Real-Time FMRI Tools & Automation in AFNI & SUMA

SSCC / NIMH & NINDS / NIH / DHHS / USA / EARTH
Why bother?

• Image quality control
  – Spikes, distortion, ghosting, noise, ...
  – Amount of motion
  – Operator error

• Functional localization
  – Localizer prior to main FMRI experiment for BCI or high-res imaging
  – Pre operative scanning
  – As Q/A in clinical settings or difficult / rare subject population
  – 'scan to criteria'

• Teaching

• Feedback and Biofeedback
  – Reduce motion
  – Alter/interfere brain function
  – Control of task/ stimulus computer
  – Classification/BCI
  – Signals in vegetative state

Cox, RW et al. 95, Cohen, MS et al. 98, Frank, J. et al 99, Voyvodic, J. 99
Yang, S. et al 08
Weiskopf, N et al. 2007
Yang, S. et al. 05
deCharms. RC. et al. 04
deCharms. RC. et al. 05
Posse S. et al. 03
LaConte SM. et al. 07
Yoo S. et al. 04
Owen AM et al 06
Image Quality Control

- Image quality control
  - Spikes, distortion, ghosting, noise, ...
  - Amount of motion

Cox, RW et al. 95,
Cohen, MS et al. 98,
Frank, J. et al. 99,
Voyvodic, J. 99
Weiskopf, N et al. 2007
Image Quality Control

- Image quality control
  - Spikes, distortion, ghosting, noise, ...
  - Amount of motion

Cox, RW et al. 95, Cohen, MS et al. 98, Frank, J. et al 99, Voyvodic, J. 99, Weiskopf, N et al. 2007
Image Quality Control

Real-time Estimation of Functional Activation

Real-time Estimation of subject movement
Reduce Motion with Feedback

- Feedback and Biofeedback
  - Reduce motion

Fig. 2 from Yang, S. et al. Neuroimage 05

Fig. 6 from Yang, S. et al. Neuroimage 05
Classification

- Classification maps high dimensional pattern into a set of classes
  - This allows a complex brain activation pattern to be identified with a set of classes or brain states.
  - Useful in to providing intuitive feedback from activation of multiple areas
  - Useful for inferring brain state

From LaConte S. – FMRI Advanced Issues ISMRM 09
Single 2 second event
From fast randomized event related FMRI

Figs. 1 and 3 from Beauchamp, M.S. et al. HBM 09
Brain Computer Interface

Stage I
- Task instruction
  - Practice of tasks and real-time fMRI
  - Generation of reference activation template

Thoughts and corresponding cursor command
- Mental calculation
- Mental speech generation
- Right hand motor imagery
- Left hand motor imagery

Activation pattern and ROI

Stage II
- Choice of cursor command
  - Thought generation
  - Real-time fMRI
  - ROI application

DSC
- Reproducibility matching

Start

Fig.1 Yoo S. et al. Neuroreport 04
The AFNI interface
The players

- **Scanner**
  - A user-supplied machine to acquire and reconstruct images in real time
• **Real Time Setup**
  
  - A user-supplied set of commands that tell AFNI what to do with incoming data
  - Can be done from shell commands or from within C code
  - Communicates with AFNI through TCP/IP socket
  - Sets up ROIs for AFNI*

*Note: RT Plugin is used to communicate with AFNI through TCP/IP socket.*
Setting up AFNI's RT plugin

• Manually
  – Good for learning and demo
Setting up AFNI's RT plugin

- Via Environment Variables
  
  ```
  setenv AFNI_REALTIME_Registration 3D:_realtime
  setenv AFNI_REALTIME_Graph Realtime
  ```
Setting up AFNI

- Manually
- Environment variables
  - See README.environment (~250 variables)
- Layout files
  - Size and position windows just so
- Via plugout_drive
  - Details will follow
- Via *image_monitor* module -drive options
  - -drive_wait 'OPEN_WINDOW axialgraph keypress=A'
  - -drive_afni 'CLOSE_WINDOW axialimage'
Demo time

  cd AFNI_data6/realtime.demos

• Motion monitoring
  tcsh demo.1.run1
  – illustrates real-time data acquisition and motion correction by AFNI

• Motion & function
  – See [demo.2.fback.0.README](http://afni.nimh.nih.gov/pub/dist/edu/data/CD.tgz) for instructions
  – Illustrates acquisition, motion correction and feedback
    • Remove option “-show_demo_gui yes” from demo.2.fback.1.receiver if it proves troublesome
ROI selection options

• Standard atlases
  • TT_Daemon :
    – Created by tracing Talairach and Tournoux brain illustrations.
    – Contributed by Jack Lancaster and Peter Fox of RIC UTHSCSA
  • CA_N27_MPM, CA_N27_ML, CA_N27_PM :
    – Anatomy Toolbox's atlases, some created from cytoarchitectonic studies of 10 human post-mortem brains
    – Contributed by Simon Eickhoff, Katrin Amunts and Karl Zilles of IME, Julich,

• FreeSurfer, subject-based
• Functional localizer
• Etc.
Standard-space atlas ROI selection

@fast_roi -region CA_N27_ML::Hip \
    -region CA_N27_ML::Amygda \
    -base TT_N27_r2+tlrc. \
    -anat doe_SurfVol_Alnd_Exp+orig. \
    -roi_grid blur_vr_run1_motor_AFB003+orig. \
    -prefix hip_amy -time

- less than 1min including skull stripping and xform to TLRC
- A couple of seconds for generating more ROIs
Atlas-based ROIs

1- Strip skull
2- Find xform to atlas space (about 40 secs, 2.5Ghz CPU)
3- Identify ROIs
4- Xform ROIs to native space (about 2 seconds)
Subject-based Anatomical ROIs

From FreeSurfer's Parcellations

surfer.nmr.mgh.harvard.edu
The players

Scanner

Real Time Setup

AFNI

RT Plugin

Plugin

Image Monitor

Real Time Receiver

Stimulus Display

• Image Monitor
  – An AFNI- or user-supplied program to wait for new images
    • AFNI-supplied programs monitor files only:
      – Imon (Monitors GE's old dreaded I files)
      – Dimon (Monitors GE's DICOM images)
      – RTfeedme (Breaks up timeseries dataset and sends it to AFNI)
    • User-supplied programs usually interface with scanner software
      – SIEMENS TRIO/ALLEGRA via functors (S. LaConte BCM, E. Stein NIDA)
    • Often only program that runs on scanner computer
      – Image Monitor sends new images or volumes to AFNI over TCP/IP socket
The players

- Scanner
- Image Monitor
- Real Time Setup
- AFNI
- RT Plugin
- Real Time Receiver
- Plugin
- Stimulus Display

- AFNI/RT plugin take incoming images/volumes and processes them per the setup instructions
  - Assemble images/volumes into time series
  - Perform image registration
  - Perform (multi*) linear regression
  - Send results to Real Time Receiver through TCP/IP socket
    - Raw, volume registered, or residual volume*
    - ROI based results
  - Send raw or processed volumes to plugins registered to receive them
    - Much faster than TCP/IP (just a data pointer is passed)
    - Plugins can also communicate with Real Time Receiver
• AFNI/RT plugin take incoming images/volumes and processes them per the setup instructions
  • Assemble images/volumes into time series
  • Perform image registration
  • Perform (multi*) linear regression
  • Send results to Real Time Receiver through TCP/IP socket
    – Raw, volume registered, or residual volume*
    – ROI based results
  • Send raw or processed volumes to plugins registered to receive them
    – Much faster than TCP/IP (just a data pointer is passed)
    – Plugins can also communicate with Real Time Receiver
The players

• Real Time Receiver (e.g. serial_helper.c or realtime_receiver.py)
  – AFNI- or User- supplied application that expects incoming data from AFNI and acts on it
    • Motion parameters
    • ROI-based data, all values or just average
    • Entire volumes of raw, or preprocessed data
    • Data from any RT plugin such as 3dsvm
  – Process incoming data to your liking
  – Optionally forward results to Stimulus Display either by serial connection, or TCP/IP*
Image Monitor (Dimon)

Dimon:
- monitor acquisition of Dicom or GE-Ifiles
- optionally write to3d script for creation of AFNI datasets
- optionally send volumes to afni's realtime plugin

find first volume (wait forever, scanning may not have started)
wait for volume:
  check every 2 seconds or every -sleep_init ms
  check slices to see if a volume is acquired
once found:
  note grid, orientation, byte order, etc.
if realtime:
  comm: open link
    try to open TCP channel to afni RT plugin
    check whether channel is ready for data
  comm: send control info
    send acquisition style (2D+zt), zorder, time pattern,
    TR, FOV, grid, datum, orientation, origin, etc.
  comm: send volume
set signal handlers, and note between-volume sleep time
for each found volume
  while no new volume is yet found
    check whether the scanner has stalled (run cancelled?)
    sleep for one TR, or -sleep_vol ms, or -sleep_frac fraction of TR
  if this is a new run
    comm: send "end of (previous) run" message
track volume statistics
check orientation
  comm: if connection not yet established, send control info
  comm: send volume
upon termination (ctrl-c or -quit and no more data)
  show run statistics
  possibly create to3d script
  comm: terminate connection
plug_realtime:
  init: register work process with afni (to be called regularly)
  plugin main: sets plugin control variables

main work process: asynchronously from main afni loop
  if new connection, initialize
  if data is bad or no new data after timeout
    write vol. to disk, plot final motion params, comm:close
  if new data: warn user and process
    process control info: TR, grid, orientation, DRIVE comds., etc.
    prepare to receive data from multiple channels
    setup new dataset
  if done with data: finish_dataset and cleanup
while there is data to read
  store into images
  if we have a full volume
    add volume to dataset
    possibly register volume to base
    update registration graph
    possibly run regression
  comm: compute and send TR data to realtime receiver
Realtime_receiver.py

set signal handlers to close all ports on exit
open incoming socket and wait for connection...
forever:

  process one run
    wait for the real-time plugin to talk to us
    check magic HELLO for type/amount of data to receive:
      only motion
      motion plus N ROI averages
      motion plus N voxel values (with coordinates, etc.)

  open outgoing serial port
  while no run termination, process one TR
    read incoming TCP data
    compute outgoing results
    write to serial port

close data ports
• SVM plugin is being modified to accept RT data
  – Given training models, classification is done in real-time
Real Time SVM*

Scanner → Real Time Setup → Image Monitor → AFNI → Real Time Receiver → RT Plugin → Plugin → Stimulus Display

*Movie generated with Real Time setup in S. LaConte et al. HBM 2007
Automation
Automating Navigation

Other applications can communicate with AFNI via a program which sends a series of commands for execution.

- Program called via “system” function (shell invocation)
- No need to manage sockets or format and transmit commands
- User Interaction with GUI is uninterrupted
Cycling trough 300 volumes

while ($cnt < 300)
plugout_drive -com "SWITCH_UNDERLAY A anat_${cnt}+orig"
-com "SWITCH_OVERLAY A anat.ns_${cnt}+orig"
-com 'OPEN_WINDOW A coronalimage opacity=0.5'
-com 'OPEN_WINDOW A axialimage keypress=v opacity=0.4'
-quit
echo "Enter new number or hit enter for next brain:"
set ans = $<
if ("$ans" == "") then
  @ cnt++
else
  set cnt = `expr $cnt + $ans`
endifend
Cycling trough 300 volumes

Loop over all volumes

while ($cnt < 300)
plugout_drive -com "SWITCH_UNDERLAY A anat_${cnt}+orig"
-com "SWITCH_OVERLAY A anat.ns_${cnt}+orig"
-com 'OPEN_WINDOW A coronalimage opacity=0.5'
-com 'OPEN_WINDOW A axialimage keypress=v opacity=0.4'
-quit

echo "Enter new number or hit enter for next brain:"
set ans = $<
if ("$ans" == "") then
    @ cnt ++
else
    set cnt = `expr $cnt + $ans``
endif
end
Cycling trough 300 volumes

Switch background volume

while ($cnt < 300)
plugout_drive -com "SWITCH_UNDERLAY A anat_${cnt}+orig"
-plcom "SWITCH_OVERLAY A anat.ns_${cnt}+orig"
-com 'OPEN_WINDOW A coronalimage opacity=0.5'
-com 'OPEN_WINDOW A axialimage keypress=v opacity=0.4'
-quit

echo "Enter new number or hit enter for next brain:"
set ans = $<
if ("$ans" == "") then
    @ cnt ++
else
    set cnt = `expr $cnt + $ans`
endif
end
Cycling trough 300 volumes

while ($cnt < 300)
plugout_drive -com "SWITCH_UNDERLAY A anat_${cnt}+orig"
-com "SWITCH_OVERLAY A anat.ns_${cnt}+orig"
-com 'OPEN_WINDOW A coronalimage opacity=0.5'
-com 'OPEN_WINDOW A axialimage keypress=v opacity=0.4'
-quit

echo "Enter new number or hit enter for next brain:"
set ans = $<
if ("$ans" == "") then
  @ cnt ++
else
  set cnt = `expr $cnt + $ans`
endif
end
Cycling through 300 volumes

while ($cnt < 300)
  plugout_drive
    -com "SWITCH_UNDERLAY A anat_${cnt}+orig"
    -com "SWITCH_OVERLAY A anat.ns_${cnt}+orig"
    -com 'OPEN_WINDOW A coronalimage opacity=0.5'
    -com 'OPEN_WINDOW A axialimage keypress=v opacity=0.4'
  -quit
  echo "Enter new number or hit enter for next brain:
  set ans = $<
  if ("$ans" == ") then
    @ cnt ++
  else
    set cnt = `expr $cnt + $ans`
  endif
end
Cycling trough 300 volumes

while ($cnt < 300)
plugout_drive 
  -com "SWITCH_UNDERLAY A anat_${cnt}+orig"
  -com "SWITCH_OVERLAY A anat.ns_${cnt}+orig"
  -com 'OPEN_WINDOW A coronalimage opacity=0.5'
  -com 'OPEN_WINDOW A axialimage keypress=v opacity=0.4'
  -quit

  echo "Enter new number or hit enter for next brain:"
  set ans = $<
  if ("$ans" == "") then
    @ cnt ++
  else
    set cnt = `expr $cnt + $ans`
  endif
end
SUMA Movie Sample

Use this link if viewing pdf. Video courtesy of Chunmao Wang.

Visual Object Recognition
Lexical Selection
Phonological Encoding
Phonetic Encoding

0ms
"Help" sources

- **Readme files**
  - README.driver
  - README.environment
  - README.realtime

- **Demo material available on:**

- **Automation Demos scripts**
  - @DriveAfni_script
  - @DriveSuma_script
  - @DO.examples
  - @Install_TSrestMovieDemo
  - @Install_InstaCorr_Demo
  - Scripts in class data under:
    - AFNI_data6/realtime.demos/

- **Sample programs**
  - rtfeedme.c
  - Dimon.c
  - serial_helper.c
  - realtime_receiver.py

- **Talk to us, we're interested in applications**