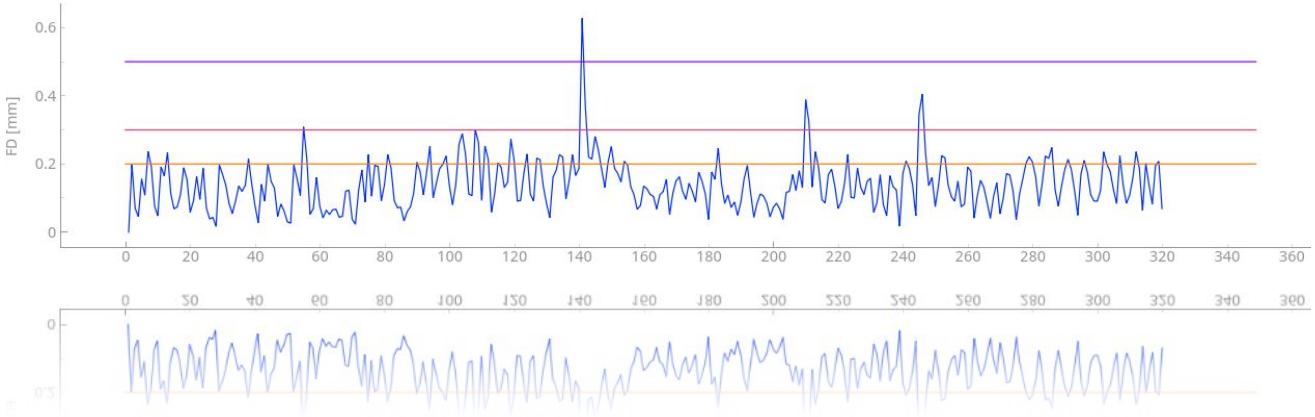
Real-time fMRI motion correction monitoring module



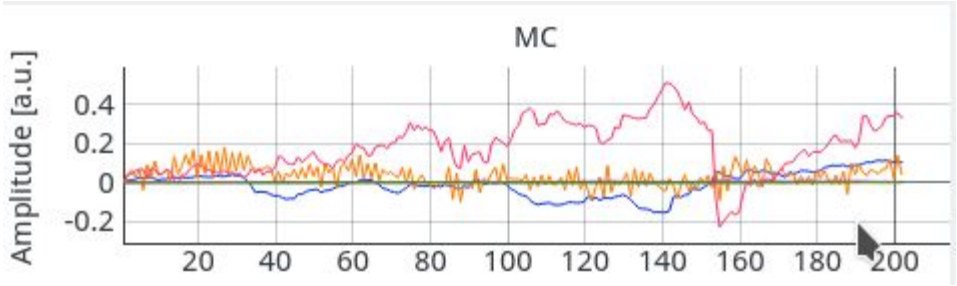
Ari Sarfatis, 12/12/2017

Problem & Objective

1. Head motion causes artifacts, which bias (f)MRI data.
2. Real-time head motion monitoring : decide immediately if a measurement/experiment should be redone or aborted & reduce overscanning
3. Task :
    a. code an online motion correction (MC) monitoring tool compatible with OpenNFT
    b. make an independent online MC monitoring module

Current MC analysis in OpenNFT

Plotting directly SPM12 realignment output (6 d.o.f.)



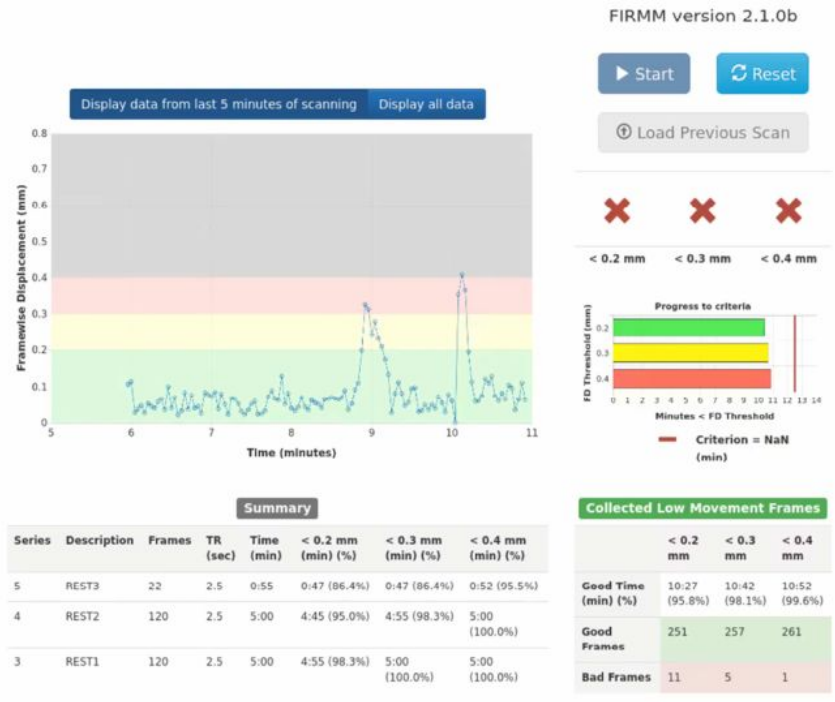
- Different scales for rotations and translations
- MC amplitude indicates absolute movement rather than frame to frame delta
- Difficult to interpret

Framewise displacement

Neuroimage. 2017 Aug 10;161:80-93. doi: 10.1016/j.neuroimage.2017.08.025. [Epub ahead of print]

Real-time motion analytics during brain MRI improve data quality and reduce costs.

Dosenbach NUF[1], Koller JM[2], Earl EA[3], Miranda-Dominguez O[3], Klein RL[4], Van AN[5], Snyder AZ[6], Nagel BJ[7], Nigg JT[7], Nguyen AL[8], Wesevich V[9], Greene DJ[10], Fair DA[11].



Series	Description	Frames	TR (sec)	Time (min)	< 0.2 mm (min) (%)	< 0.3 mm (min) (%)	< 0.4 mm (min) (%)
5	REST3	22	2.5	0:55	0:47 (86.4%)	0:47 (86.4%)	0:52 (95.5%)
4	REST2	120	2.5	5:00	4:45 (95.0%)	4:55 (98.3%)	5:00 (100.0%)
3	REST1	120	2.5	5:00	4:55 (98.3%)	5:00 (100.0%)	5:00 (100.0%)

	< 0.2 mm	< 0.3 mm	< 0.4 mm
Good Time (min) (%)	10:27 (95.8%)	10:42 (98.1%)	10:52 (99.6%)
Good Frames	251	257	261
Bad Frames	11	5	1

- Combine motion correction data into one scalar indicator : **Framewise Displacement (FD)**
- FD represents “the sum of the absolute head movements in all six rigid body directions from frame to frame”.
- “Motion related artifacts are strongly correlated with measures of FD”

DEMO

1. Reset DEMO
2. Install OpenNFT only
3. Launch OpenNFT on data, show clickable window
    a. show FD & correlation with high mvt
4. Show parameters in .ini
5. Quit OpenNFT
6. Launch fdm, install stand alone
7. select watch folder, start
8. break reading, show parameter window, resume
9. break & launch SPM ! & then regression batch, resume
10. wait end & add files manually to folder
11. launch custom batch -> show batch file
12. quit fdm
13. reopen watch folder

Demonstration of FD addon in OpenNFT

1. Self-installs
2. Integrates and interfaces with OpenNFT
3. Computes FD online
4. Shows legacy MC data
5. Shows absolute displacement
6. Reads settings (threshold(s), radius,..) from openNFT .ini experiment setup files
7. Popup window to switch between views and see all MC data at a glance

Demonstration of stand alone MC monitoring

1. Written in Python (PyQt + PyQtGraph)
2. Connected to Matlab shared engine running SPM12
3. Watches folder (watchdog) for new .dcm files
4. Self-installs
5. Computes FD online and displays absolute displacement
6. play/pause
7. Set parameters from UI
8. Regresses activity with FD
9. Allows running custom SPM batches on monitored files
10. Saves MC data at the end of the experiment
11. Loads data from previous experiments

Next steps

- Make it as user-friendly as possible
  - Clean and document the code
  - Program more defensively
  - Possible to resume past experiment
- Make it useful for as many users as possible
  - Add support for more formats (MINC, ECAT, PAR/REC)
  - Expand syntax for custom SPM batches
  - Improve regression batch (very basic at the moment)
- Explore other head movement metrics ?
  - FD -> need to take image acquisition time into account when setting the thresholds
- Other ideas?

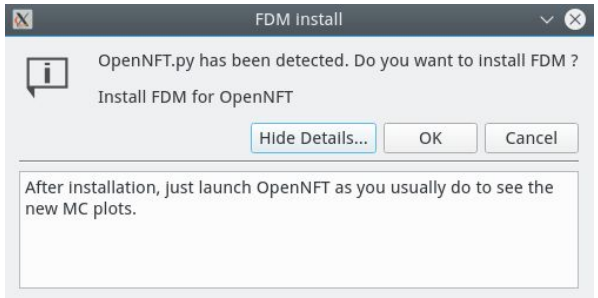
The Future of FIRMM

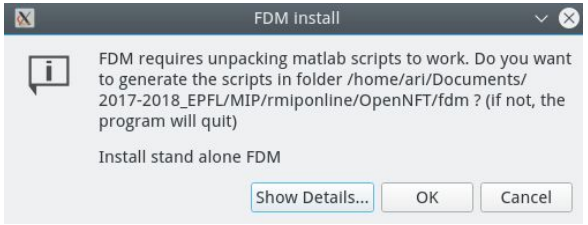
We would love to get user feedback on which features we should implement next.

Possible future features

- [ ] Screenshots
- [ ] Incomplete FOV detection
- [ ] DVARS
- [ ] vNav (structural navigator images)
- [ ] Slice viewer (data alignment)
- [ ] Web GUI section to adjust settings
- [ ] Customizable FD plot display (control axis limits)
- [ ] More flexible FD plot display (toggle between individual series)
- [ ] Increased access control and user roles
- [ ] Faster image processing
- [ ] Choice of plot: FD or all 6 directions
- [ ] Sonstiges: _____

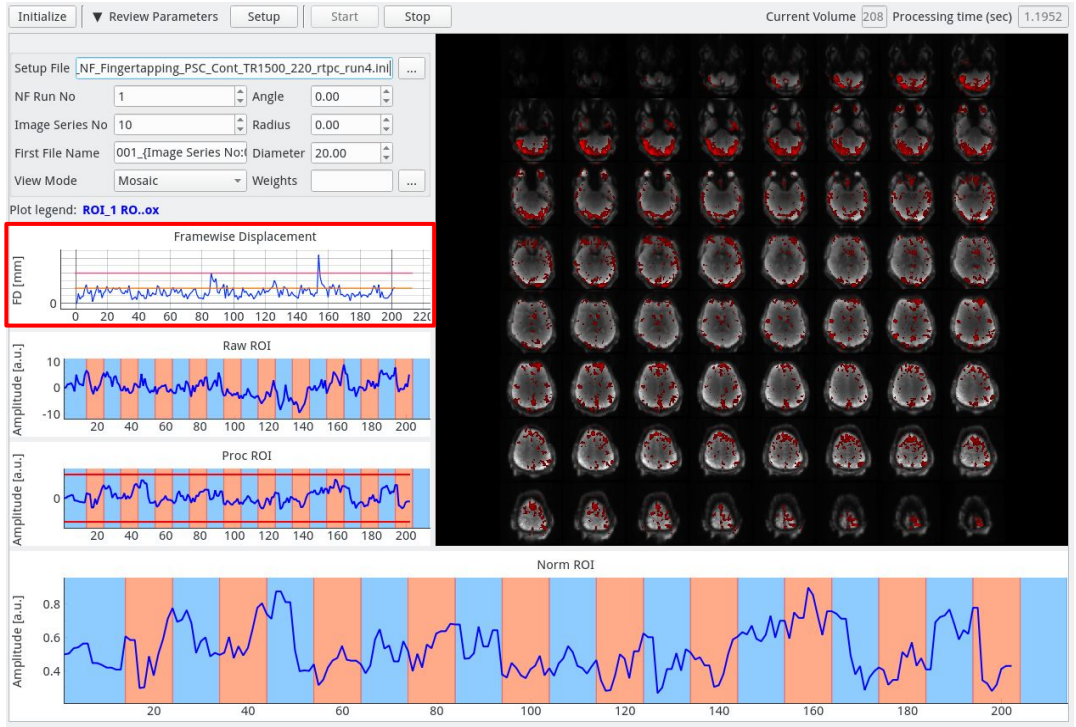
The module detects the presence of OpenNFT and installs itself



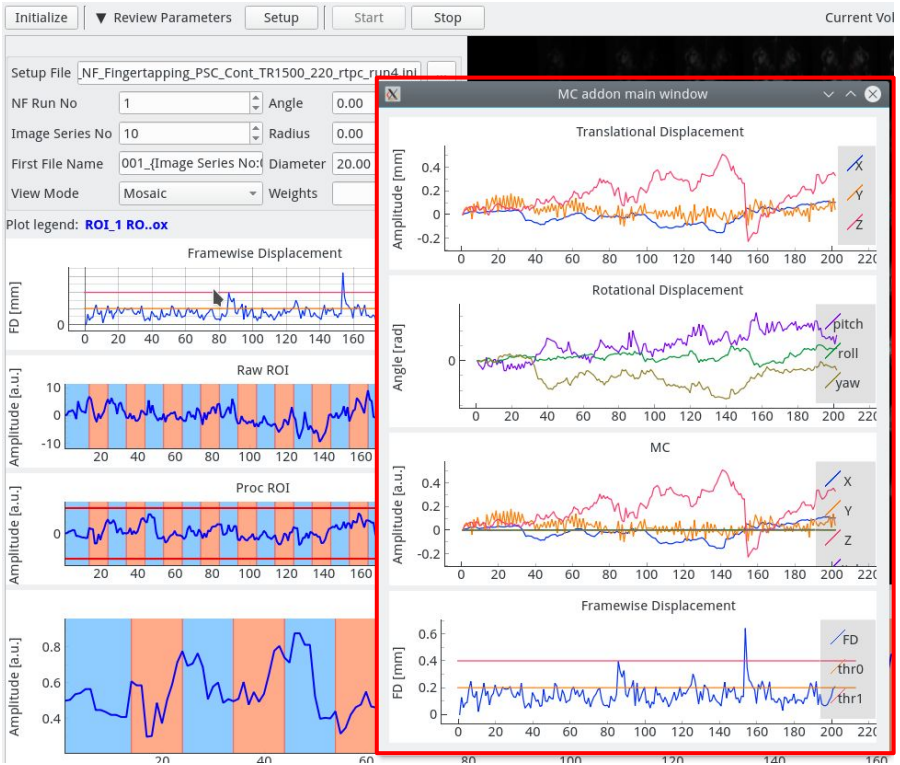



```
pg.setConfigOption('foreground',
                   self.palette().color(QPalette.Foreground))
#self.plotBgColor = (210, 210, 210)
self.plotBgColor = (255, 255, 255)

self.roiImageView = self.createRoiImageView()

self.orthView = projview.ProjectionsWidget(self)
self.layoutOrthView.addWidget(self.orthView)
self.orthView.setVisible(False)

### EDITED BY fdm. Manual modification may break fdm (un)installation and operation
#self.mcPlot = self.createMcPlot()
self.fdmod = fdm.fd_OpenNFT(self)
self.mcPlot = self.fdmod.createMcPlot()
### end of fdm EDITED BLOCK

(self.rawRoiPlot,
 self.procRoiPlot,
 self.normRoiPlot) = self.createRoiPlots()

self.settingFileName = config.ROOT_PATH
self.appSettings = QSettings(self)

self.iteration = 1
self.preiteration = 0
self.pendingFilename = ''
self.isCalculateDcm = False# todo: rename to computeModelInProgress
self.isMainLoopEntered = False
self.typicalFileSize = 0
self.mainLoopLock = threading.Lock()
self.displayData = None
self.displayQueue = queue.Queue()
```

The module integrates into OpenNFT's interface



Double Clicking the MC graph opens a selection window



Displayed thresholds and other parameters can be set in an OpenNFT config file