

The Art and Pitfalls of fMRI Preprocessing

An SPM Perspective (Extended Version for Tutorial)

Lars Kasper

June 14th, 2015

MR-Technology Group & Translational Neuromodeling Unit

Institute for Biomedical Engineering
University of Zurich and ETH Zurich





fMRI preprocessing has catastrophic consequences on study outcomes - if it fails. However, such failures are hard to detect, due to the complexity of preprocessing pipelines and the sheer amount of image data to be reviewed. Thus, an automatic way of reproducing, documenting and assessing complex workflows is key to a successful preprocessing strategy.

This talk demonstrates how SPM caters for this need by one of its main user interface features: the Batch Editor. First, we see how its modular structure and input/output definitions allow for setting up the whole preprocessing pipeline at once in a reproducible fashion that can be easily reviewed, and applied to different subjects. Modules such as realignment, slice-timing correction, co-registration, and normalization via unified

segmentation are covered. We introduce one specific preprocessing pipeline with its rationale, and a clear decision tree for when to deviate from it. Secondly, the Batch Editor greatly facilitates quality assurance by recruiting the versatile (but somewhat hidden) visualization capabilities of SPM, to create graphical reports of each preprocessing step. Finally, we highlight how to integrate non-standard preprocessing modules into SPM via the Batch Editor, by the example of physiological noise correction with a custom-built toolbox (TAPAS PhysIO).

The format of this talk will incorporate the graphical user interface wherever possible, including walk-throughs and screen-shots.

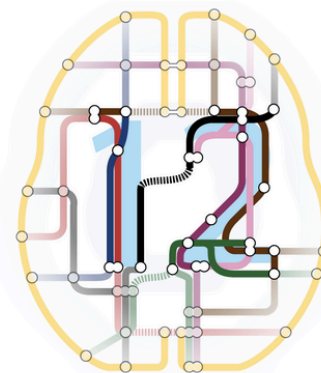
Statistical Parametric Mapping



- What is SPM?
 - SPM = Statistical Parametric Mapping
 - Developed at the Functional Imaging Laboratory (FIL) at UCL, London
 - Much more than fMRI (E/MEG/PET)
- Philosophy
 - Free Software (GPL), Matlab-based
 - script language, also for non-programmers
 - Simple debugging, interactive operation via command window, powerful editor
 - Easy access/modification of source code
 - Combination with self-written custom code

Welcome to SPM12

Please refer to this version as "[SPM12](#)" in papers and communications.



The SPM12 [Manual](#) and [Release Notes](#) are available as PDF documents in the *man* directory of your SPM installation.

[Updates](#) will be made available from time to time and advertised on the [SPM mailing list](#). You can also check for updates by clicking [here](#).

We would love to hear your comments or bug reports - please contact us at [<fil.spm@ucl.ac.uk>](mailto:fil.spm@ucl.ac.uk).

SPM is developed under the auspices of the Functional Imaging Laboratory (FIL), the Wellcome Trust Centre for Neuroimaging (WTCN), in the Institute of Neurology at University College London

Command Window

```

  /_/_/_/_/_/_/_/
  \_/_/_/_/_/_/_/  Statistical Parametric Mapping
  (_/(_/(_/_/_/_)  SPM12 - http://www.fil.ion.ucl.ac.uk

SPM12: v6225
=====
Initialising SPM
SPM present working directory:
```



- Download example code and data presented in this talk:
 - <http://www.tnu-zurich.org/team/lars-kasper/>
 - Section: Talk and Lecture Materials
- SPM12 (Statistical Parametric Mapping)
 - <http://www.fil.ion.ucl.ac.uk/spm/software/spm12/>
 - developed by the Functional Imaging Lab, UCL, London
- TAPAS PhysIO Toolbox (SPM or Matlab standalone)
 - <http://www.translationalneuromodeling.org/tapas/>
 - Documentation & Example Data (Philips/Siemens/GE):
 - <http://www.translationalneuromodeling.org/software/documentation/>
 - <http://www.translationalneuromodeling.org/software/tapas-data/>



- 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

- **Setting up a Preprocessing Pipeline in SPM:**

The Batch Editor

- SPM Overview & Vocabulary
- Preprocessing Theory (SPM-specific)
- Interface/Philosophy: Modules/Dependencies/Data
- Monitoring and Comparing Pipelines
- Multi-subject Pipelines
- Integrating Own Code and Toolboxes:
Physiological Noise Modeling and Evaluation

Overview of SPM for fMRI



Setup Pipeline

Quality Monitoring

Multi-Subject

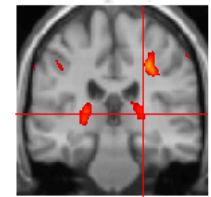
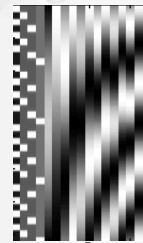
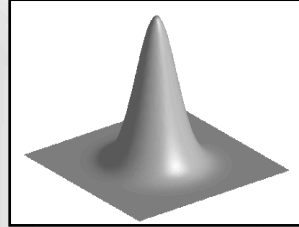
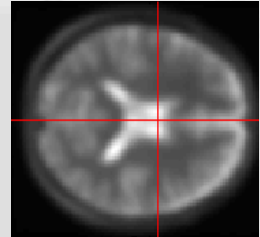
Noise Modeling

Image time-series

Kernel

Design matrix

Statistical parametric map (SPM)



Realignment

Smoothing

General linear model

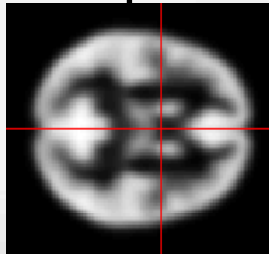
Temporal Filtering
Physiological Noise Correction

Spatial Preprocessing

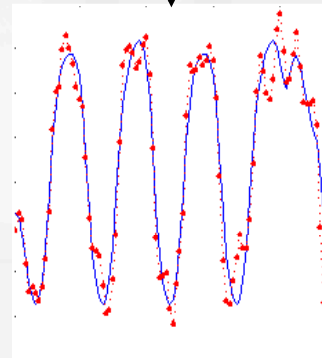
Normalisation

Statistical inference

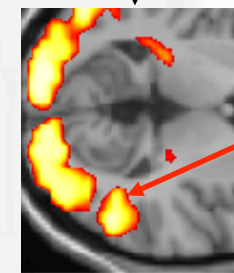
Random field theory



Template



Parameter estimates



$p < 0.05$

The SPM GUI

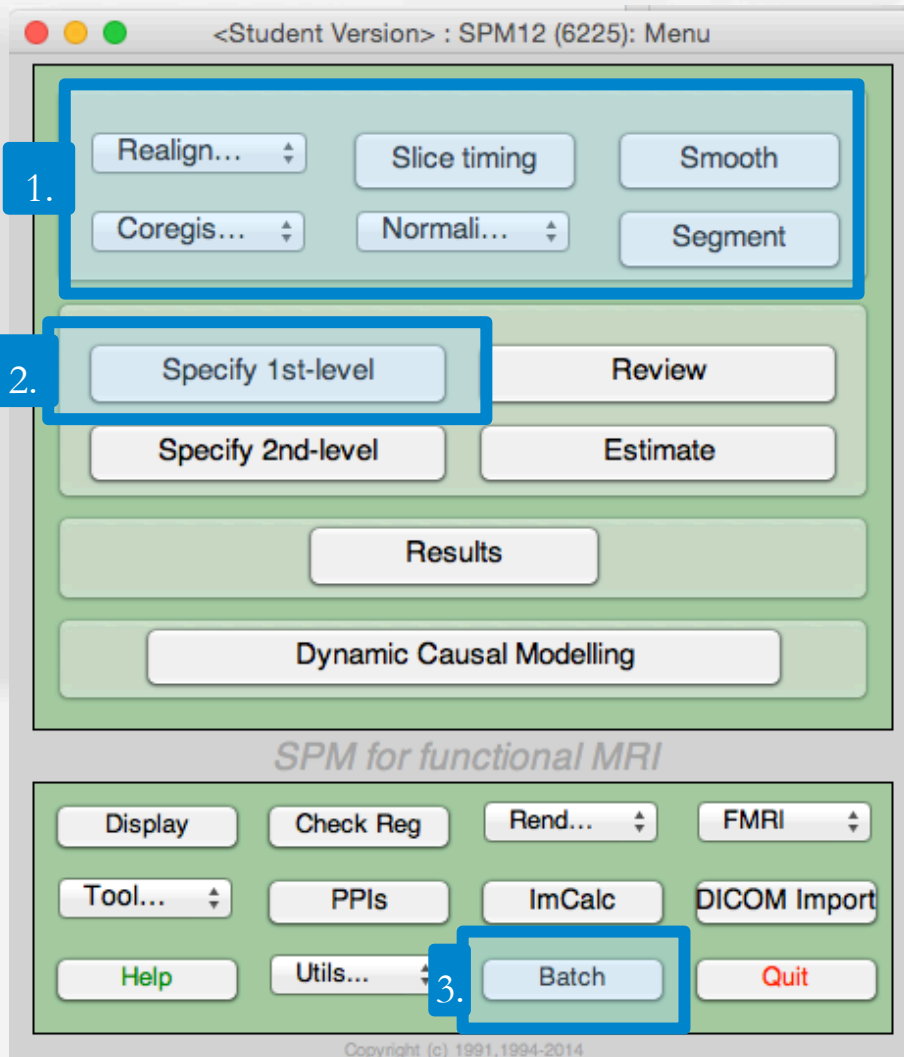


Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling



■ Preprocessing

- Realignment
- Slice-Timing Correction
- Co-Registration
- Unified Segmentation & Normalisation
- Smoothing...

■ Noise Modeling

- Physiological Confound Regressors

■ The Batch Editor



- SPM uses different names for different modes of image registration
- depending on input images and allowed transformations

Realignment

- Intra-modal image registration
 - e.g. functional images
- rigid body transformations
 - translation/rotation

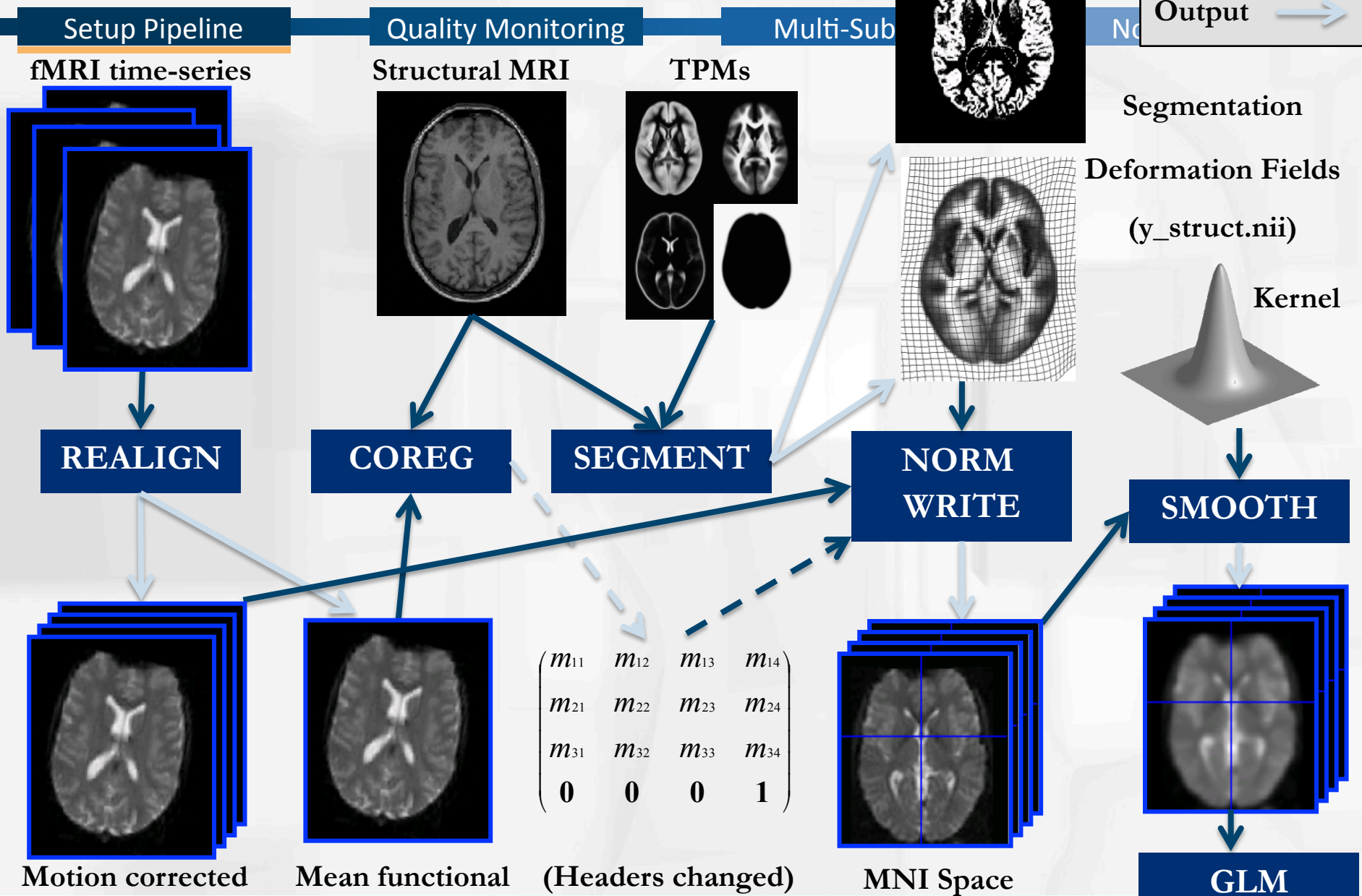
Co-Registration

- Inter-modal registration
 - e.g. T1/T2 contrast
 - functional to structural image
- affine transformations
 - rigid body
 - stretching/shearing

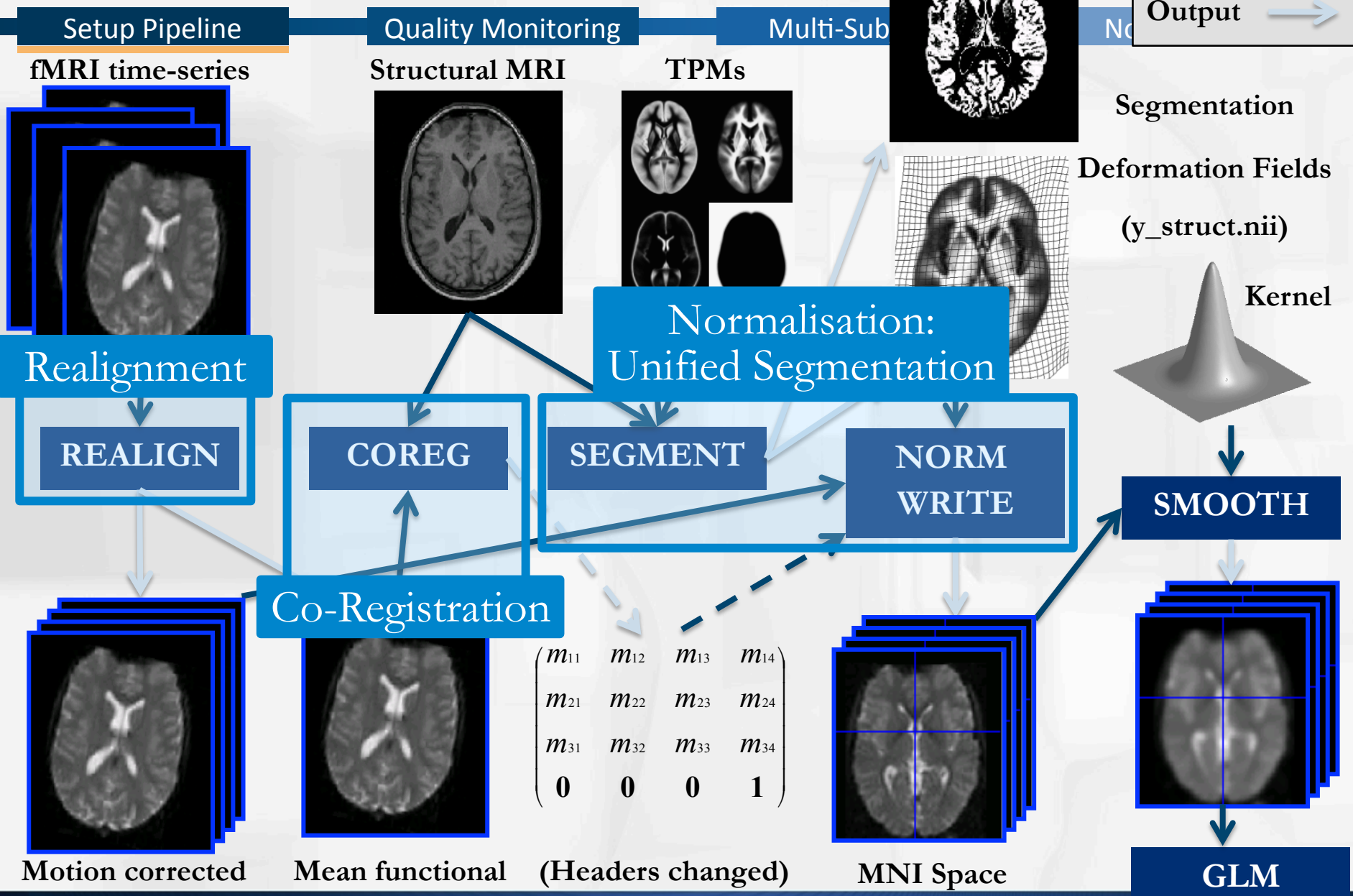
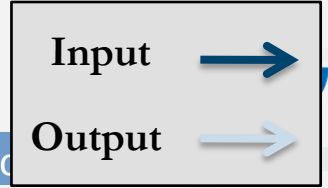
Normalisation

- Multi-modal registration
 - e.g. T1 and/or T2
 - structural image(s) to template
- non-linear transformations
 - voxel-wise mapping (deformation fields)

SPM Preprocessing Pipeline



SPM Preprocessing Pipeline



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

Theory: Co-Registration



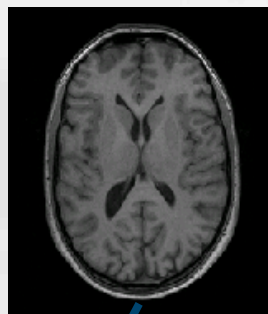
Setup Pipeline

Quality Monitoring

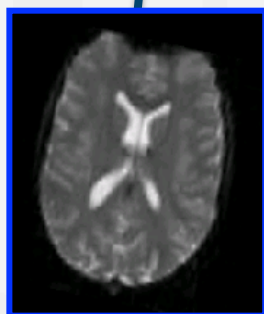
Multi-Subject

Noise Modeling

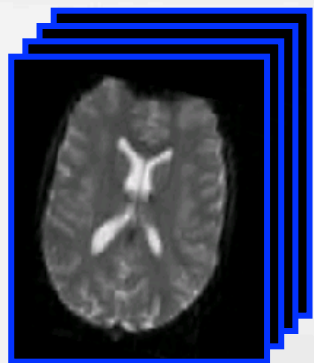
Structural MRI



COREG



Mean functional (Headers changed)



Motion corrected

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

- Aligns mean functional image to structural image
- Affine transformation: translations, rotations, scaling, shearing
- Objective function: mutual information, since contrast different
 - Optimisation via Powell's method: conjugate directions, line search along parameters

Typically only transformation matrix changed (no reslicing)

Theory: Coreg Joint Histogram



Setup Pipeline

Quality Monitoring

Multi-Subject

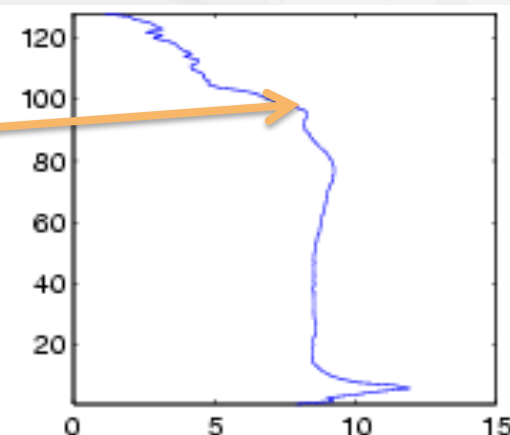
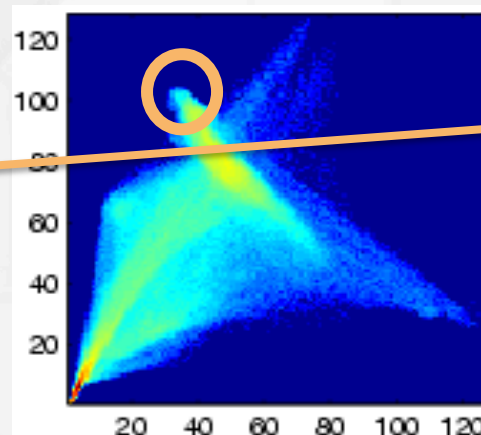
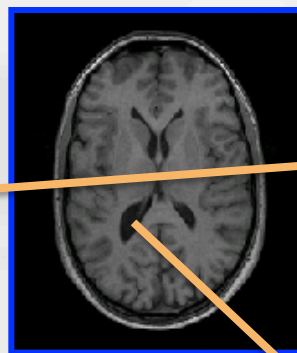
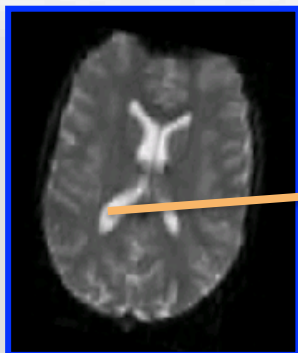
Noise Modeling

Mean functional

Anatomical MRI

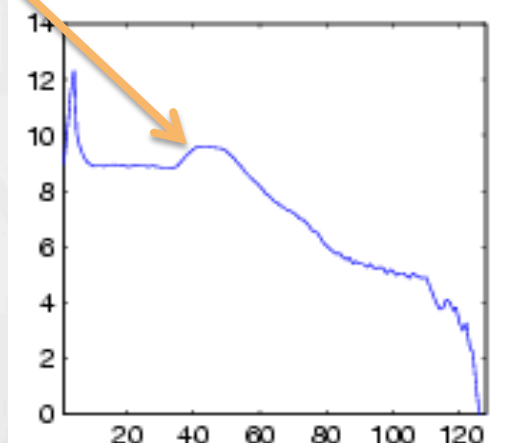
Joint Histogram

Marginal Histogram



■ Joint and marginal Histogram

- Quantify how well one image predicts the other
 - how much shared information
- Joint probability distribution estimated from joint histogram
- Aligned voxels = crisp peaks in joint histogram



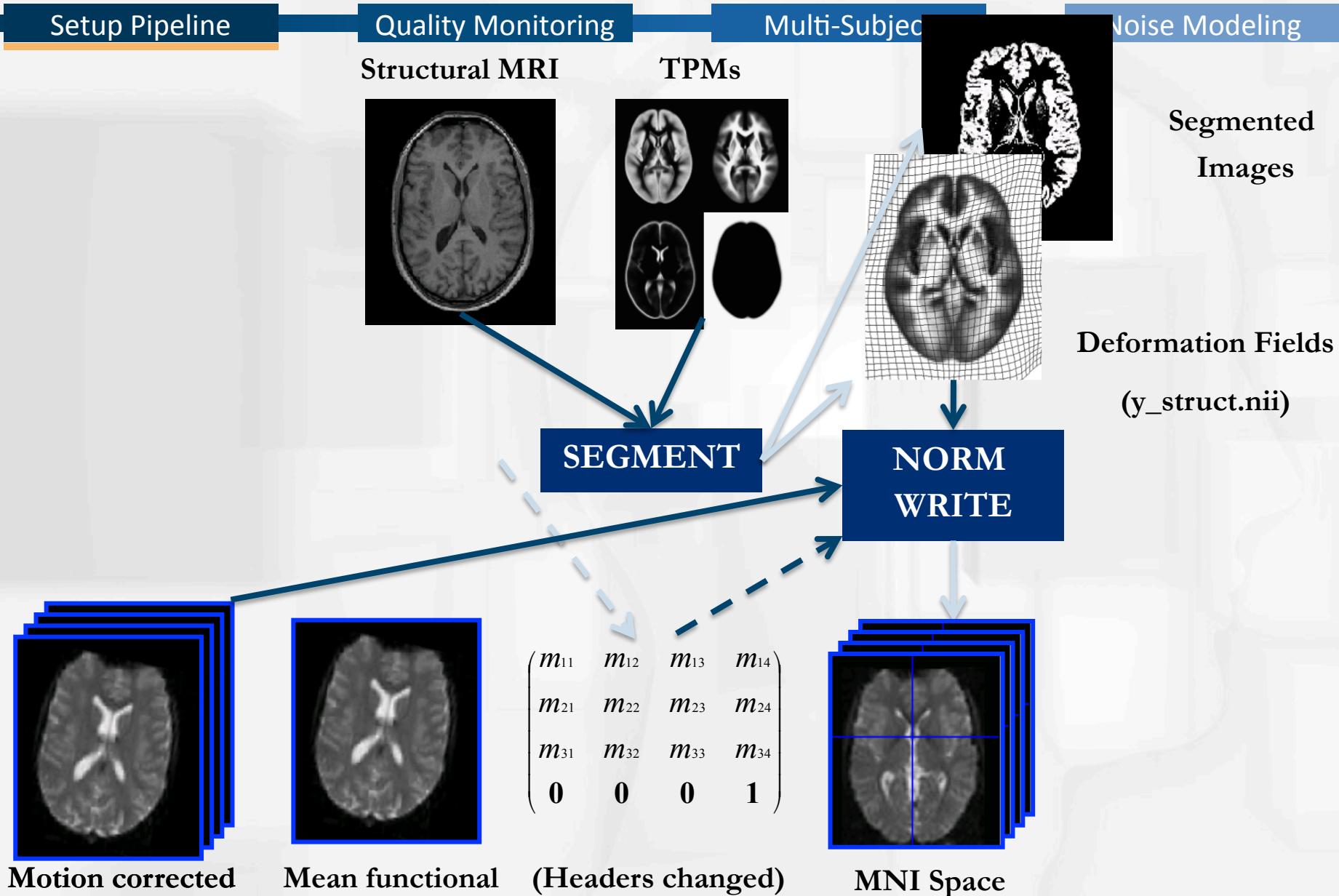
intensity bins
structural

intensity bins
functional

Joint Histogram:
 $h(i_f, i_s)$

Count of voxels who have intensity i_f in functional and i_s in structural image

Theory: Unified Segmentation





- Why is normalisation difficult?
 - No simple similarity measure, a lot of possible transformations...
 - Different Imaging Sequences (Contrasts, geometry distortion)
 - Noise, artefacts, partial volume effects
 - Intensity inhomogeneity (bias field)
 - **Normalisation** of segmented tissues is more robust and precise than of original image
 - Tissue **segmentation** benefits from spatially aligned tissue probability maps (of prior segmentation data)
- ➔ Motivates a unified model of segmentation/normalisation

Theory: Unified Model Segmentation



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

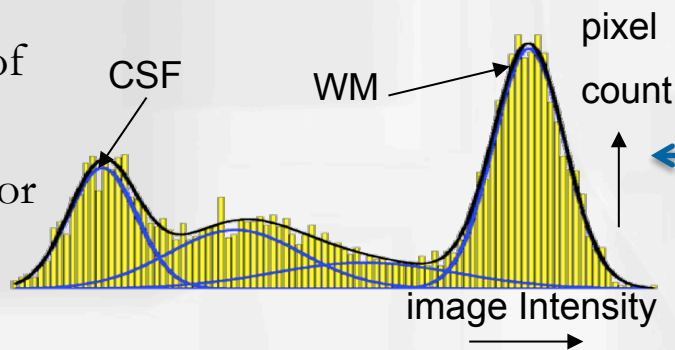
- Bayesian generative model¹ of voxel intensities from tissue class probabilities

[1] Ashburner & Friston (2005), *Neuroimage*

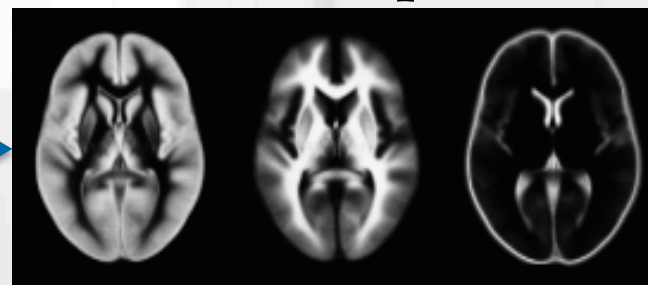
- probability of intensity of a certain pixel being at specific value

Gaussian Mixture Model

probability of intensity in given voxel for tissue class



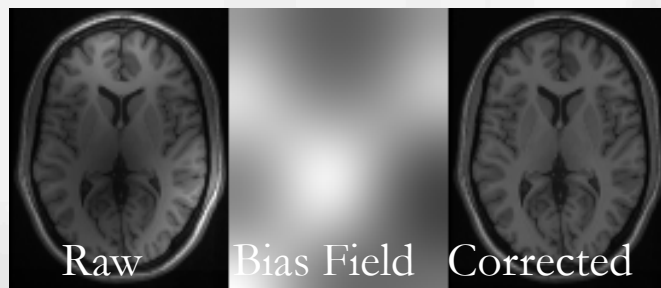
Prior: Tissue probability maps



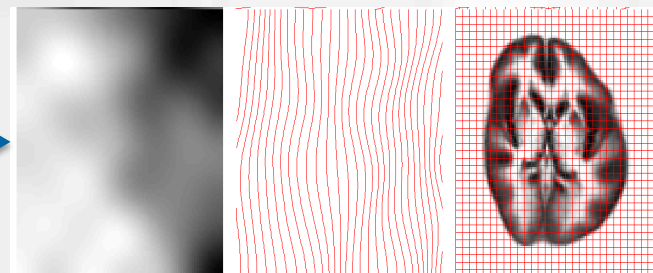
TPMs in MNI space

Bias Field

coil inhomogeneities

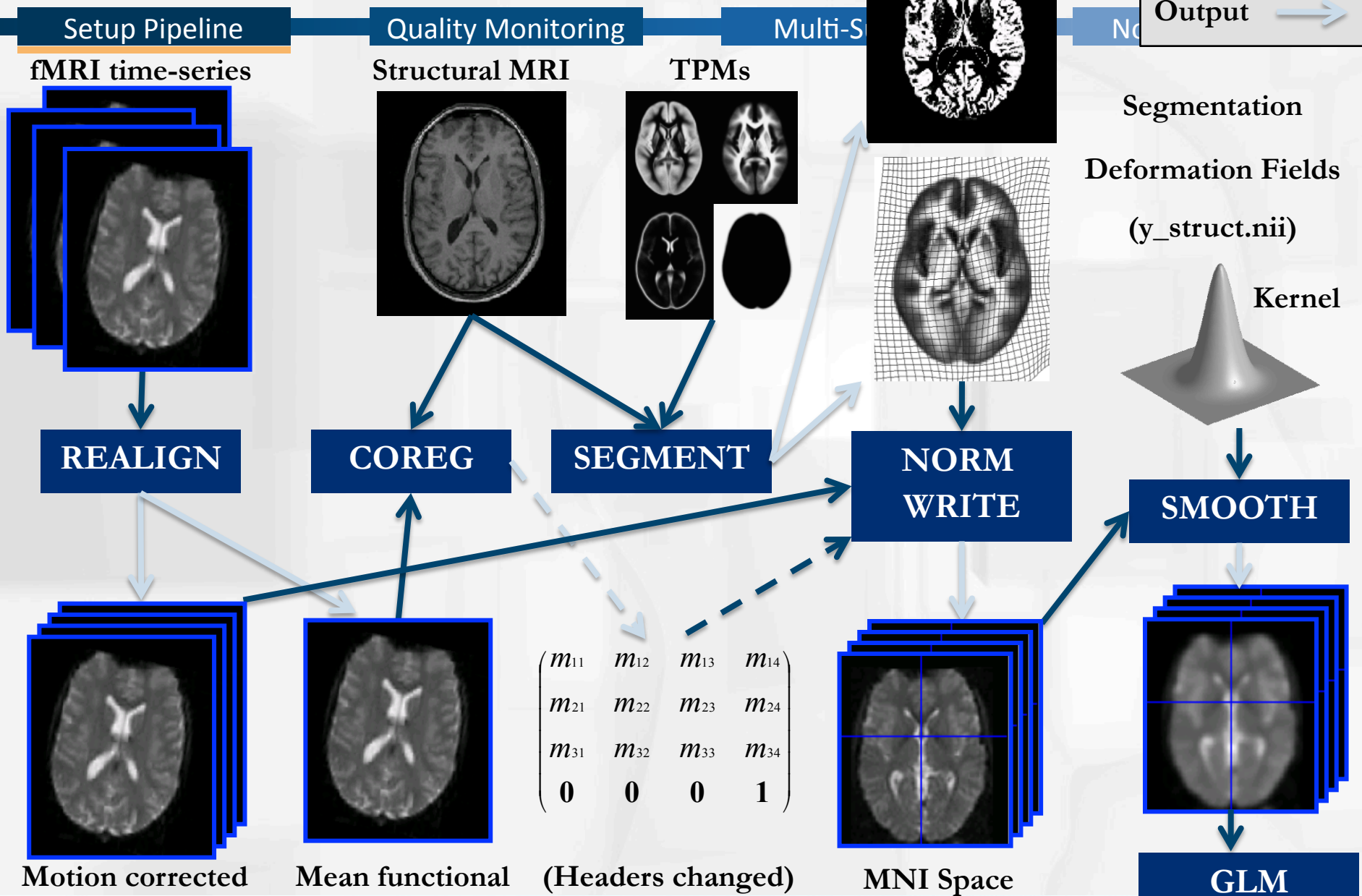


Deformation Fields



discrete cosine transforms

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



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

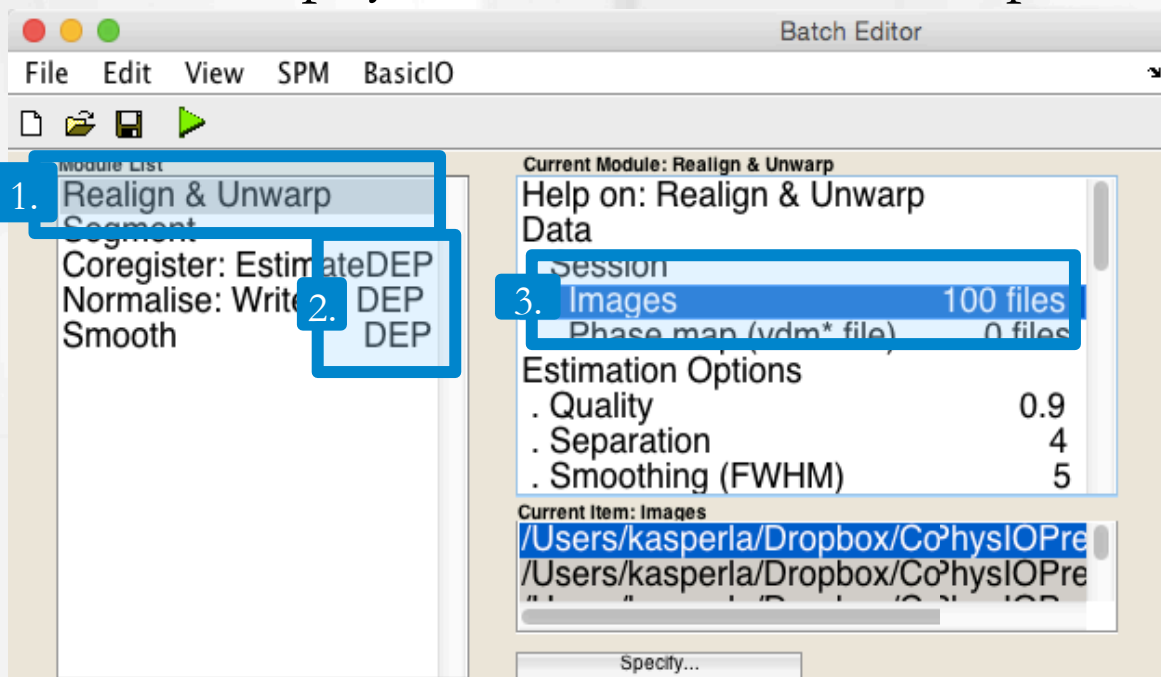
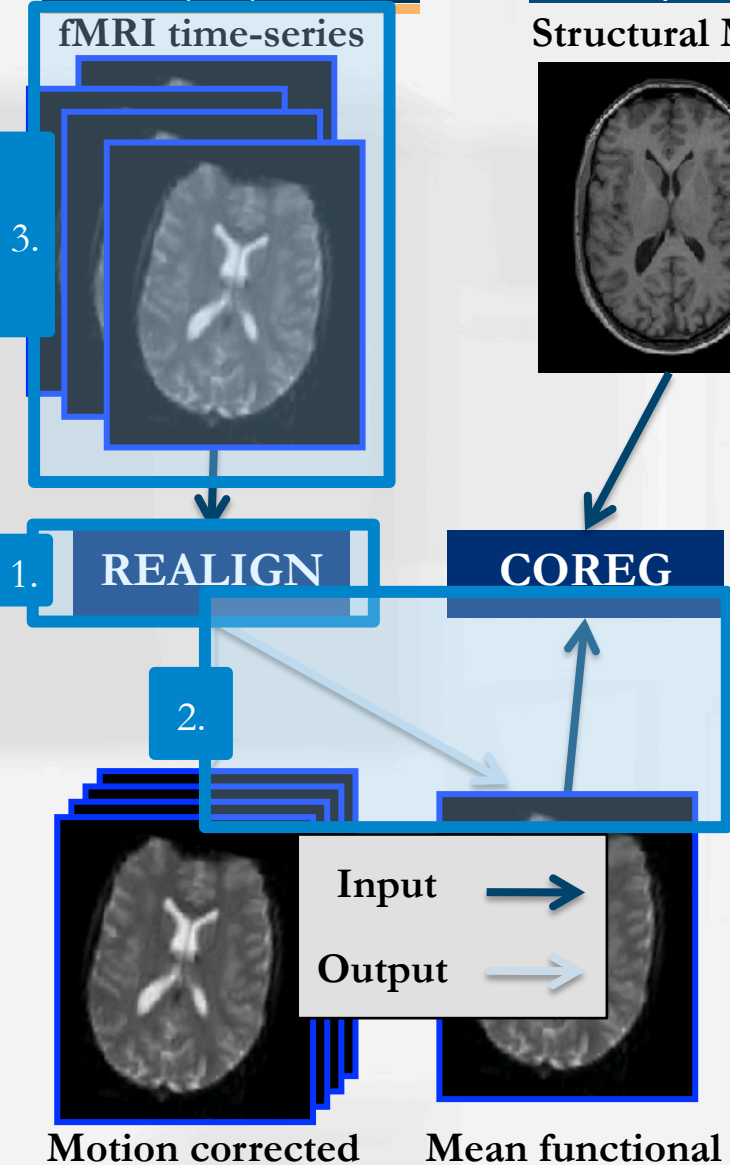
fMRI time-series

Structural MRI

Elements of a Pipeline

1. Modules
2. Dependencies
3. Data (Subject-specific)

Philosophy: Batch = Materialised Pipeline



The Batch Editor in SPM



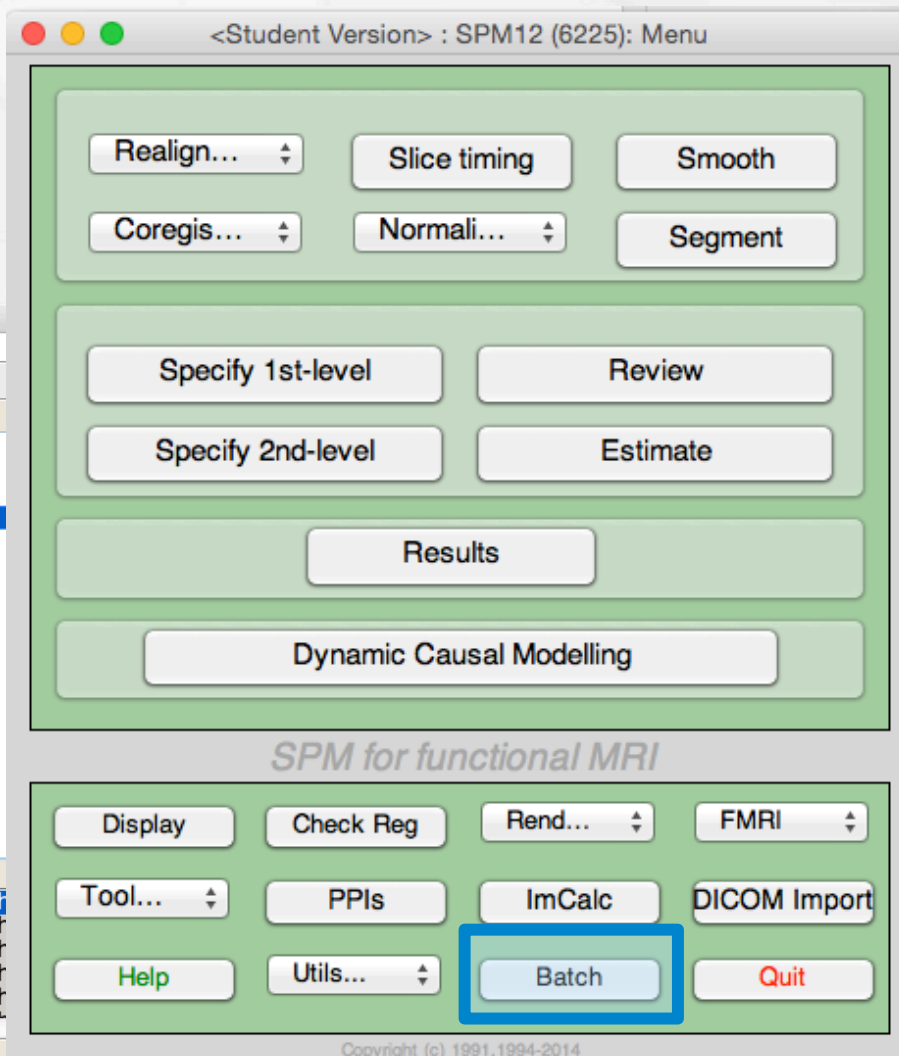
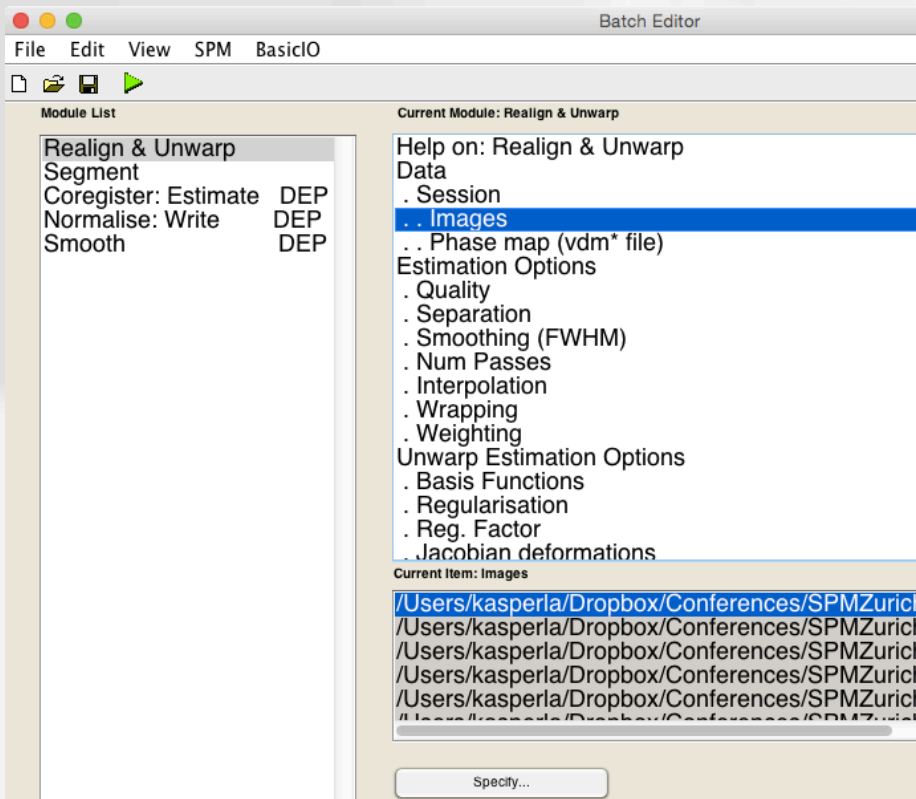
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Developer: V. Glauche
- Command line usage:
`spm_jobman`



Demo 1



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Simple preprocessing pipeline for fMRI (G. Ridgway)
 - Based on `spm12/batches/preproc_fmri_simplified.m`
 - Data: Social Learning (A. Diaconescu^[1], TNU Zurich, Philips 3T)
- After download:
 1. Unzip archive `examples_physio_short.zip`
 2. Open Matlab, run `code/init_reset_example.m`
 - `subject01`-folder created, Batches filled with right data (filenames and path) already
 3. Load `subject01/batches/demo01_simple_batch_preproc/batch_spm_preproc_fmri_simplified.m` in Batch Editor
 - Either via GUI or `spm_jobman('initcfg'); spm_jobman('interactive', 'batch_spm_preproc_fmri_simplified.m');`
 4. Run Batch (Press Play)

[1] Diaconescu (2014), PLoS CB

Demo 1 – GUI Batch Editor



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

Batch Editor

File Edit View SPM BasicIO

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
- . 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

Current Item: Images

[/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproc](#)

[/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproc](#)

[/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproc](#)

[/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproc](#)

[/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproc](#)

[/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproc](#)

[/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreproc](#)

Specify...

Demo 1 - Dependencies



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

Batch Editor

File Edit View SPM BasicIO

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

Specify... Dependency

Source Image

Demo 1 – Hands-On Movie



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

The screenshot displays the MATLAB R2015a environment. The top menu bar includes 'MATLAB', 'Window', and 'Help'. The main interface is divided into several panes:

- File Explorer (Left):** Shows the current folder path: `/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreprocessing/example_physio_short`. It lists files and folders including 'code', 'raw', and 'text'. The 'code' folder contains files like `create_multi_subject...`, `get_multiple_conditio...`, `init_reset_example.m`, and `update_absolute_path...`. The 'raw' folder contains subfolders like 'batches' and files like `fmri.nii.gz`, `onsets.txt`, `phys.log`, and `struct.nii.gz`. The 'text' folder contains `readme.txt`.
- Editor (Center):** Displays the contents of `readme.txt`. The text is as follows:

```
1 % ECG 3T FMRI
2 % =====
3 %
4 % To run this example, do the following:
5 % 0) Make sure you successfully installed SPM 12 from
6 %    http://www.fil.ion.ucl.ac.uk/spm/software/spm12/
7 %    => check via typing spm_fmri in Matlab
8 %
9 %
10 % 1) Install the PhysIO Toolbox for Physiological Noise Correction
11 %    a) Download from www.translationalneuromodeling.org/tapas
12 %    copy all files in
13 %       PhysIO_r<CurrentVersion>/code
14 %    to a new sub-folder of your SPM12-folder
15 %    ... /spm12/toolbox/PhysIO
16 %    (NOT the matlab/toolbox folder !!!)
17 %    b) Start Matlab and SPM12 and check whether in the batch editor,
18 %       a new drop-down option exists in
19 %       SPM -> Tools -> TAPAS PhysIO Toolbox
20 %    c) Close SPM12
21 %
22 % 2) Run code/init_reset_example.m in Matlab
23 %    (subfolder of this file's location)
24 %    => This creates a subject folder, a sub-folder structure with
25 %       all data, and automatically updates paths for your computer
```
- Command Window (Bottom):** Shows the MATLAB prompt `>> |` with a cursor.
- Search Documentation (Top Right):** A search bar with the text 'Startup ArtPitfallsSpm' and a search icon.

Demo 1 - Results



Setup Pipeline

Quality Monitoring

Multi-Subject

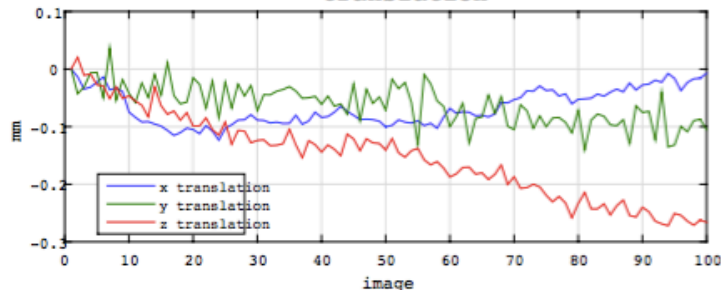
Noise Modeling

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

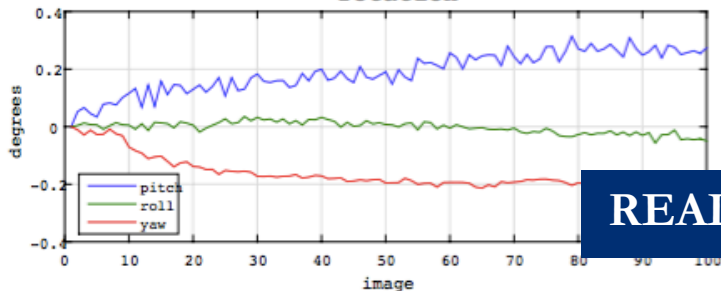
Image realignment

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

translation



rotation



REALIGN

Normalised Mutual Information Coregistration

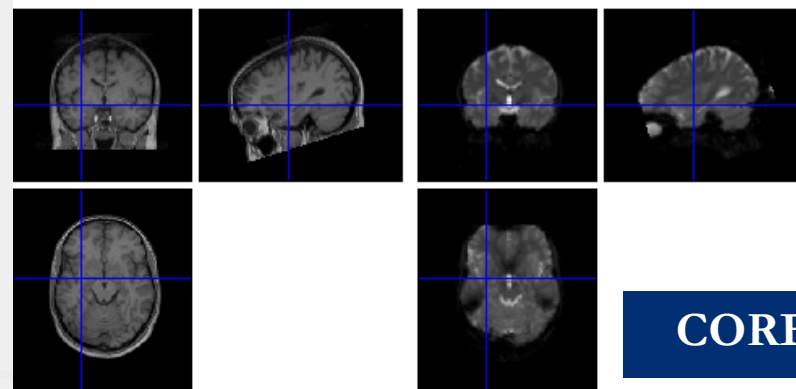
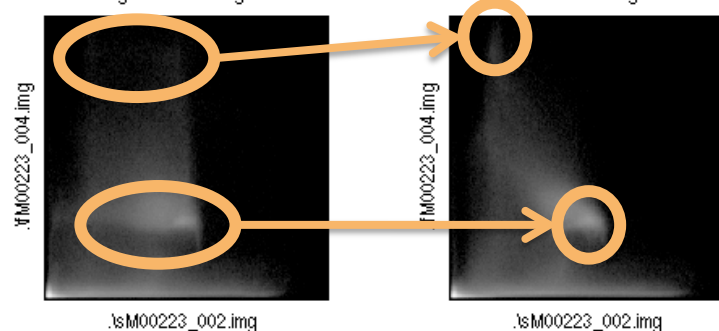
$$X1 = 3.000*X + 0.019*Y + 0.017*Z + 28.728$$

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

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

Original Joint Histogram

Final Joint Histogram



COREG

- 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



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Pipelines → Automatisation of Preprocessing
- When something goes wrong...how do we even notice?
 - Monitoring, but: cumbersome, when lots of data
 - Thus: Automate quality monitoring as well via pipelines
- Required: Suitable performance measures
 - Single image: visual inspection geometry/contrast/noise/SNR
 - Time series
 - Mean ⇒ Artifact levels (localization)
 - SD ⇒ Fluctuation levels (consistent over whole time series)
 - $SNR = \text{Mean}/SD$ ⇒ sensitivity for signal changes of interest, e.g. BOLD
 - Diff images (max(abs(diff)) or odd vs even) ⇒ outlier detection, image noise

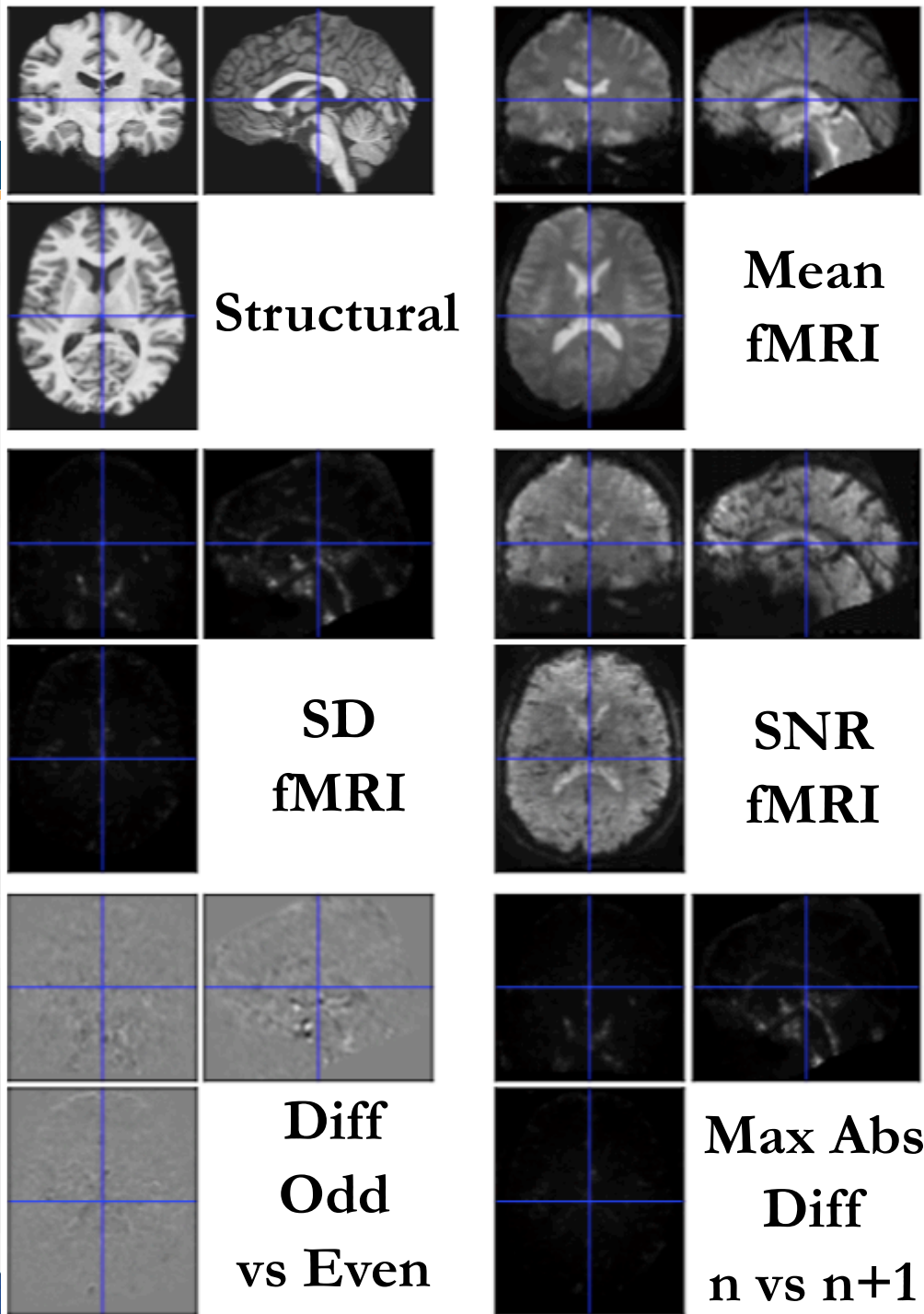
Welvaert (2013), PLoS One *Friedman/Glover (2006), JMIR*

Statistical Images

Setup Pipeline

Quality Monitoring

- Structural: visual inspection geometry/contrast/noise/SNR
- Functional
 - Mean => Artifact levels (localization)
 - SD => Fluctuation levels (consistent over whole time series)
 - SNR = Mean/SD => sensitivity for signal changes of interest, e.g. BOLD
 - Diff images (max(abs(diff)) or odd vs even) => outlier detection, image noise

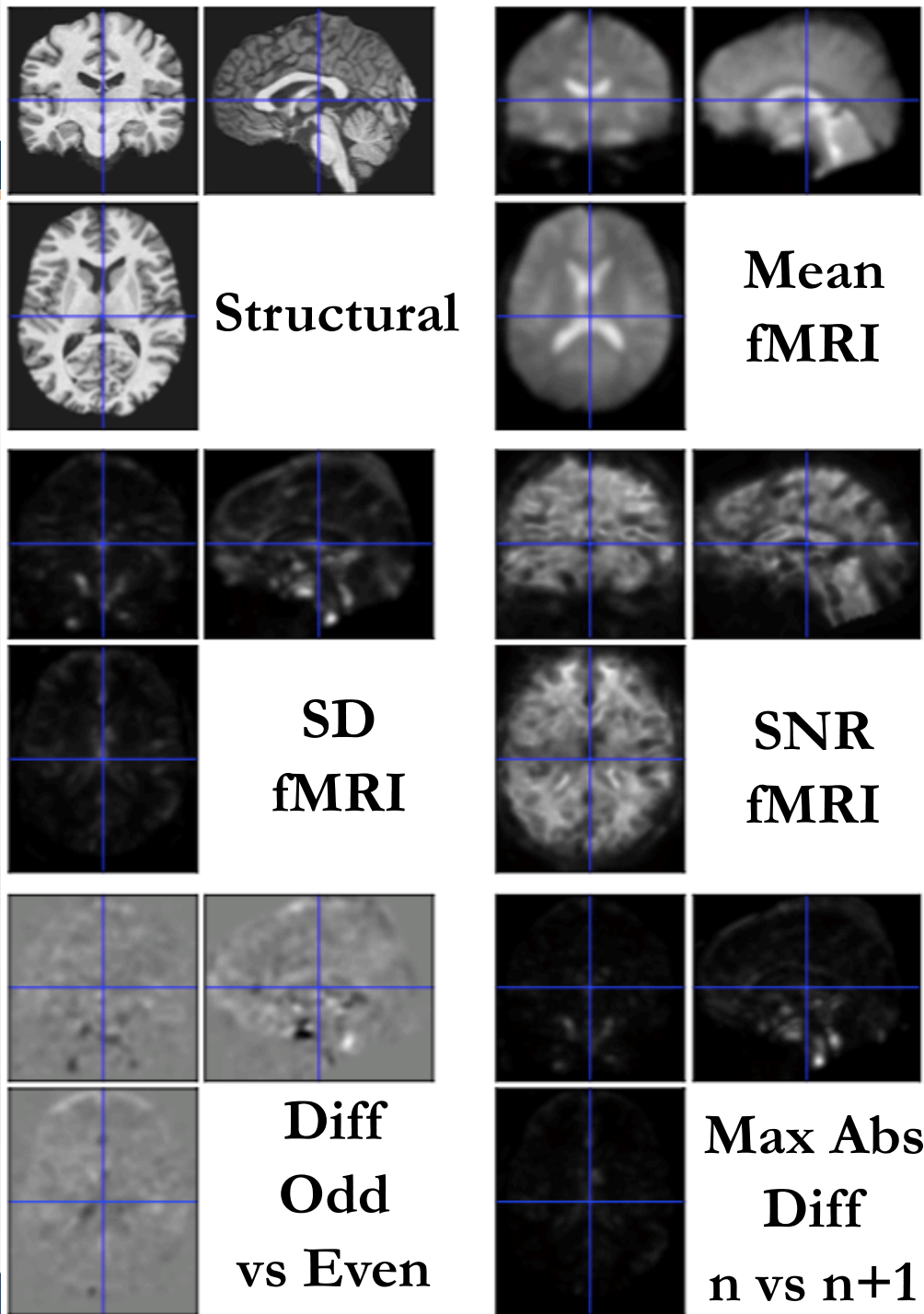


Statistical Images

Setup Pipeline

Quality Monitoring

- Structural: visual inspection geometry/contrast/noise/SNR
- Functional
 - Mean => Artifact levels (localization)
 - SD => Fluctuation levels (consistent over whole time series)
 - SNR = Mean/SD => sensitivity for signal changes of interest, e.g. BOLD
 - Diff images (max(abs(diff)) or odd vs even) => outlier detection, image noise



fMRI = Acquiring Movies



Setup Pipeline

Quality Monitoring

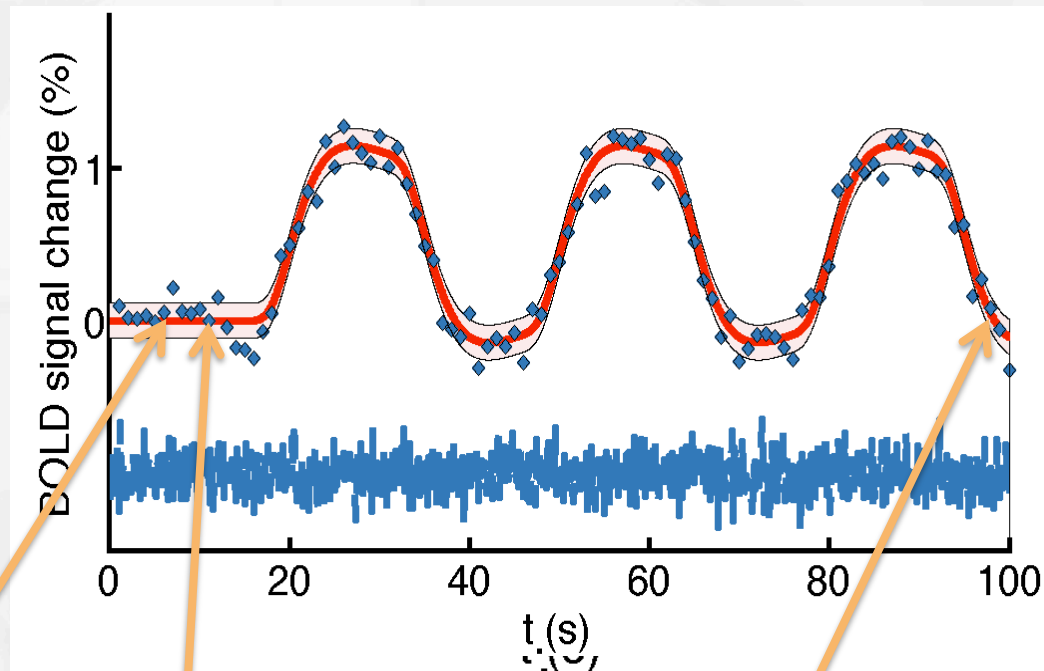
Multi-Subject

Noise Modeling

- The Localized Time-series is the Fundamental Information Unit of fMRI

Signal: Fluctuation through Blood oxygen level dependent (BOLD) contrast

Noise: All other fluctuations



- Run/Session: Time Series of Images



scan 1



...



scan N

time

Why Temporal SNR?



Setup Pipeline

Quality Monitoring

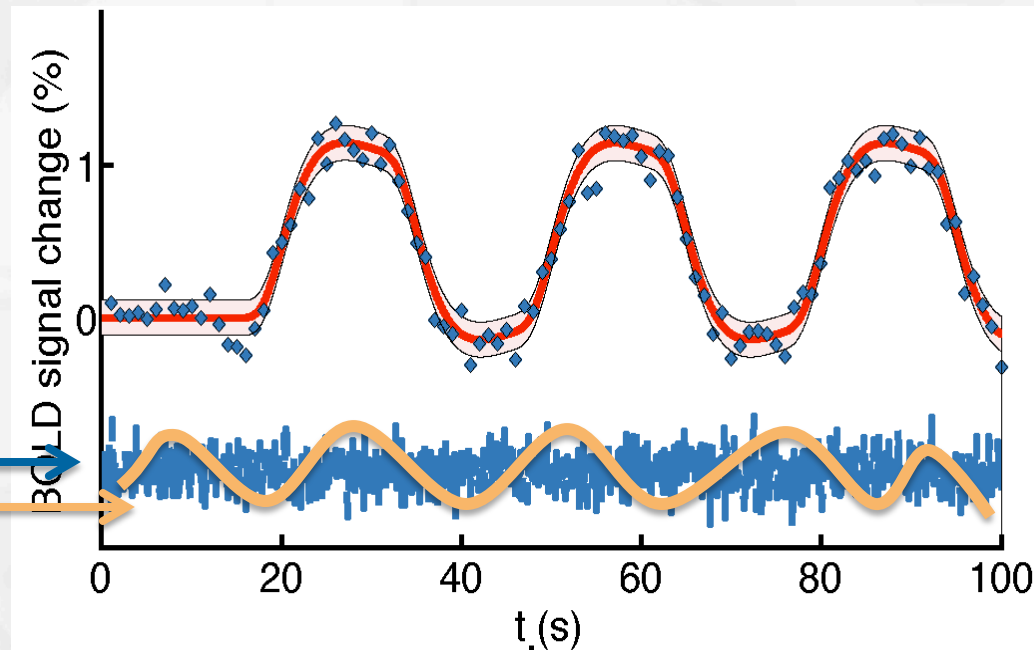
Multi-Subject

Noise Modeling

- Thermal Noise
 - temporally uncorrelated
 - reduced SNR → risk of false negatives
 - Remedy: Spatial Smoothing

Noise: All other fluctuations

- “Structured” Noise
 - temporally correlated
 - reduced SNR → risk of false negatives
 - correlated with task → risk of false positives
 - Remedy: Noise Modelling (e.g. GLM)



Inference = Signal-To-Noise

$$t = \frac{\beta}{\sqrt{\hat{\sigma}_\varepsilon^2 (X^T X)^{-1}}} = \frac{\beta \|\mathbf{x}\|}{\hat{\sigma}_\varepsilon}$$

$$F = \frac{N - M}{M_1} \cdot \frac{(\sigma_S^2 + \sigma_N^2) - \sigma_N^2}{\sigma_N^2}$$



- Complete preprocessing & monitoring pipeline for fMRI
 - Based on `spm12/batches/preproc_fmri`
 - Includes `batch_report_quality.m` to visualize and save quality measures after each preprocessing step
- Run via
 1. Cleanup processed data of subject01: `init_reset_example(2)`
 2. Load `subject01/batches/demo02_simple_batch_preproc/batch_preproc_fmri_report_quality.m`
 3. Either via GUI or `spm_jobman('interactive', 'batch_preproc_fmri_report_quality.m');`
 - Alternative: Run batch directly from command line: `spm_jobman('run', 'batch_preproc_fmri_report_quality.m');`

Demo 2 - GUI



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

Batch Editor

File Edit View SPM BasicIO

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

Specify... Dependency

Demo 2 - Video



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

QuickTime Player File Edit View Window Help

SPM12 (6225): Menu

HOME PLOTS

New Shortcut Organize Shortcuts Quick Access

MANAGE

Current Folder

Name

Folder

- batches
- demo01_simple
- demo02_compare
- demo03_multi
- demo04_stats
- spm_2015Jun04
- fMRI
 - fmri.mat
 - fmri.nii
 - fmri_uw.mat
 - meanufmri.nii
 - rp_fmri.txt
 - swufmri.nii
 - ufmri.mat
 - ufmri.nii
 - wufmri.nii
- glm
- logs
 - onsets.txt
 - phys.log
 - report_quality
- struct
 - mstruct.nii
 - struct.nii
 - struct_seg8.mat
 - v_struct.nii

demo02_compare_batch_quali

SPM for functional MRI

Display Check Reg Re... FM...

To... PPIs ImCalc DICOM Import

Help Util... Batch Quit

MATLAB R2015a - acad

PUBLISH VIEW

SPM12 (6225): Graphics

File Edit View Insert Tools Desktop Window SPM Figure Help

Normalised Mutual Information Coregistration

X1 = 3.288°X + 0.023°Y - 0.114°Z - 4.830
Y1 = -0.000°X + 1.763°Y - 1.027°Z + 56.693
Z1 = 0.040°X + 0.540°Y + 3.284°Z + 45.494

Original Joint Histogram Final Joint Histogram

./fmri/meanufmri.nii ./struct/mstruct.nii

./struct/mstruct.nii ./struct/mstruct.nii

Command Window

>>

Click and drag to move the document tabs...

Demo 2 - Output



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

■ 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

SPM Plotting: CheckReg-Magic



Setup Pipeline

Quality Monitoring

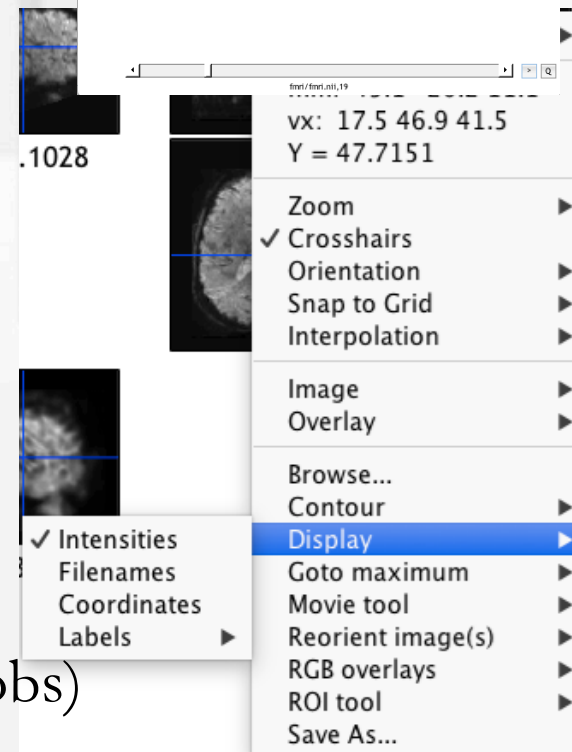
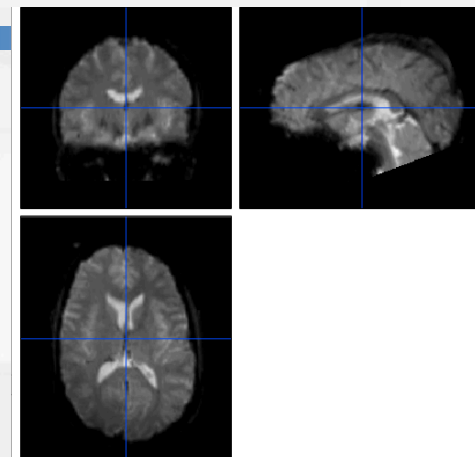
Multi-Subject

- 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)





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

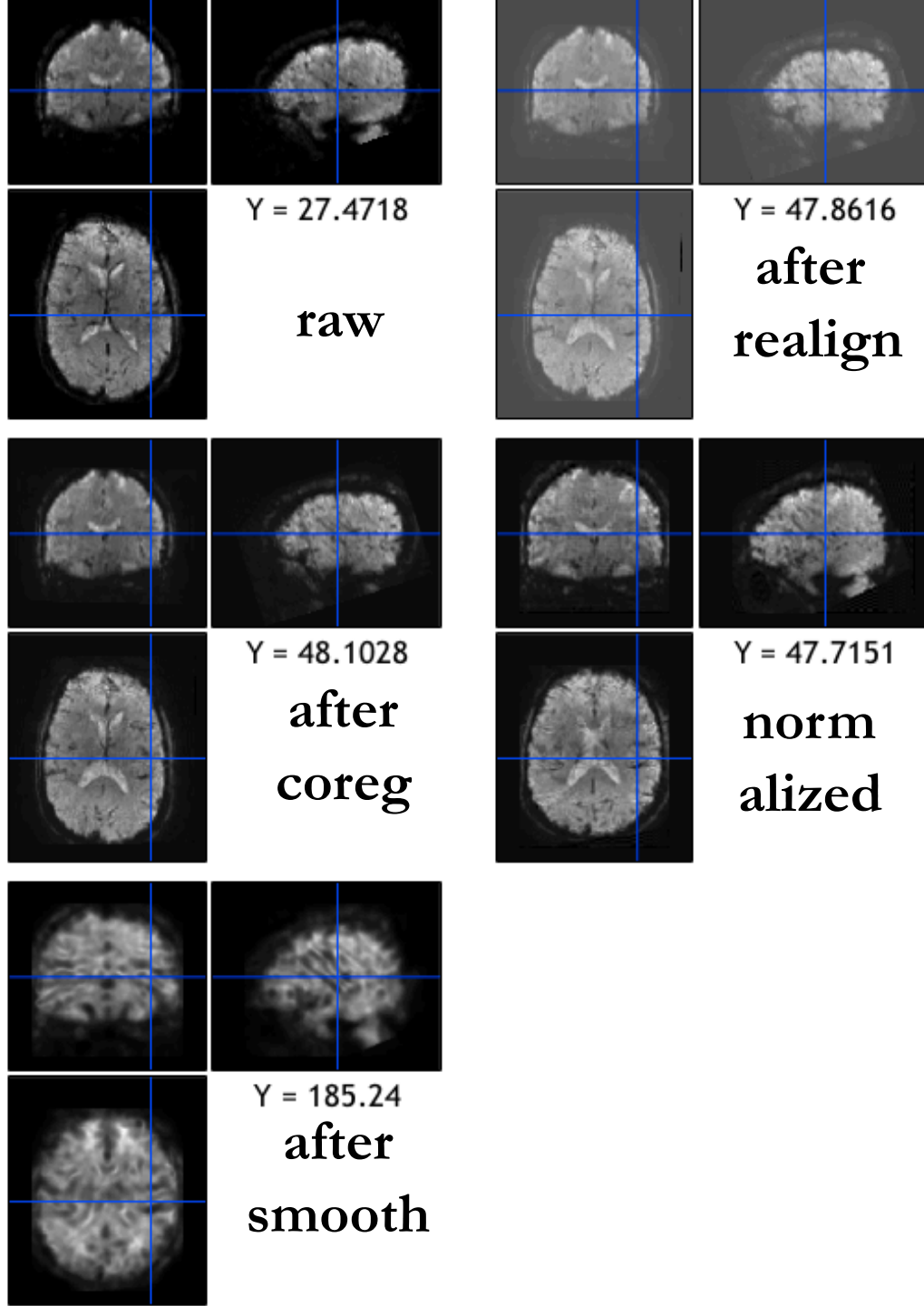
```
dirs = {'00_raw', '01_realigned',  
       '02_coregistered', '03_normalized',  
       '04_smoothed'}';  
files = 'snr.nii';  
fpFiles = strcat(dirs, '/', files);  
spm_check_registration(fpFiles{:})
```

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

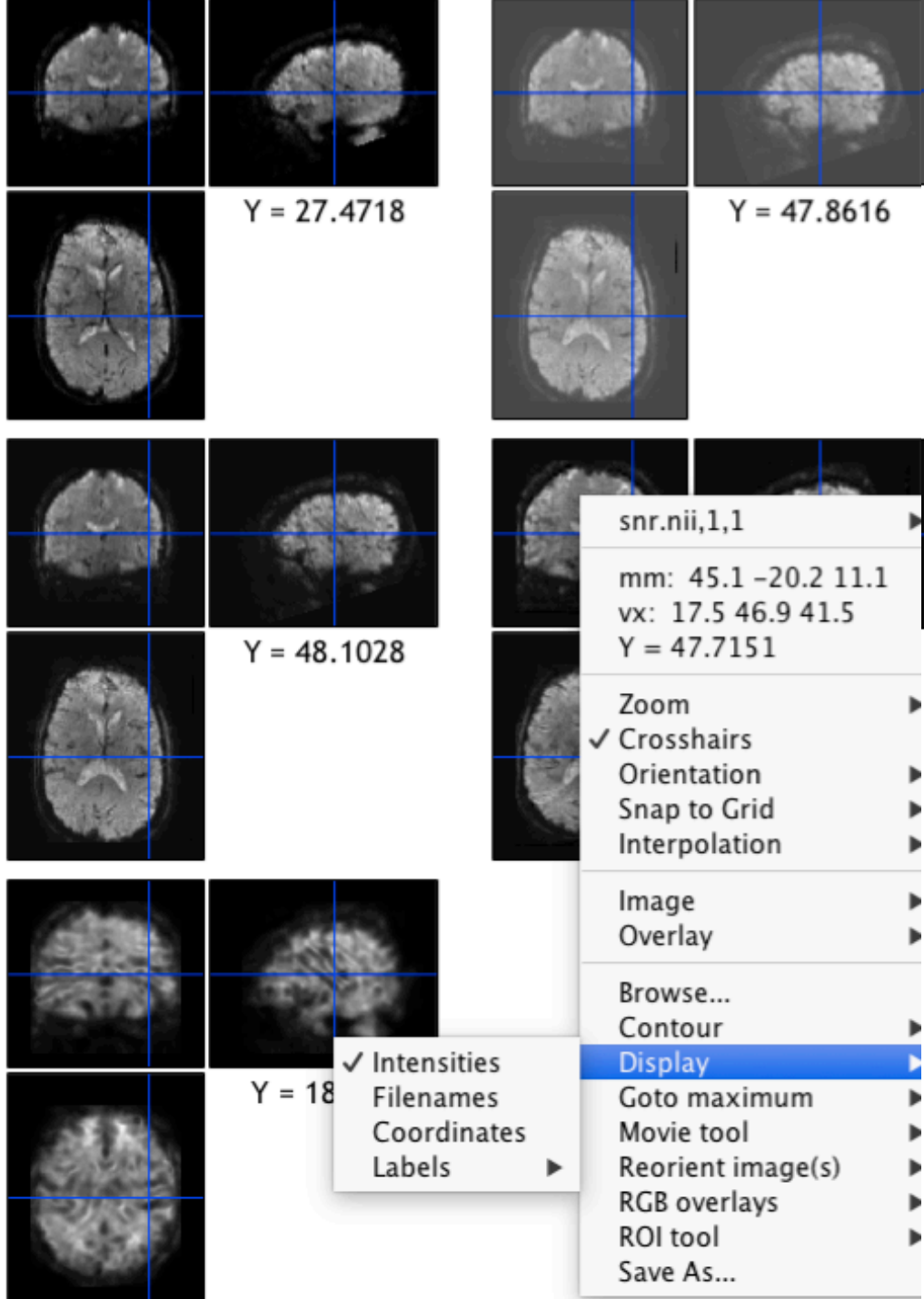


Plotting Example

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



- Task: Spotting unusual quality report images
 - `run demo02.../batch_run_artefact_subjects.m`
 - executes `code/create_artefact_subjects.m`
 - then: executes `demo02-preprocessing` batch for each subject
 - View for individual subjects: `report_quality.ps`
- What went wrong here?
 - subjectA1
 - subjectA2
 - subjectA3

Quiz...



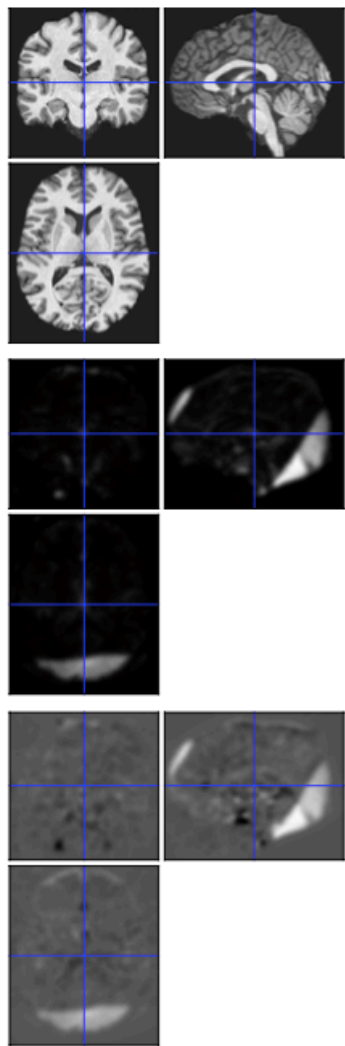
Setup Pipeline

Quality Monitoring

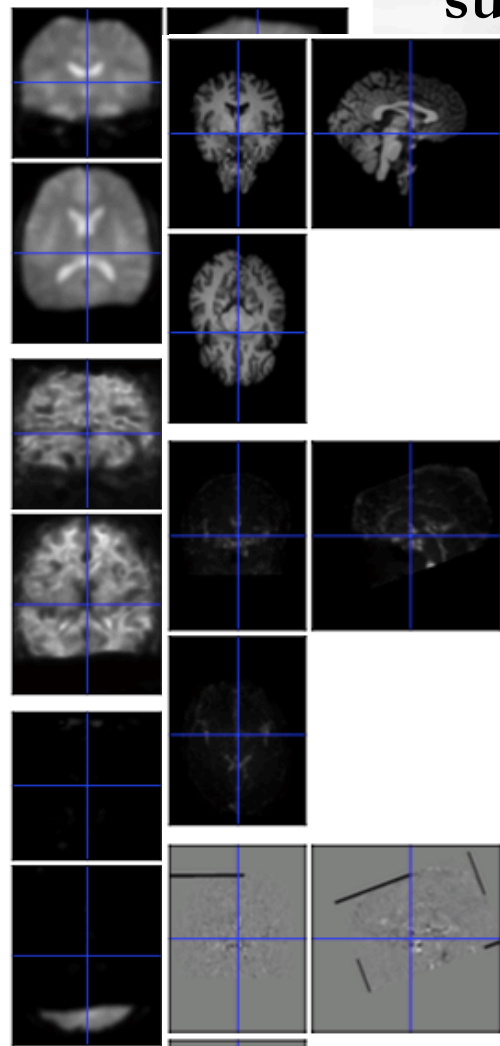
Multi-Subject

Noise Modeling

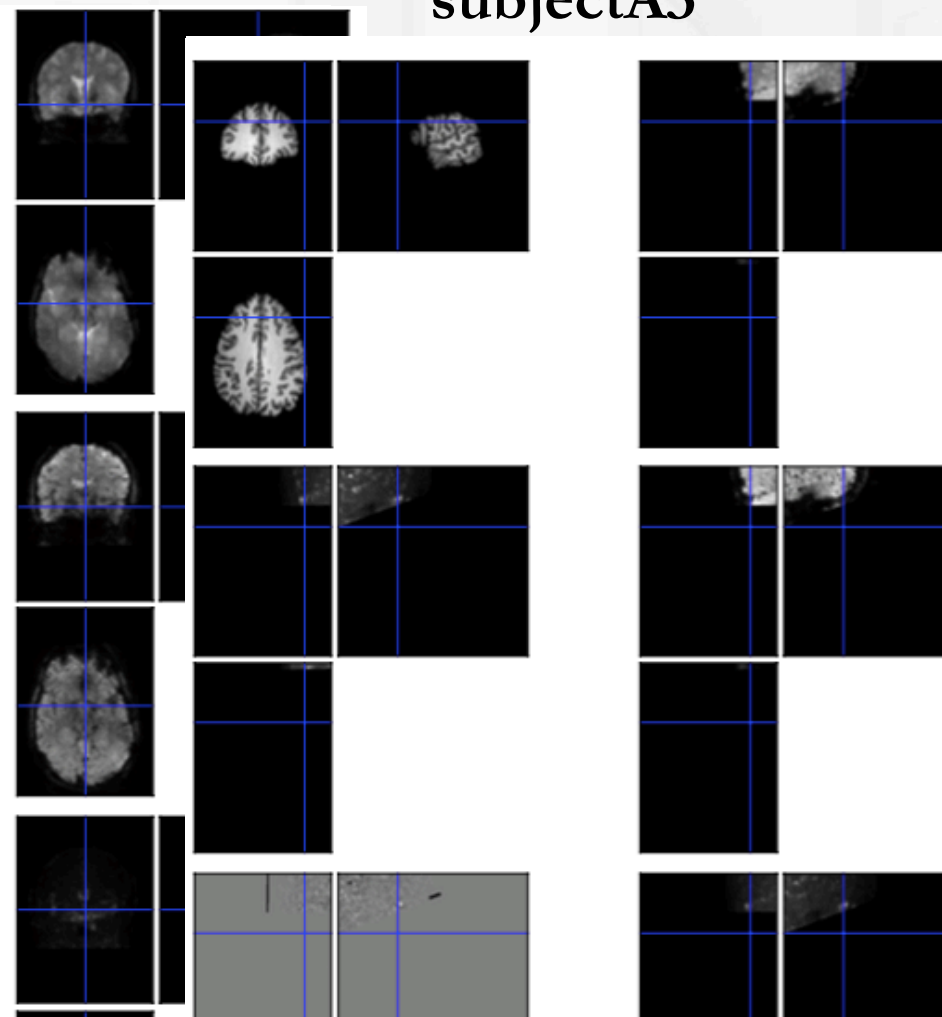
subjectA1



subjectA2



subjectA3



- 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

Comparing Pipelines



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- When running different preprocessing pipelines on the same subject:
 - Always start from raw nifti data (structural and functional)
 - Copy it to new processing folder, e.g.
 - `subject01/preproc_alternative/fmri/`
`/struct/`
 - Alter original batch file and save it under new name
 - Swapping preprocessing steps is not simple in the batch editor
 - Solution: save as .m-file and edit in Matlab
 - helper script (code folder):
`reorder_matlabbatch(fileBatch, indicesBefore, indicesAfter)`
 - Run different batch files via `run_job`

Comparing Pipelines



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Example: When to use Slice timing correction?
- Run:
 - `demo02_compare_batch_quality/batch_compare_preproc`
- this executes:
 - `batch_compare_template_stc_realign`
 - slice timing correction before realignment
 - `batch_compare_template_realign_stc`
 - realignment before slice timing
- Note: Proper Assessment only after statistical analysis possible

- 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



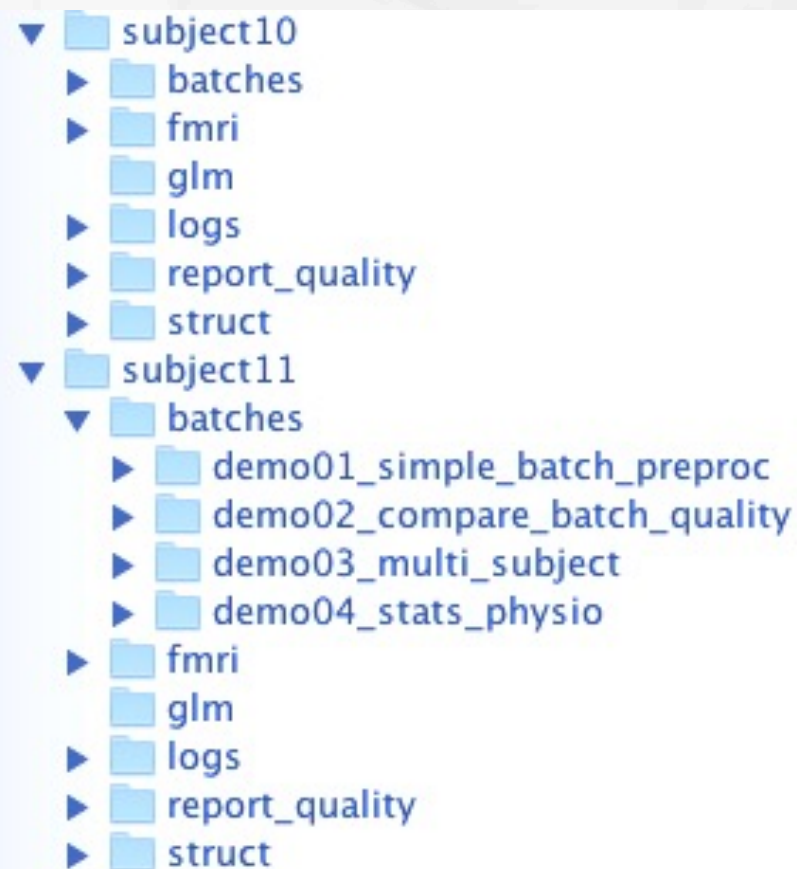
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- One Study Folder
 - sub-folder for code, templates for subject-specific batches
 - modules and dependencies stay the same
- One Folder for each subject
 - therein: same sub-folder structure
 - reduces subject-specific data specification to changing the subject folder
 - batches/ => subject batch with data spec
 - fmri/ => functional data per session
 - struct/ => structural data



- glm/ => statistical analyses
- logs/ => behavioural response files, physiological recordings

Looping over subjects



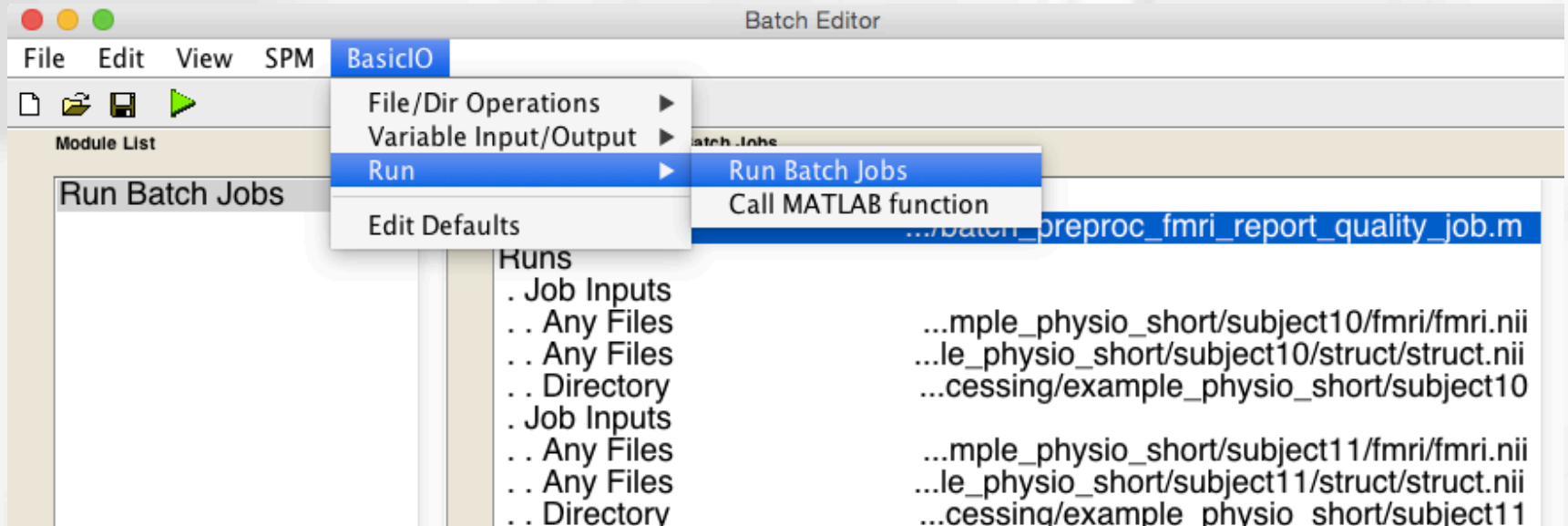
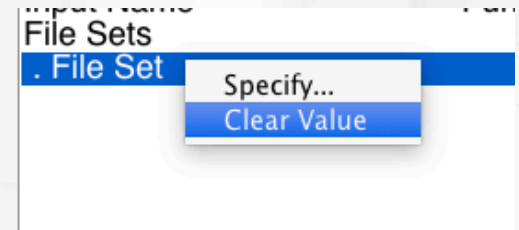
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

1. Test and optimise batch for 1 subject
2. Clear subject-specific data inputs from batch (right click on entry, <-X), save as template job
3. Create “Run Batch Job”
 - calls job-batch several times, filled with different inputs



Looping over subjects - Scripting



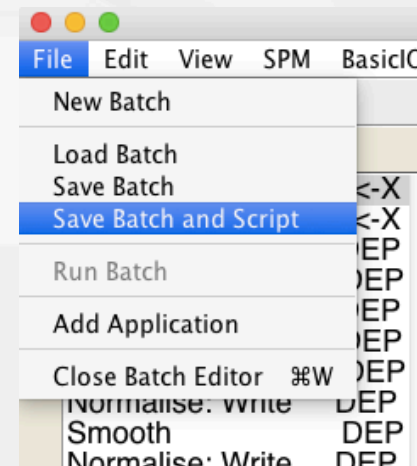
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Alternative: After removing subject-specific data, choose “Save batch and script”
 - creates `_job.m`-file, same as usual batch script (now: without data)
 - creates `.m`-file as a Matlab script with a multi-subject loop and specified missing input variables



```
1 % List of open inputs
2 % Named File Selector: File Set - cfg_files
3 % Named File Selector: File Set - cfg_files
4 % Make Directory: Parent Directory - cfg_files
5 nrun = X; % enter the number of runs here
6 jobfile = {'/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreprocessing/example_physio
7 jobs = repmat(jobfile, 1, nrun);
8 inputs = cell(3, nrun);
9 for crun = 1:nrun
10     inputs{1, crun} = MATLAB_CODE_TO_FILL_INPUT; % Named File Selector: File Set - cfg_files
11     inputs{2, crun} = MATLAB_CODE_TO_FILL_INPUT; % Named File Selector: File Set - cfg_files
12     inputs{3, crun} = MATLAB_CODE_TO_FILL_INPUT; % Make Directory: Parent Directory - cfg_files
13 end
14 spm('defaults', 'FMRI');
15 spm_jobman('run', jobs, inputs{:});
```

TODO

Demo 3 – Example Loop Script



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

```
% 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/PracticalsPhysIOPreprocessir
jobs = repmat(jobfile, 1, nrun);
inputs = cell(3, nrun);
for crun = 1:nrun
    pathSubject = fullfile(pathStudy, dirSubjectArray{crun});
    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{:});
```


Multi-subject Quality Monitoring



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Concatenate `spm_<date>.ps` report files
- Use `spm_check_registration` to plot same statistics for different subjects
 - same way as for quality monitoring, after different preprocessing steps
 - e.g. did SNR image before smoothing change from subject to subject => possible coil failure



- Performs Demo 2 (complete preproc + quality report) for multiple subjects (subject10 and subject11)
 - create subject folders: Run `code/create_multi_subject_data.m`
- Run via
 1. Load “job”: `subject10/batches/demo03_multi_subject/batch_preproc_fmri_report_quality_job.m` - **Inspect!**
 2. Load & Run `batch_preproc_fmri_report_quality_gui.m`
 - in same folder; use *Batch* Editor to inspect
 3. Alternative: Open in *Matlab* Editor, inspect construction of subject dependent file names, run in Matlab GUI:
`batch_preproc_fmri_report_quality_run.m`

- 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

Executing Custom Matlab Code



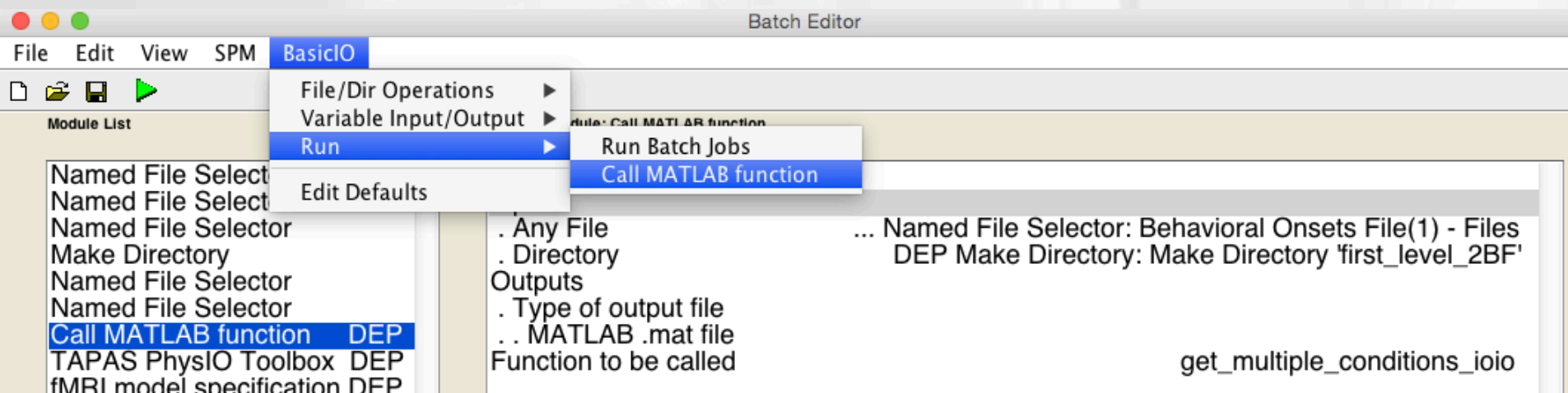
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Call Matlab script/function within Batch Editor
 - BasicIO => Call Matlab Function
 - Specify number and type of input parameters (also dependencies)
- Use, if own code must run at certain point in pipeline
 - e.g. extra realignment parameters 12/24
 - Otherwise: Matlab Editor/Command Window more efficient



Physiological Noise Correction

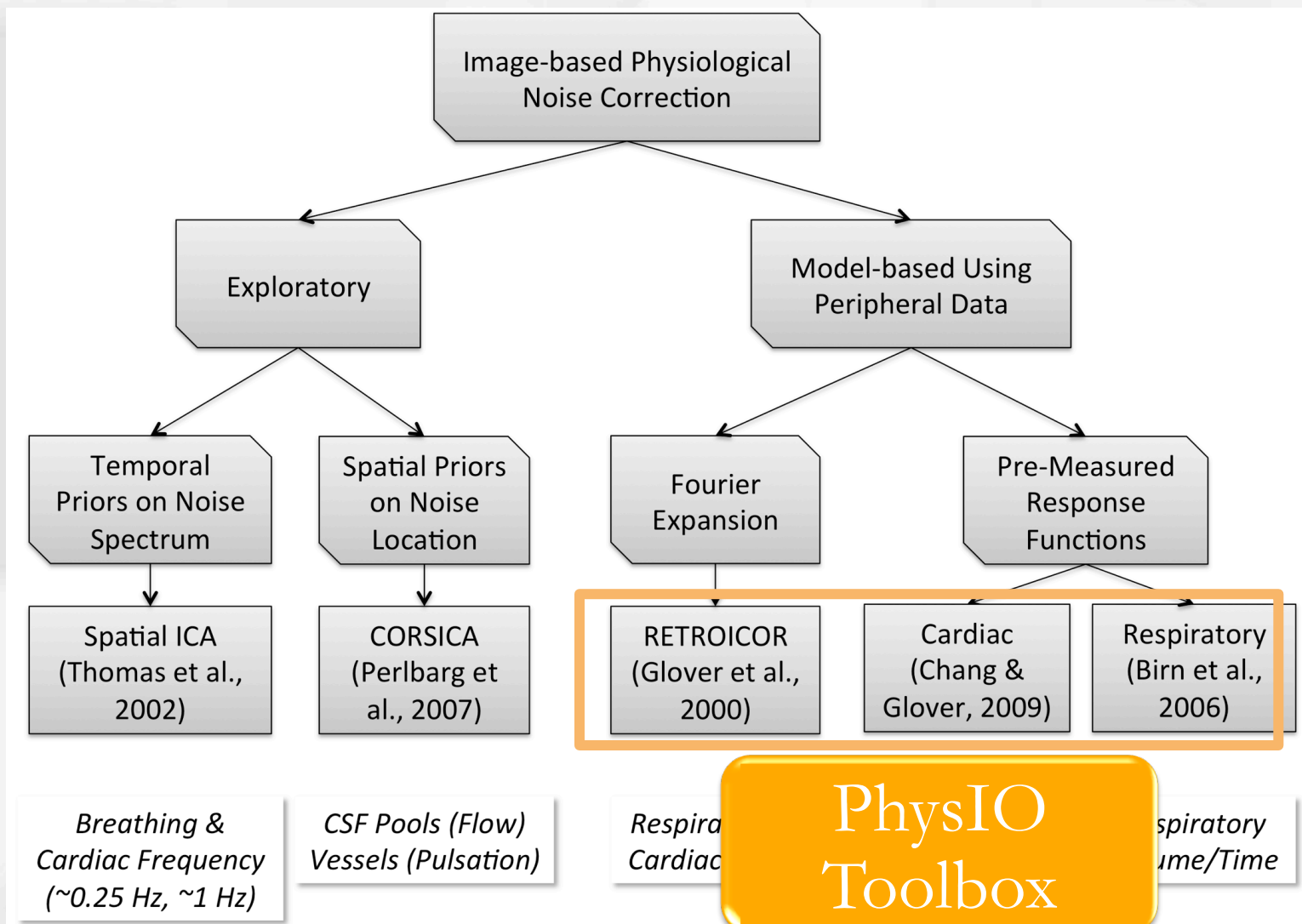


Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling



PhysIO
Toolbox

Model-based Noise Correction SPM



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

1. Physiological Monitoring

2. Preprocessing of Physiological Data

3. Model time series physiological noise

4. Noise Reduction and Assessment

Peripheral Devices

PhysIO Toolbox

SPM

ECG, PPU

→ Cardiac cycle

Breathing belt

→ Respiratory cycle

Confound regressors

Workflow PhysIO Toolbox



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

Read logfiles

- Of peripheral physiological data
- Vendor-specific

Preprocess physiological data

- Filter noisy ECG & detect cardiac pulses
- Hand-pick missing pulses

Model time series physiological noise

- RETROICOR
- Respiratory Volume
- Heart Rate

Include confound regressors (GLM)

- Multiple_regressors file for SPM

Finally:

Check Influence of Physiological Noise (Correction) on Data

- SPM
- F-contrast on 1st and second level

PhysIO: SPM Batch Interface

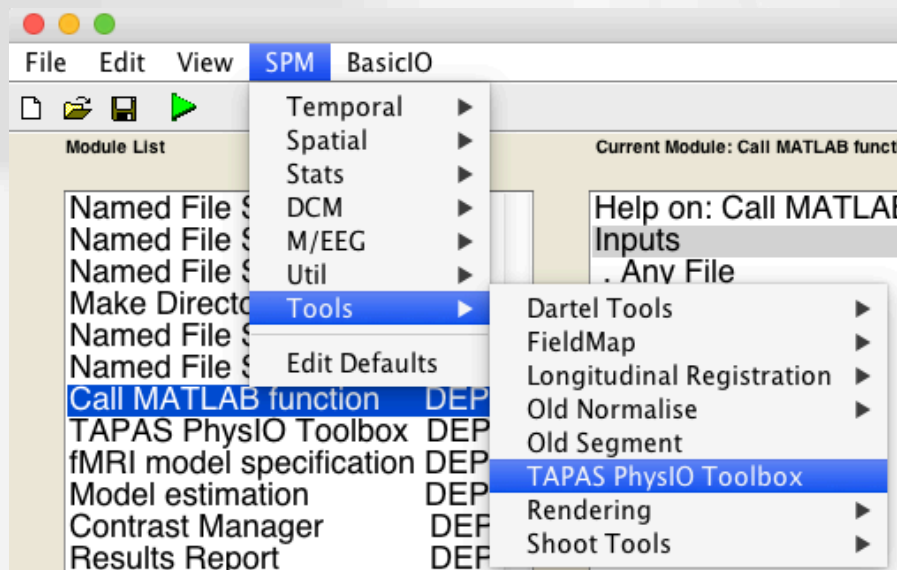


Setup Pipeline

Quality Monitoring

Multi-Subject

- Copy PhysIO/code/* .m to spm12/toolbox/physio to appear in Batch Editor
 - Menu SPM => Tools
 - works with dependencies etc.



```
Current Module: TAPAS PhysIO Toolbox
Help on: TAPAS PhysIO Toolbox
save_dir
log_files
. vendor
. log_cardiac
. log_respiration
. log_scan_timing
. sampling_interval
. relative_start_acquisition
sqpar (Sequence timing parameters)
. Nslices
. NslicesPerBeat
. TR
. Ndummies
. Nscans
. onset_slice
. time_slice_to_slice
. Nprep
thresh (Thresholding parameters for de-noising)
. Scan/Physlog Time Synchronization
. .nominal
. cardiac
. . modality
. . Initial Detection of Heartbeats
. . load_from_logfile
. . Post-hoc Selection of Cardiac Pulses
. . Off
model
. type
. order
. . cardiac
. . respiratory
. . cardiac X respiratory
. . orthogonalise
. input_other_multiple_regressors
. output_multiple_regressors
verbose
. level
. fig_output_file
. use_tabs
```

Peripheral Measurement

Versatile Read-In

Scan Time Synchronization

Preprocessing of Physiological Data

Noise Modelling

Noise Correction (via SPM GLM)

Pipeline Noise Modeling/Report



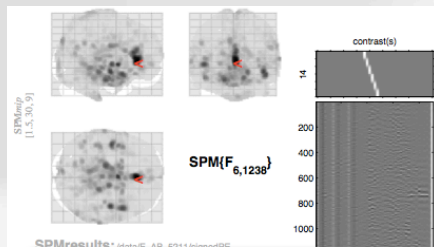
Setup Pipeline

Quality Monitoring

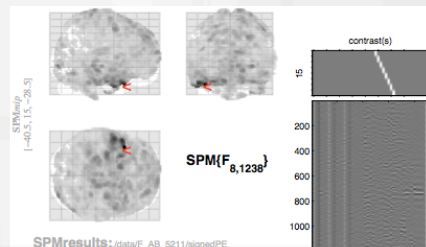
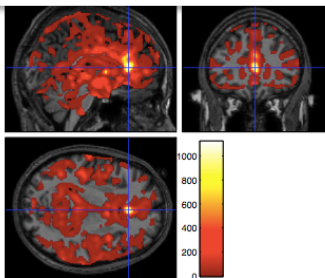
Multi-Subject

Noise Modeling

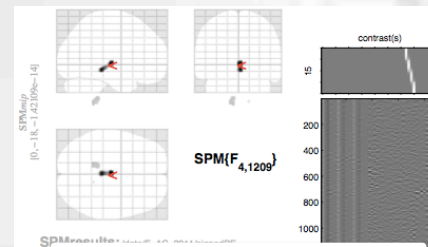
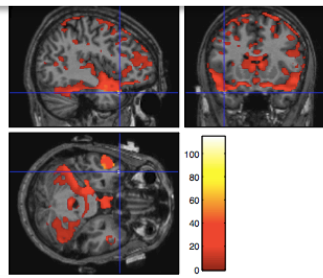
- Model physiological noise as nuisance regressors
- Check noise correction via explained variance (F-contrasts)
 - Automatic contrast creation and report (glass brain): Batch Editor
SPM => Stats => Contrast Manager/Results Report
 - With underlays: `tapas_physio_report_contrasts`



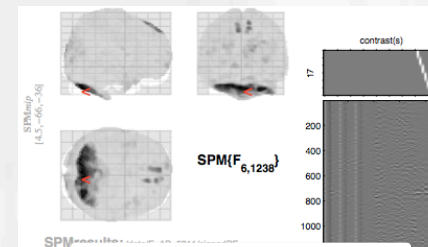
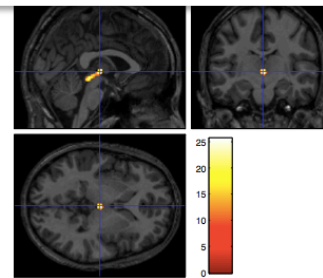
Cardiac



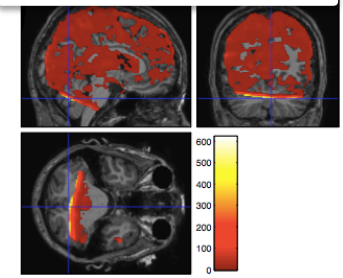
Respiratory



Card x Resp



Movement





- 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

Batch Editor

File Edit View SPM BasicIO



Module List

- 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

Current Module: TAPAS PhysIO Toolbox

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
- . align_scan

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)

Current Item: save_dir

Reference from

Make Directory: Make Directory 'first_level_2BF'

Specify...

Dependency

Demo 4 -Video



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

The screenshot displays the MATLAB R2015a environment. The top menu bar includes 'HOME', 'PLOTS', 'APPS', and 'SHORTCUTS'. The main workspace is divided into three panes:

- Current Folder:** Shows a directory tree with folders like 'batches', 'fmri', 'glm' (selected), 'logs', 'report_quality', and 'struct'. The 'glm' folder is highlighted with a blue selection bar.
- Editor:** Displays the contents of a 'README.txt' file. The text provides instructions for running the example, including steps to install SPM 12 and the PhysIO Toolbox for Physiological Noise Correction. It mentions downloading from <http://www.fil.ion.ucl.ac.uk/spm/software/spm12/> and the PhysIO Toolbox from www.translationalneuromodeling.org/tapas.
- Command Window:** Shows the execution of SPM12. The output includes the SPM12 version (v6225), the date and time (23:32:06 - 10/06/2015), and the message 'Initialising SPM :done'. It also shows the current working directory: '/Users/kasperla/Dropbox/Conferences/SPMZurich15/PracticalsPhysIOPreprocessing/example_phy'.

- 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 and Matlab script (subj.-loop)

- Integrating Own Code & Physiological Noise Correction

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

Thank you for your attention!



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- ...and these people for their support in the preparation:
 - Ged Ridgway
 - Guillaume Flandin
 - Klaas Enno Stephan
 - John Ashburner
 - Andreea Diaconescu
 - Steffen Bollmann
- Questions:
 - Now 😊
 - Poster #3736 (PhysIO): Wed, 12:45pm
 - kasper@biomed.ee.ethz.ch

■ Setting up a Preprocessing Pipeline in SPM:

The Batch Editor

- SPM Vocabulary
 - What is SPM? GUI
 - What is Realign/Co-Registration/Normalisation via Unified Segmentation
 - Dealing with Headers (Estimate vs Write/Reslice), 3D vs 4D frames
- Preprocessing Theory (SPM-specific)
 - Co-Registration Joint Histogram
 - Unified Segmentation: Algorithm, TPMs, Why does it “normalise”?
- Batch Editor Philosophy: Modules/Dependencies/Data
 - History, V. Glauche, `spm/batches (Ged)`, `preproc_fmri_simplified.m`
 - Finally: Process Diagram and corresponding pipeline, run!



- **Monitoring and Comparing Pipelines**
 - Performance Measures: Mean/SD/SNR/Diff Images
 - Rationale from time series, partial volume effects
 - mean = artifact, SD = sensitivity loss, SNR = combined, $\text{mean}(\text{abs}(\text{diff}))$
 - SPM Plotting Routines and Automatic Reporting
 - `spm_print`, `spm_check_registration`
 - Spotting Failed Pipelines
 - `fmri.mat` with 1 vol (or spike in FFT?), anatomy mirrored for coreg/segment
 - Comparing Alternative Pipelines
 - slice timing correction no/before/after realign: `preproc_fmri.m`
 - deep or broad folder tree for different pipelines?

■ Multi-subject Pipelines

- Staying on Top: Organisation of Data
 - Same folder structure for different subjects, suggested naming
- Looping Pipelines over Subjects' Data
 - Unfilled job dependencies (<-X) as input parameters
 - Skipping subjects
 - No GUI for cluster (`spm_get_defaults('cmdline',1)`) => howto output?
- Tips for Efficient Performance Monitoring
 - Compiling large PS files with abundant plotting
 - Scripts for re-loading quality report images (`spm_check_registration`)

- **Integrating Own Code and Toolboxes:**

Physiological Noise Modeling and Evaluation

- Executing Custom Matlab Code within the Pipeline
 - e.g. creation of `multiple_conditions`, extra motion regressors (12/24)
- The TAPAS PhysIO Toolbox
 - Overview which noise models implemented
 - Focus on robustness, multi-vendor compatibility
 - Modules/Parameters colored and in Batch Editor
- Automatic (Noise) Modeling and Contrast Reporting
 - Contrast Report => Glass Brain & Table as `.ps`
 - `tapas_physio_report_contrasts` => F-contrast, showing typical distributions and structural underlay



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

B. Allowed Transformations

Rigid-Body

Affine

Non-linear

REALIGN

COREG

SEGMENT

**NORM
WRITE**

C. Similarity Measure

Mean-squared
Difference

Mutual
Information

Tissue Class
Probability

D. Optimisation

Exact Linearized
Solution

Conjugate Direction
Line Search

Iterated Conditional Modes
(EM/Levenberg-Marquardt)

Theory: Good Co-Registration Output



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Voxels of same tissue have same intensity in MR-contrast
- In a 2nd MR contrast, this intensity may be different, but still the same among all voxels of same tissue type

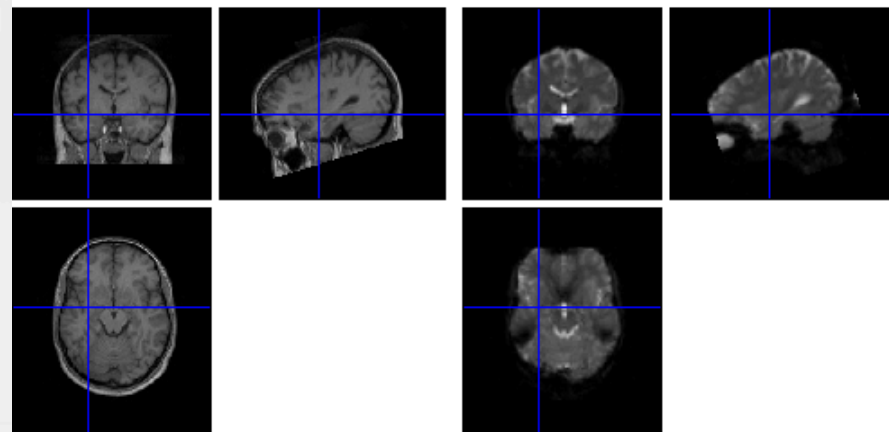
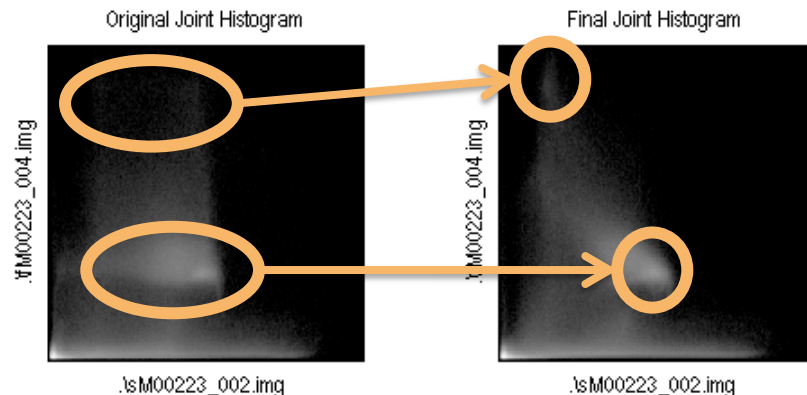
➔ Aligned voxels in 2 images = crisp peaks in joint histogram

Normalised Mutual Information Coregistration

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

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

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



Theory: Unified Model Segmentation



Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- SPM12 implements a generative model¹ of voxel intensity from tissue class probabilities
 - Principled Bayesian probabilistic formulation
 - Segmentation by inverting a Gaussian mixture model
- Deformations of prior tissue probability maps (TPMs, priors) part of the model
 - The inverse transformation that aligns the TPMs normalises the structural image(s)
 - Non-linear deformations are constrained by regularisation factors
- Bias correction (coil inhomogeneity) within the model

[1] Ashburner & Friston (2005), *Neuroimage*

Mixture of Gaussians



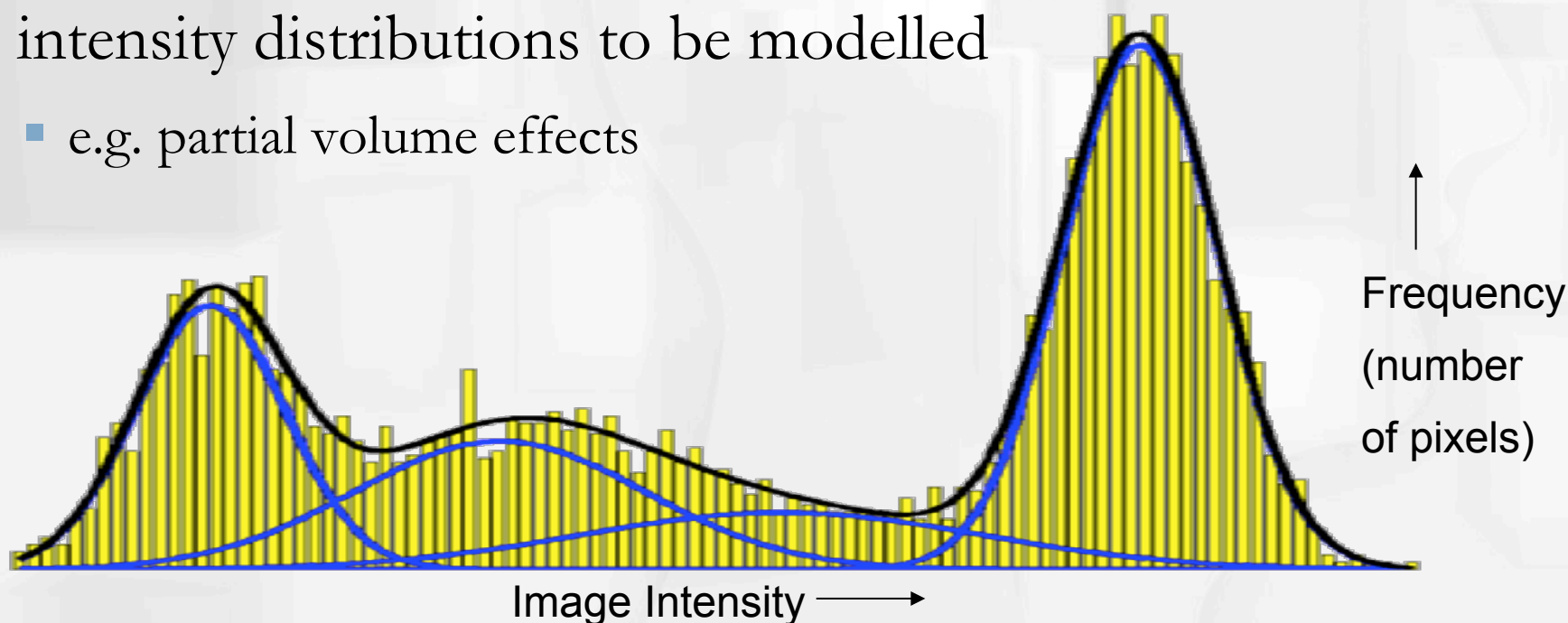
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Classification is based on a Mixture of Gaussians model, which represents the intensity probability density by a number of Gaussian distributions.
- Multiple Gaussians per tissue class allow non-Gaussian intensity distributions to be modelled
 - e.g. partial volume effects



Tissue Probability Maps



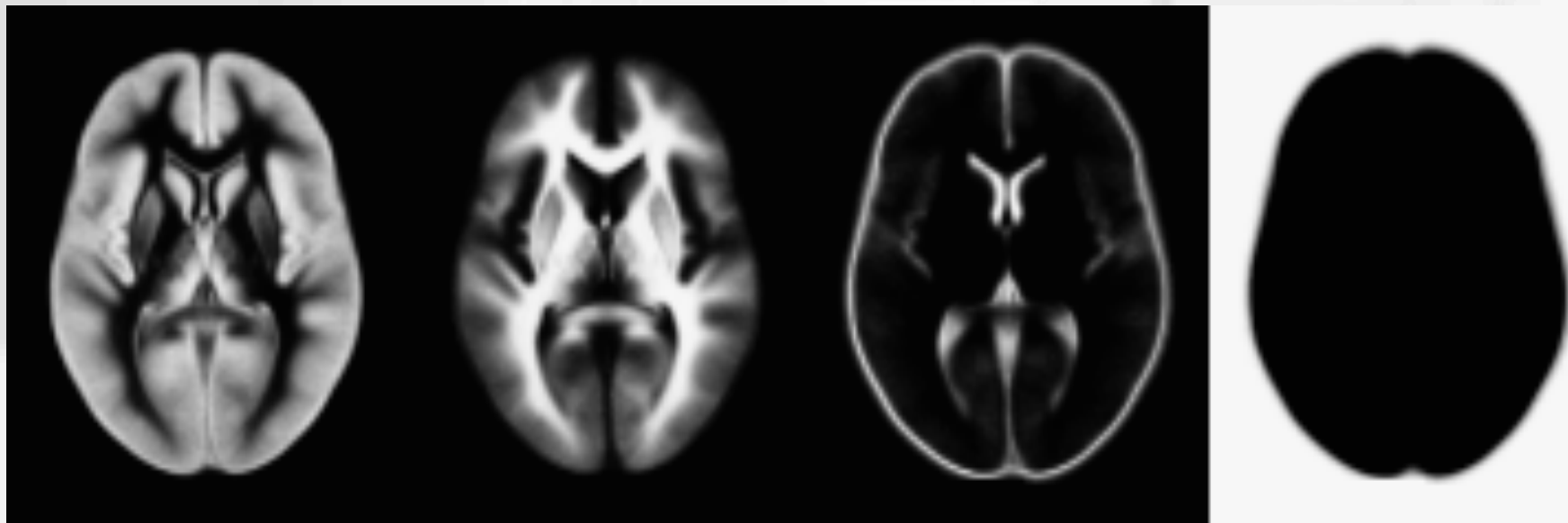
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Tissue probability maps (TPMs) are used as the prior, instead of the proportion of voxels in each class



ICBM Tissue Probabilistic Atlases. These tissue probability maps were kindly provided by the **International Consortium for Brain Mapping**

Deforming the Tissue Probability Maps



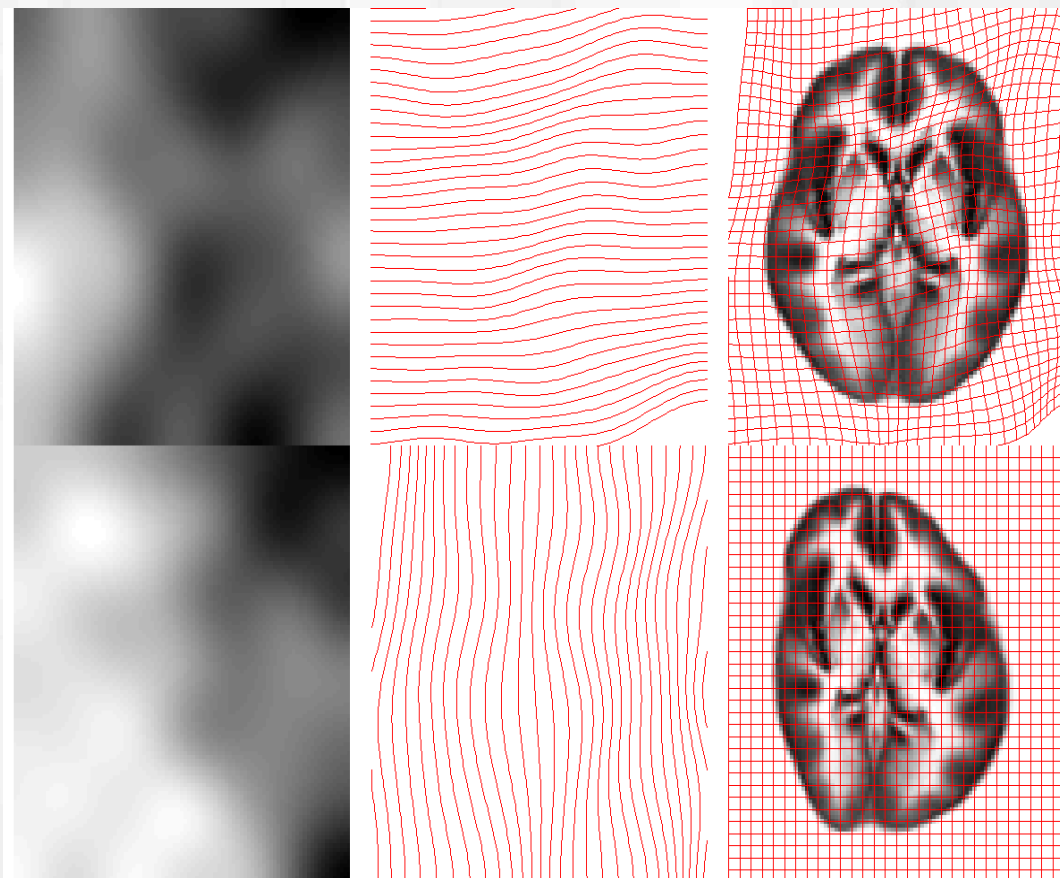
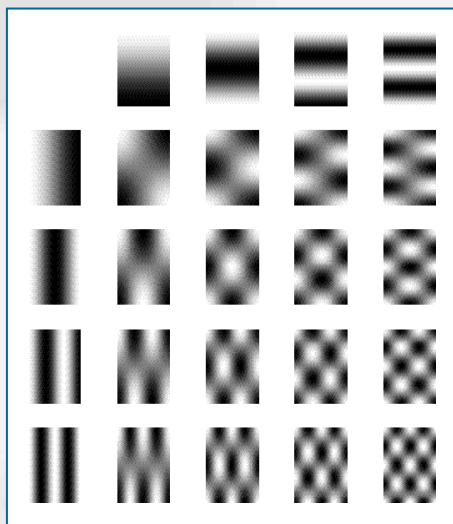
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Tissue probability map templates are warped to match the subject
- The inverse transform warps to the TPMs



Headers (Map Voxel-to-World)



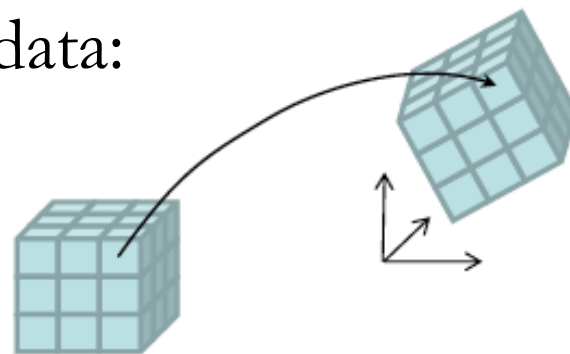
Setup Pipeline

Quality Monitoring

Multi-Subject

Noise Modeling

- Data:
 - 3D images are made up of voxels.
 - Voxel intensities are stored on disk as lists of numbers.
- Header: Meta-information about the data:
 - image dimensions (list to 3D)
 - “voxel-to-world mapping” (affine)
 - from: data coordinates (voxel column i , row j , slice k)
 - to: a real-world position (x,y,z mm) in a coordinate system (e.g. Scanner or MNI)
- SPM changes headers only where possible
 - to avoid interpolation in realign and co-register (estimate)
 - enforce rewriting the data (by interpolation) by “reslice”/“write”



E. Reslicing/Interpolation



Setup Pipeline

Quality Monitoring

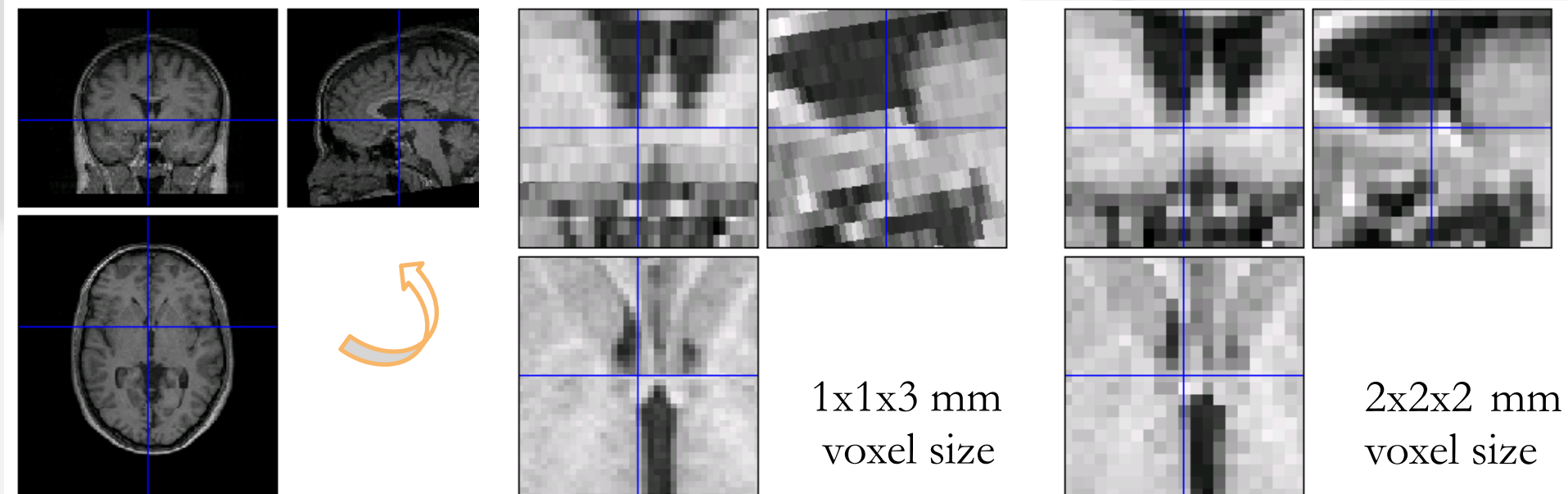
Multi-Subject

Noise Modeling

- Images have to be saved as voxel intensity list on disk
- After applying transformation parameters, data is re-sampled onto same grid of voxels as reference image

Reoriented

Resliced



- Always take care whether you specify a nifti-file (containing a 4D dataset, including different volumes) or a nifti-frame (one 3D volume)
 - file: `subject01/fmri/fmri.nii`
 - frame: `subject01/fmri/fmri.nii,1` (or `,15` etc.)
- The Batch Editor is not always consistent in what it needs, i.e. just the file name or a list of *all* individual frames
- This is one the most common error sources using preprocessing pipelines