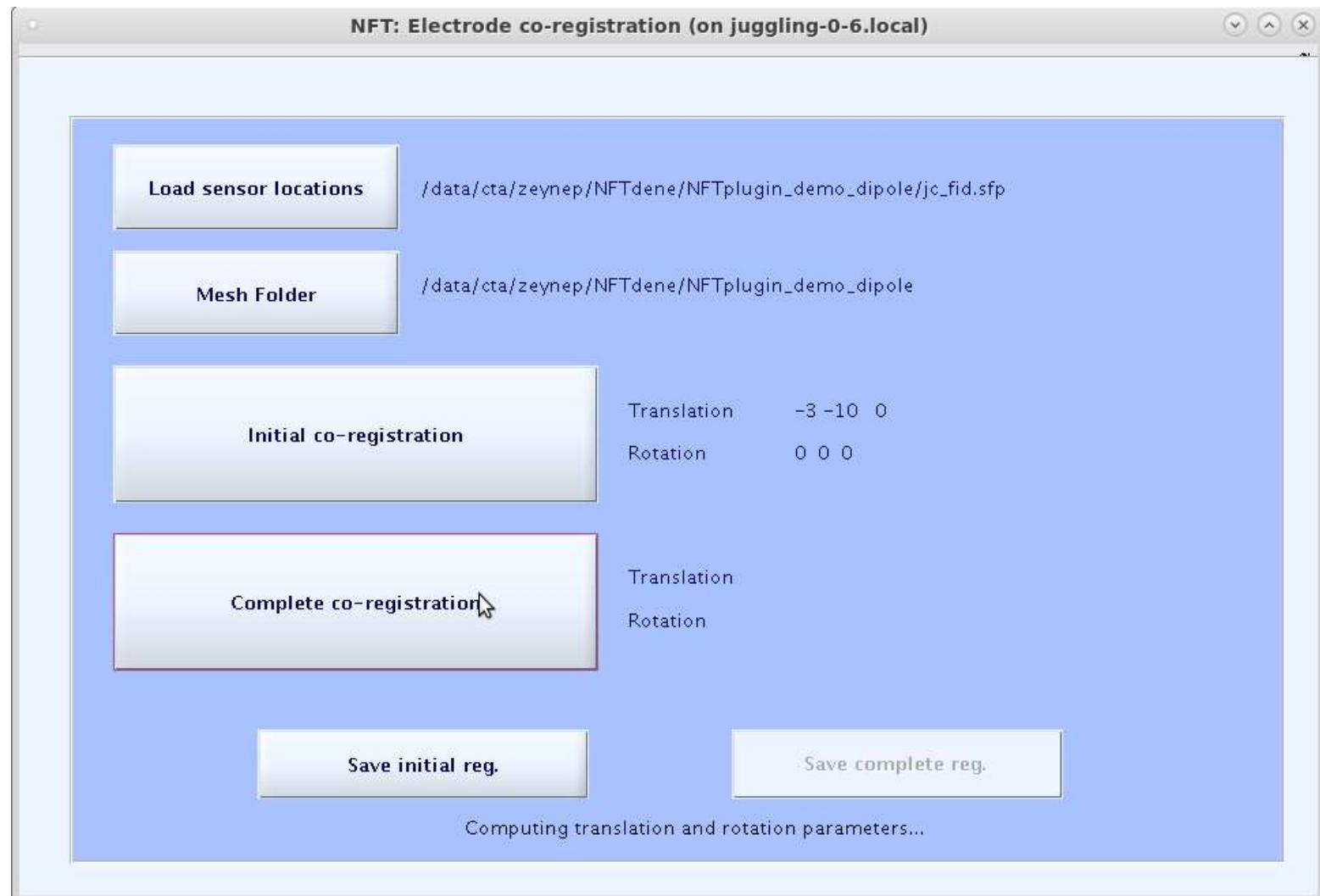
Figure: Co-registered electrode locations (initial) (on juggling-0-6 (oral))

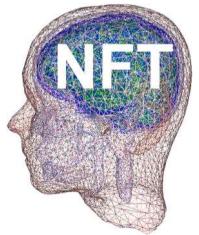
File Edit View Insert Tools Desktop Window Help



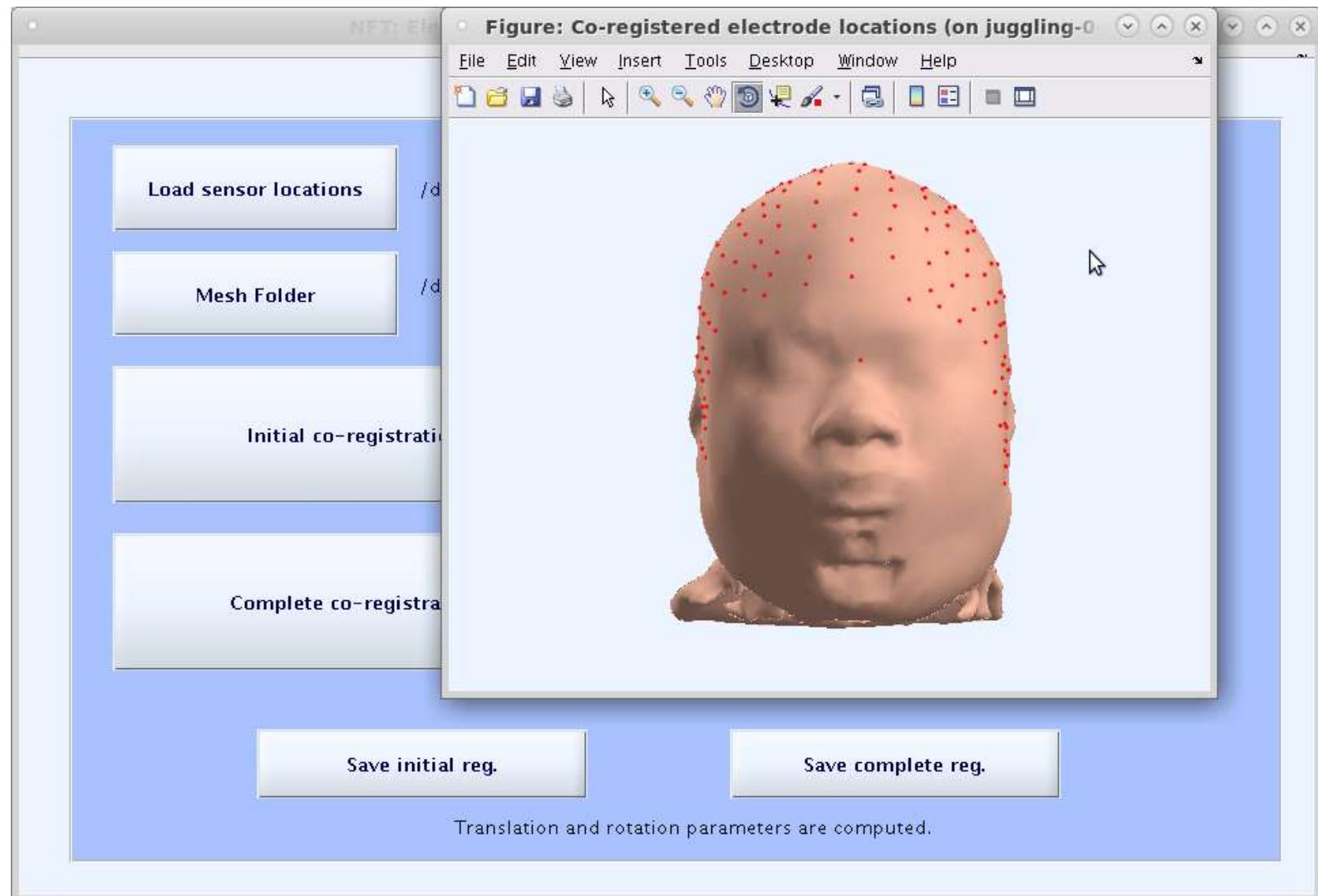


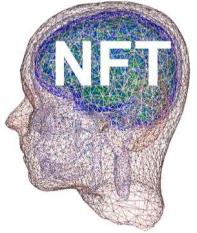
Electrode co-registration



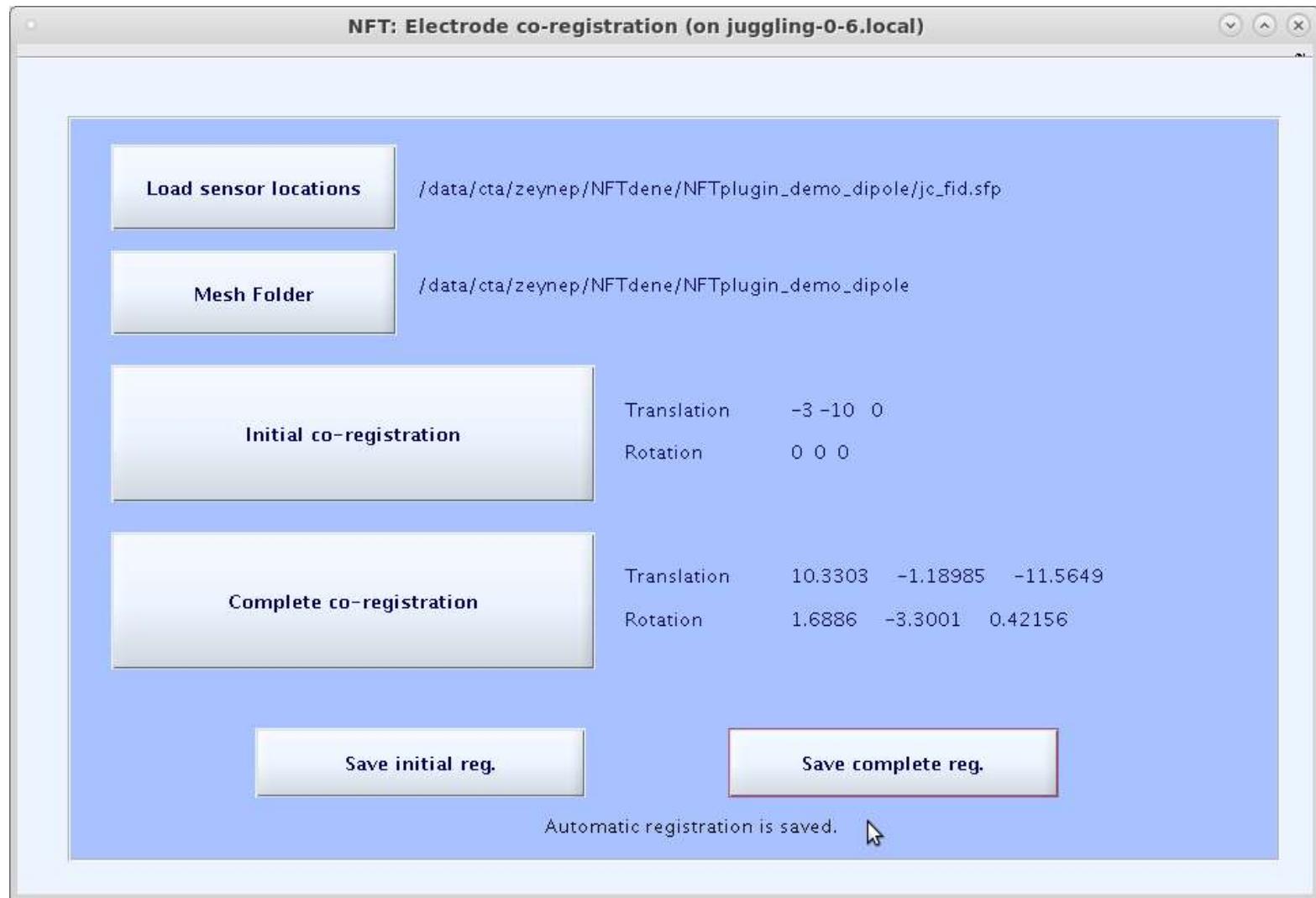


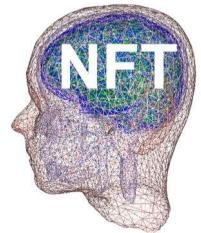
Electrode co-registration





Mesh generation





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

sens =

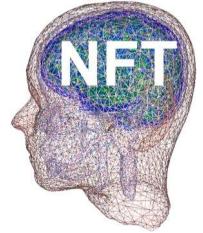
    fn: '/data/cta/zeynep/NFTdene/JC/jc_fid.sfp'
    eloc: [1x208 struct]
    pnt: [208x3 double]
    ind: [1x208 double]
    param: [1x1 struct]

>> sens.param

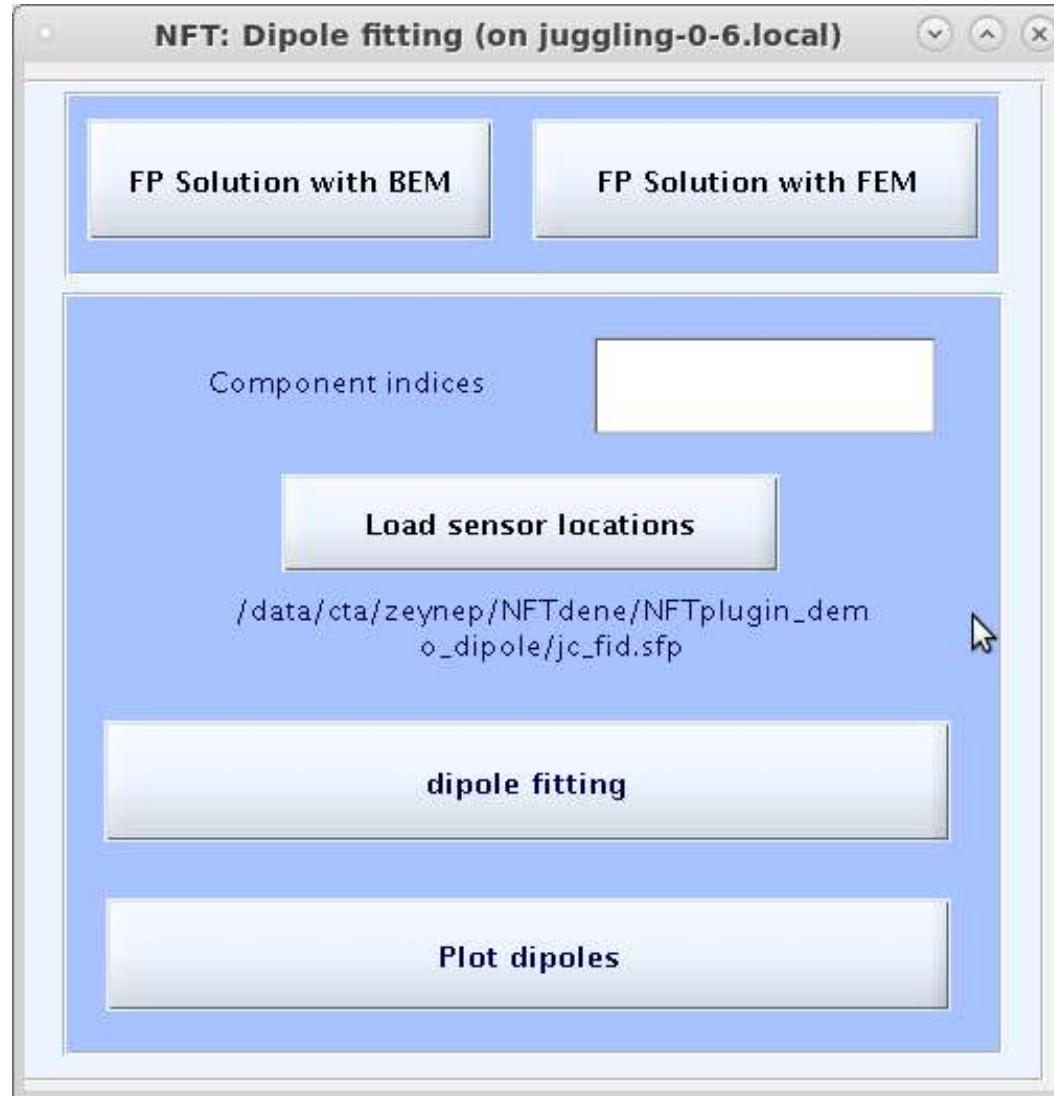
ans =

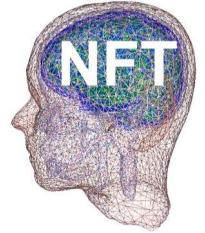
    init: [-3 -10 0 0 0 0 1 1 1]
    auto: [10.3303 -1.1899 -11.5649 1.6886 -3.3001 0.4216]

>> |
```

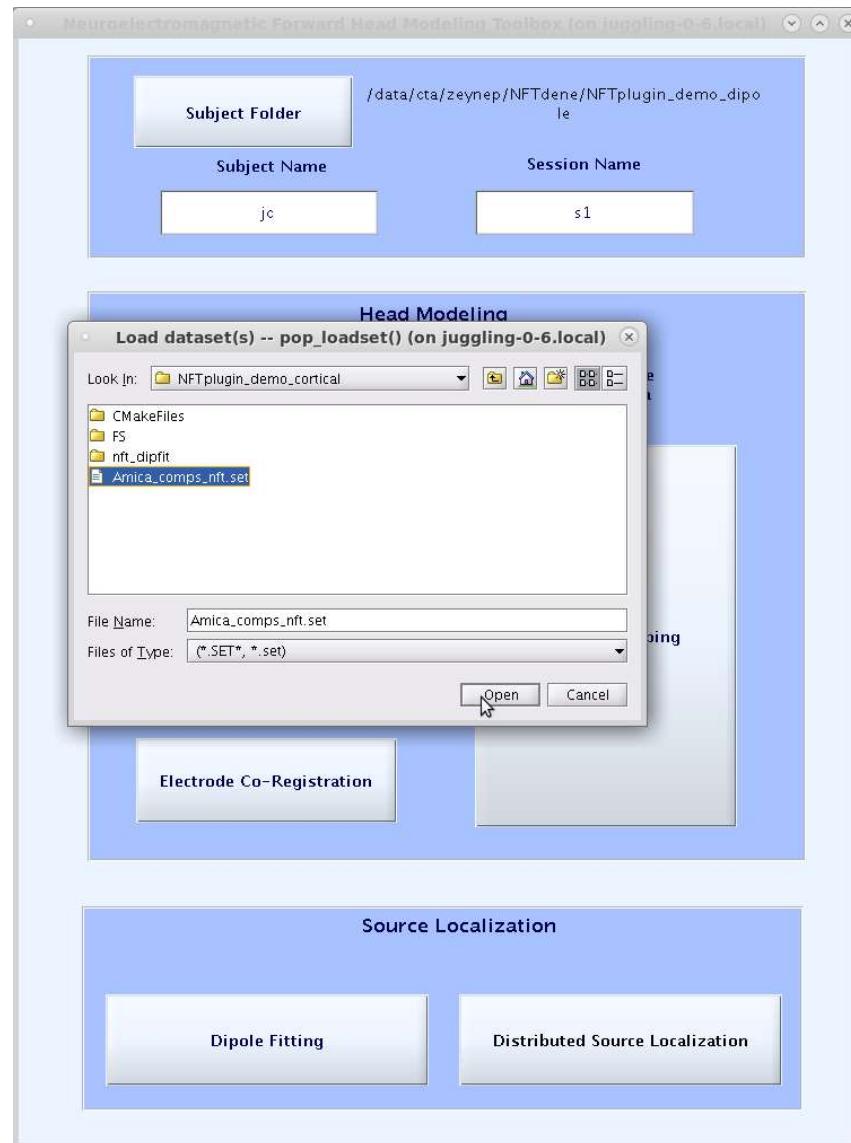


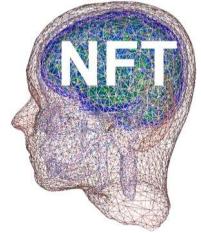
Dipole source localization





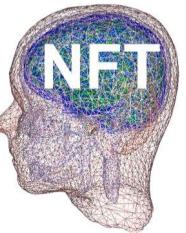
Select EEG data





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 Model Generation

NFT: Forward problem solution (on juggling-0-6.local)

BEM Mesh Info

jc	Mesh Name
Show Mesh	
4	Number of Layers
16151	Number of Nodes
32286	Number of Elements
3	Number of Nodes/Element

BEM Model

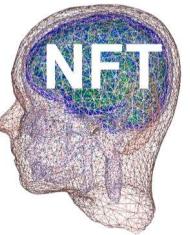
jc	Model Name		
Enter conductivity values:			
0.33	Scalp	0.0132	Skull
0.33	Brain	1.79	CSF
<input checked="" type="checkbox"/> Modified (Isolated Problem Approach)			
Create Model			
Value Changed!			

Session

s1	Session Name
Load Sensors	
<input checked="" type="radio"/> Mesh Coordinates	Load
<input type="radio"/> Mesh Node List	Show Sensors
Generate transfer matrix	
Value Changed!	

Forward Problem Solution

Load Source Space
Compute Lead Field Matrix
Plot Potential Distribution
For Dipole <input type="text"/>



Forward Problem Solution with BEM

Forward Model Generation

NFT: Forward problem solution (on juggling-0-6.local)

BEM Mesh Info

jc	Mesh Name
Show Mesh	
4	Number of Layers
16151	Number of Nodes
32286	Number of Elements
3	Number of Nodes/Element

BEM Model

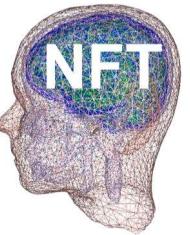
jc	Model Name		
Enter conductivity values:			
0.33	Scalp	0.0132	Skull
0.33	Brain	1.79	CSF
<input checked="" type="checkbox"/> Modified (Isolated Problem Approach)			
Create Model			
Value Changed!			

Session

s1	Session Name
Load Sensors	
<input checked="" type="radio"/> Mesh Coordinates	Load
<input type="radio"/> Mesh Node List	Show Sensors
Generate transfer matrix	
Value Changed!	

Forward Problem Solution

Load Source Space
Compute Lead Field Matrix
Plot Potential Distribution
For Dipole



Forward Problem Solution with BEM

Forward Model Generation

NFT: Forward problem solution (on juggling-0-6.local)

BEM Mesh Info

jc	Mesh Name
<input type="button" value="Show Mesh"/>	
4	Number of Layers
16151	Number of Nodes
32286	Number of Elements
3	Number of Nodes/Element

BEM Model

jc	Model Name		
Enter conductivity values:			
0.33	Scalp	0.0132	Skull
0.33	Brain	1.79	CSF
<input checked="" type="checkbox"/> Modified (Isolated Problem Approach)			
<input type="button" value="Create Model"/>			
BEM Model Created			

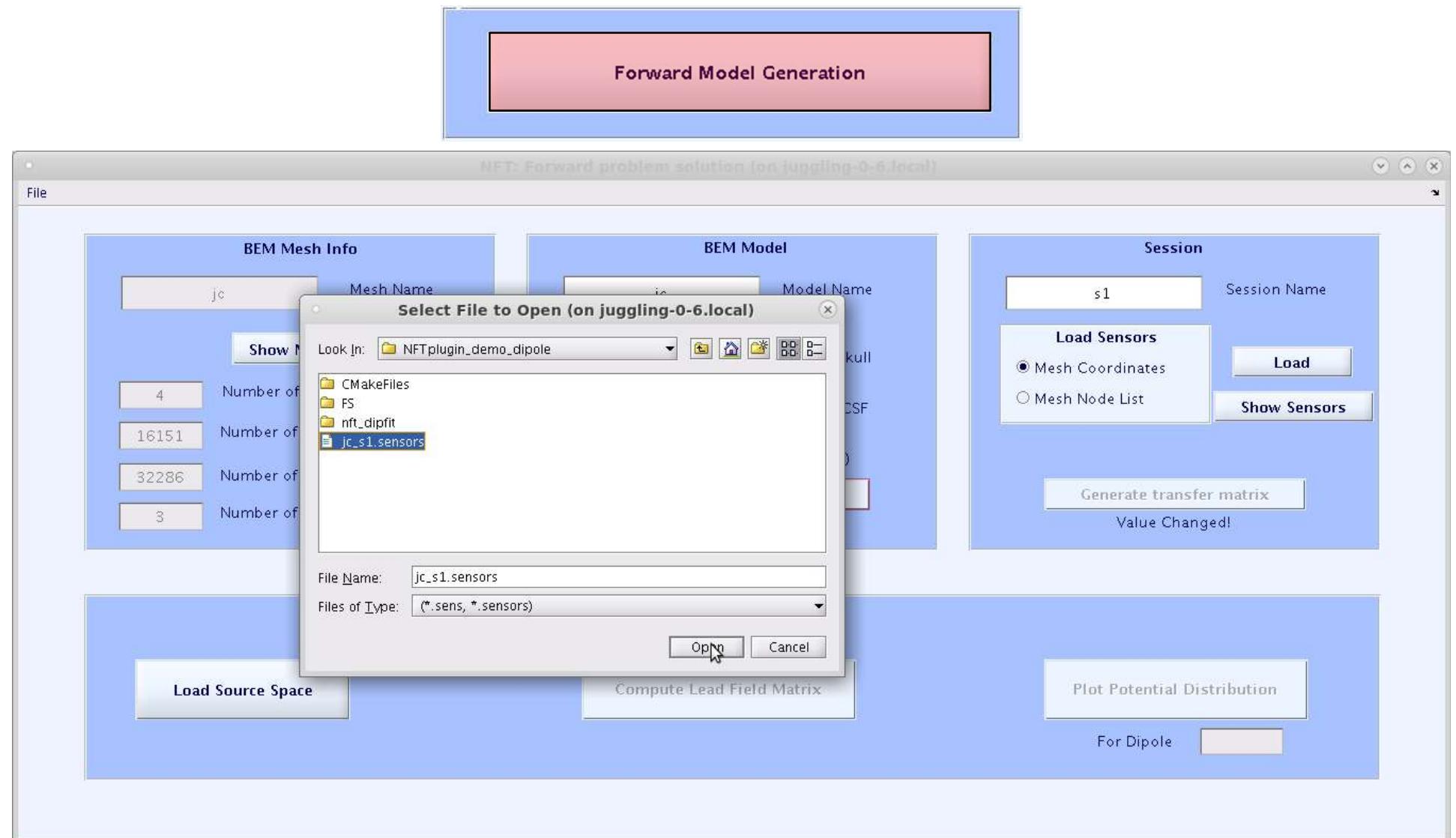
Session

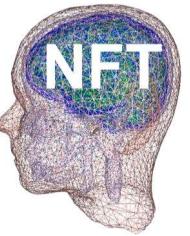
s1	Session Name
Load Sensors	
<input checked="" type="radio"/> Mesh Coordinates	<input type="button" value="Load"/>
<input type="radio"/> Mesh Node List	<input type="button" value="Show Sensors"/>
<input type="button" value="Generate transfer matrix"/>	
Value Changed!	

Forward Problem Solution

<input type="button" value="Load Source Space"/>
<input type="button" value="Compute Lead Field Matrix"/>
<input type="button" value="Plot Potential Distribution"/>
For Dipole <input type="text"/>

Forward Problem Solution with BEM





Forward Problem Solution with BEM

Forward Model Generation

NFT: Forward problem solution (on juggling-0-6.local)

BEM Mesh Info

jc	Mesh Name
<input type="button" value="Show Mesh"/>	
4	Number of Layers
16151	Number of Nodes
32286	Number of Elements
3	Number of Nodes/Element

BEM Model

Enter conductivity values:

0.33	Scalp	0.0132	Skull
0.33	Brain	1.79	CSF

Modified (Isolated Problem Approach)

BEM Model Loaded

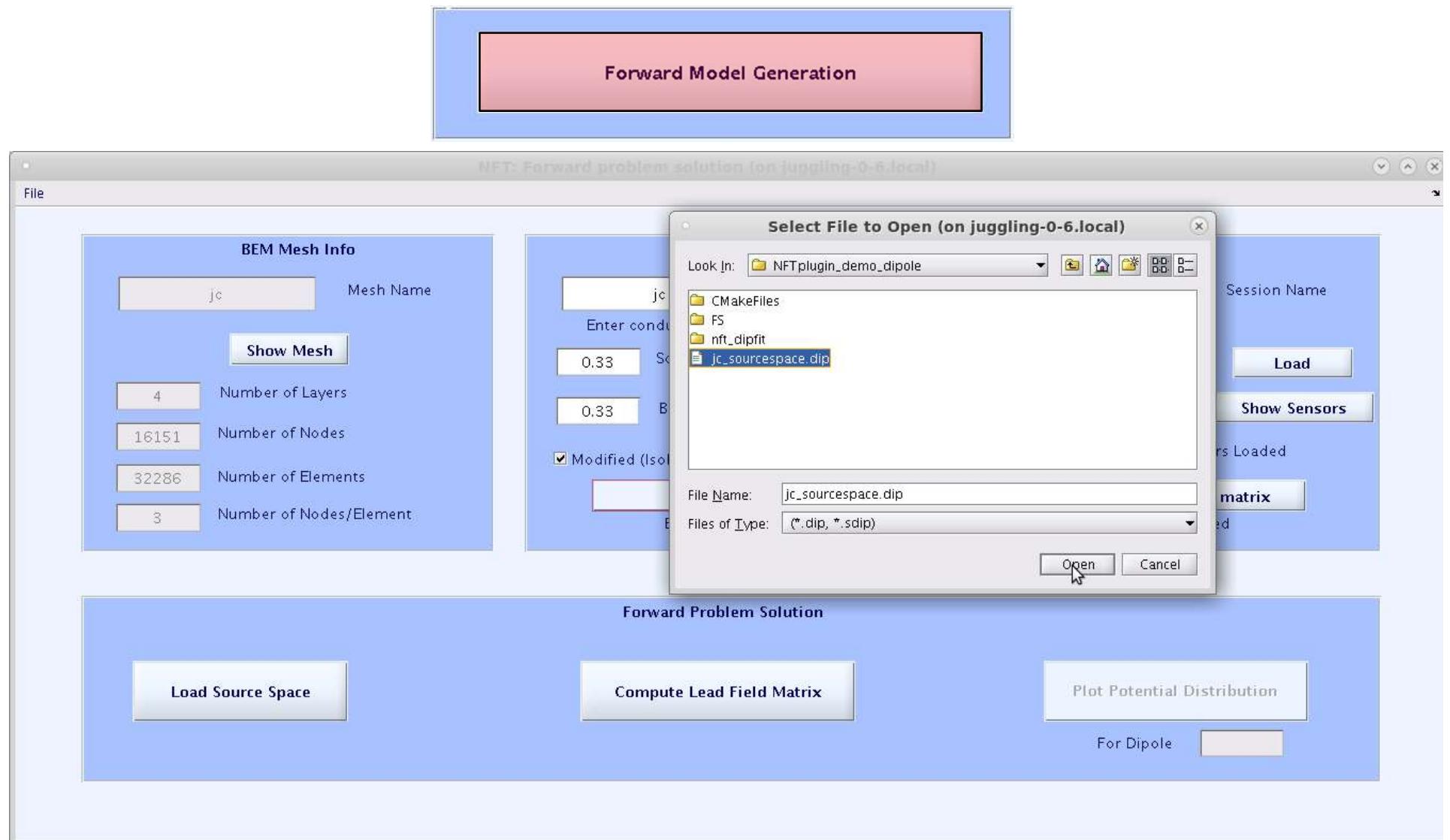
Session

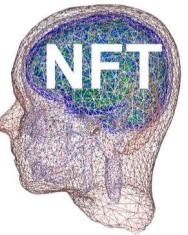
s1	Session Name
<input type="radio"/> Mesh Coordinates	<input type="button" value="Load"/>
<input type="radio"/> Mesh Node List	<input type="button" value="Show Sensors"/>
208	Sensors Loaded
<input type="button" value="Generate transfer matrix"/>	Session Loaded

Forward Problem Solution

For Dipole

Forward Problem Solution with BEM





Forward Problem Solution with BEM

Forward Model Generation

NFT: Forward problem solution (on juggling-0-6.local)

BEM Mesh Info

jc	Mesh Name
Show Mesh	
4	Number of Layers
16151	Number of Nodes
32286	Number of Elements
3	Number of Nodes/Element

BEM Model

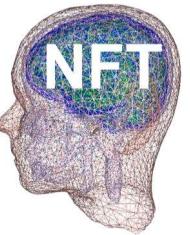
jc	Model Name		
Enter conductivity values:			
0.33	Scalp	0.0132	Skull
0.33	Brain	1.79	CSF
<input checked="" type="checkbox"/> Modified (Isolated Problem Approach)			
Create Model			
BEM Model Loaded			

Session

s1	Session Name
Load Sensors	
<input checked="" type="radio"/> Mesh Coordinates	Load
<input type="radio"/> Mesh Node List	Show Sensors
208	Sensors Loaded
Generate transfer matrix	
Session Loaded	

Forward Problem Solution

Load Source Space
7479 Dipoles Loaded
Compute Lead Field Matrix
Plot Potential Distribution
For Dipole <input type="text"/>



Forward Problem Solution with BEM

Forward Model Generation

NFT: Forward problem solution (on juggling-0-6.local)

BEM Mesh Info

jc	Mesh Name
Show Mesh	
4	Number of Layers
16151	Number of Nodes
32286	Number of Elements
3	Number of Nodes/Element

BEM Model

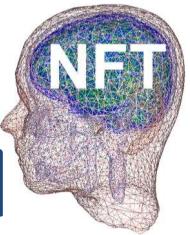
jc	Model Name		
Enter conductivity values:			
0.33	Scalp	0.0132	Skull
0.33	Brain	1.79	CSF
<input checked="" type="checkbox"/> Modified (Isolated Problem Approach)			
Create Model			
BEM Model Loaded			

Session

s1	Session Name
Load Sensors	
<input checked="" type="radio"/> Mesh Coordinates	Load
<input type="radio"/> Mesh Node List	Show Sensors
208	Sensors Loaded
Generate transfer matrix	
Session Loaded	

Forward Problem Solution

Load Source Space
7479 Dipoles Loaded
Compute Lead Field Matrix
LFM Computed
Plot Potential Distribution
For Dipole



Forward Problem Solution with BEM

Forward Model Generation

NFT: Forward problem solution (on juggling-0-6.local)

BEM Mesh Info

jc	Mesh Name
Show Mesh	
4	Number of Layers
16151	Number of Nodes
32286	Number of Elements
3	Number of Nodes/Element

BEM Model

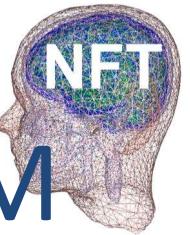
jc	Model Name		
Enter conductivity values:			
0.33	Scalp	0.0132	Skull
0.33	Brain	1.79	CSF
<input checked="" type="checkbox"/> Modified (Isolated Problem Approach)			
Create Model			
BEM Model Loaded			

Session

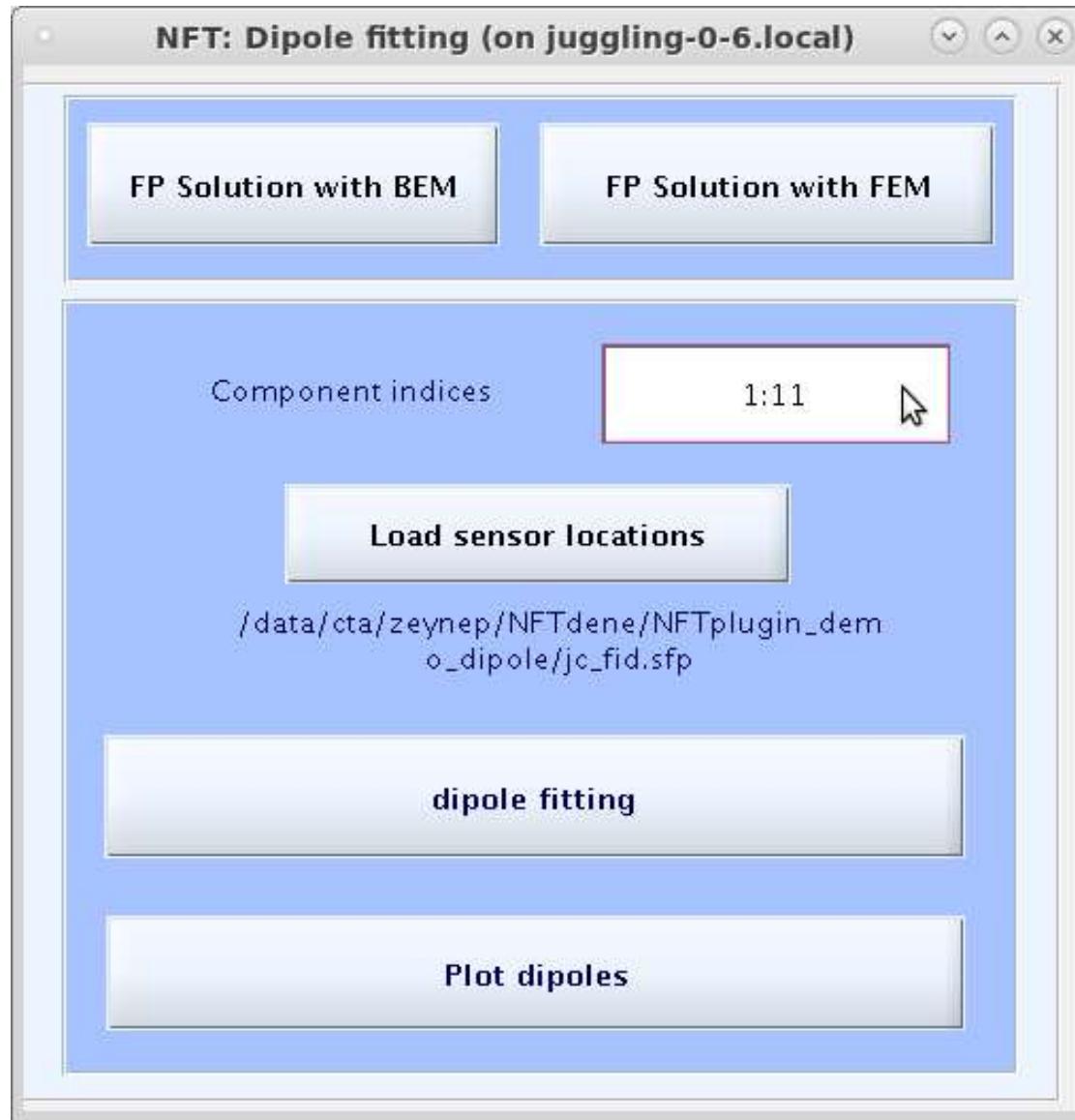
s1	Session Name
Load Sensors	
<input checked="" type="radio"/> Mesh Coordinates	Load
<input type="radio"/> Mesh Node List	Show Sensors
208	Sensors Loaded
Generate transfer matrix	
Session Loaded	

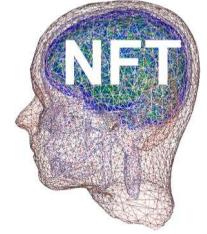
Forward Problem Solution

Load Source Space
7479 Dipoles Loaded
Compute Lead Field Matrix
LFM Computed
Plot Potential Distribution
For Dipole



Inverse Problem Solution with BEM

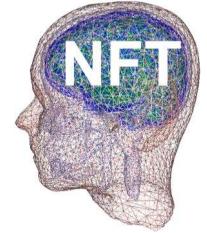




Output

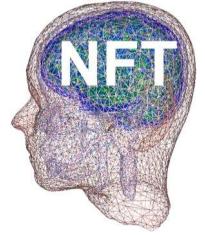
- ◆ Dipole source localization is saved in EEG structure, under EEG.etc.nft.
- ◆ After source localization with NFT, you can continue using EEGLAB;

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

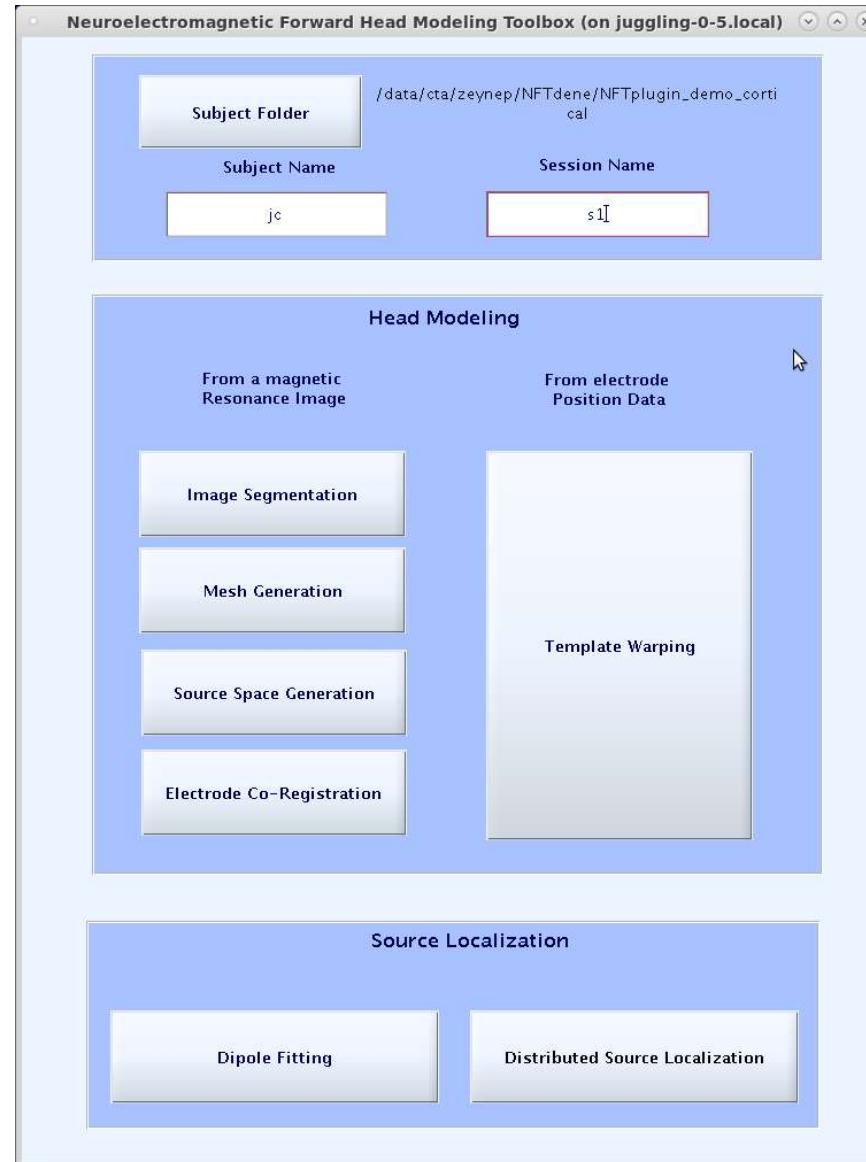


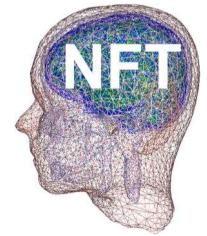
Distributed Source localization

Go to the folder NFTplugin_demo_cortical
addpath

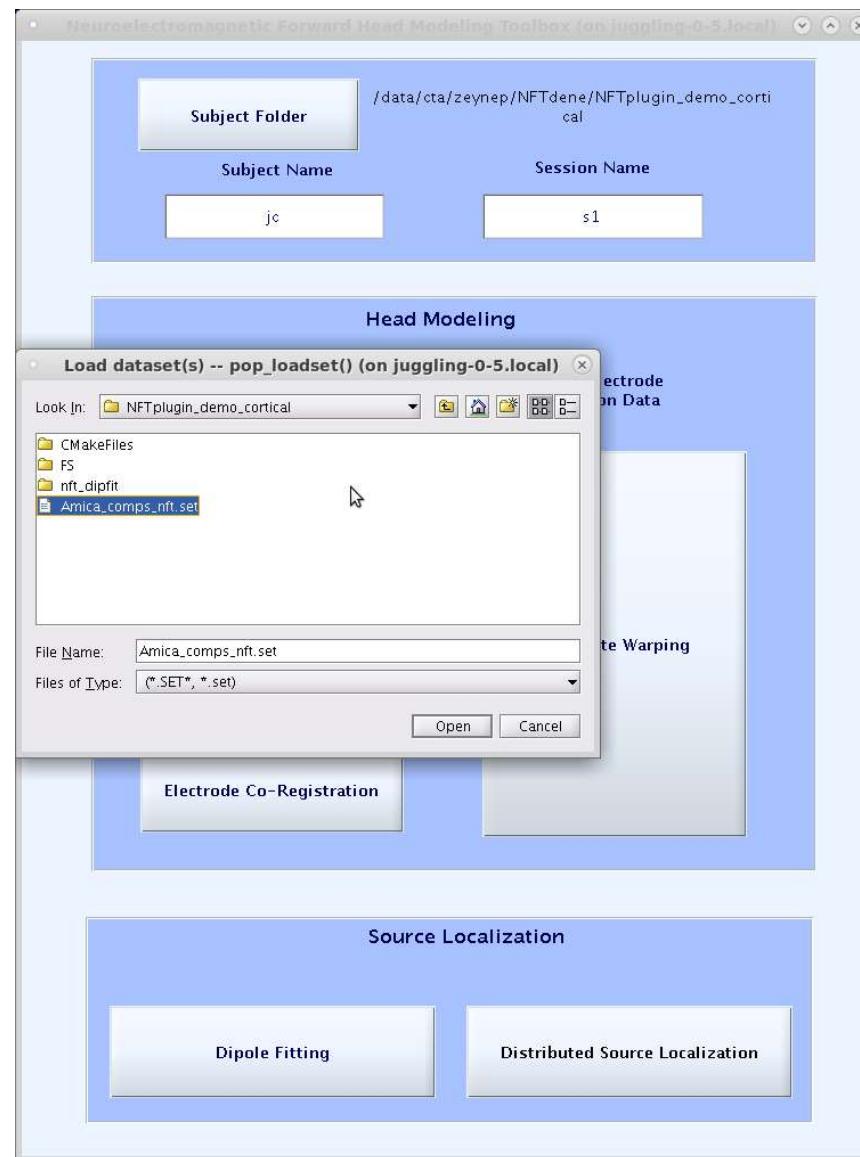


Distributed Source Localization



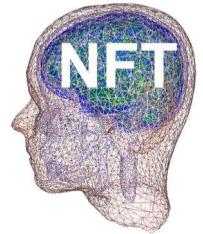


Select EEG data

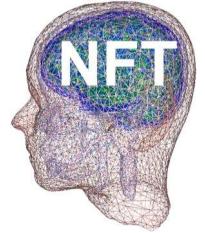




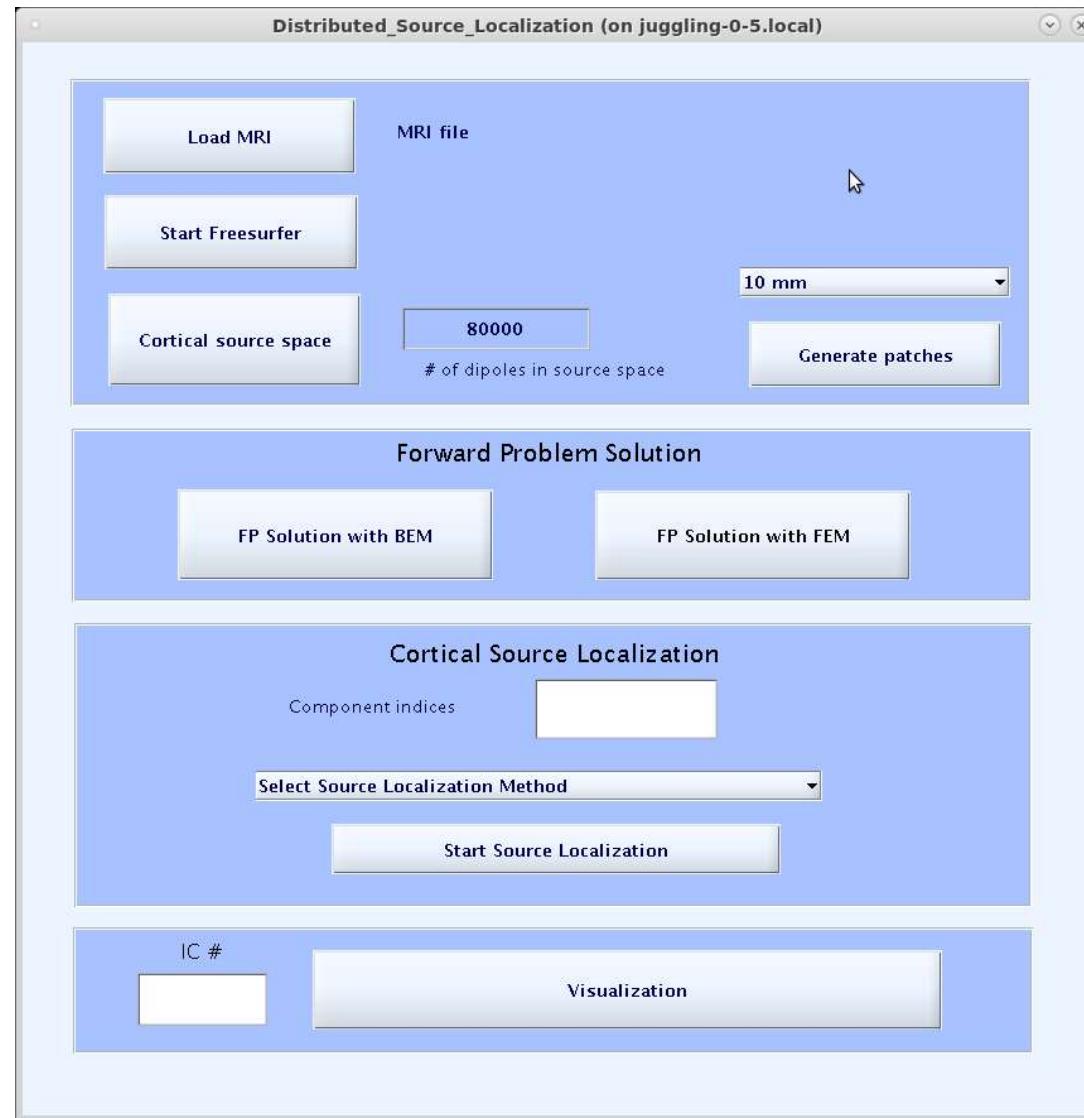
NIST – Generation of a cortical source space

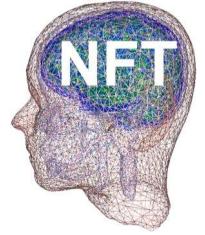


- ◆ Load Freesurface cortical surface
- ◆ Downsample to 80,000 vertices
- ◆ Co-register with the NFT brain surface
- ◆ Re-generate NFT head model
- ◆ Calculate normals for each vertex on the cortical surface
- ◆ Save the cortical source space as: **Subject_name FS_ss.dip**
- ◆ Calculate node area of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

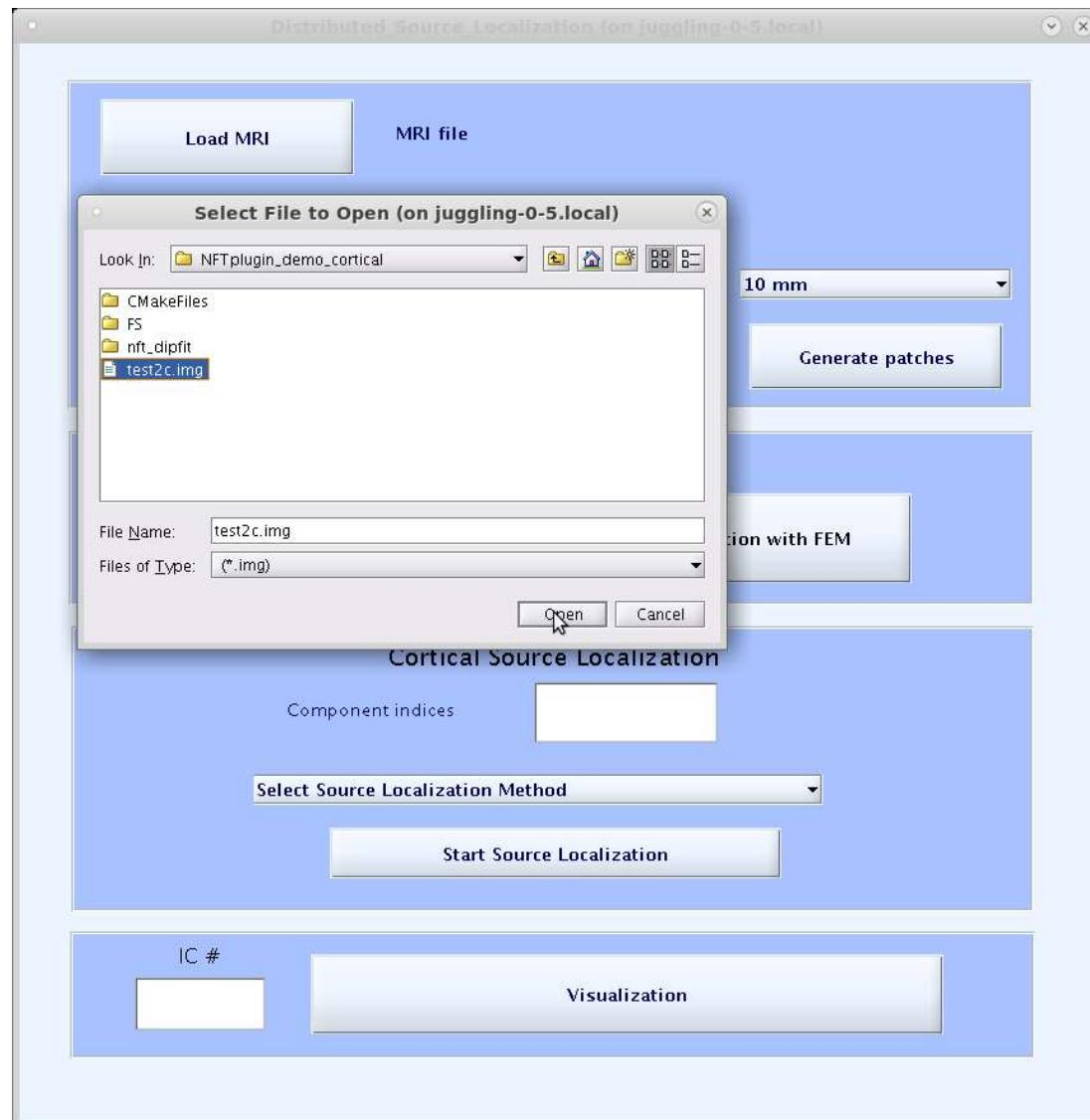


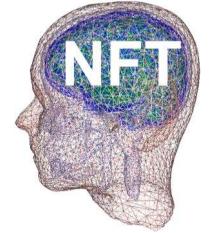
Distributed Source Localization



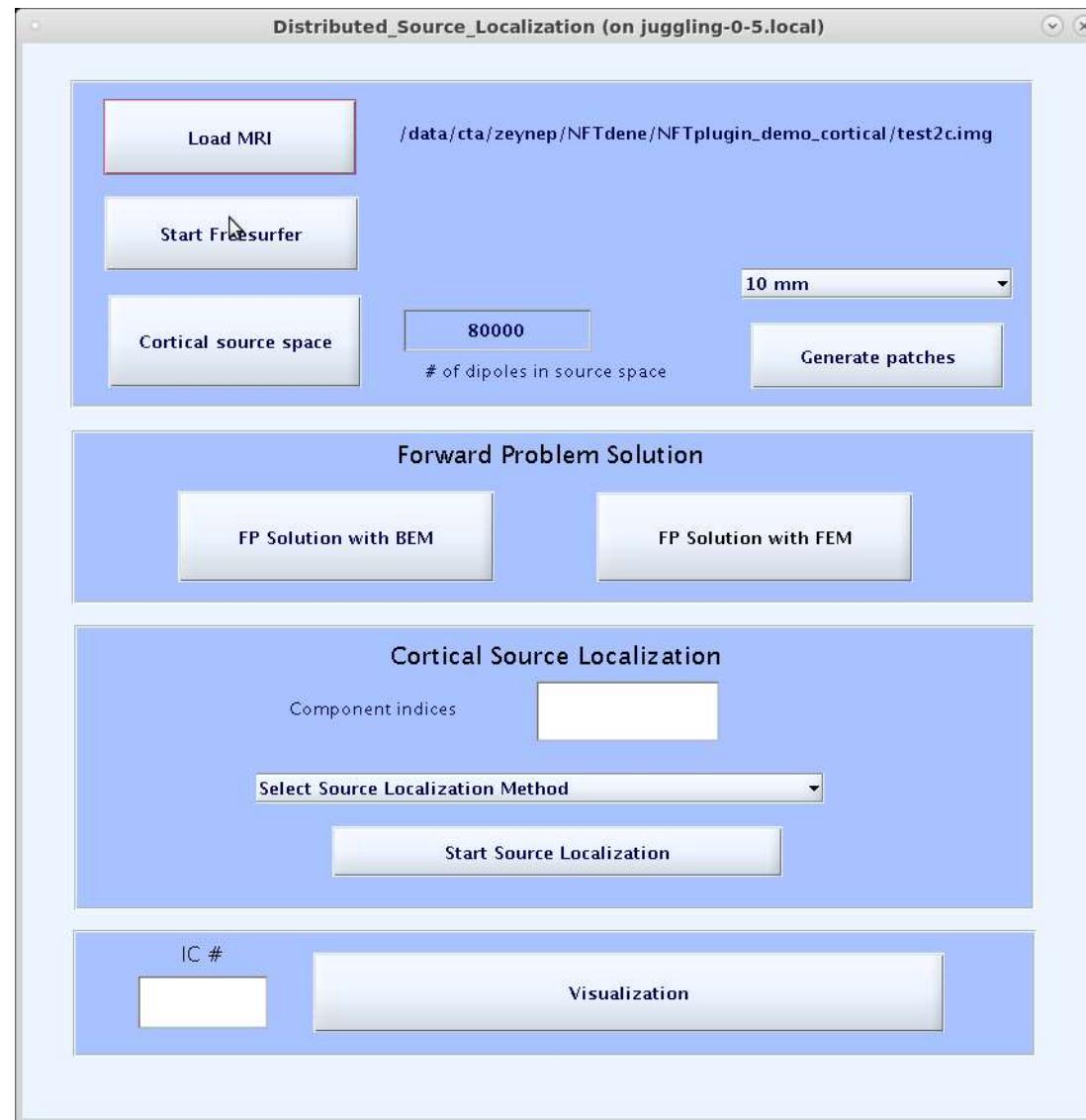


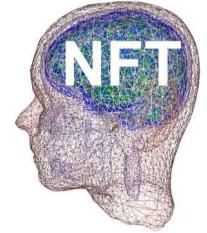
Load MRI





Run Freesurfer





Distributed Source Localization (on juggling-0-5.lor)

<input type="button" value="Load MRI"/>	/data/cta/zeynep/NFTdene/NFTplugin_demo_cortical/test2c.img
<input type="button" value="Start Freesurfer"/>	Running Freesurfer completed!
<input type="button" value="Cortical source space"/>	<input type="text" value="80000"/> # of dipoles in source space
	<input type="button" value="Generate patches"/> <input type="button" value="10 mm"/>

Forward Problem Solution

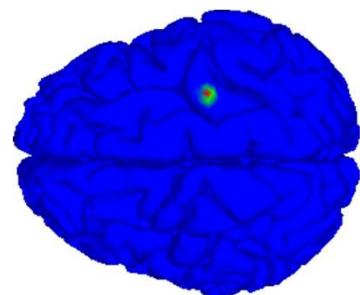
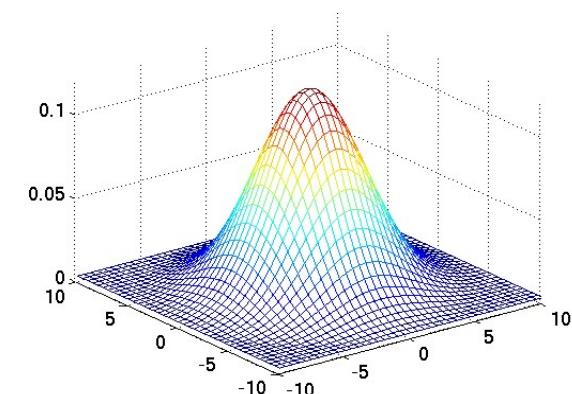
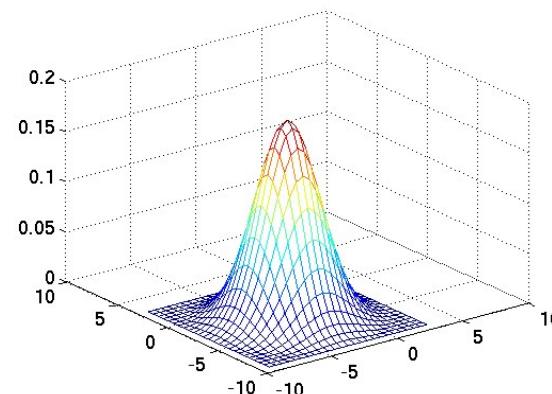
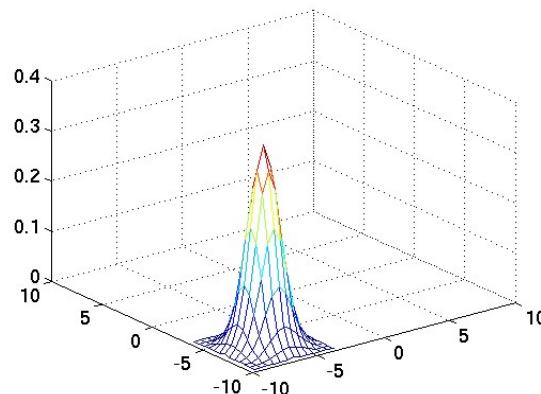
<input type="button" value="FP Solution with BEM"/>	<input type="button" value="FP Solution with FEM"/>
---	---

Cortical Source Localization

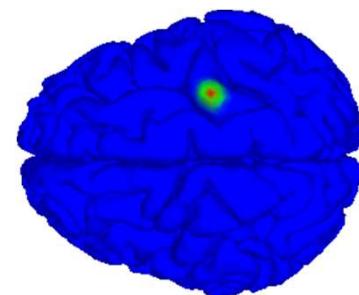
Component indices	<input type="text"/>
Select Source Localization Method	<input type="button"/>
<input type="button" value="Start Source Localization"/>	

IC # Visualization

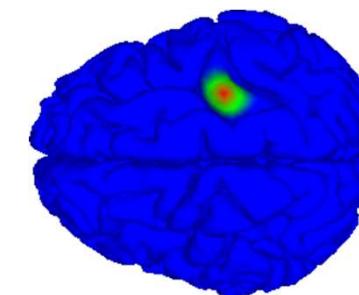
NIST – Patch generation



3 mm patch



6 mm patch



10 mm patch

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!

Cortical source space 80000 # of dipoles in source space 3,6,10 mm Generate patches

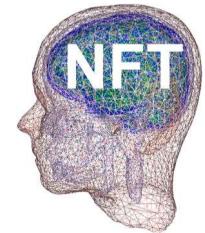
Forward Problem Solution

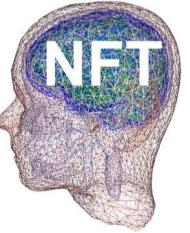
FP Solution with BEM FP Solution with FEM

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

Conductivity values:

Scalp

0.0132

Skull

Brain

Sensors

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

Sensors

Sensors Loaded

Create Session

No Session

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

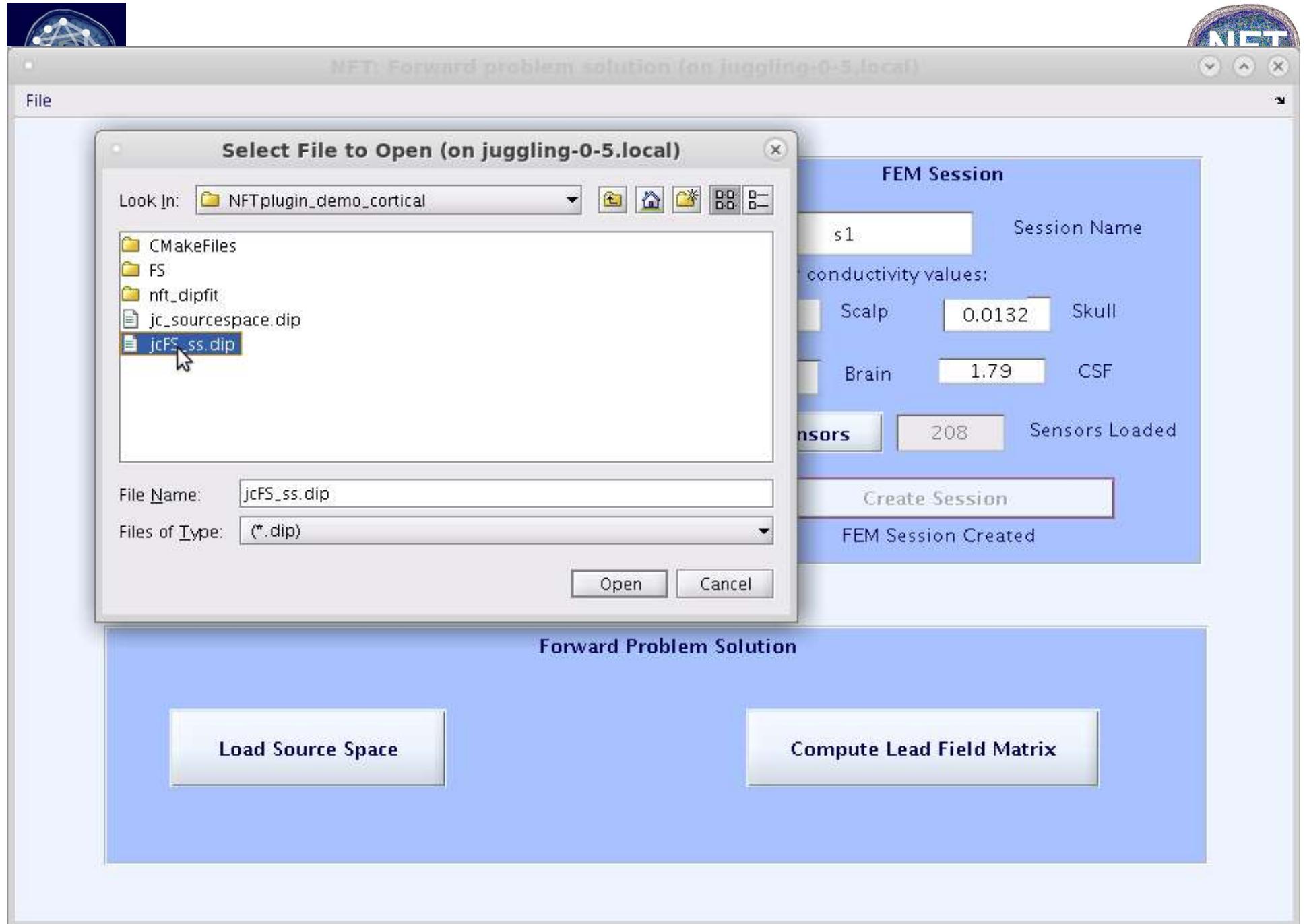
Create Session

Value Changed!

Forward Problem Solution

Load Source Space

Compute Lead Field Matrix





NFT: Forward problem solution (on juggling-0-5.local)

File

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!

Cortical source space 80000 # of dipoles in source space 3,6,10 mm Generate patches

Forward Problem Solution

FP Solution with BEM FP Solution with FEM

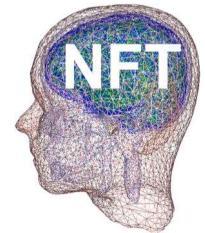
Cortical Source Localization

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!

Cortical source space 80000 # of dipoles in source space 3,6,10 mm Generate patches

Forward Problem Solution

FP Solution with BEM FP Solution with FEM

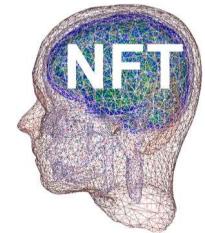
Cortical Source Localization

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!

Cortical source space 80000 # of dipoles in source space 3,6,10 mm Generate patches

Forward Problem Solution

FP Solution with BEM FP Solution with FEM

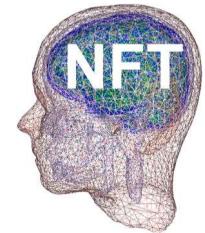
Cortical Source Localization

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!

Cortical source space 80000 # of dipoles in source space Generate patches 3,6,10 mm

Forward Problem Solution

FP Solution with BEM FP Solution with FEM

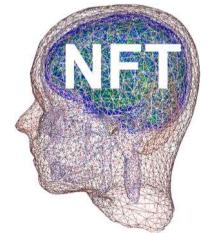
Cortical Source Localization

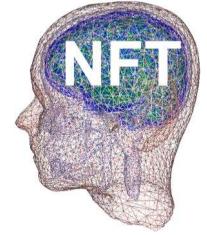
Component indices 1:11

Sparse compact and smooth (SCS) method

Start Source Localization

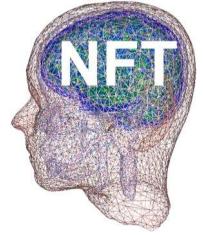
IC # 2 Visualization





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



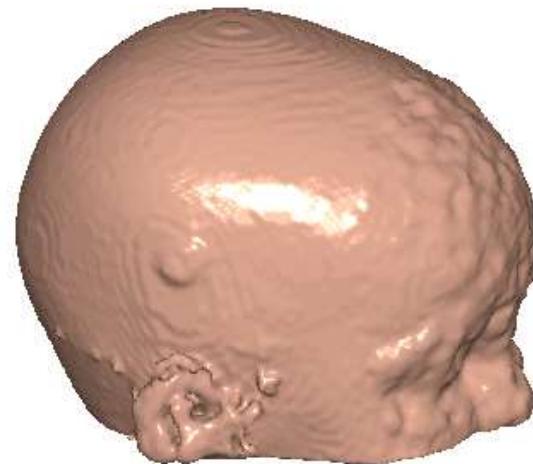
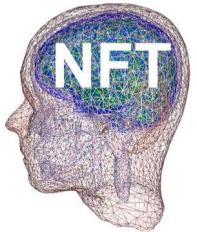
1,441,777
tetrahedral volume
elements



1,025,643
tetrahedral volume
elements



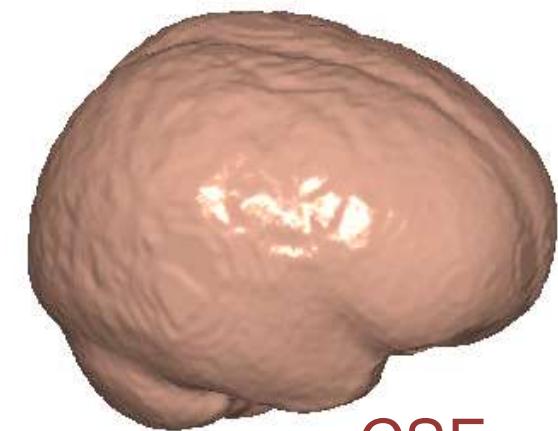
Richards Database for child head modeling



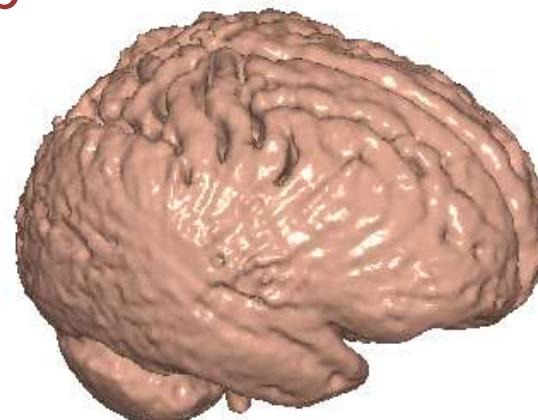
scalp



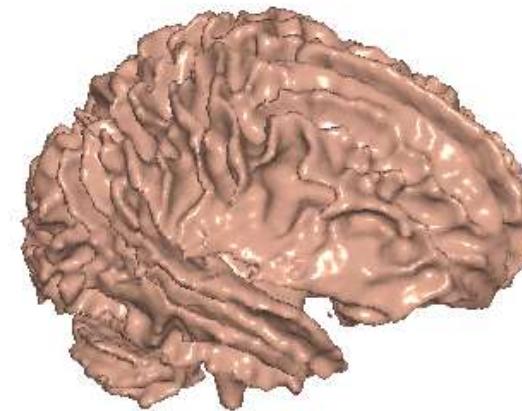
skull



CSF

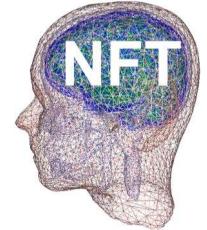


brain

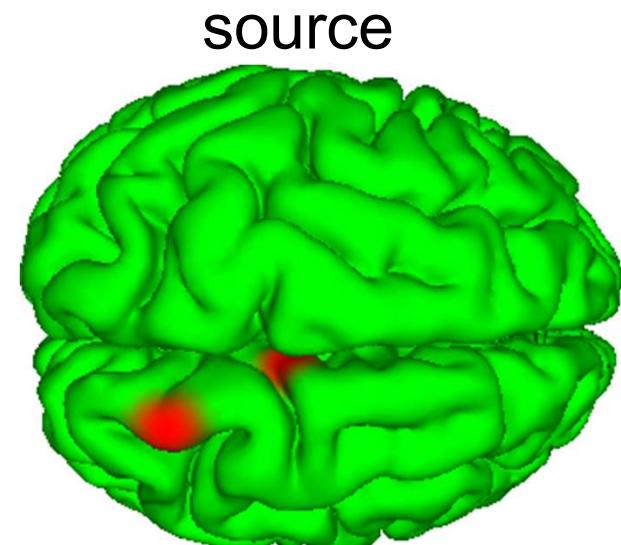


white matter

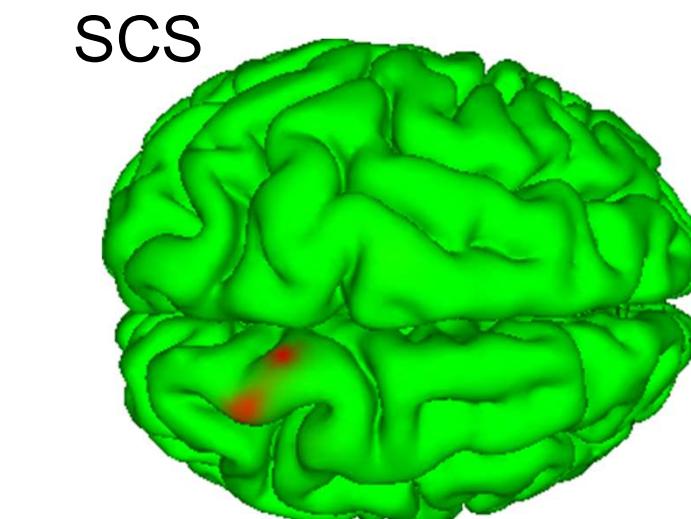
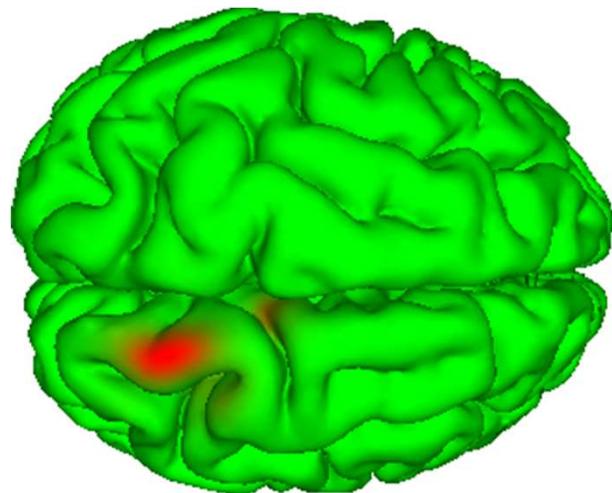
5-layer template BEM head model for three-year olds.



Source localization results

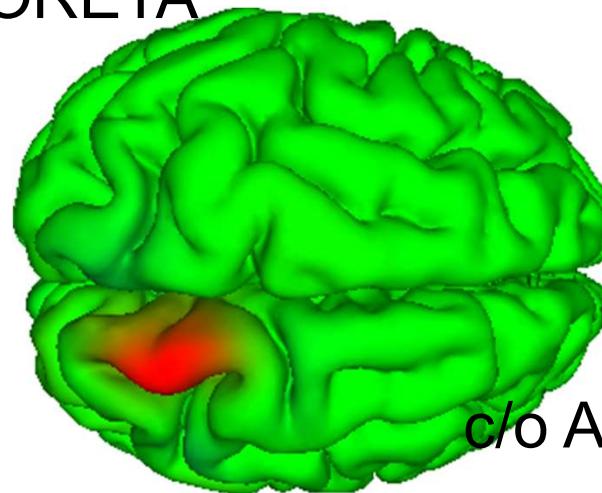


Patch-based SBL

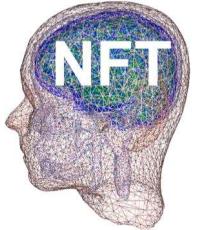


Cheng Cao, 2012

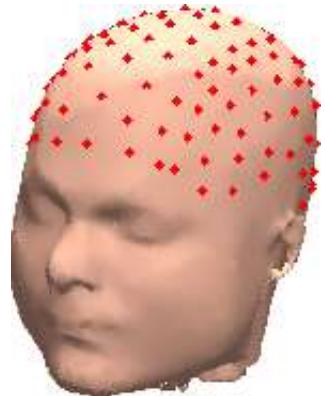
sLORETA



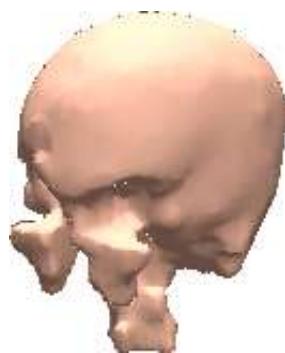
c/o A. Ojeda



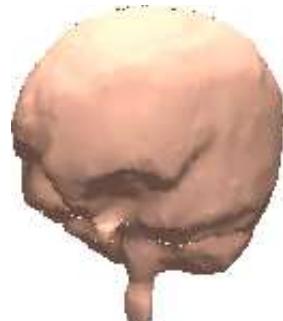
EEG source localization



scalp



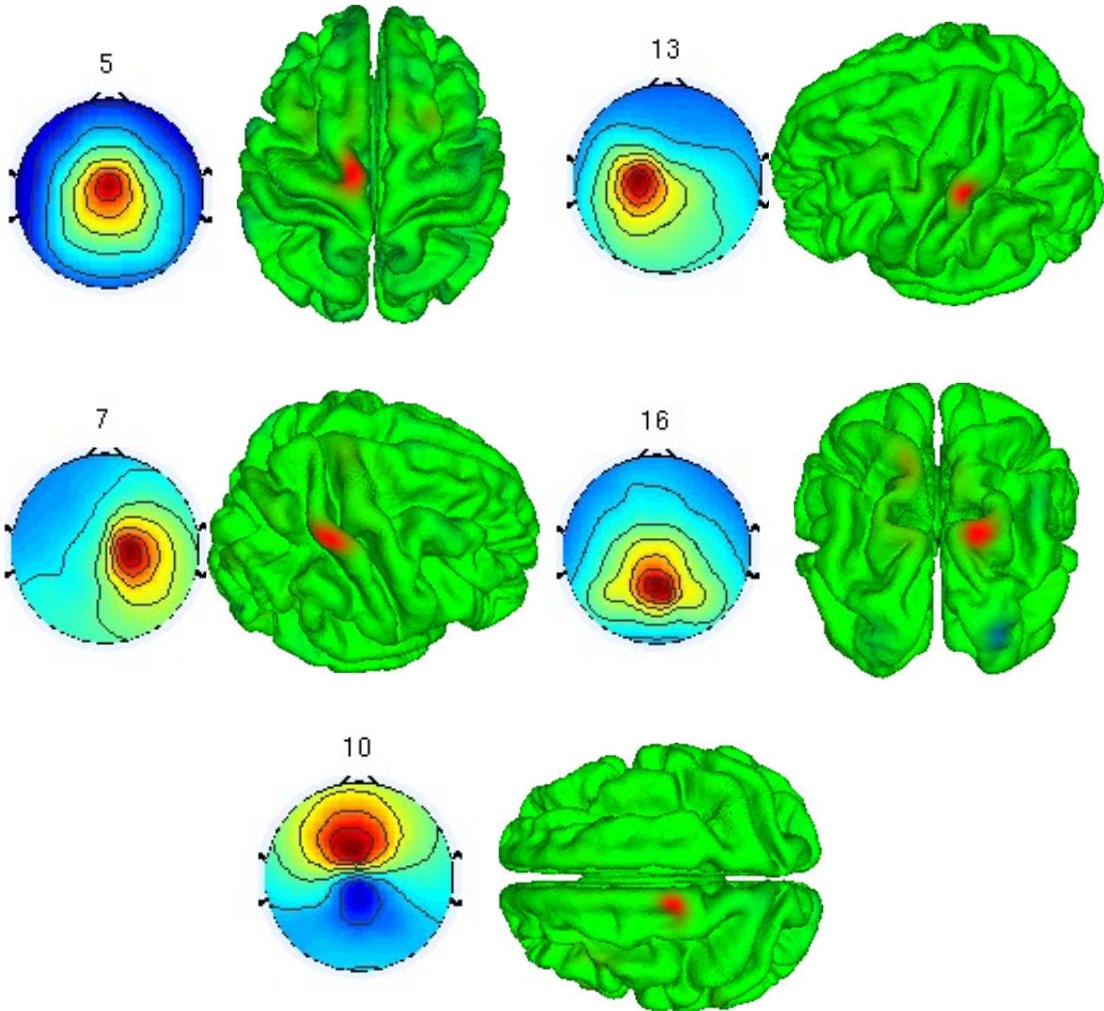
skull



CSF

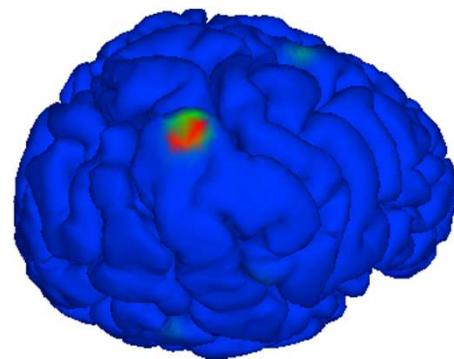
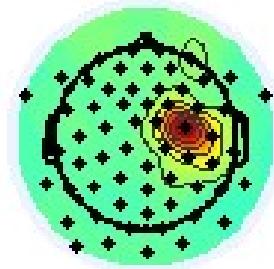


brain

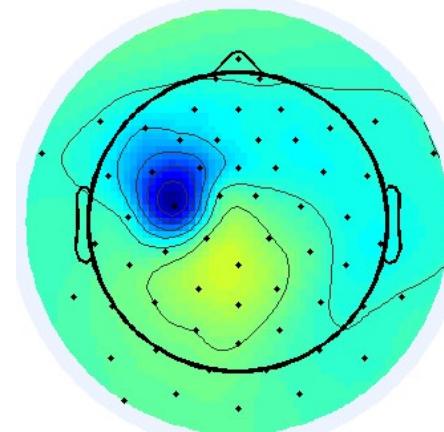


Source localization for 1-year old

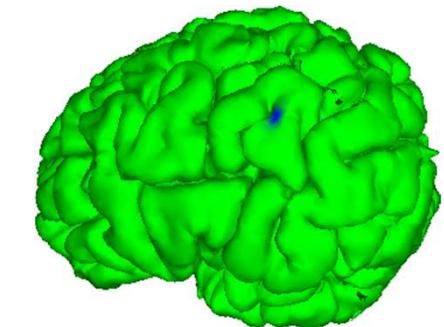
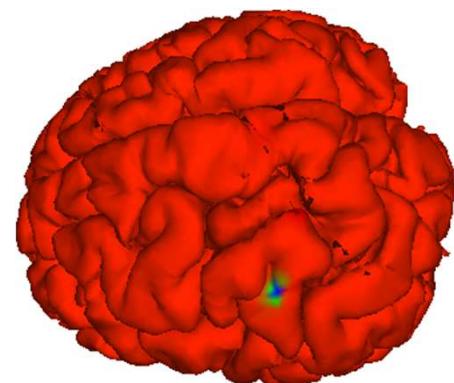
44

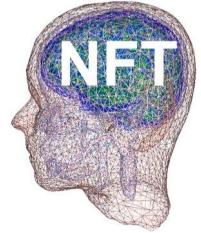


IC 22 from SFIC R136.12 epochs



22

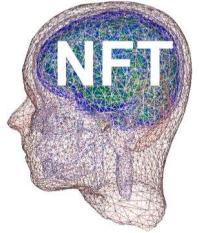




Visualization

- ◆ 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

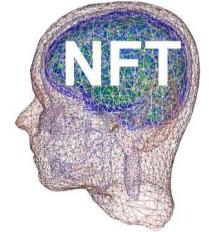


NFT Matlab Scripts

- ◆ Start EEGLAB and set your parameters:

eeglab

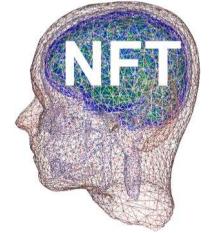
```
EEG = pop_loadset('filename',eeg_file,'filepath',eeg_path);
[ALLEEG, EEG, CURRENTSET] = eeg_store( ALLEEG, EEG, 0 );
% set 'of' (output folder), subject_name,
% session_name, and elec_file
subject_name = 'SubjectA';
session_name = 's1';
nl = 4; % number of layers
plotting = 1;
comp_index = 1:20;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



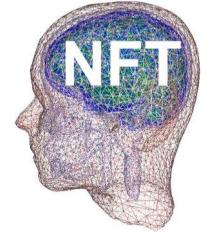
NFT Matlab Scripts

- ◆ 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);
```

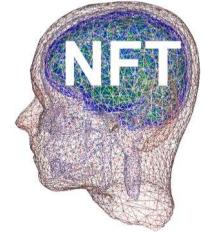
```
dip1 = nft_inverse_problem_solution(subject_name,  
session_name, of, EEG, comp_index, plotting,  
elec_file)
```



NFT Matlab Scripts

- ◆ FEM warping mesh

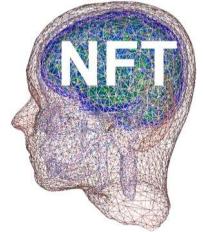
```
session_name='s1_fem';
nft_warping_mesh(subject_name, session_name,
elec_file, nl, of,0,1);
nft_fem_forward_problem_solution(subject_name,
session_name, of);
dip1 = nft_inverse_problem_solution(subject_name,
session_name, of, EEG, comp_index, plotting,
elec_file)
```



NFT Matlab Scripts

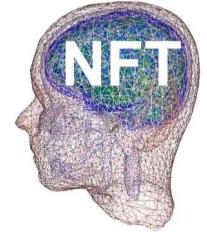
- ◆ Set NFT dipole structure to EEGLAB dipole structure

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



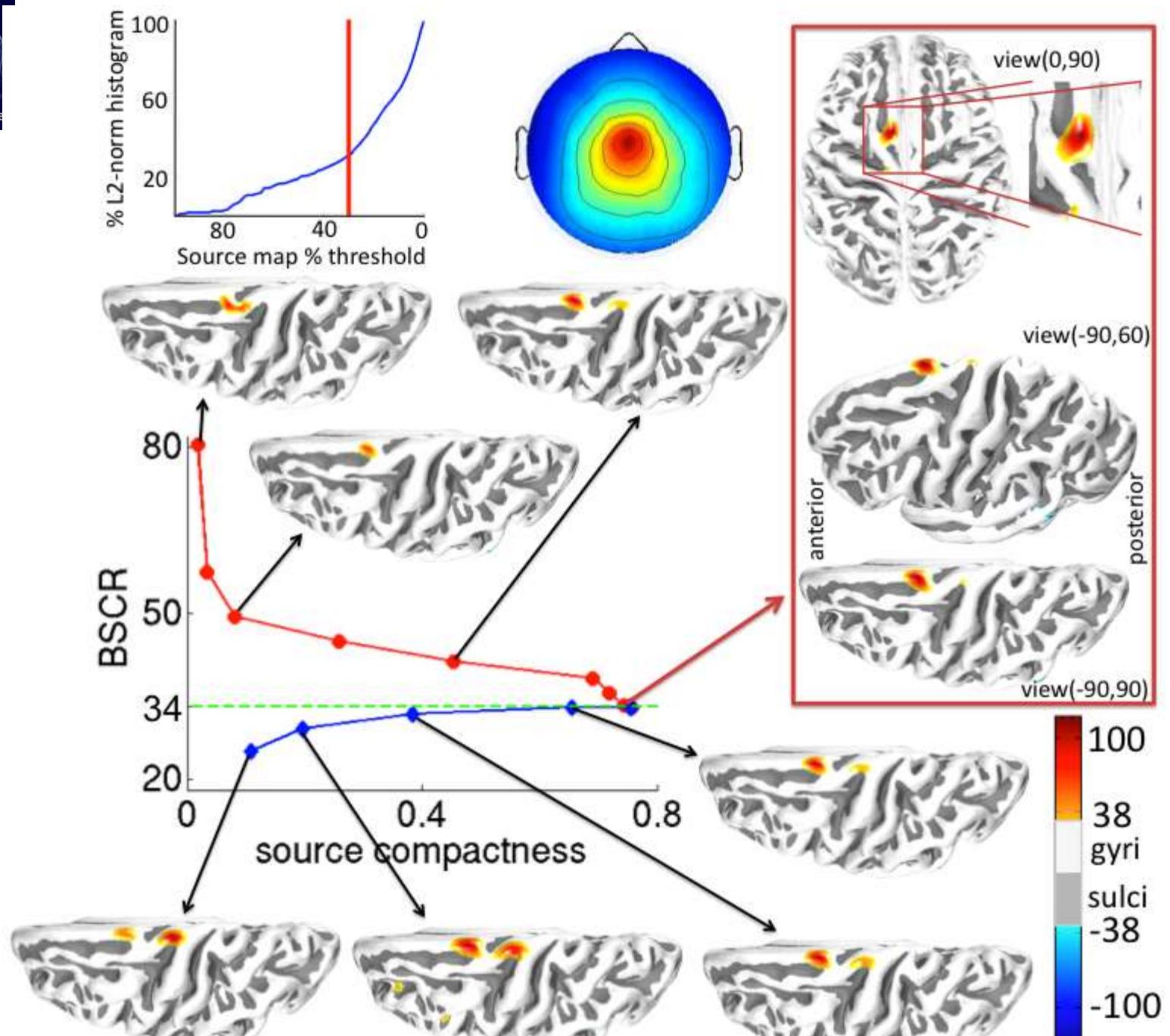
NFT download and reference

- ◆ <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.

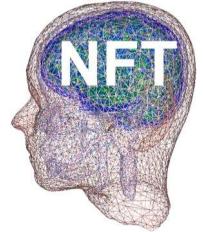


SCALE

- ◆ 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.

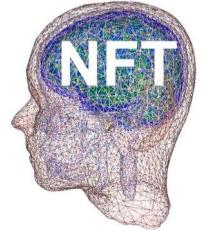
4-layer head model is used: scalp, skull, CSF, and brain

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