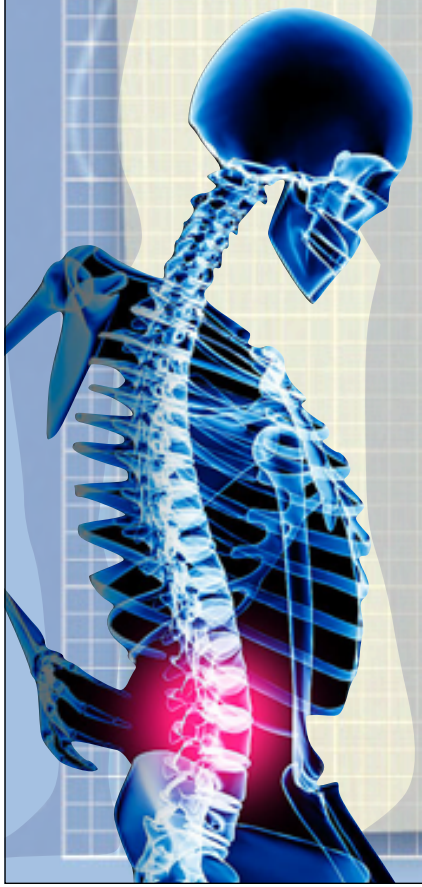
Realignment

Essentials of fMRI Analysis

Justin Hulbert 20/7/2011

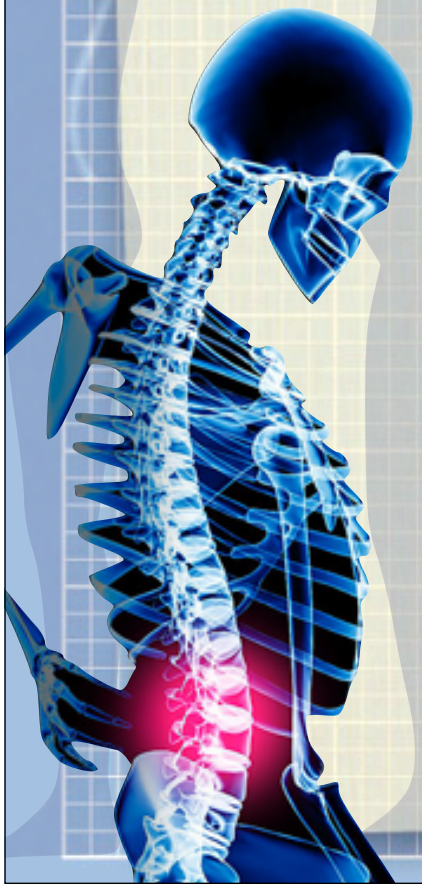


Realignment

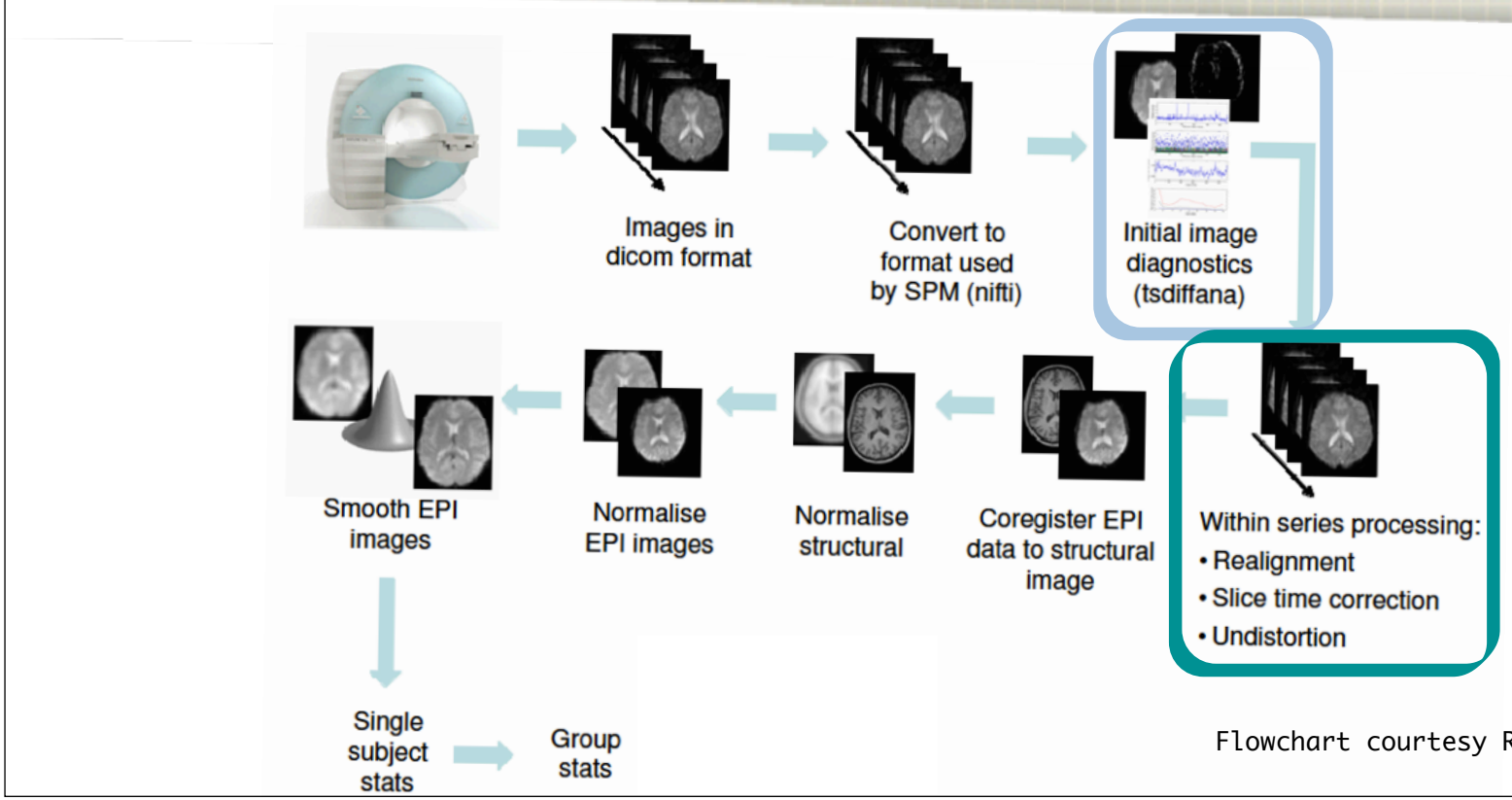
Essentials of fMRI Analysis

Justin H.

20/7/2011



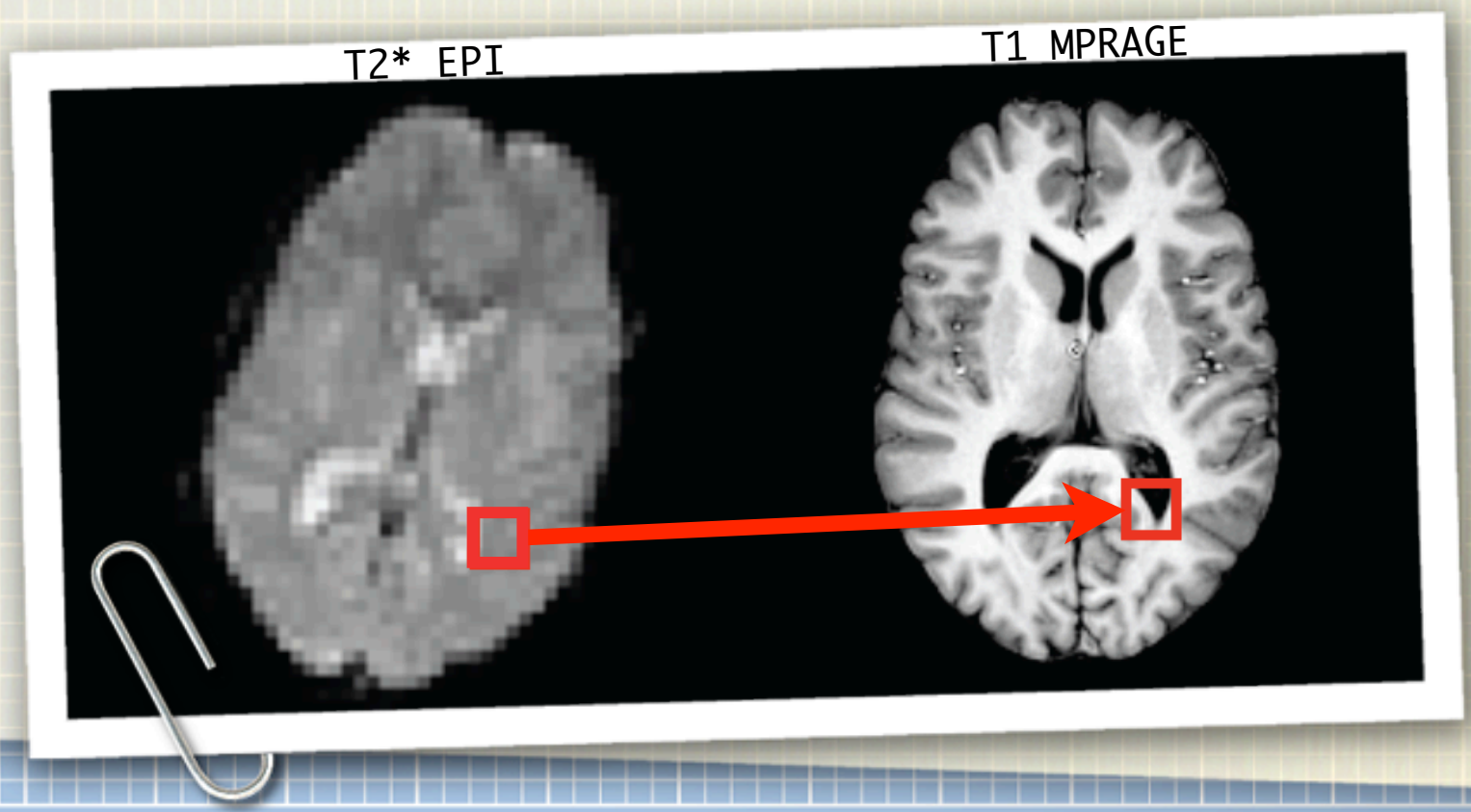
Data Analysis: A Birdseye View



Flowchart courtesy Russell Thompson

Where Is My Mind?

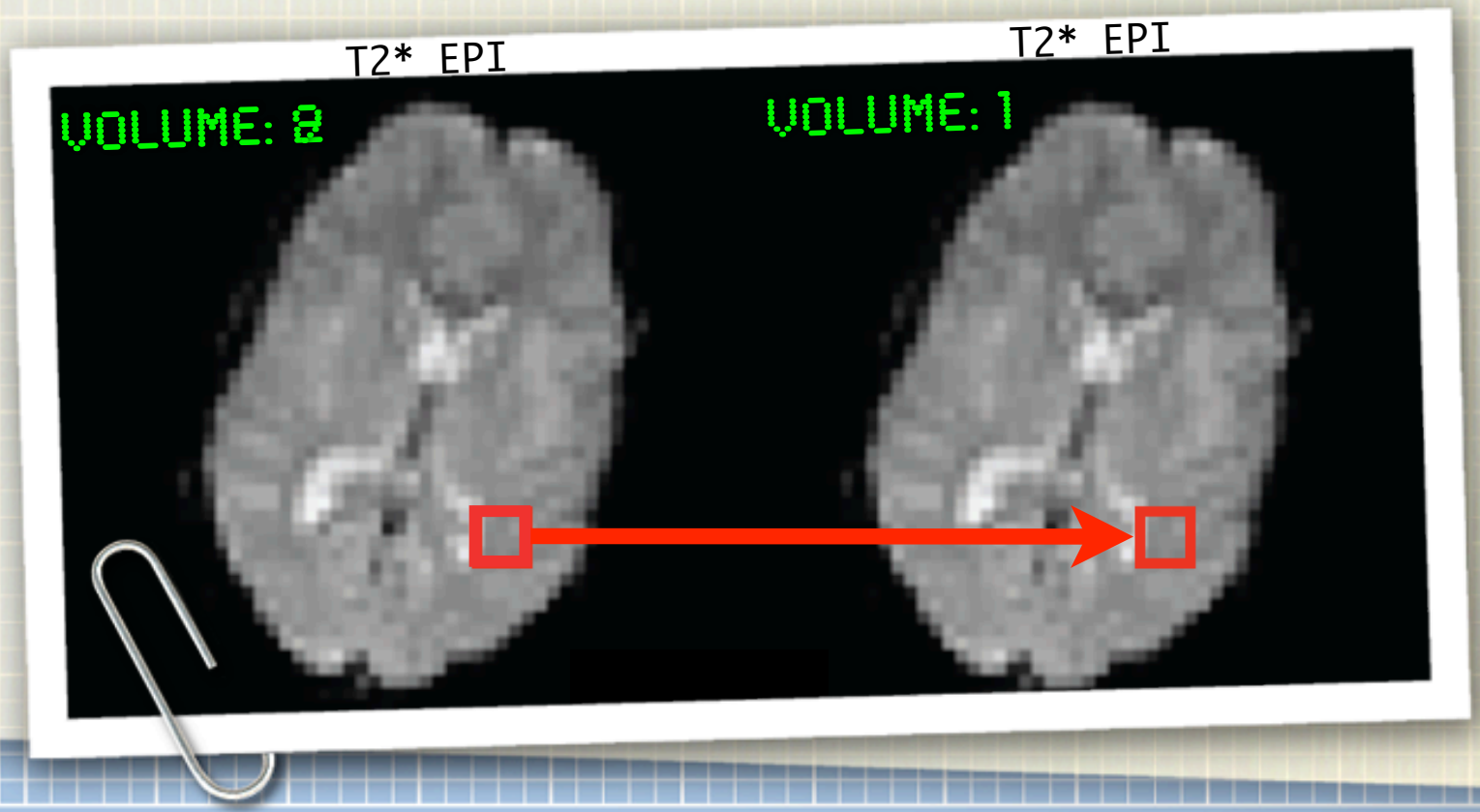
Co-Registration: Aligning two images of different modalities from the same individual




THE
REALIGNMENT
ZONE

The Problem

CSF <<Voxel Contents May Shift>> Gray Matter



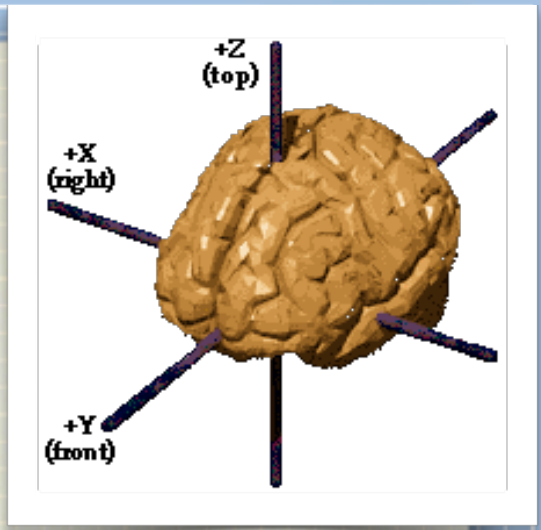
The Goal: Compare Like to Like

Remove movement arti[e]facts in fMRI time-series

- Increases SNR by reducing residual noise
- Reduces likelihood of false positive due to task-correlated motion
  - But can hurt sensitivity to real effects

Six Rigid Body Transformation Parameters:

3 Flavo[u]rs of Translation



Side-to-Side

Translation in X

Front-to-Back

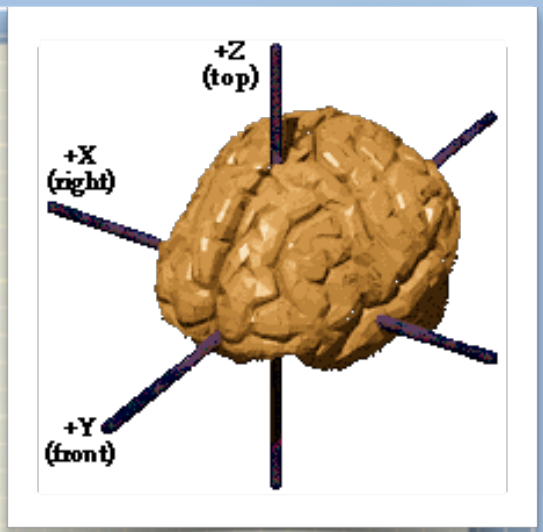
Translation in Y
(Different than Zoom)

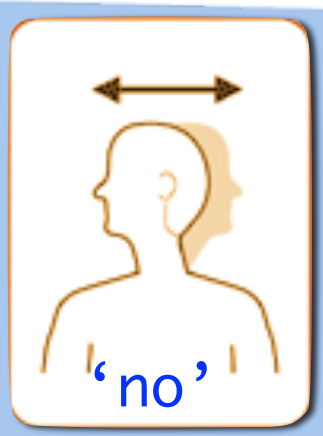
Top-to-Bottom

Translation in Z [Z]

Six Rigid Body Transformation Parameters:

3 Flavo[u]rs of Rotation



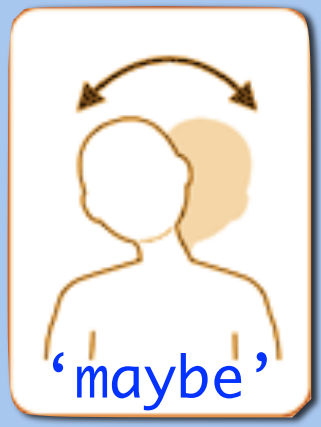


Yaw

Rotation around Z [Z]

Pitch

Rotation around X



Roll

Rotation around Y

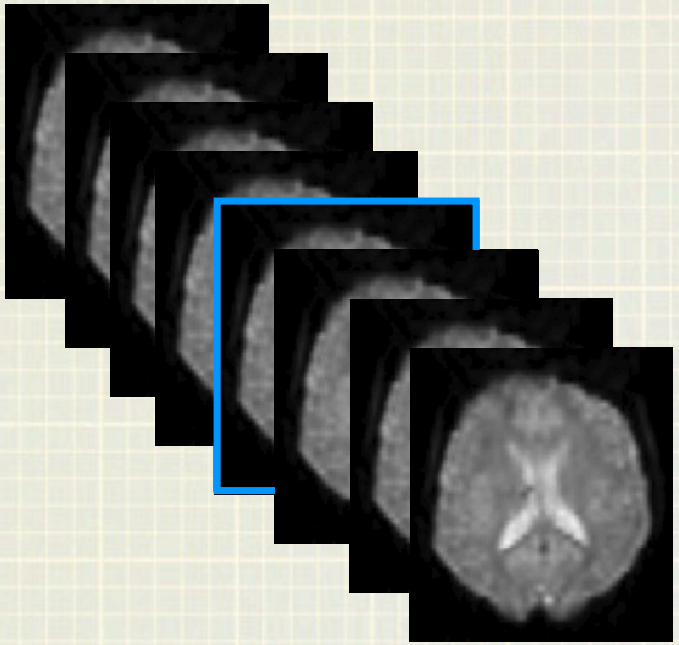
Solutions*

1) Prevention
   a. Comfortably lock down
      - Padding, bite bar,...
   b. Instructions
      - Lie still!
      - Don't talk b/t runs!
      - Minimiz[s]e swallowing!
   c. Design
      a. Keep scan time reasonable
2) Realignment (Motion Correction)
   a. Prospective Movement Correction (PACE)
      - Only rec. for real-time
   b. Offline pre-processing
   c. Include motion parameters
3) Reject bad data

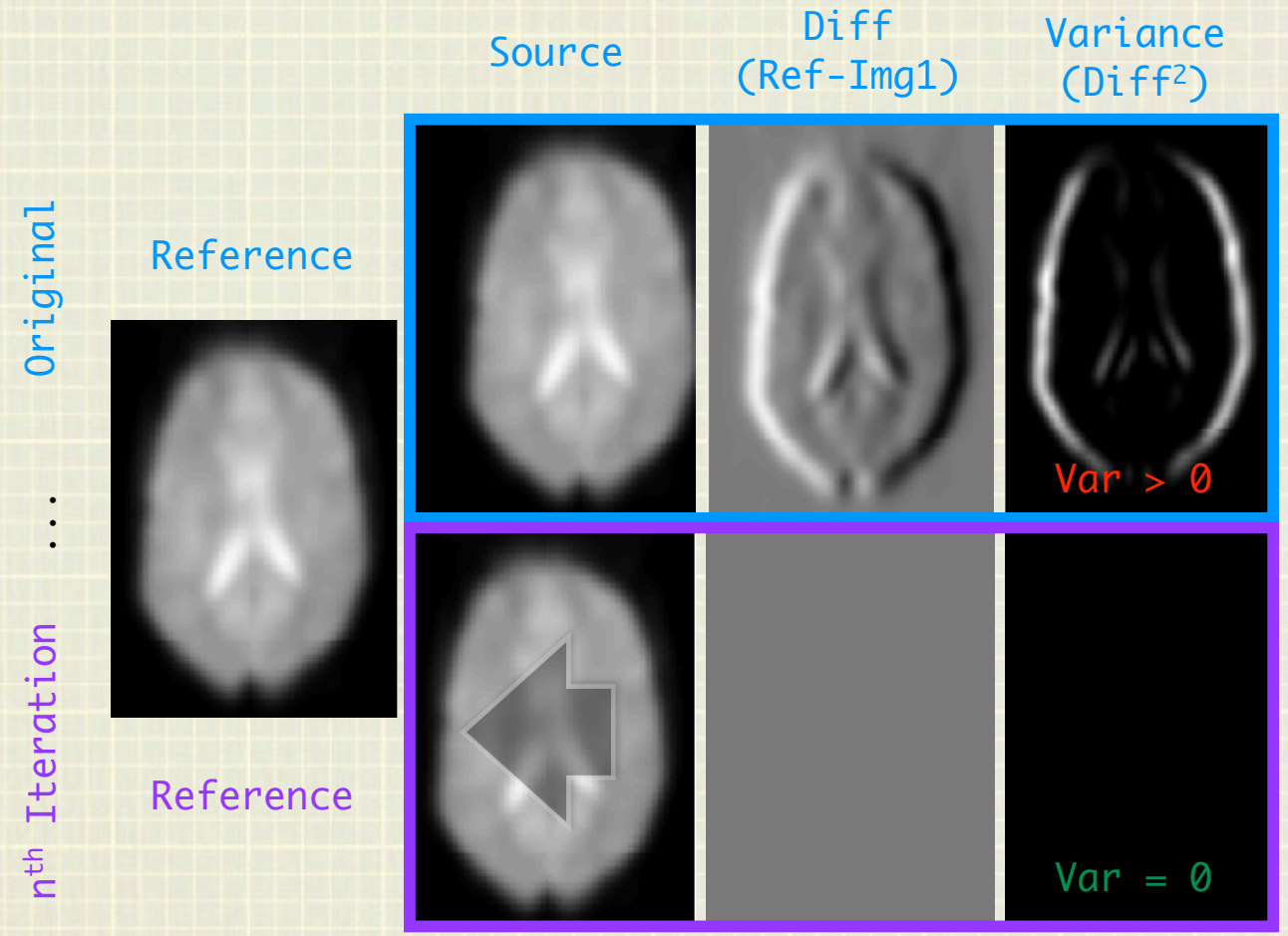
*nothing's perfect

DIY: Realignment

1. Choose a reference image
   - Not necessarily the very first (dummy scans)
   - Should be representative & reasonably close in time to the structural
2. Registration
   - Estimate the 6 transformation parameters b/t each image & reference
     - Minimiz[s]e variance
3. Apply transformation (re-slice)
   - Re-sample each imaging according to the transformation parameters
   - *NOT necessary now for every image if later normaliz[s]ing

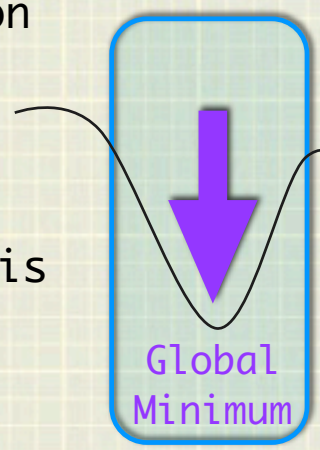


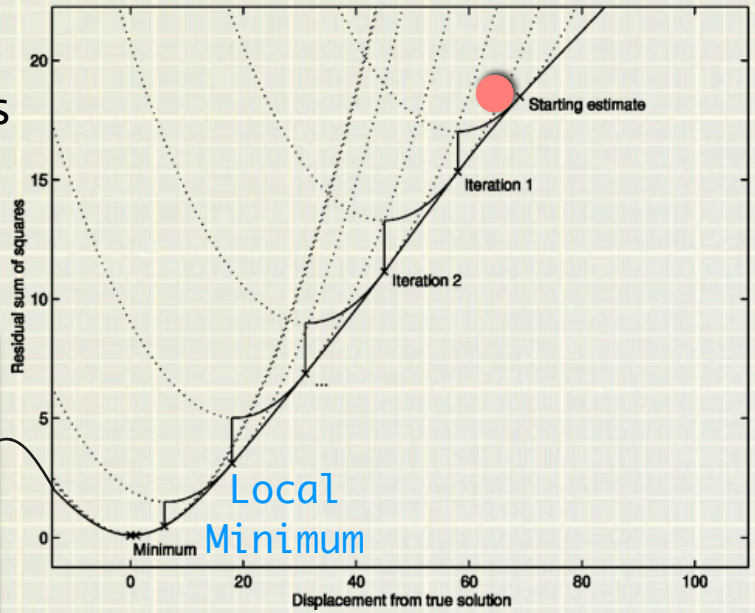
Optimiz[s]ing the Cost Function



Gauss-Newton Algorithm

- Provides a systematic way of modifying the parameters at each iteration
- Attempts to minimiz[s]e mean of squared difference between two images
  - Minimum is estimated by fitting a quadratic at each iteration
- We slowly build up a full transformation matrix out of this





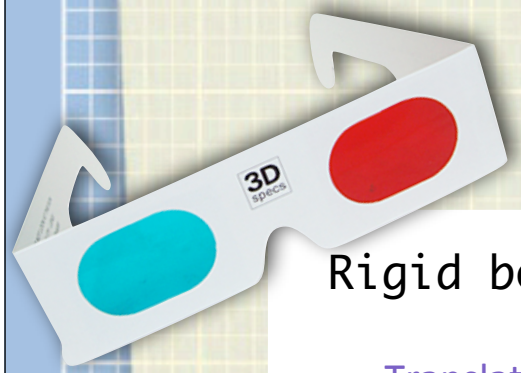
(1) Registration

Affine transformations can be represented in matrix form:

Transformation (mapping) Parameters

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix}$$

New = Original



Rigid body transformations paramateriz[s]ed by:

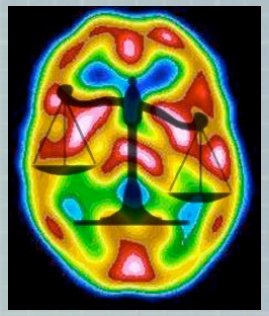
Translations × Pitch about X axis × Roll about Y axis × Yaw about Z axis

$$\begin{pmatrix} 1 & 0 & 0 & X_{trans} \\ 0 & 1 & 0 & Y_{trans} \\ 0 & 0 & 1 & Z_{trans} \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) & 0 \\ 0 & -\sin(\Phi) & \cos(\Phi) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \cos(\Theta) & 0 & \sin(\Theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\Theta) & 0 & \cos(\Theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \cos(\Omega) & \sin(\Omega) & 0 & 0 \\ -\sin(\Omega) & \cos(\Omega) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

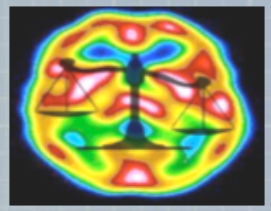
<<SPM does it from right to left

Order of Operations

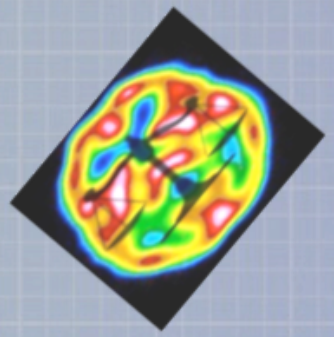
Original



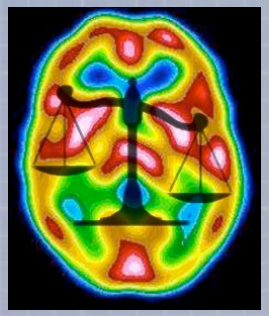
50° around Y



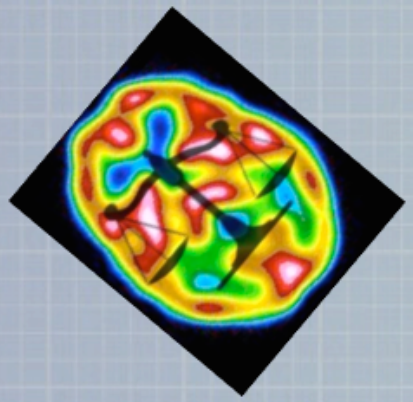
50° around Z



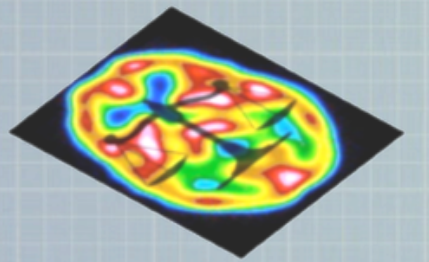
Original



50° around Z

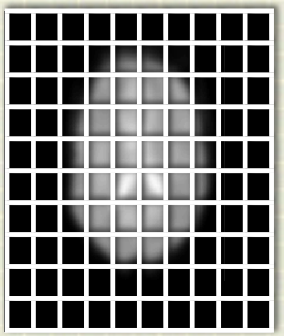


50° around Y

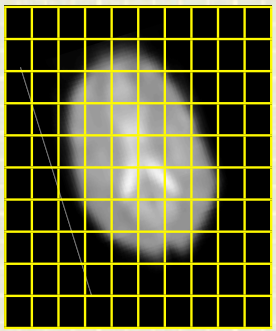


(2) Transformation (Reslicing)

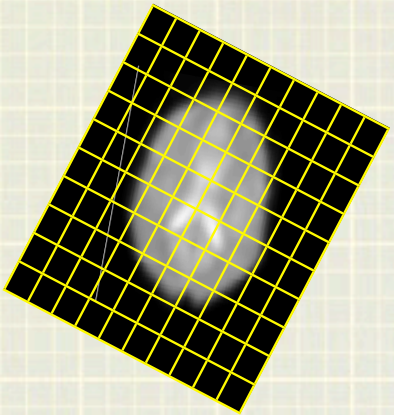
Reference

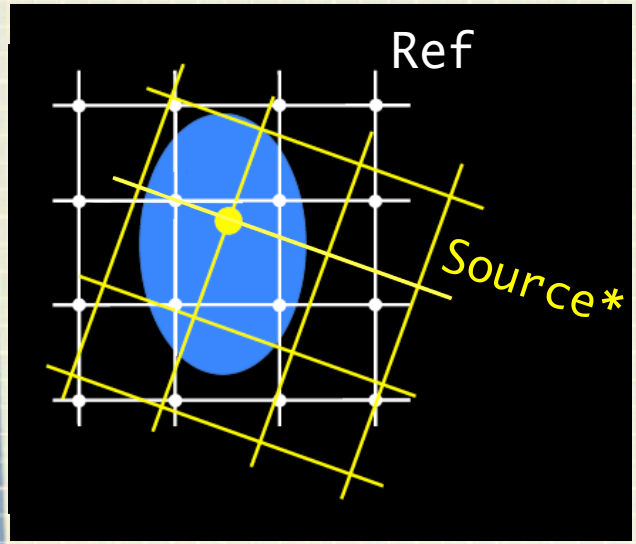


Source



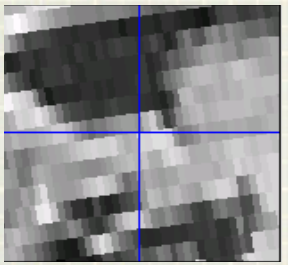
Source*



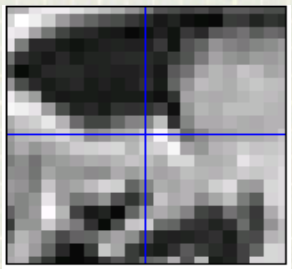


- Transform according to the estimated parameters & resample to match the reference grid
- To do this, we need estimate intensity values between grid points
  - So...interpolate!

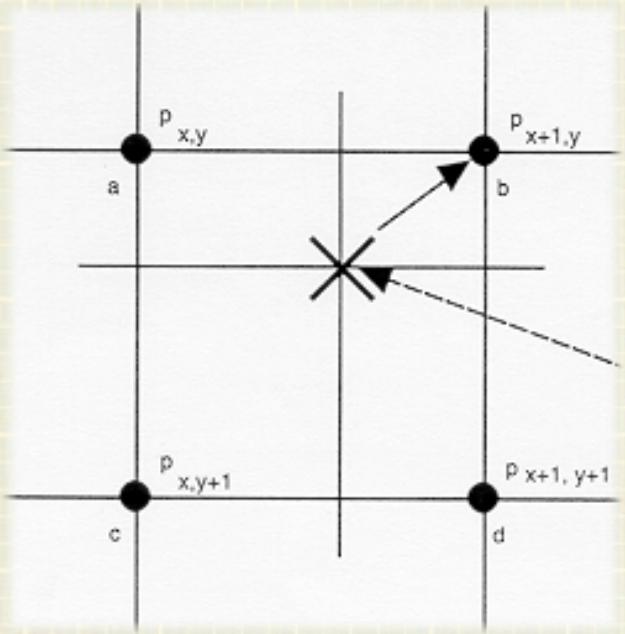
Reoriented



Re-sliced

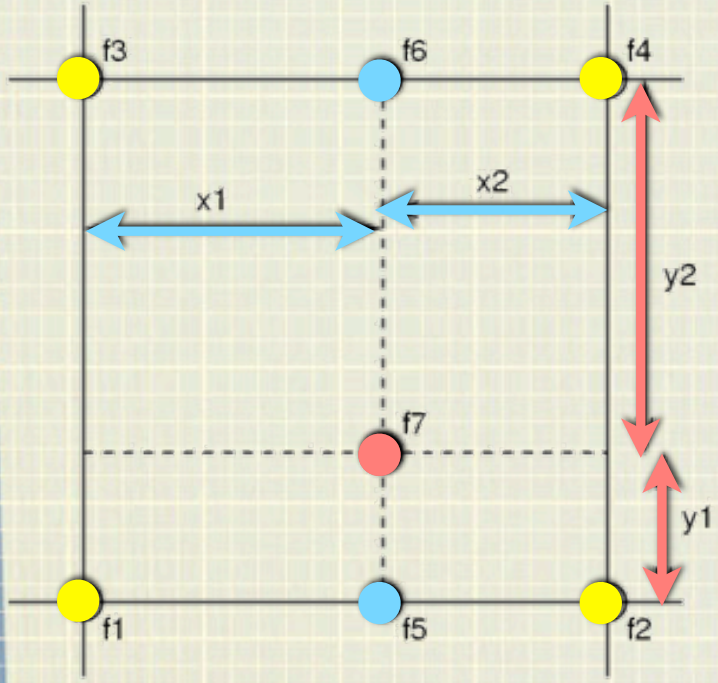


Nearest Neighbo[u]r (Zero-Order Hold)



- Takes value of closest voxel
- Original voxel intensities preserved
- Very fast
- But...image is degraded considerably
  - “Blocky” images

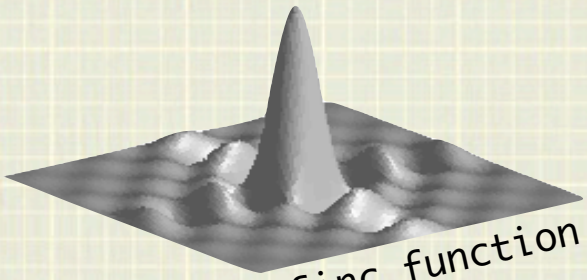
Trilinear Interpolation (First-Order Hold)



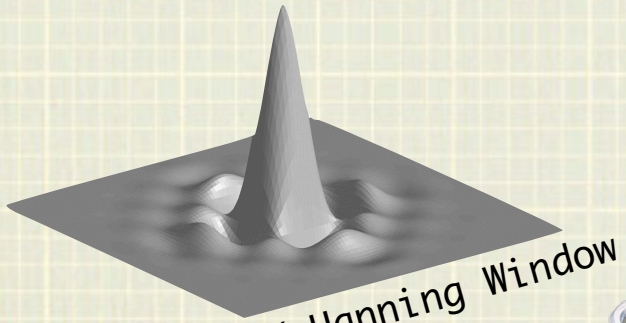
Represented in 2D for illustration

- Takes the weighted average of the neighbo[u] ring voxels
  - $f_5 = f_1x_2+f_2x_1$
  - $f_6 = f_3x_2+f_4x_1$
  - $f_7 = f_5y_2+f_6y_1$
- Slower but less blocky than nearest neighbo[u]r
- Loses some high frequency information (smoothing)

Windowed Sinc Interpolation



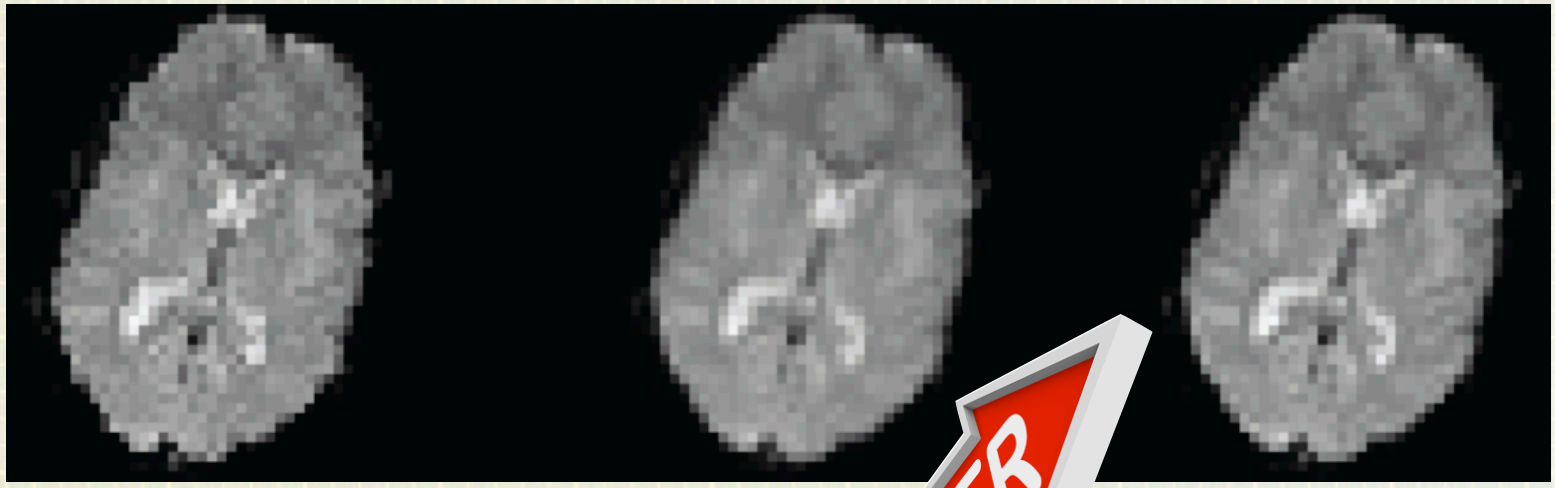
Sinc function



w/ Hanning Window

- Sinc interpolation gives results closest to a Fourier interpolation (which is ideal) but in real space
  - Convolve sinc function center[e]d on the point to be resampled
  - Theoretically, every voxel in sample is used, but an approx. using subset of near neighbo[u]rs speeds things up
    - Hence the ‘window’
- Greatly reduces arti[e]facts, but S-L-O-W
- Interpolation performed on each dimension sequentially

Results of Interpolation

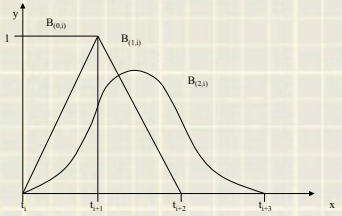


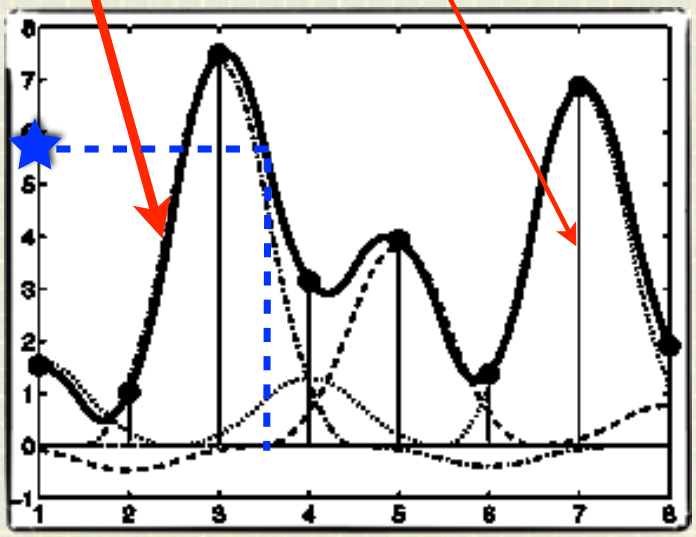
Nearest Neighbo[u]r

Trilinear

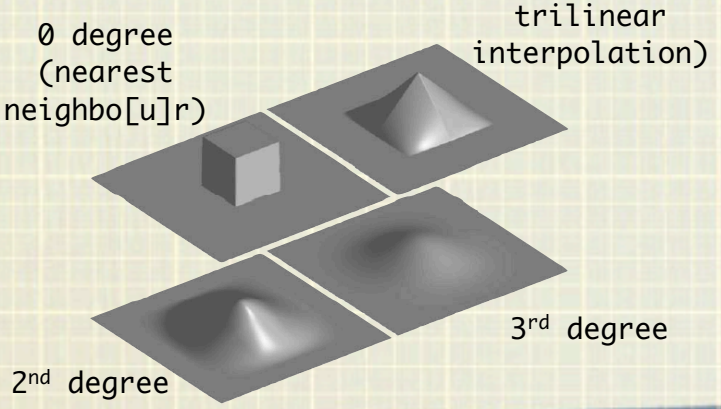
Sinc

B[asis]-splines

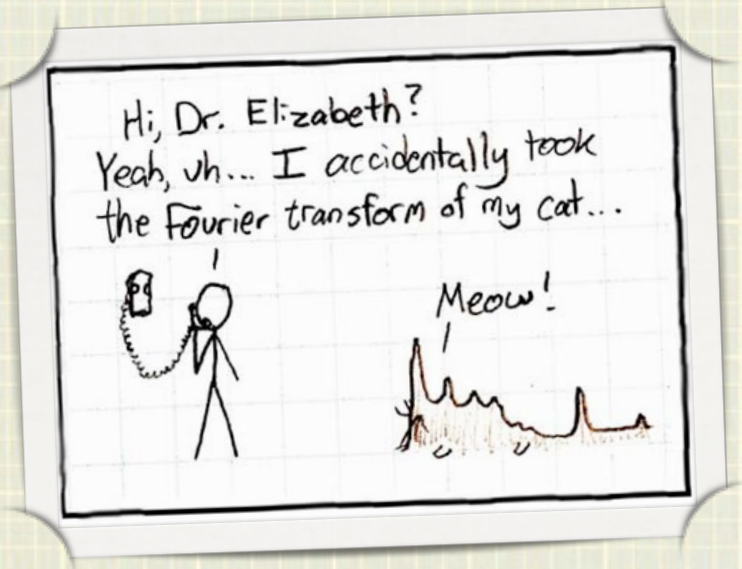




- A form of ‘generalized interpolation’
  - First transforms image into basis functions before applying local convolution
  - Re-sampling involves computing linear combination of functions
  - Done sequentially along each dimension
  - Far more efficient than classical interpolation



Fourier Methods

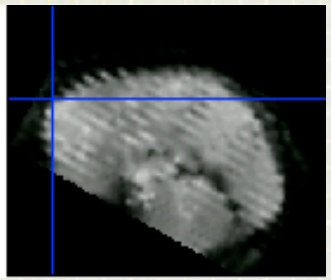


- Faster than higher-order interpolation
- Uses fast Fourier transforms
  - Convolution performed rapidly in Fourier space
- However, can only handle translations (not really built for rotations currently)

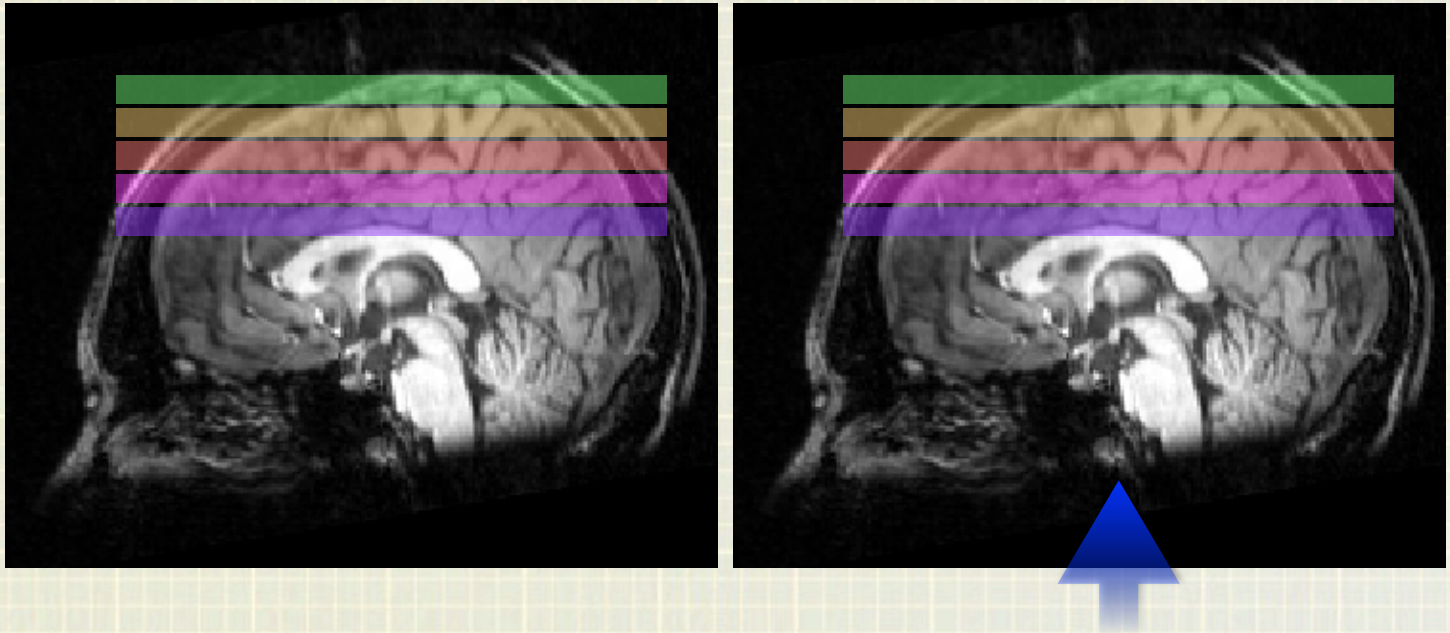
Residual Errors

- Post-realignment, still variance due to movement left over:
  - Can be due to shifts between and within slice acquisition
  - Interpolation arti[e]facts
  - Non-linear distortions due to inhomogeneities of magnetic field...
- Spin-history changes (@ their worst when acquired interleaved)
  - Residual magnetiz[s]ation effects of previous scans
  - Movement may make effective TR longer/shorter for some slices
- Adding motion parameters may help

Spin History 'Striping'



The Limits of Realignment



- Make sure you have good coverage of ROI
- If expect this problem, perhaps use PACE

SPM Walkthrough

1) Realign ("Est & Reslice")

2) Select images for each session. Don't include dummies; First should be 'good'

3) Speed (0.001) vs. Quality (1)

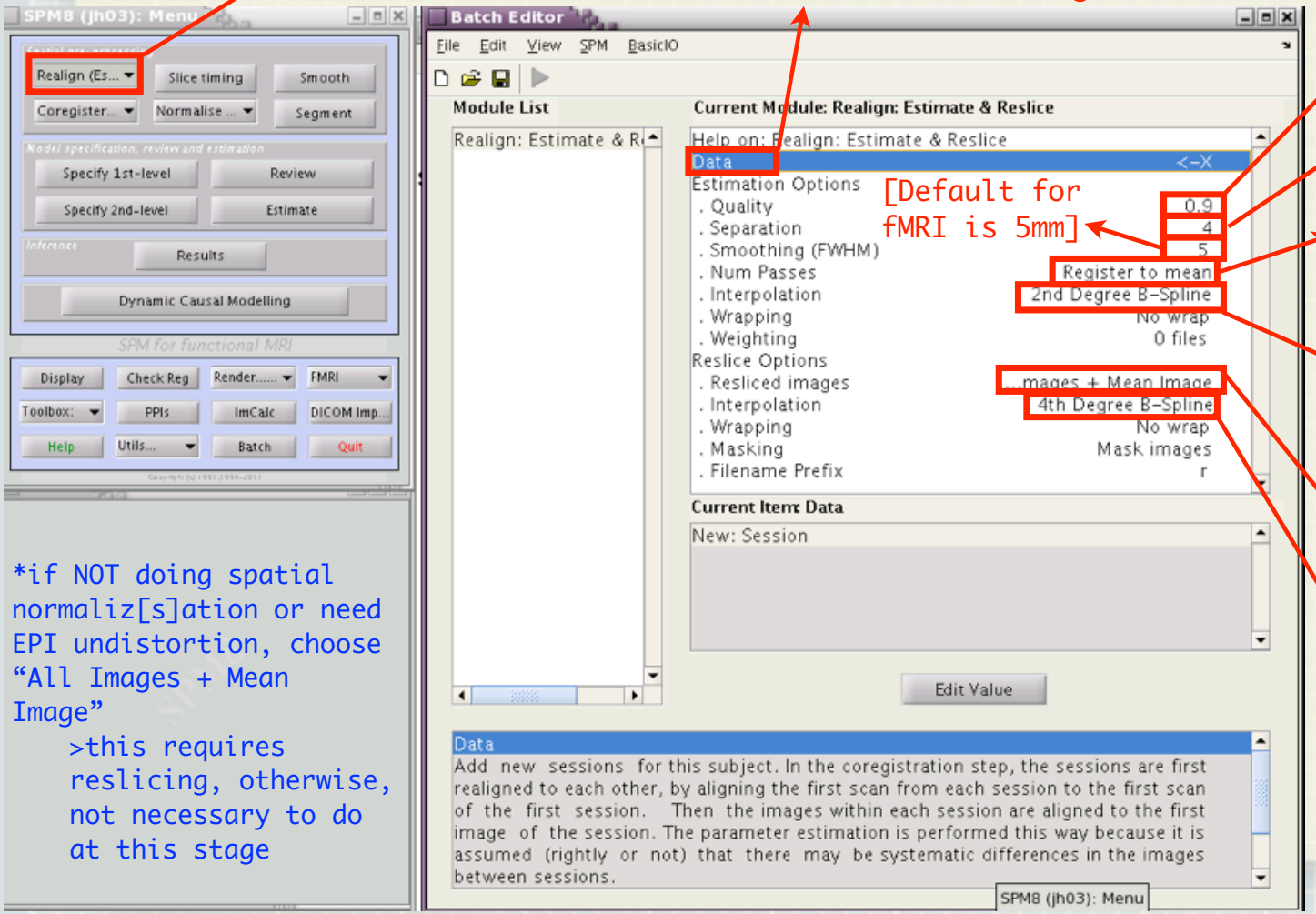
4) Separation (mm) b/t points sampled in ref (smaller = better, slower)

5) Use 1st image or run registration 2x

6) Higher degree = better, slower (uses more neighbo[u]ring voxels

7) Create 'Mean image only'*

8) Set interpolation degree (higher=slower/ more neighbo[u]rs)

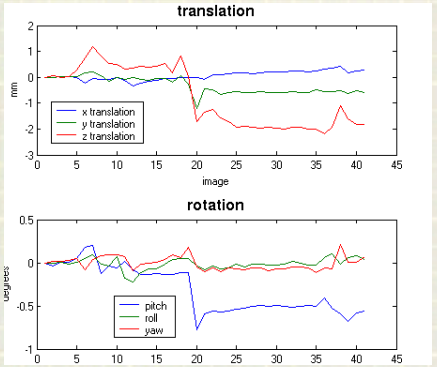


[Default for fMRI is 5mm]

*if NOT doing spatial normaliz[s]ation or need EPI undistortion, choose "All Images + Mean Image"

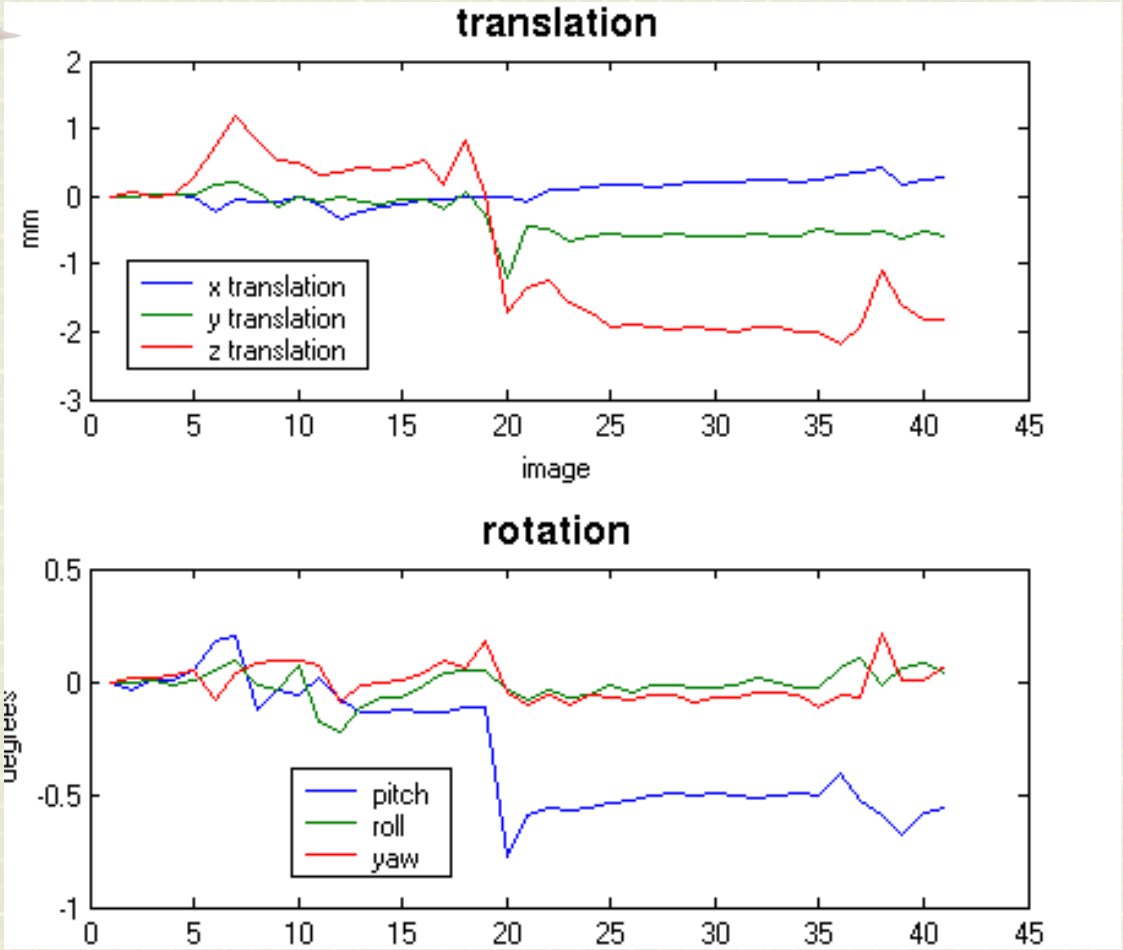
>this requires reslicing, otherwise, not necessary to do at this stage

SPM Outputs



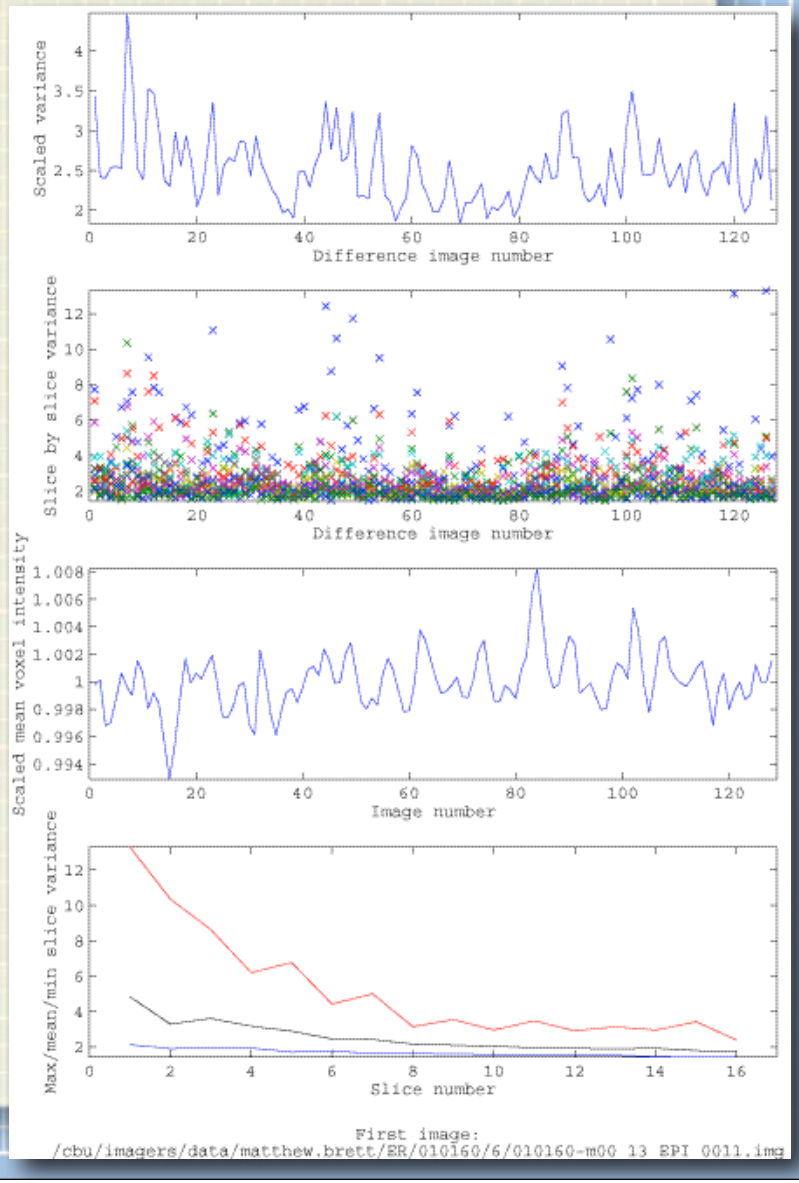
- spm.ps
  - Graphs of the estimated motion
- rp_[firstimage].txt
  - all realignment parameters to realign to the first image file
  - Number of rows = number images in run
    - Need this for later stats & can be modeled as confounds
- means*.nii
  - Mean of the realigned/resliced images, used in coregistration
- r*.nii
  - Resliced images (only required if NOT doing spatial normaliz[s]ation or IF planning to apply EPI undistortion)

Estimated Motion



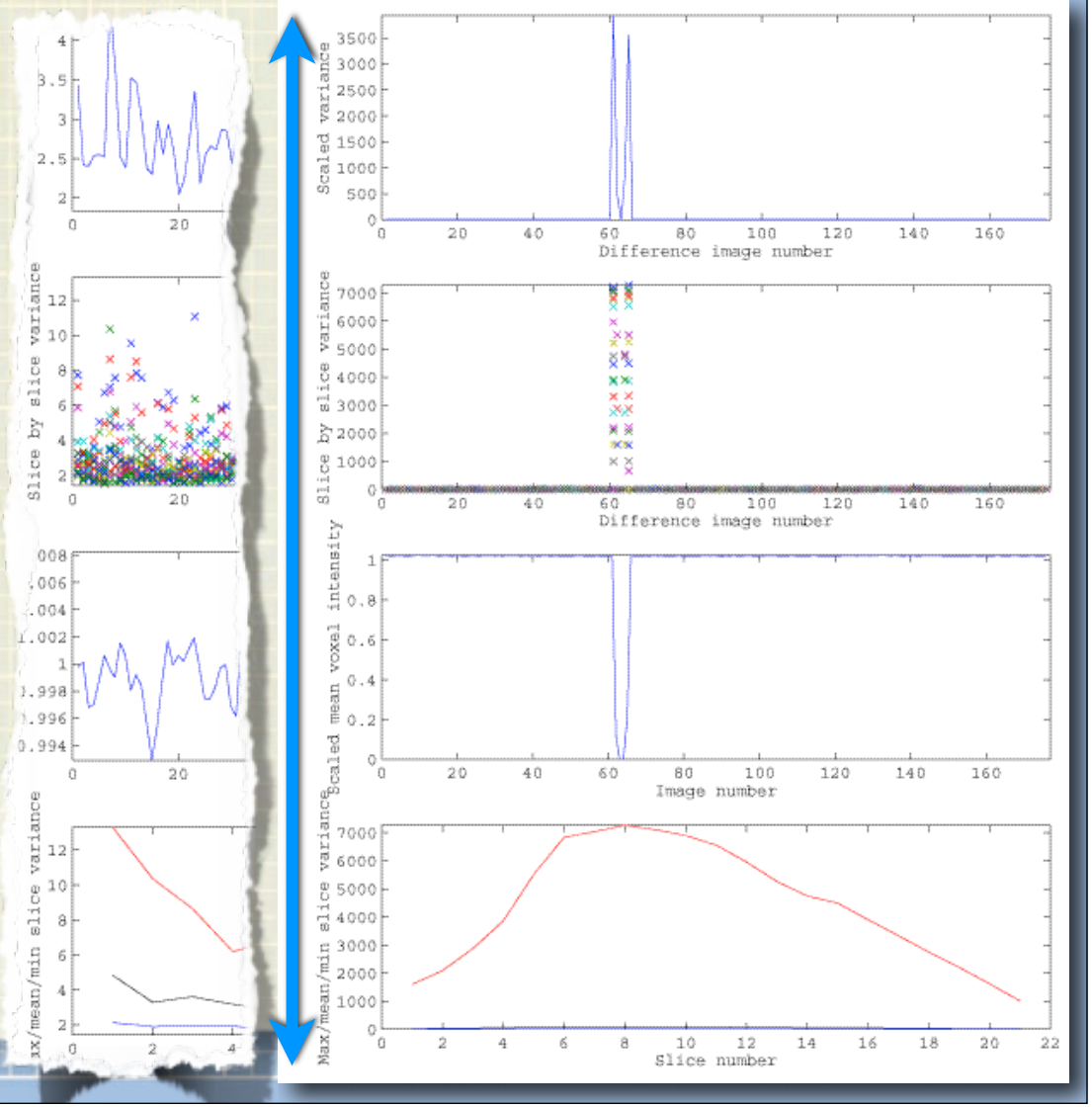
tsdiffana: Typical Data

- Run before realignment
- Available to outsiders: http://imaging.mrc-cbu.cam.ac.uk/imaging/DataDiagnostics
  - >>tsdiffana
    - Select files for one session & press "done"
    - Say "no" when asked if should write difference images
- Writes mean, stdev, and slice-to-slice variance



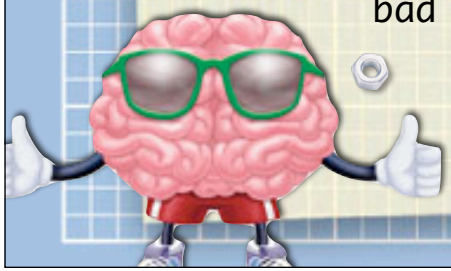
fsdiffana: Problem Data

- Here the gradient coil stopped working for a few volumes in the middle of the run
- Notice the scale:



Rules of Thumb

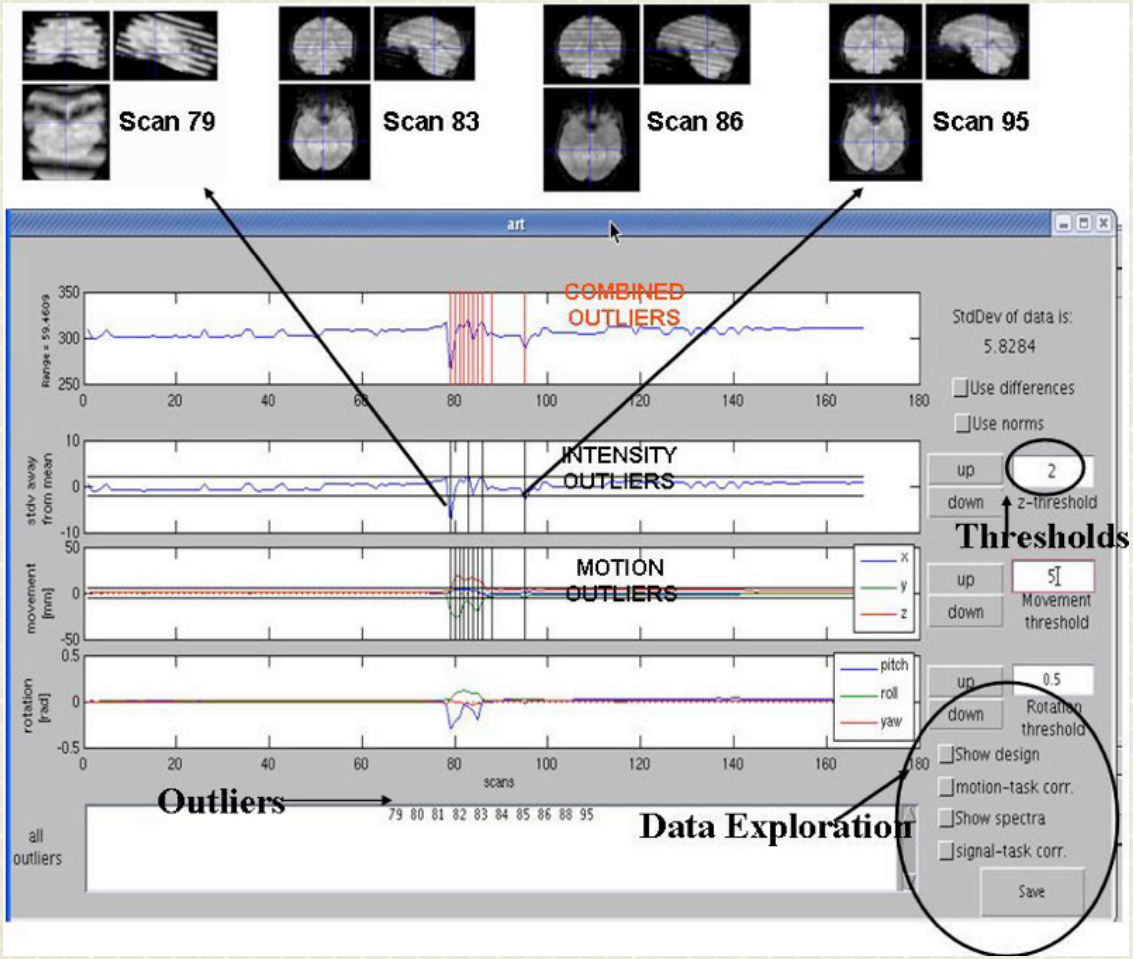
- No simple rules to define 'too much' motion, but...
  - >2-3mm or 2 degrees (.38 radians) in any dimension, exclude subject (i.e., ~voxel size)
    - Ardekani et al. (2001) find SPM can handle up to 10mm summed across dimensions
    - If >.5 mm, consider including movement parameters (will eat up some DoFs, but typically have plenty)
  - Inside the brain, 1mm shifts can cause 3-5% signal change
    - At the edges, 3-5% change is caused by 1/20th of a pixel shift (~187 microns)!
- General points:
  - Rotational data harder to correct
  - Motion during scan session worse than in between
  - If motion correlated w/ task, can you redesign & rerun?
- Most researchers don't recommend removing/replacing/'fixing' bad slices/scans
  - But it is possible to come up with nuisance regressors for affected volumes (e.g. for epilepsy)



Slice Timing <> Motion Correction?

- If you run slice timing correction 1st then abrupt movements between scans will cause it to interpolate between different brain regions :(
- If you run motion correction 1st, slices no longer necessarily correspond to acquisition order, so timing correction won't be appropriate :(
  - Problem is worse if you have interleaved slice ordering, in which case you should run slice time correction 1st & select slice timing corrected images for realignment
- CBU acquires in sequence (down), and many prefer to run motion correction 1st under this system
  - But whether this matters hasn't been investigated fully

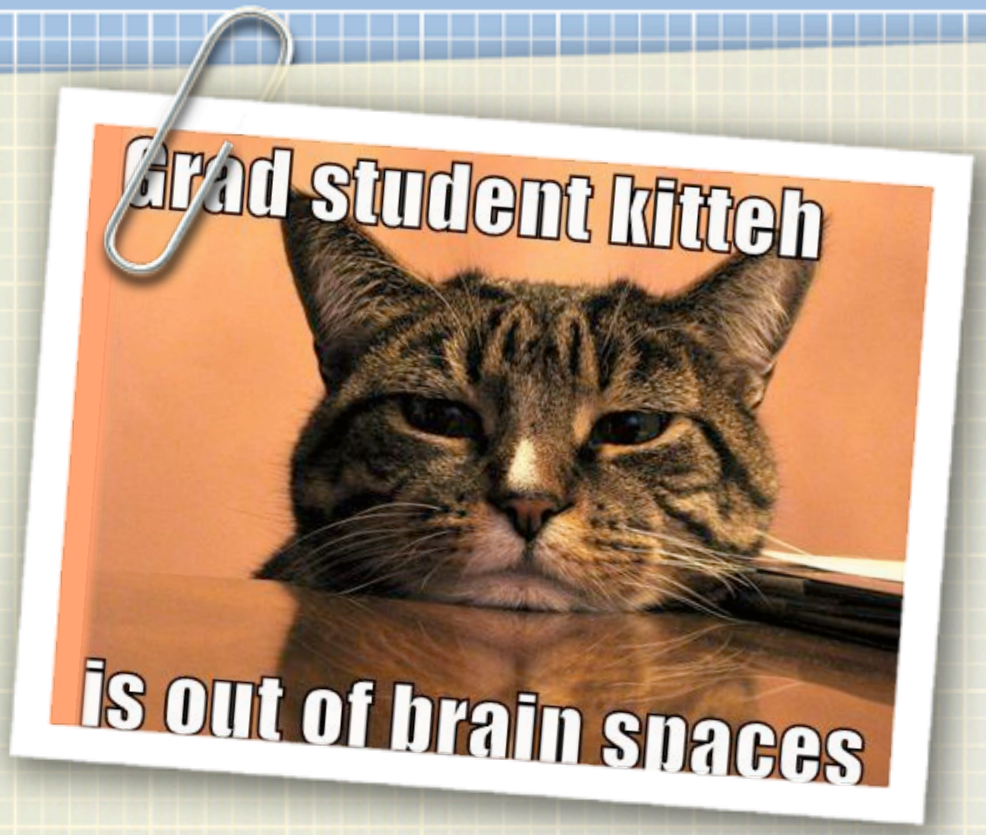
Artifact Detection Toolbox (ART)



http://web.mit.edu/swg/software.htm

Acknowledgements

- SPM Methods for Dummies
  - www.fil.ion.cul.ac.uk/spm/doc
- FSL Course
  - www.fmrib.ox.ac.uk/fslcourse
- CBU Imaging wiki
  - imaging.mrc-cbu.cam.ac.uk/imaging/CbuImaging
- Jody Culham's fMRI for Newbies
  - www.fmri4newbies.com
- Russell Thompson's fMRI Basics Course
- Florida's citrus growers



Warp Speed Ahead...