fMRI Preprocessing & Noise Modeling

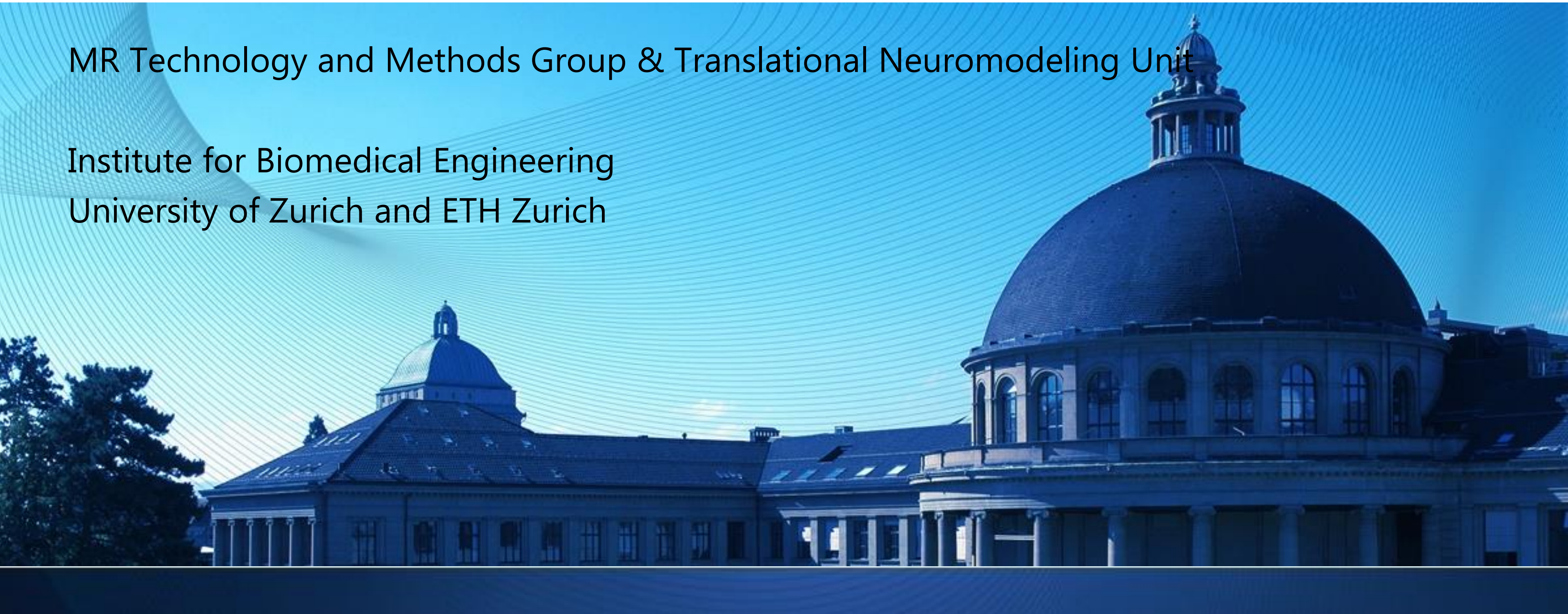
An SPM Tutorial

Lars Kasper

October 10th/ October 17th, November 21 2017

MR Technology and Methods Group & Translational Neuromodeling Unit

Institute for Biomedical Engineering
University of Zurich and ETH Zurich



Scan Parameters - Your Data

Scan Parameter	Value
TR (s)	2.2
TE (ms)	EPI:36; B0 map: 2 and 4.3
nSlices	32
Slice order	Ascending ([1:32])
nVolumes	145
nDummies	3
Voxel size (mm^3)	2 x 2 x 3
Bandwidth / pixel (Hz)	29.5 (=1/Total readout duration)
Fat shift (=Blip direction)	Posterior (-1)

Outline

- Setting up a Preprocessing Pipeline in SPM:
  The Batch Editor

Setup Pipeline
DEMO 1

- Monitoring and Comparing Pipelines:
  Quality Assurance

Quality Monitoring
DEMO 2

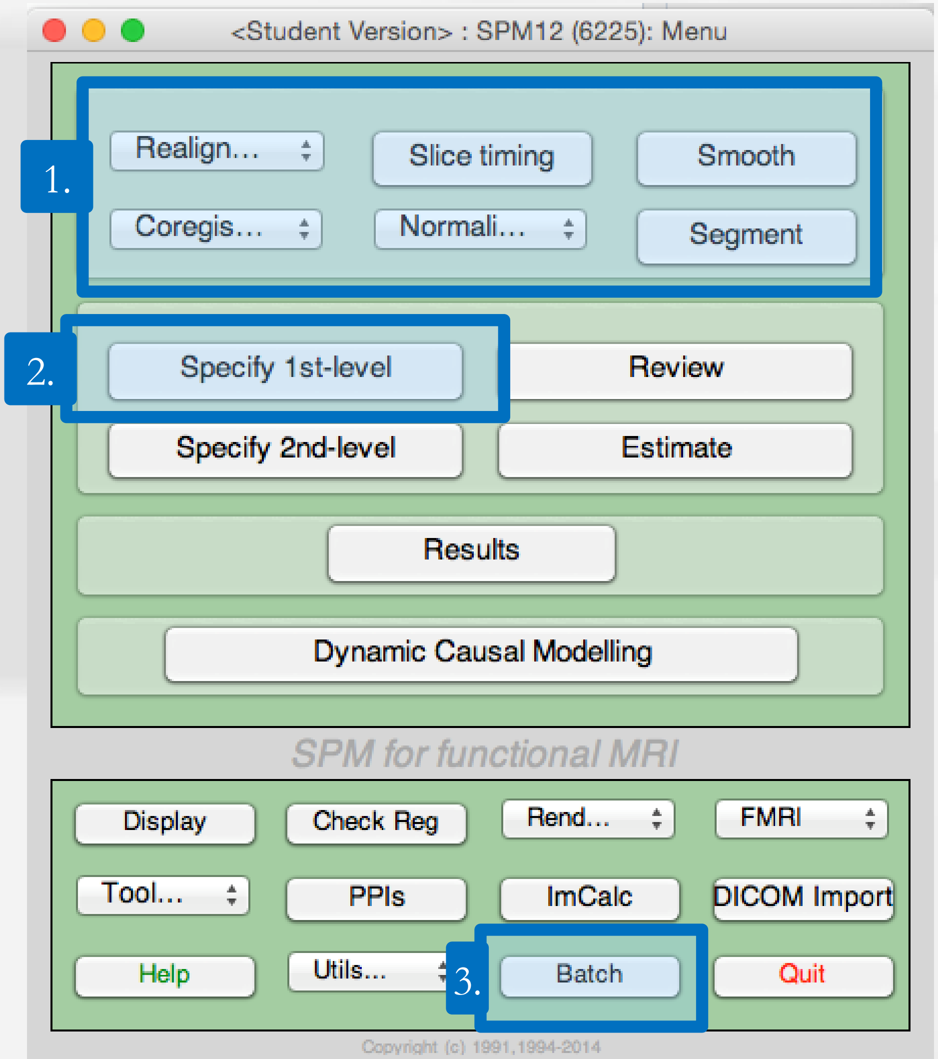
- Multi-subject Pipelines:
  Automatisation

Multi-Subject
DEMO 3

- Integrating Own Code and Toolboxes:
  Physiological Noise Modeling and Evaluation

Noise Modeling
DEMO 4

The SPM GUI

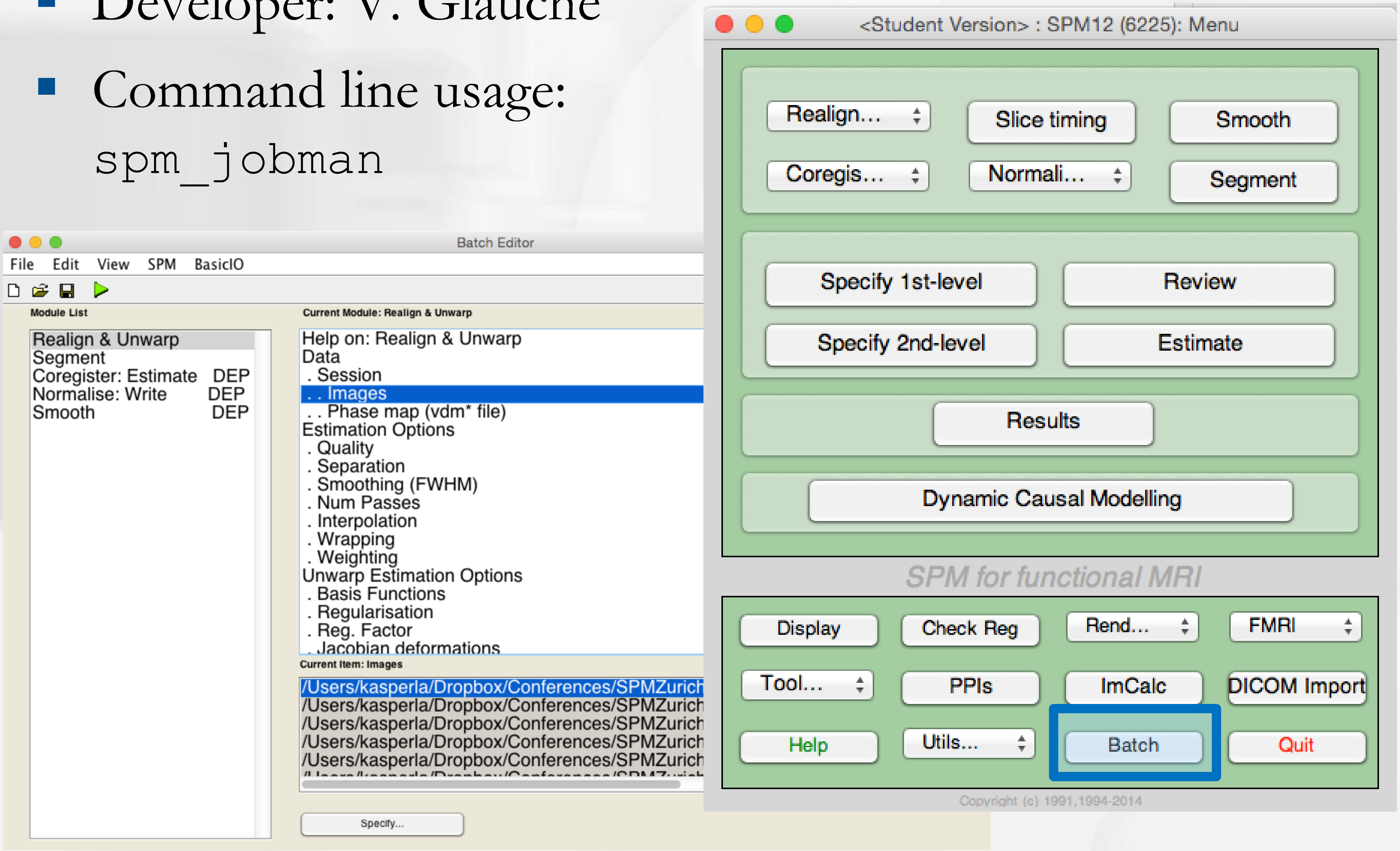


- Preprocessing
  - Realignment
  - Slice-Timing Correction
  - Co-Registration
  - Unified Segmentation & Normalisation
  - Smoothing...
- Noise Modeling
  - Physiological Confound Regressors
- The Batch Editor

The Batch Editor in SPM

- Developer: V. Glauche
- Command line usage:

  `spm_jobman`



Module List

```
Realign & Unwarp
Segment
Coregister: Estimate   DEP
Normalise: Write       DEP
Smooth                 DEP
```

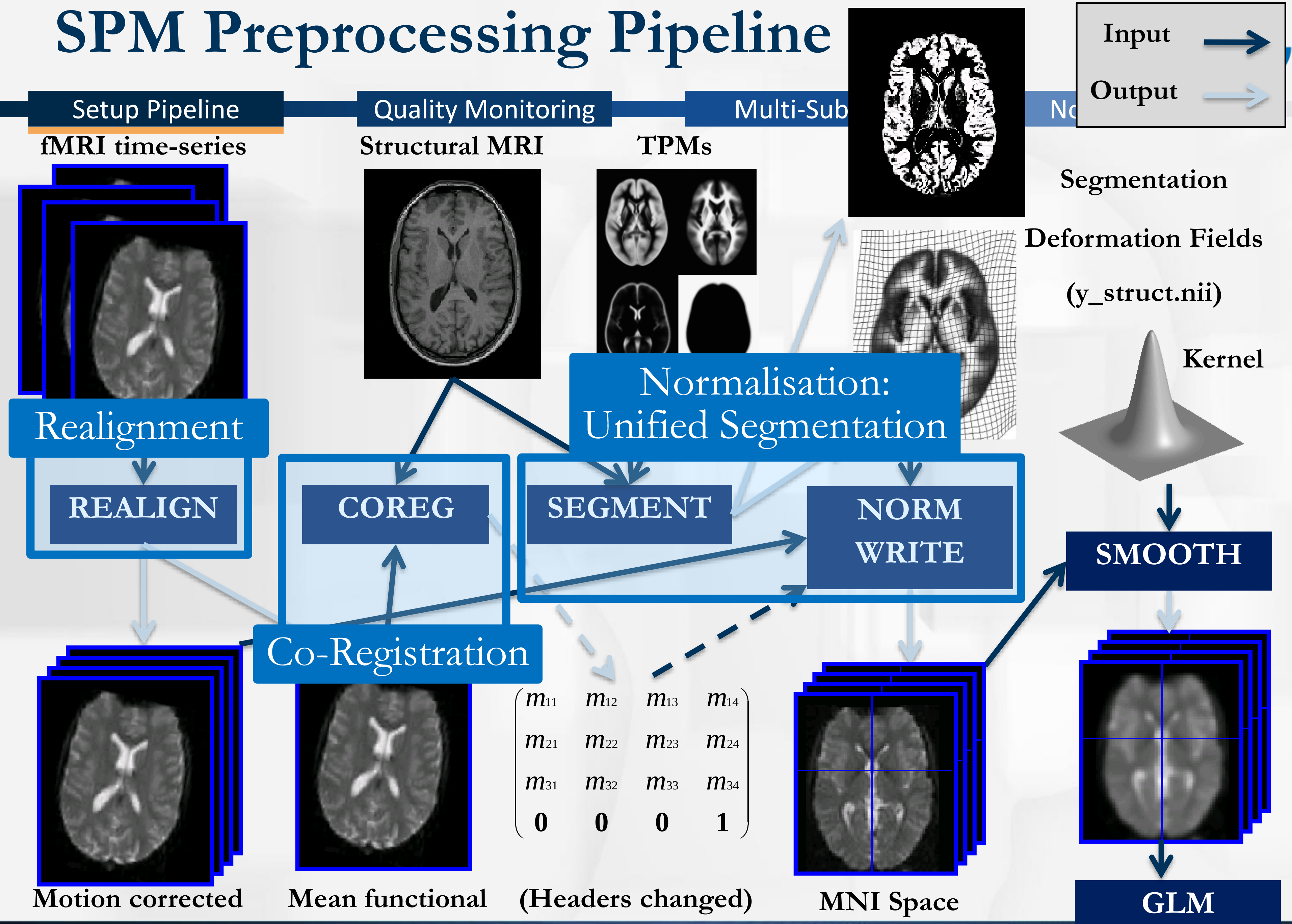
Current Module: Realign & Unwarp

```
Help on: Realign & Unwarp
Data
 . Session
 . . Images
 . . Phase map (vdm* file)
Estimation Options
 . Quality
 . Separation
 . Smoothing (FWHM)
 . Num Passes
 . Interpolation
 . Wrapping
 . Weighting
Unwarp Estimation Options
 . Basis Functions
 . Regularisation
 . Reg. Factor
 . Jacobian deformations
```

Current Item: Images

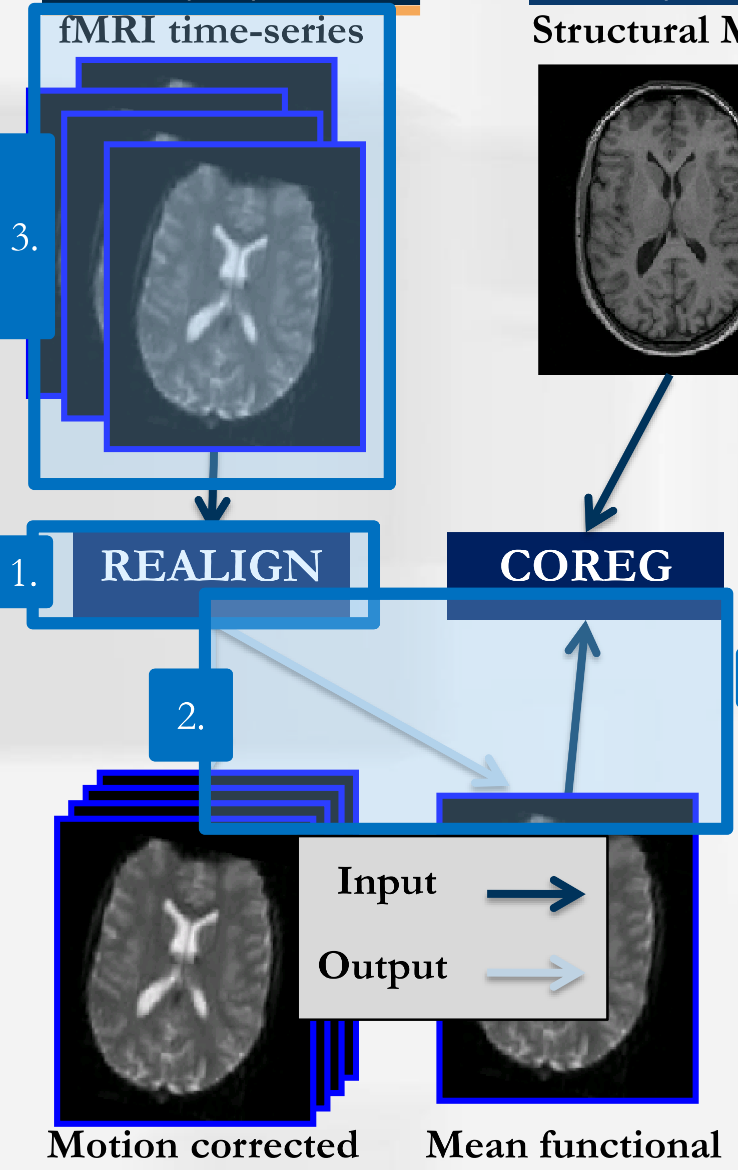
```
/Users/kasperla/Dropbox/Conferences/SPMZurich
/Users/kasperla/Dropbox/Conferences/SPMZurich
/Users/kasperla/Dropbox/Conferences/SPMZurich
/Users/kasperla/Dropbox/Conferences/SPMZurich
/Users/kasperla/Dropbox/Conferences/SPMZurich
```

SPM Preprocessing Pipeline

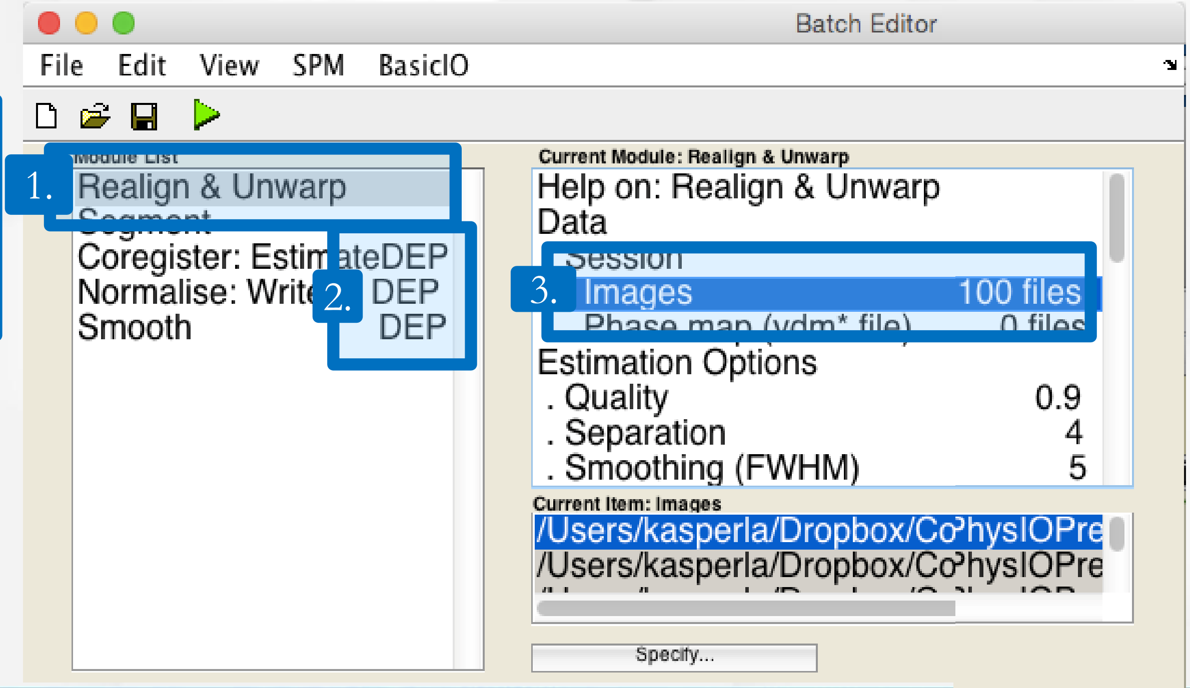


$$\begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

From Pipelines to Batches



- Elements of a Pipeline
  1. Modules
  2. Dependencies
  3. Data (Subject-specific)
- Philosophy: Batch = Materialised Pipeline



Module List

Realign & Unwarp
Segment
Coregister: Estimate DEP
Normalise: Write DEP
Smooth DEP

Current Module: Realign & Unwarp

Help on: Realign & Unwarp
Data
. Session
. Images 100 files
. Phase map (vdm* file) 0 files
Estimation Options
. Quality 0.9
. Separation 4
. Smoothing (FWHM) 5

Current Item: Images

/Users/kasperla/Dropbox/CoPhysIOPre
/Users/kasperla/Dropbox/CoPhysIOPre

Demo 0

- Setting up the data for all demos & checking it
  - Source: Methods & Models 2017 course
- After download:
  1. Unzip archive `AllSubjects.zip`
     - location: `<YourPath>/StudyName/Data/Raw/`
  2. Open Matlab, add Code/tnufmri path to it
     - `Addpath(genpath(<YourPath>/2017/Code/tnufmri`
     - Alternatively: download TNU fMRI preprocessing pipeline from `git@tnurepository.ethz.ch:pipelines/tnufmri.git`
  3. Inspect raw functional *and* structural data carefully
     - Required for each subject
     - Check all files and **Reorient** to anterior commissure via Display button
     - Preferred viewer for time series: SPM CheckReg (s. next page)

Demo 0: CheckReg-Magic

- Recommended Plotting: Check Reg
  - via Batch Editor: SPM => Util => Check Reg.
  - Matlab command line:

    ```
    spm_check_registration(‘img1.nii’, ‘img2.nii’, …)
    ```

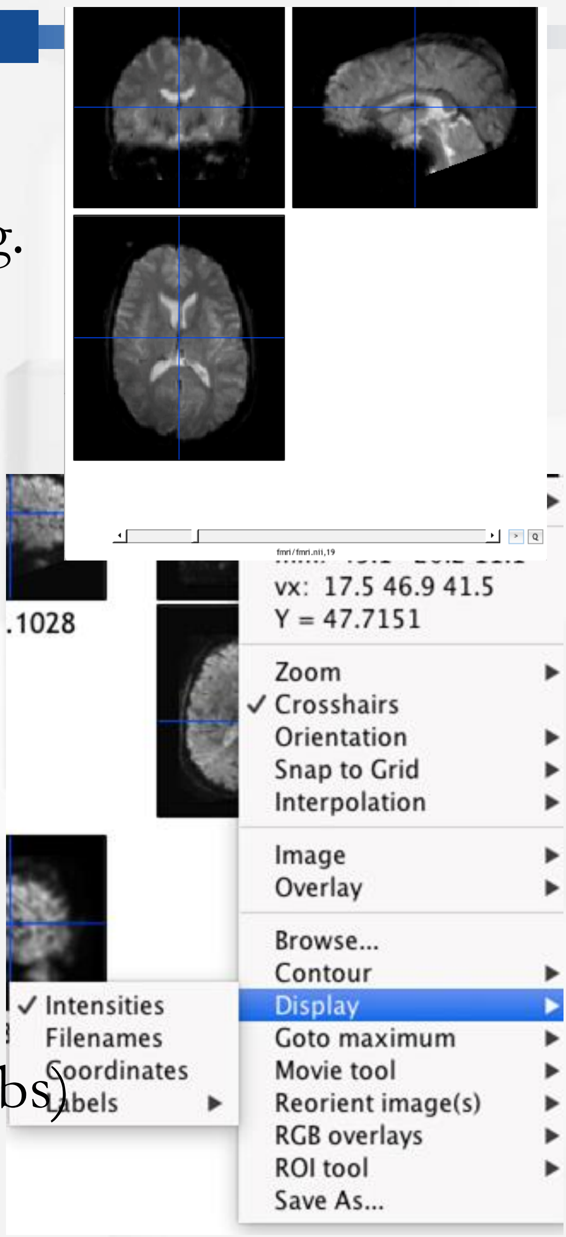
  - 4D NIFTI files:
    - show individual image tiles:

      ```
      spm_check_registration(‘fmri.nii,1’, ‘fmri.nii,2’, …)
      ```

    - show movie (**NEW**! SPM12):

      ```
      spm_check_registration(‘fmri.nii’)
      ```

  - **Right click** reveals amazing features (edges, anatomic labels, header info, contrast, add blobs)

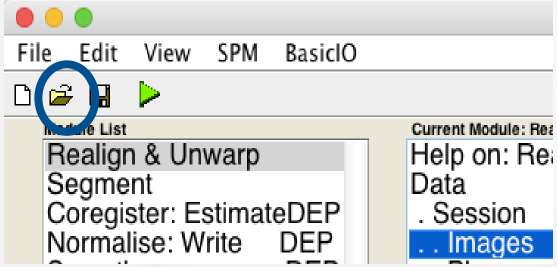


Demo 1

- Default preprocessing pipeline for fMRI
  - Standard recommendation @ TNU
    - Slice Timing Correction
    - Realign
    - Coregister
    - Segment (Structural Image)
    - Skullstrip Brain (via ImCalc)
    - Normalize-Write (Structural Image)
    - Normalize-Write (Functional Image)
    - Smooth
- Inspection: Load into Batch Editor

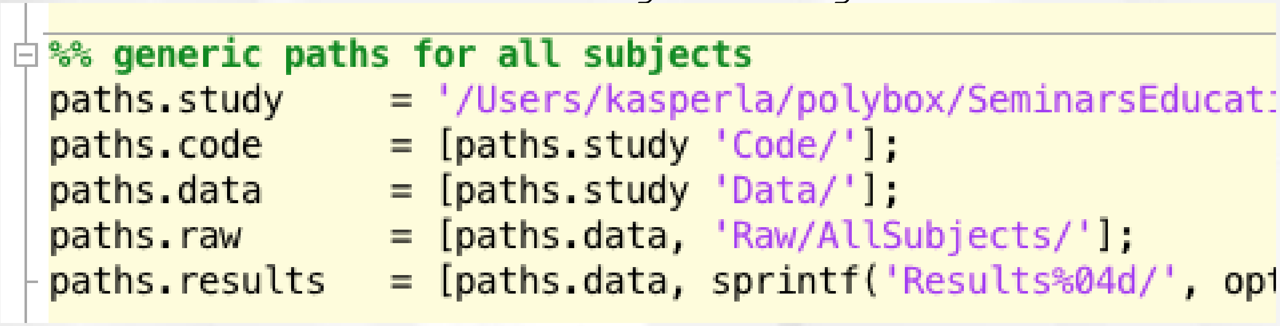
  `tnufmri/Preprocessing/batch_preproc_fmri_slicetiming.m`

    - via GUI or `spm_jobman('initcfg'); spm_jobman('interactive',... 'batch_preproc_fmri_slicetiming.m' );`
  - Run via "Run Batch" (Press Play)
    - will only work after updating paths to you system!
    - Slice Timing: Update Session 1 and 2 (fmri01/02.nii)
    - Segment: Update Channel 1 (struct.nii), and TPM locations



Demo 1

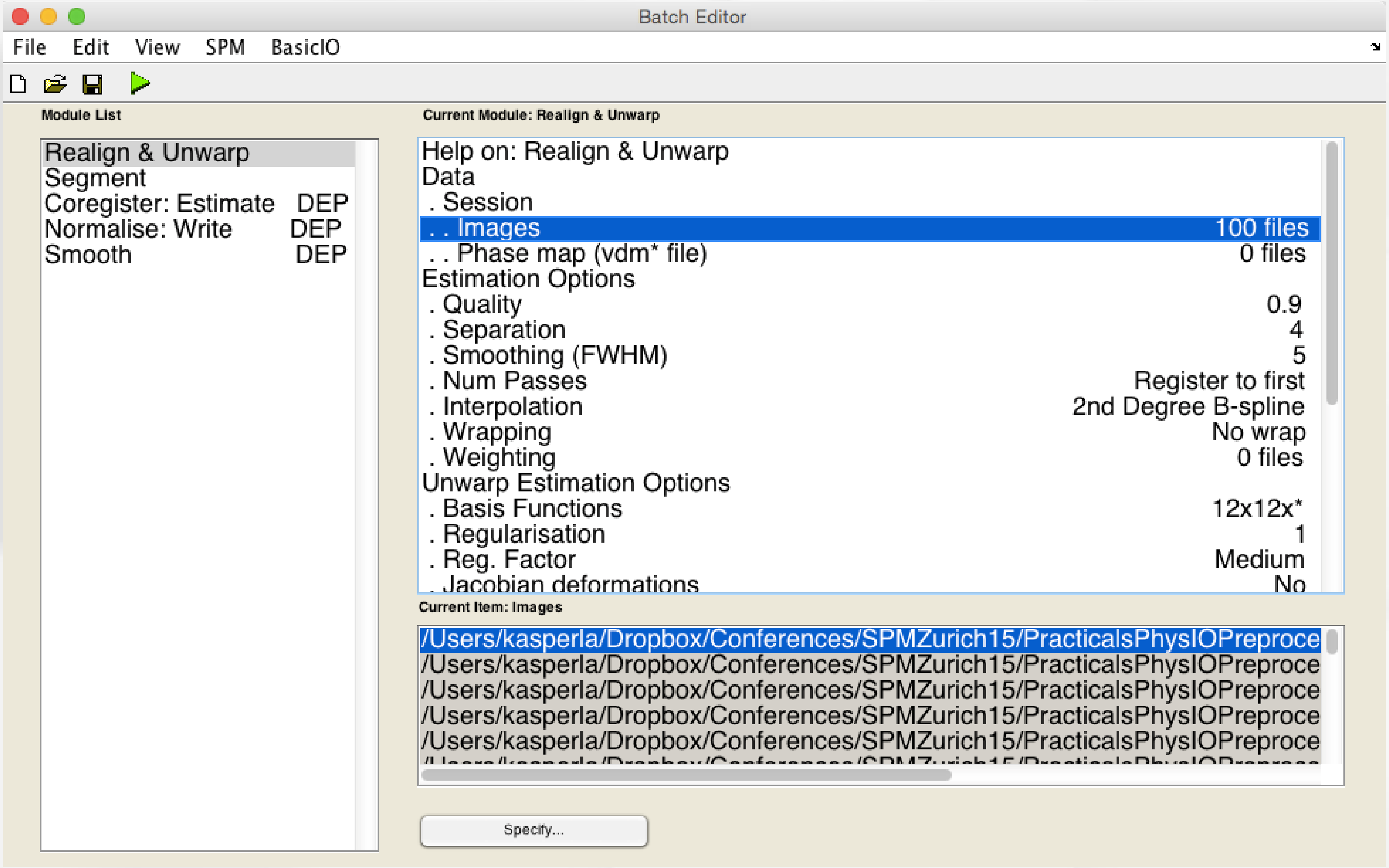
- Alternative: Pipeline Scripting in Matlab
  - update paths in `Code/tnufmri/tnufmri get subject details.m`



```
%% generic paths for all subjects
paths.study      = '/Users/kasperla/polybox/SeminarsEducat
paths.code       = [paths.study 'Code/'];
paths.data       = [paths.study 'Data/'];
paths.raw        = [paths.data, 'Raw/AllSubjects/'];
paths.results    = [paths.data, sprintf('Results%04d/', op
```

  - run in command window: `tnufmri_main_single_subject();`
    - parameters `tnufmri_main_single_subject(subjectId, preprocId, iProcessingSteps);`
    - Default options: subjectId = 1, preprocId = 1, iProcessingSteps = [1 3 4];
      - `tnufmri_main_single_subject(1,1,[1 3 4]);`
  - Examples:
    - Try displaying a video of the raw fmri01 (mean-corrected): iProcessingSteps = [1 2]
    - Try pipeline for another subject (Sub05): subjectId = 5

Demo 1 – GUI Batch Editor

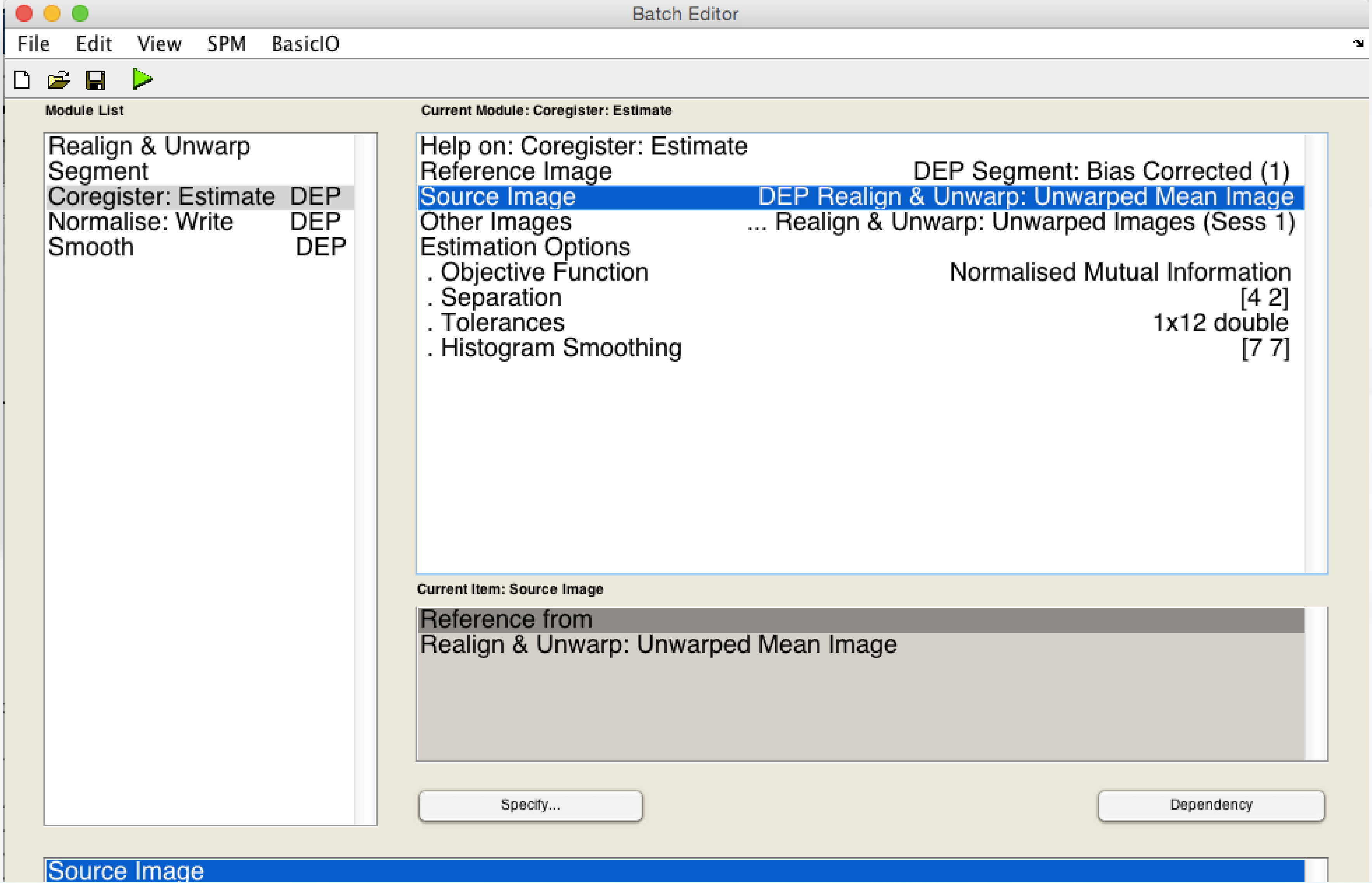


Realign & Unwarp	
Segment	
Coregister: Estimate	DEP
Normalise: Write	DEP
Smooth	DEP

Help on: Realign & Unwarp	
Data	
. Session	
. . Images	100 files
. . Phase map (vdm* file)	0 files
Estimation Options	
. Quality	0.9
. Separation	4
. Smoothing (FWHM)	5
. Num Passes	Register to first
. Interpolation	2nd Degree B-spline
. Wrapping	No wrap
. Weighting	0 files
Unwarp Estimation Options	
. Basis Functions	12x12x*
. Regularisation	1
. Reg. Factor	Medium
. Jacobian deformations	No

```
/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproce
/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproce
/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproce
/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproce
/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproce
```

Demo 1 - Dependencies



Module List

- Realign & Unwarp
- Segment
- Coregister: Estimate DEP
- Normalise: Write DEP
- Smooth DEP

Current Module: Coregister: Estimate

Help on: Coregister: Estimate

Reference Image	DEP Segment: Bias Corrected (1)
Source Image	DEP Realign & Unwarp: Unwarped Mean Image
Other Images	... Realign & Unwarp: Unwarped Images (Sess 1)
Estimation Options	
. Objective Function	Normalised Mutual Information
. Separation	[4 2]
. Tolerances	1x12 double
. Histogram Smoothing	[7 7]

Current Item: Source Image

Reference from
Realign & Unwarp: Unwarped Mean Image

Source Image

Demo 1 - Results

- Automatic status plots saved in `spm_<date>.ps`

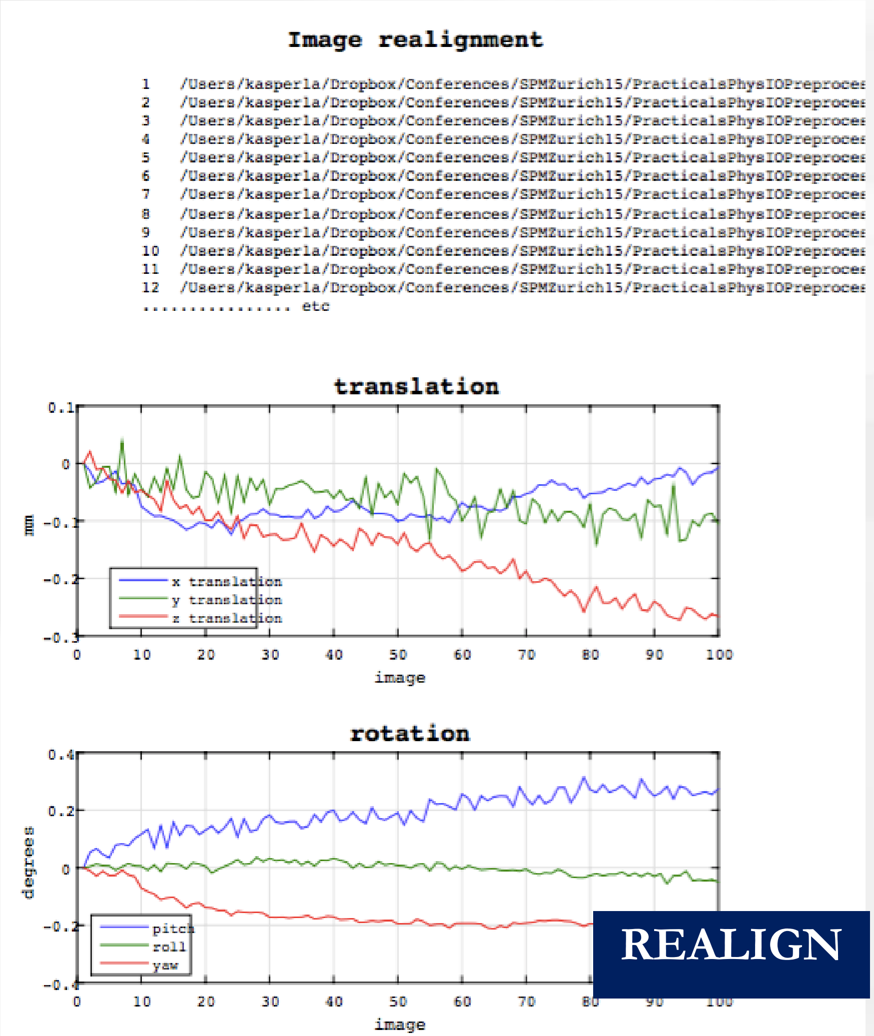
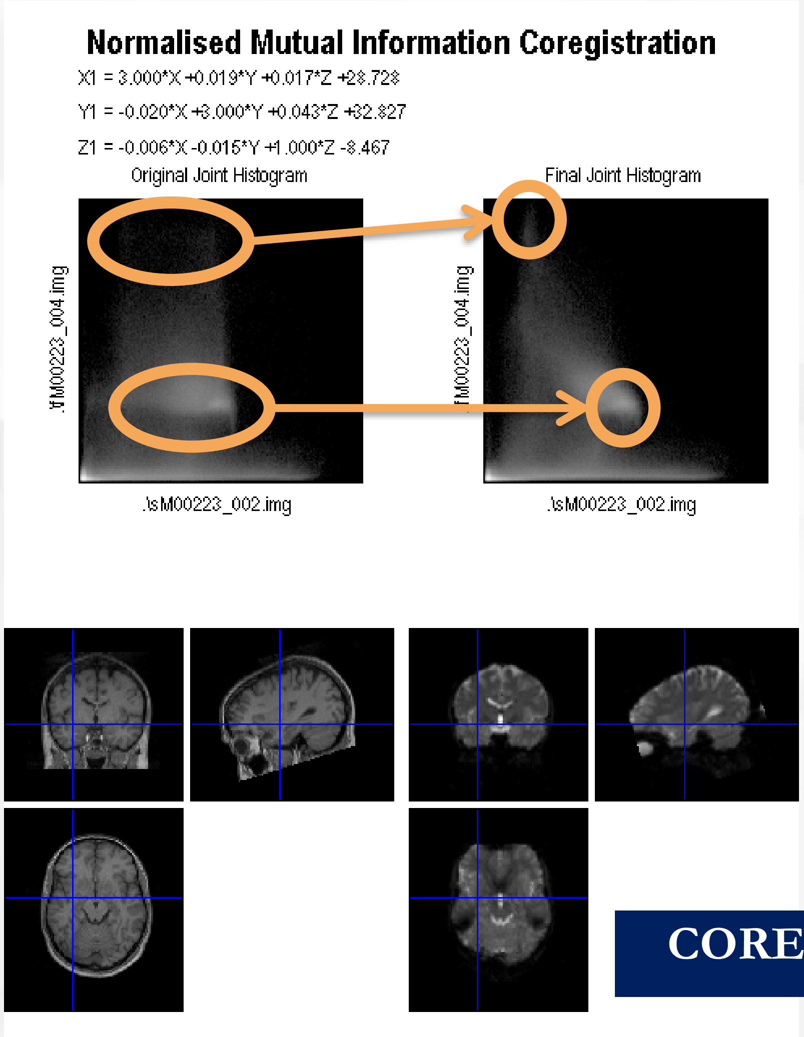


Image realignment

```
1    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
2    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
3    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
4    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
5    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
6    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
7    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
8    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
9    /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
10   /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
11   /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
12   /Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproces
................. etc
```

REALIGN



Normalised Mutual Information Coregistration

X1 = 3.000*X +0.019*Y +0.017*Z +28.728

Y1 = -0.020*X +3.000*Y +0.043*Z +32.827

Z1 = -0.006*X -0.015*Y +1.000*Z -8.467

COREG

Outline

- Setting up a Preprocessing Pipeline in SPM: The Batch Editor
- **Monitoring and Comparing Pipelines**
    - Performance Measures: Mean/SD/SNR/Diff Image
    - SPM Plotting Routines and Automatic Reporting
    - Spotting Failed Pipelines
    - Comparing Alternative Pipelines
- Multi-subject Pipelines
- Integrating Own Code and Toolboxes: Physiological Noise Modeling and Evaluation

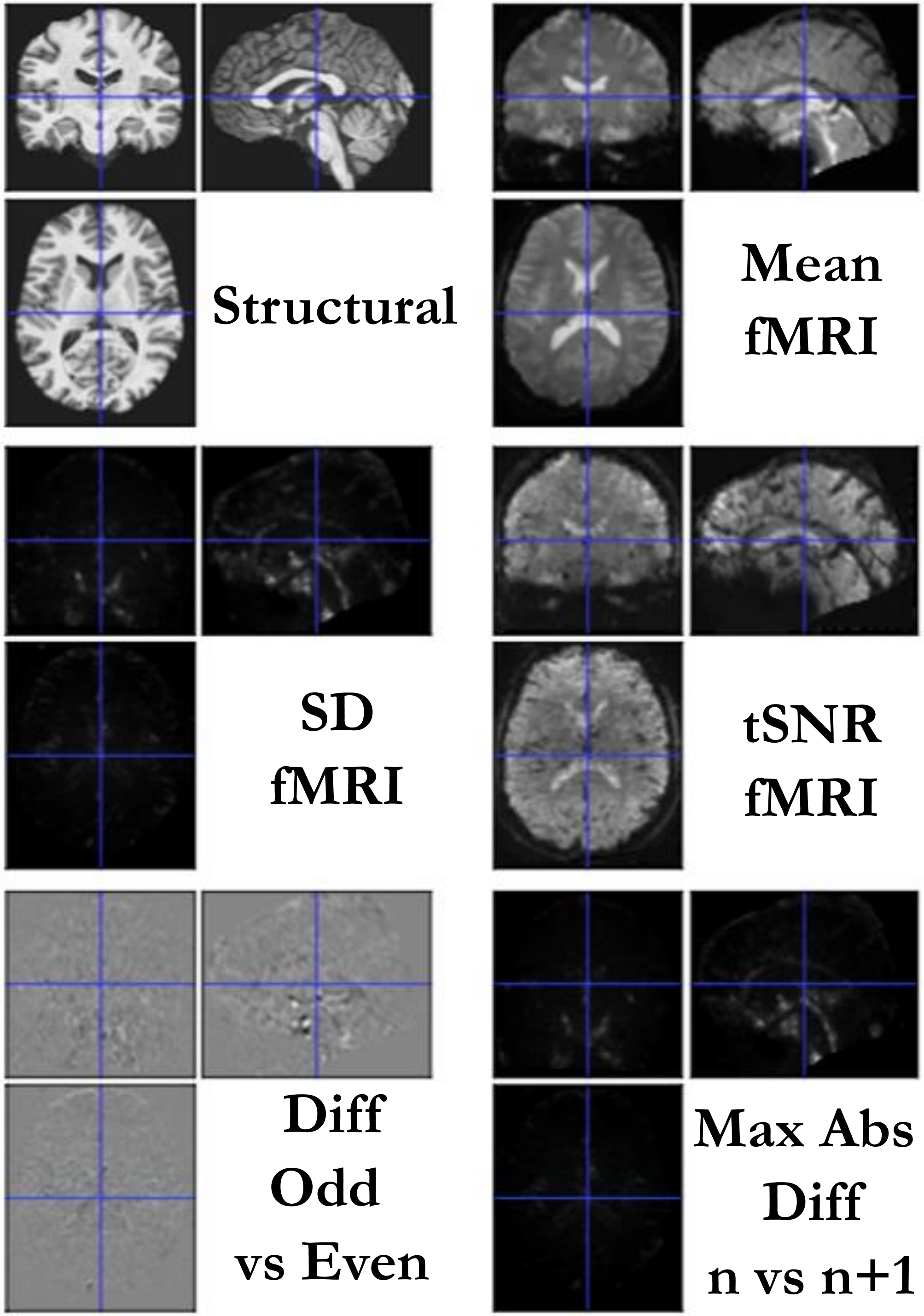
Pipeline Monitoring

- Pipelines → Automatisation of Preprocessing
- When something goes wrong…how do we even notice?
  - Monitoring, but: cumbersome, when lots of data
  - Thus: Automatise quality monitoring as well via pipelines
- Required: Suitable performance measures
  - Single image: visual inspection geometry/contrast/noise/SNR
    - structural image
  - Time series: Statistical Images (Mean, SD, tSNR, max(abs(diff)))
    - functional images

Welvaert (2013), PLoS One *Friedman/Glover (2006), JMRI*

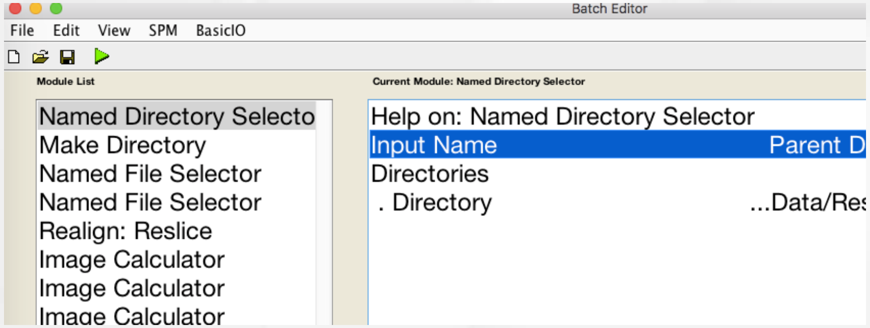
Statistical Images

- Structural Image:
  visual inspection geometry/
  contrast/noise/SNR
- Functional Time Series:
  - Mean => Artifact levels (localization)
  - SD => Fluctuation levels
  - tSNR = Mean/SD
    => sensitivity for BOLD signal changes
  - Diff = max(abs(diff)) or odd – even
    => outlier detection, image noise



Demo 2

- Quality monitoring batch
  - Computes all statistical images suggested, displays and prints to ps
  - Includes `batch_report_quality.m` to visualize and save quality measures after each preprocessing step
- Run via
  1. Load Batch `tnufmri/Quality/batch_report_quality.m`
  2. Enter Directory to save statistical images and PS-report
  3. Select Functional / Structural Images (Named File Selectors)
     - check final before smoothing
       - functional: wuafmri01.nii
       - structural: wBrain.nii
     - in principle to do after each step…



Module List

Named Directory Selecto
Make Directory
Named File Selector
Named File Selector
Realign: Reslice
Image Calculator
Image Calculator
Image Calculator

Current Module: Named Directory Selector

Help on: Named Directory Selector
Input Name Parent D
Directories
. Directory ...Data/Res

Demo 2b

- Combine Demo 1 and 2 to create quality reports after each preprocessing step
    - multiple runs of batch_report_quality
    - with dependencies for input functional/structural
    - statistical images saved in different folders (01_slicetiming, …)
    - ps-figure prints concatenated

Demo 2b - Output

- Output:
  - `subject01/report_quality/report_quality.ps`
    - PostScript file with all output plots, generated from the following nifti image files
  - `report_quality/00_raw/` => raw time series statistics

    `mean.nii` => mean of time series (per pixel)

    `sd.nii` => standard deviation (per pixel)

    `snr.nii` => mean/sd (per pixel)

    `diffOddEven.nii` => SumOddImages – SumEvenImages

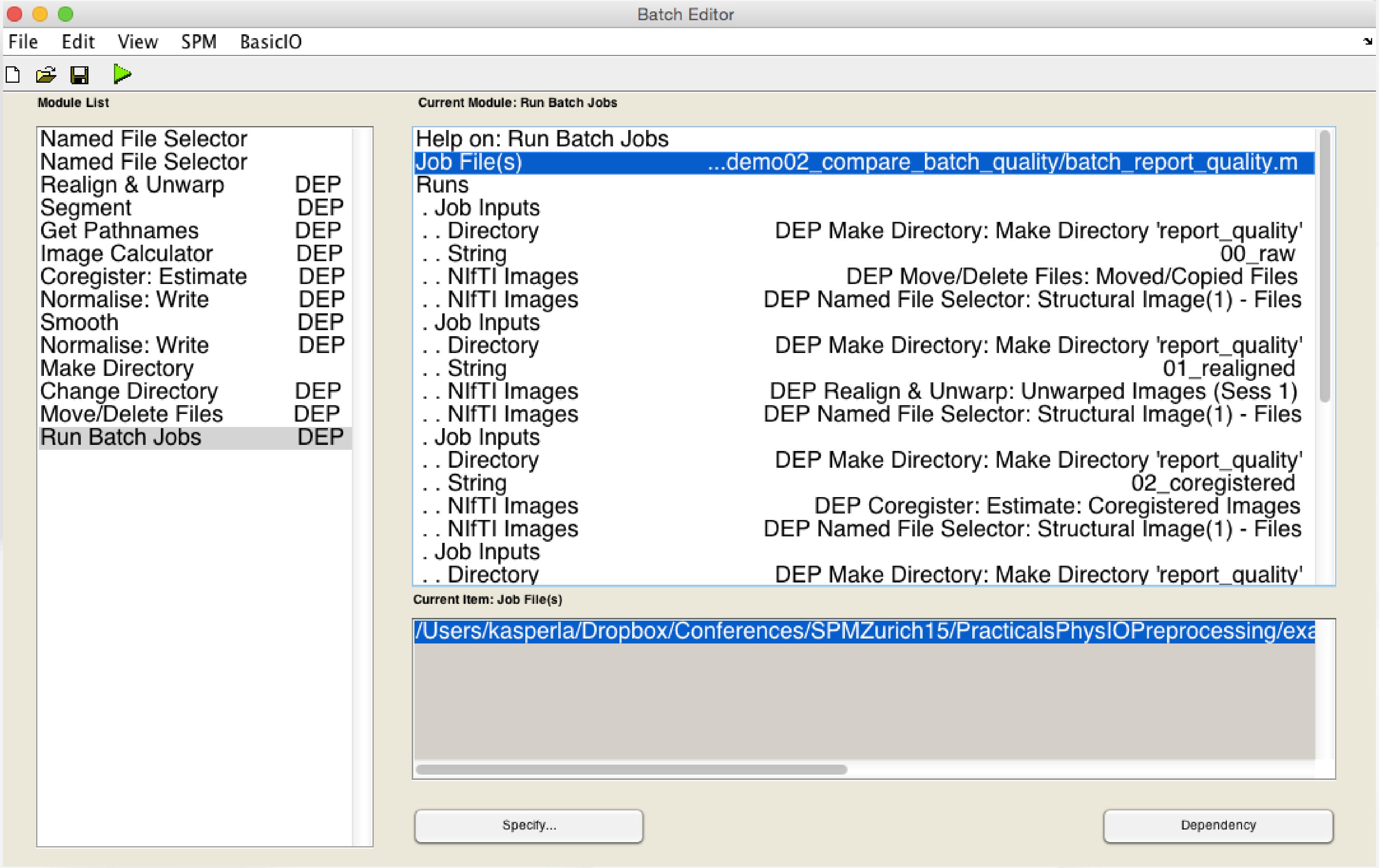
    `maxAbsDiff` => maximum delta image (vol n vs n+1)

    `subject01/01_realigned` => realigned time series stats

    …

    `subject01/04_smoothed` => smoothed time series stats

Demo 2 - GUI



Module List

Named File Selector	
Named File Selector	
Realign & Unwarp	DEP
Segment	DEP
Get Pathnames	DEP
Image Calculator	DEP
Coregister: Estimate	DEP
Normalise: Write	DEP
Smooth	DEP
Normalise: Write	DEP
Make Directory	
Change Directory	DEP
Move/Delete Files	DEP
Run Batch Jobs	DEP

Current Module: Run Batch Jobs

Help on: Run Batch Jobs	
Job File(s)	...demo02_compare_batch_quality/batch_report_quality.m
Runs	
. Job Inputs	
. . Directory	DEP Make Directory: Make Directory 'report_quality'
. . String	00_raw
. . NIfTI Images	DEP Move/Delete Files: Moved/Copied Files
. . NIfTI Images	DEP Named File Selector: Structural Image(1) - Files
. Job Inputs	
. . Directory	DEP Make Directory: Make Directory 'report_quality'
. . String	01_realigned
. . NIfTI Images	DEP Realign & Unwarp: Unwarped Images (Sess 1)
. . NIfTI Images	DEP Named File Selector: Structural Image(1) - Files
. Job Inputs	
. . Directory	DEP Make Directory: Make Directory 'report_quality'
. . String	02_coregistered
. . NIfTI Images	DEP Coregister: Estimate: Coregistered Images
. . NIfTI Images	DEP Named File Selector: Structural Image(1) - Files
. Job Inputs	
. . Directory	DEP Make Directory: Make Directory 'report_quality'

Current Item: Job File(s)

/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreprocessing/exa

Plotting Example

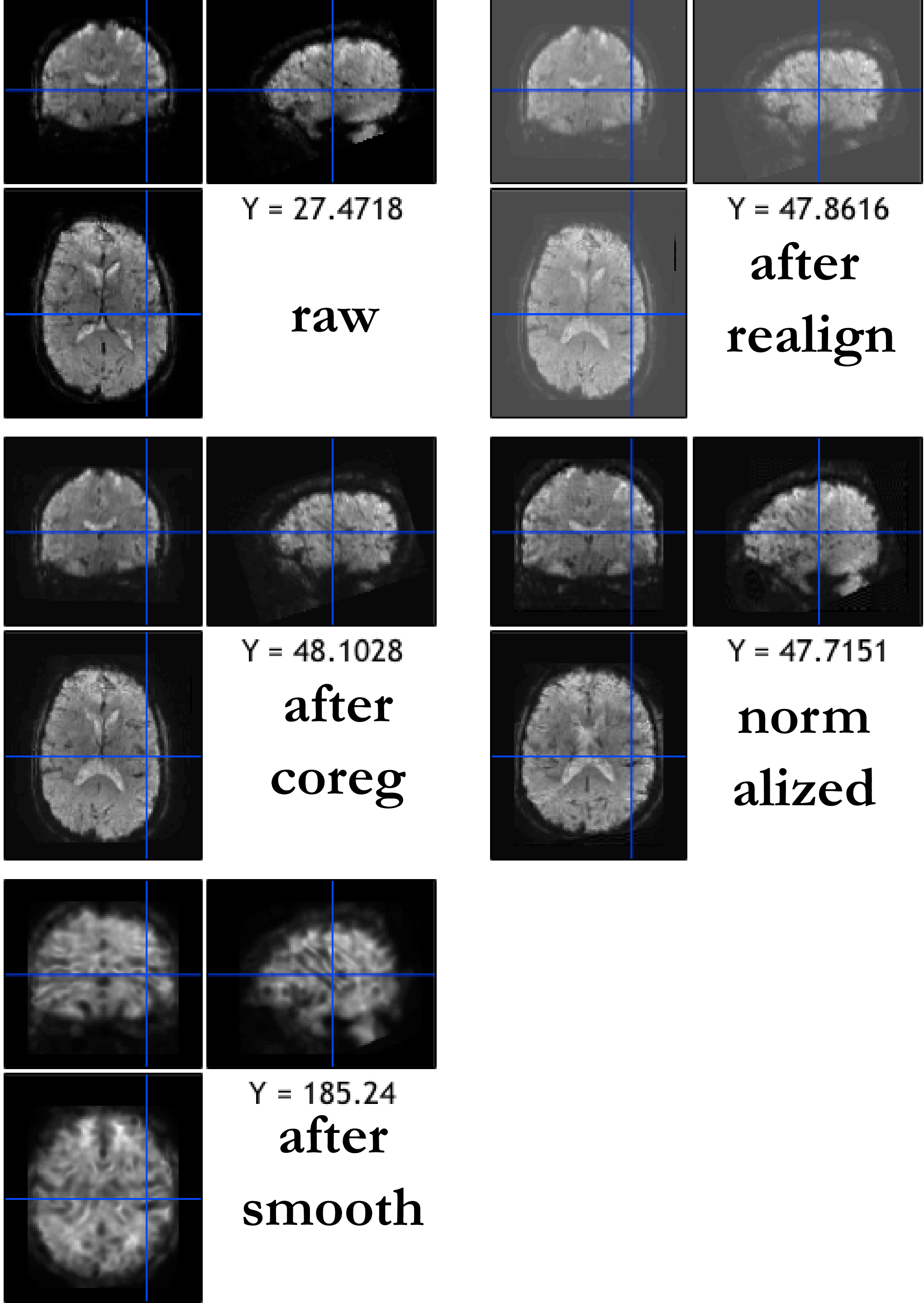
- Example: Comparing temporal SNR throughout preprocessing
  - Temporal SNR per pixel in functional image time series after each preprocessing step
  - Command line code:

```
dirs = {'00_raw', '01_realigned',
'02_coregistered', '03_normalized',
'04_smoothed'}';
files = 'snr.nii’;
fpFiles = strcat(dirs, '/', files);
spm_check_registration(fpFiles{:})
```

Plotting Example

Setup Pipeline | **Quality Monitoring**

- Example: Temporal SNR per pixel in functional image time series after each preprocessing step
- Right click reveals amazing features
- Result: Increased SNR through realignment and smoothing

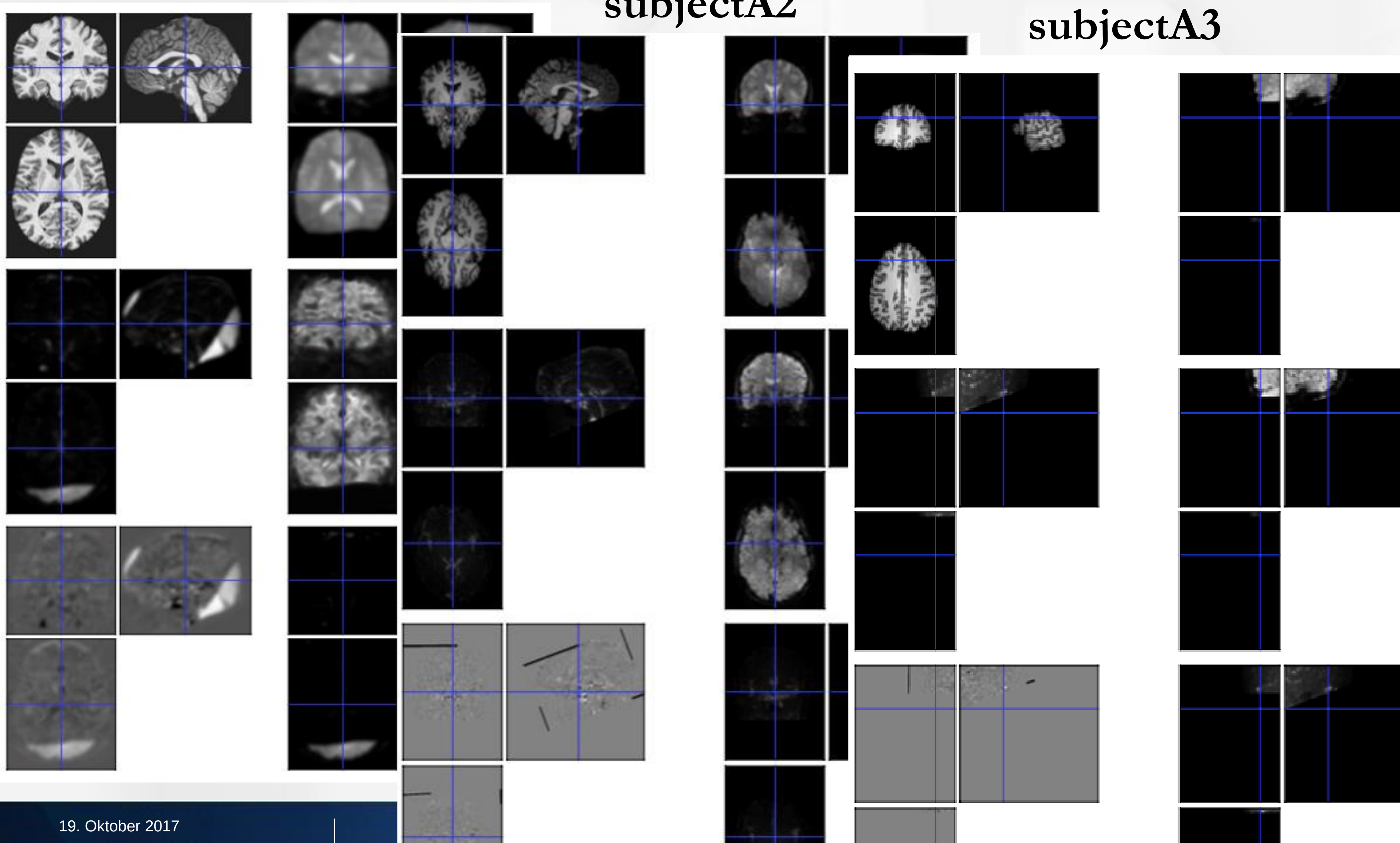


Quiz…What went wrong here?

subjectA1

subjectA2

subjectA3



Quiz…

- Task: Spotting unusual quality report images
- What went wrong here?
  - subjectA1
    - realignment failed
    - one volume rotated (30 degrees around x-axis)
  - subjectA2
    - segmentation failed
    - structural mirrored compared to template
  - subjectA3
    - co-registration failed
    - functional image shifted by 10 cm (x, y, and z) compared to structural

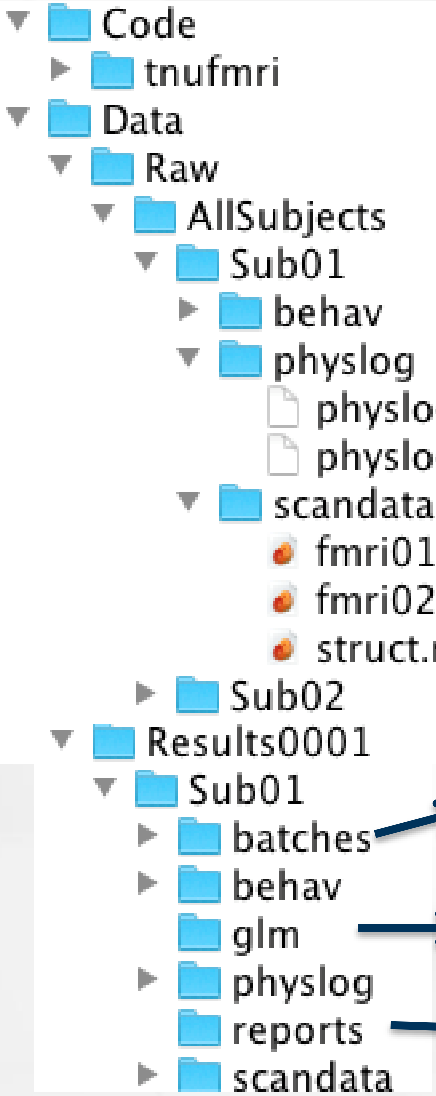
Outline

Setup Pipeline | Quality Monitoring | Multi-Subject | Noise Modeling

- Setting up a Preprocessing Pipeline in SPM: The Batch Editor
- Monitoring and Comparing Pipelines
- **Multi-subject Pipelines**
  - Staying on Top: Organisation of Data
  - Looping Pipelines over Subjects' Data
  - Tips for Efficient Performance Monitoring
- Integrating Own Code and Toolboxes: Physiological Noise Modeling and Evaluation

Data Organisation

- One Study Folder
  - sub-folder for code, e.g. templates for subject-specific batches
    - modules and dependencies stay the same
  - raw subject data: Data/Raw
  - processed subject data: Data/Results<preprocId>
- In Raw/Results:
  - One folder per subject
  - therein: same sub-folder structure
  - reduces subject data specification to changing the subject folder



```
▼ Code
  ▶ tnufmri
▼ Data
  ▼ Raw
    ▼ AllSubjects
      ▼ Sub01
        ▶ behav
        ▼ physlog
            physlog_run01.log
            physlog_run02.log
        ▼ scandata
            fmri01.nii
            fmri02.nii
            struct.nii
      ▶ Sub02
  ▼ Results0001
    ▼ Sub01
      ▶ batches
      ▶ behav
        glm
      ▶ physlog
        reports
      ▶ scandata
```

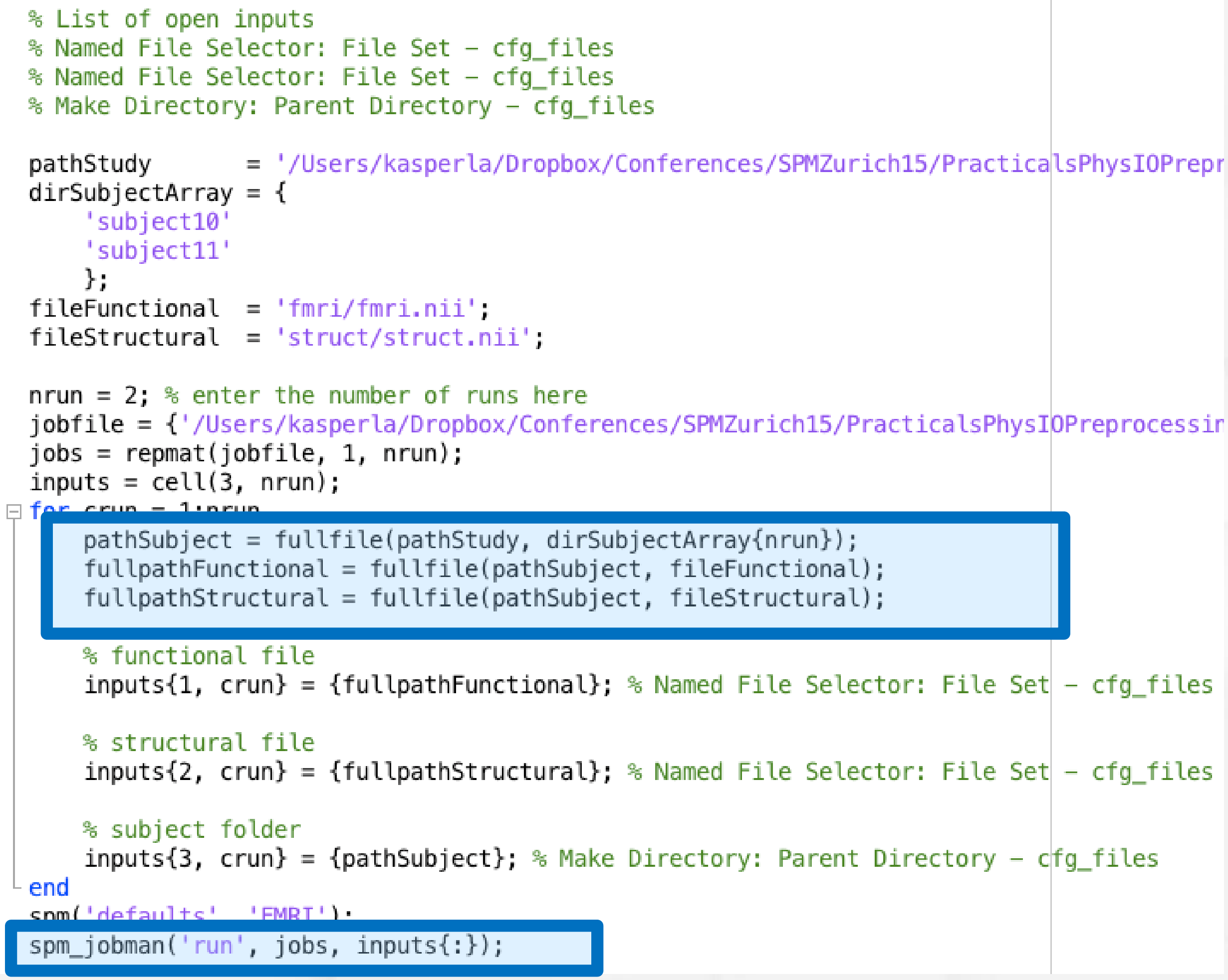
batches → executed batches for subject

glm → 1st level analysis

reports → quality measures, ps-files…

Multi-Subject = Looping

- All we have to change in batch: input data to subject specifics
- Then: Loop over subjects
- Easy, if files are named analogously



```
% List of open inputs
% Named File Selector: File Set - cfg_files
% Named File Selector: File Set - cfg_files
% Make Directory: Parent Directory - cfg_files

pathStudy       = '/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPrepr
dirSubjectArray = {
    'subject10'
    'subject11'
    };
fileFunctional  = 'fmri/fmri.nii';
fileStructural  = 'struct/struct.nii';

nrun = 2; % enter the number of runs here
jobfile = {'/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreprocessin
jobs = repmat(jobfile, 1, nrun);
inputs = cell(3, nrun);
for crun = 1:nrun
    pathSubject = fullfile(pathStudy, dirSubjectArray{nrun});
    fullpathFunctional = fullfile(pathSubject, fileFunctional);
    fullpathStructural = fullfile(pathSubject, fileStructural);

    % functional file
    inputs{1, crun} = {fullpathFunctional}; % Named File Selector: File Set - cfg_files

    % structural file
    inputs{2, crun} = {fullpathStructural}; % Named File Selector: File Set - cfg_files

    % subject folder
    inputs{3, crun} = {pathSubject}; % Make Directory: Parent Directory - cfg_files
end
spm('defaults', 'FMRI');
spm_jobman('run', jobs, inputs{:});
```

Demo 3 – Example Loop Script

Setup Pipeline | Quality Monitoring | Multi-Subject | Noise Modeling

- Using pipelining scripts, it becomes even easier: `tnufmri_main_multi_subjects(subjectIds, preprocId, iProcessingSteps)`
- Basically a loop over `tnufmri_main_single_subject`

Outline

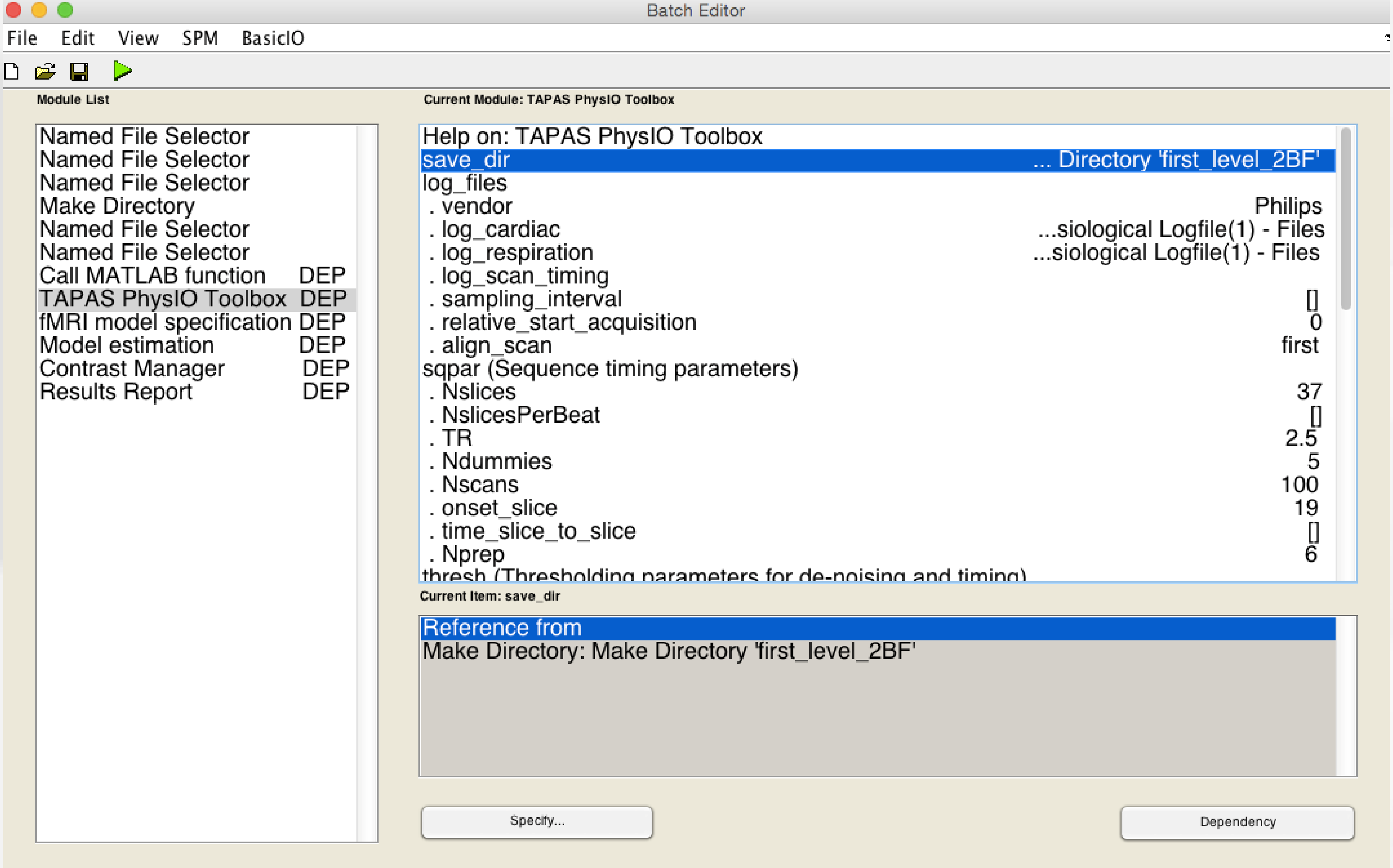
- Setting up a Preprocessing Pipeline in SPM: The Batch Editor
- Monitoring and Comparing Pipelines
- Multi-subject Pipelines
- **Integrating Own Code and Toolboxes: Physiological Noise Modeling and Evaluation**
  - Executing Custom Matlab Code within the Pipeline
  - The TAPAS PhysIO Toolbox
  - Automatic (Noise) Modeling and Contrast Reporting

Demo 4

- Performs whole single-subject analysis except preprocessing
  - creates behavioral regressors (visual stimuli/button responses)
    - `multiple_conditions.mat`
  - creates nuisance regressors, including RETROICOR via the TAPAS PhysIO Toolbox
    - `multiple_regressors.mat`
  - Sets up 1st level analysis GLM & estimates it
  - Estimates F-contrasts of interests and plots them to .ps file
- Run via
  1. `spm_jobman(‘interactive’, subject01/batches/demo04_stats_physio/batch_physio_glm_contrasts.m)`

Demo 4

Setup Pipeline | Quality Monitoring | Multi-Subject | Noise Modeling



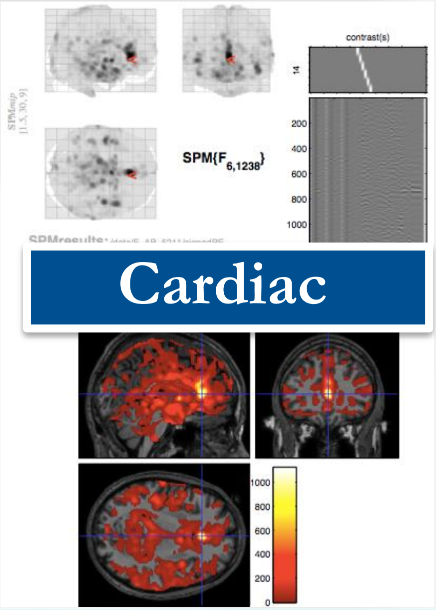
```
Named File Selector
Named File Selector
Named File Selector
Make Directory
Named File Selector
Named File Selector
Call MATLAB function       DEP
TAPAS PhysIO Toolbox       DEP
fMRI model specification   DEP
Model estimation           DEP
Contrast Manager           DEP
Results Report             DEP
```

```
Help on: TAPAS PhysIO Toolbox
save_dir                                  ... Directory 'first_level_2BF'
log_files
 . vendor                                                        Philips
 . log_cardiac                           ...siological Logfile(1) - Files
 . log_respiration                       ...siological Logfile(1) - Files
 . log_scan_timing
 . sampling_interval                                                   []
 . relative_start_acquisition                                           0
 . align_scan                                                       first
sqpar (Sequence timing parameters)
 . Nslices                                                             37
 . NslicesPerBeat                                                      []
 . TR                                                                 2.5
 . Ndummies                                                             5
 . Nscans                                                             100
 . onset_slice                                                         19
 . time_slice_to_slice                                                 []
 . Nprep                                                                6
thresh (Thresholding parameters for de-noising and timing)
```

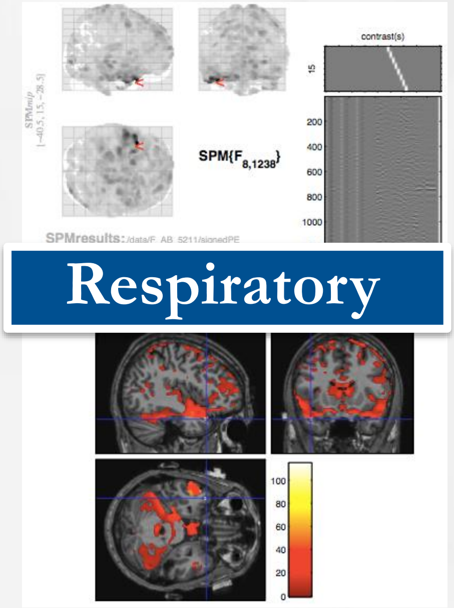
```
Reference from
Make Directory: Make Directory 'first_level_2BF'
```

Pipeline Noise Modeling/Report

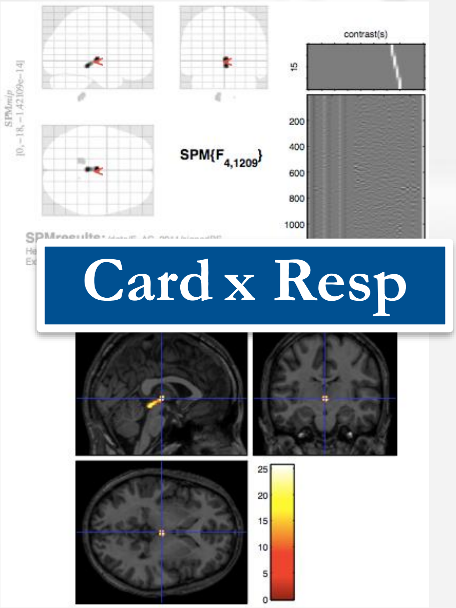
- Model physiological noise as nuisance regressors
- Check noise correction via explained variance (F-contrasts)
  - Automatic contrast creation and report (glass brain): Batch Editor SPM => Stats => Contrast Manager/Results Report
  - With underlays: `tapas_physio_report_contrasts`



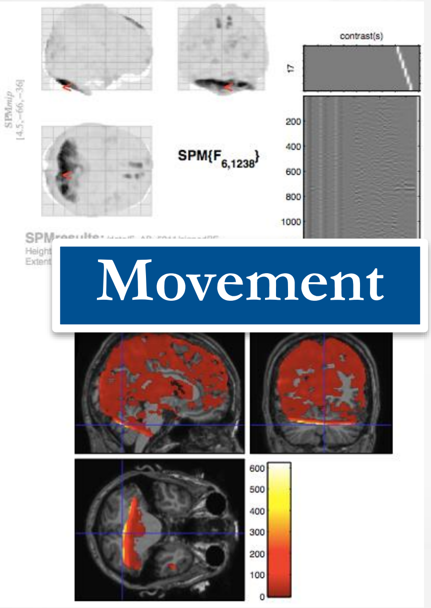
Cardiac



Respiratory



Card x Resp



Movement

Conclusion

- Setting up a Preprocessing Pipeline in SPM:
  The Batch Editor

  …*best way for reproducible, documented preprocessing*
- Monitoring and Comparing Pipelines

  …*within Batch Editor, CheckReg Tool, Stat Images per step*
- Multi-subject Pipelines

  …*via template batch, same sub-folders & Matlab script (subj.-loop)*
- Integrating Own Code & Physiological Noise Correction

  …*by calling Matlab functions within Batch Editor, e.g. PhysIO Toolbox*