NFT & NIST
Neuroelectromagnetic Forward and Inverse Head Modeling Toolbox

Zeynep AKALIN ACAR
17th EEGLAB Workshop, San Diego
November, 2013
◆ 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

From a magnetic Resonance Image
- Image Segmentation
- Mesh Generation
- Source Space Generation
- Electrode Co-Registration

T1-weighted

Segmentation

Electrode Registration

BEM mesh
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
Select current folder as subject folder
Enter “SubjectA” as subject name
Enter “s1” as the session name
Image Segmentation

NFT: MR segmentation

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: (135, 135, Set)
   - Fill level: 0.4
   - Threshold: 0.4

4. Outer Skull Segmentation
   - Center of one eye: (110, Set)

5. Inner Skull Segmentation

Save Results
- Output Folder: /data/projects/zeynep/common/home/zeynef/p/id/dename

Display Image
- MR image
- Fibered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask
Image Segmentation

NFT: MR segmentation

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: 66
   - White matter seed point: 135
   - Fill level: 0.4
   - Threshold: 0.4

4. Outer Skull Segmentation
   - Center of one eye: 110

5. Inner Skull Segmentation

Save Results

Display Image:
- MR image
- Fibered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Load image
Segmentation

Select an image in analyze format
Segmentation

Run filtering
Segmentation

NFT: MR segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: x = 135, y = 135, z = 110
   - White matter seed point: x = 0.4, y = 0.4, z = 0.4

4. Outer Skull Segmentation
   - Center of one eye: x = 110

5. Inner Skull Segmentation

Display Image:
- MR image
- Filtered image
- Scalp mask
- Brain mask
- Outer skull mask
- Inner skull mask

Save Results:
- Output Folder
- Filtered Image
- Segmentation
Segmentation

View filtered image

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: 66
   - White matter seed point: 135

4. Outer Skull Segmentation
   - Center of one eye: 110

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/zeynek/comm\on/home/zeynek/io/deneme
- Filtered Image
- Segmentation

Image is filtered!
Segmentation

Click ‘Next’ for scalp segmentation.
Segmentation

Click ‘Run’ for scalp segmentation.
Image Segmentation
Segmentation

View scalp mask
Segmentation

Click ‘Next’ for brain segmentation.
Segmentation

Selection of cerebellar low point
Segmentation

Selection of a WM point

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations
   - Image diffusion

2. Scalp Segmentation

3. Brain Segmentation
   - 67
   - Cerebellar low point

4. Outer Skull Segmentation
   - 110
   - Center of one eye

5. Inner Skull Segmentation
   - Scalp segmented!
Segmentation

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: 67
   - White matter seed point: 172
   - Fill level: 0.4
   - Threshold: 0.4

4. Outer Skull Segmentation
   - Center of one eye: 110

5. Inner Skull Segmentation

Set

Swartz Center for Computational Neuroscience
Segmentation

Click ‘Run’ for brain segmentation
Segmentation

Change thresholds if there is need.
Segmentation

View brain mask

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: 67
   - White matter seed point: 172, 158
   - Fill level: 0.1
   - Threshold: 0.1

4. Outer Skull Segmentation
   - Center of one eye: 110

5. Inner Skull Segmentation

File

(x, y, z) = (150, 172, 158)
Segmentation

Click ‘Next’ for skull segmentation
Segmentation

Select a slice for eyes
Segmentation

NFT: MR segmentation

Image Segmentation
- Swap L/R
- Check inhomogeneity

1. Anisotropic Filtering
   - Number of iterations
   - Image diffusion

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point
   - White matter seed point

4. Outer Skull Segmentation
   - 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
- Filtered Image
- Segmentation
Segmentation

Click ‘Run’ for skull segmentation.
Segmentation

Click on the eyes
Segmentation

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: 67
   - White matter seed point: (172, 158, 150)
   - Fill level: 0.1
   - Threshold: 0.1

4. Outer Skull Segmentation
   - Center of one eye: 95

5. Inner Skull Segmentation

Save Results

Output Folder: /data/projects/zeynep/communication/home/zevnei/0/deneme

Filtered Image: Segmentation
Segmentation
Segmentation

View skull segmentation

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: 67
   - White matter seed point: 172
   - Fill level: 0.1
   - Threshold: 0.1

4. Outer Skull Segmentation
   - Center of one eye: 95

5. Inner Skull Segmentation

Save Results

Output Folder: /data/projects/zeinep/comm/ronevee/io/deneme

Filtered Image
Segmentation

Skull segmented!
Segmentation

Click ‘Next’ for CSF segmentation
Segmentation

Click ‘Run’ for CSF segmentation
Segmentation

NFT: MR segmentation

Image Segmentation
- Anisotropic Filtering
  - Number of iterations: 5
  - Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: (x, y, z) = (172, 158, 150)
   - White matter seed point: (x, y, z) = (0.1, 0.1)
   - Fill level: [0, 1]
   - Threshold: [0, 1]

4. Outer Skull Segmentation
   - Center of one eye: (x, y, z) = (95, 0.1)

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/jo/dename
- Filtered Image
- Segmentation

Segmentation complete!
View CSF segmentation
Segmentation

Save filtered image
Segmentation

NFT: MR segmentation

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: 67
   - White matter seed point: 172, 158, 150
   - Fill level: 0.1
   - Threshold: 0.1

4. Outer Skull Segmentation
   - Center of one eye: 95

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/zeyp/communication/home/zeyp/01/02/denarae
- Filtered Image
- Segmentation

Filtered image saved as SubjectA_filtered.mat
Segmentation

Save segmentation
Segmentation

NFT: MR segmentation

Image Segmentation

1. Anisotropic Filtering
   - Number of iterations: 5
   - Image diffusion: 3

2. Scalp Segmentation

3. Brain Segmentation
   - Cerebellar low point: x=67, y=158, z=150
   - White matter seed point: x=172, y=158, z=150
   - Fill level: 0.1
   - Threshold: 0.1

4. Outer Skull Segmentation
   - Center of one eye: z=95

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/zeyenp/comm_on/home/zeyenp/io/dename
- Run
- Next

Segmentation saved as SubjectA_segments.mat
Image Segmentation

```
>> dir SubjectA
SubjectA_mri.mat    SubjectA_segments.mat

>> load SubjectA_mri
>> mri

mri =

   dim: [256 256 255]
   xgrid: [1x256 double]
   ygrid: [1x256 double]
   zgrid: [1x256 double]
   anatomy: [256x256x255 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
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

From a magnetic Resonance Image

- Image Segmentation
- Mesh Generation
- Source Space Generation
- Electrode Co-Registration
Electrode Co-registration

From a magnetic Resonance Image

- Image Segmentation
- Mesh Generation
- Source Space Generation
- Electrode Co-Registration

NFT: Electrode co-registration

Load sensor locations

Electrode file name

/data/projects/zeynep/common/home_zeynep/jo/dene_me/dene_real

Mesh Folder

Initial co-registration

Translation
Rotation

Complete co-registration

Translation
Rotation

Save initial reg.

Save complete reg.
Electrode Co-registration
Electrode Co-registration

From a magnetic Resonance Image

- Image Segmentation
- Mesh Generation
- Source Space Generation
- Electrode Co-Registration
Electrode Co-registration
Electrode Co-registration
Electrode Co-registration

From a magnetic Resonance Image

Image Segmentation

Mesh Generation

Source Space Generation

Electrode Co-Registration

Load sensor locations
/data/projects/zeynep/common/home_zeynep/jo/denege/jop3_raw.el

Mesh folder
/data/projects/zeynep/common/home_zeynep/jo/denege/dene_real

Initial co-registration
Translation 0.25 1.5708
Rotation 0.25 -1.5708

Complete co-registration

Save initial reg.

Save complete reg.

Computing translation and rotation parameters...
Electrode Co-registration

From a magnetic Resonance Image

- Image Segmentation
- Mesh Generation
- Source Space Generation
- Electrode Co-Registration
Head Modeling from Electrode Position Data

- Warp a template mesh to electrode positions
  - When no MR images are available
  - Non-rigid thin-plate spline warping
Template Warping

NFT: Template head model warping

Load sensor data

Output Folder: /data/projects/zeyrep/common/home_zeyrep/jo/dene/mni

MNI head model

Warped MNI head model

Start warping
Template Warping

![Diagram of template warping interface]

- Load sensor data
- Output Folder
- Select File to Open:
  - dene_mni
  - dene_real
  - jop3.elp
  - jop3_raw.elp

- Start warping

Note: The diagram shows a software interface for warping templates in computational neuroscience applications.
Template Warping

NFT: Template head model warping

Load sensor data

/data/projects/zeyrep/common/home_zeyrep/jo/dene/mc/jcp3_left_slp

Output Folder

/data/projects/zeyrep/common/home_zeyrep/jo/dene/mc/dene_mni

MNI head model

Warped MNI head model

4 # of layers (3 or 4)

Start Warping
Template Warping
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 Problem Solution with BEM
Forward Problem Solution with BEM
Forward Problem Solution with BEM
Forward Problem Solution with BEM
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.
Dipole Fitting

- Requires EEGLAB integration to access Component indices.

- Uses FieldTrip in EEGLAB for dipole fitting.
Warping Demo

- If the subject does not have an MRI, a template head model (MNI) can be warped to electrode locations:
  - Go to *NFT-2.4_demo_warping* folder
  - Start NFT
  - Enter *SubjectA* as subject name and *s1* as session name
  - Select the current folder as the subject folder
  - Click on *Warping*
  - After warping, continue with Forward and Inverse modeling.
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;
  ```
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**: 1,441,777 tetrahedral volume elements
- **12 months**: 1,025,643 tetrahedral volume elements
Richards Database for child head modeling

5-layer template BEM head model for three-year olds.
NIST – Neuroelectromagnetic Inverse Source Localization Toolbox

- Generates a cortical source space using Freesurfer
- Generates patches of cortex
- Solves distributed source localization using methods:
  - Patch-based Sparse Bayesian Learning (SBL) method
  - Sparse, compact, and smooth (SCS) method
NIST main window
NIST – Load MRI

Load the same pre-processed MRI that is loaded in NFT segmentation
Freesurfer will generate a folder named “FS” in the subject folder. All Freesurfer outputs will be saved in that directory.

ps: Freesurfer has to be installed in the system.
NIST – Generation of a cortical source space

The Freesurface cortical surface will be co-registered with the NFT head model.
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.
NIST – Patch generation

Select patch size:
- 3, 6, 10 mm
- 10 mm
- 6 mm
- 3 mm
NIST has options to generate Gaussian patches with 10 mm, 6 mm, 3 mm in radius.
NIST – Source Localization

NIST has 2 source localization methods:
1. Patch-based Sparse Bayesian Learning method
2. Sparse, compact, and smooth (SCS) method

We plan to add more!
Source localization results

source

Patch-based SBL

SCS

sLORETA

Cheng Cao, 2012

c/o A. Ojeda
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
◆ Start EEGLAB and set your parameters:

```matlab
  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)
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
  
  ```matlab
  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)
  ```
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