

SPM5 Manual

The FIL Methods Group

Functional Imaging Laboratory
Wellcome Department of Imaging Neuroscience
Institute of Neurology, UCL
12 Queen Square, London WC1N 3BG, UK
November 30, 2005
<http://www.fil.ion.ucl.ac.uk/spm/>

Contents

I Temporal processing	13
1 Slice Timing	15
1.1 Data	16
1.1.1 Sessions	16
1.2 Number of Slices	16
1.3 TR	16
1.4 TA	16
1.5 Slice order	16
1.6 Reference Slice	16
II Spatial processing	17
2 Realign	19
2.1 Realign: Estimate	19
2.1.1 Data	19
2.1.2 Estimation Options	20
2.2 Realign: Reslice	21
2.2.1 Images	21
2.2.2 Reslice Options	21
2.3 Realign: Estimate & Reslice	21
2.3.1 Data	22
2.3.2 Estimation Options	22
2.3.3 Reslice Options	23
3 Realign & Unwarp	25
3.1 Data	27
3.1.1 Session	28
3.2 Estimation Options	28
3.2.1 Quality	28
3.2.2 Separation	28
3.2.3 Smoothing (FWHM)	28
3.2.4 Num Passes	28
3.2.5 Interpolation	29
3.2.6 Wrapping	29
3.2.7 Weighting	29
3.3 Unwarp Estimation Options	29
3.3.1 Basis Functions	29
3.3.2 Regularisation	29
3.3.3 Reg. Factor	29
3.3.4 Jacobian deformations	29
3.3.5 First-order effects	30
3.3.6 Second-order effects	30
3.3.7 Smoothing for unwarp (FWHM)	30
3.3.8 Re-estimate movement params	30

3.3.9	Number of Iterations	30
3.3.10	Taylor expansion point	30
3.4	Unwarp Reslicing Options	30
3.4.1	Reslices images (unwarp)?	30
3.4.2	Interpolation	31
3.4.3	Wrapping	31
3.4.4	Masking	31
4	Coreg	33
4.1	Coreg: Estimate	33
4.1.1	Reference Image	33
4.1.2	Source Image	34
4.1.3	Other Images	34
4.1.4	Estimation Options	34
4.2	Coreg: Reslice	34
4.2.1	Image Defining Space	34
4.2.2	Images to Reslice	34
4.2.3	Reslice Options	34
4.3	Coreg: Estimate & Reslice	35
4.3.1	Reference Image	35
4.3.2	Source Image	35
4.3.3	Other Images	35
4.3.4	Estimation Options	35
4.3.5	Reslice Options	36
5	Segment	37
5.1	Data	38
5.2	Output Files	38
5.2.1	Grey Matter	41
5.2.2	White Matter	41
5.2.3	Cerebro-Spinal Fluid	41
5.2.4	Bias Corrected	41
5.3	Custom	41
5.3.1	Tissue probability maps	41
5.3.2	Gaussians per class	42
5.3.3	Affine Regularisation	42
5.3.4	Warping Regularisation	42
5.3.5	Warp Frequency Cutoff	42
5.3.6	Bias regularisation	42
5.3.7	Bias FWHM	43
5.3.8	Sampling distance	43
6	Normalise	45
6.1	Normalise: Estimate	46
6.1.1	Data	46
6.1.2	Estimation Options	46
6.2	Normalise: Write	47
6.2.1	Data	47
6.2.2	Writing Options	48
6.3	Normalise: Estimate & Write	48
6.3.1	Data	48
6.3.2	Estimation Options	49
6.3.3	Writing Options	50
7	Smooth	51
7.1	Images to Smooth	51
7.2	FWHM	51

III fMRI Statistics	53
8 fMRI model specification	55
8.1 Timing parameters	55
8.1.1 Units for design	57
8.1.2 Interscan interval	57
8.1.3 Microtime resolution	57
8.1.4 Microtime onset	57
8.2 Data & Design	57
8.2.1 Subject/Session	57
8.3 Factorial design	60
8.3.1 Factor	61
8.4 Basis Functions	61
8.4.1 Canonical HRF	61
8.4.2 Other basis sets	61
8.5 Model Interactions (Volterra)	62
8.6 Directory	62
8.7 Global normalisation	62
8.8 Explicit mask	62
8.9 Serial correlations	63
8.10 Reviewing your design	63
9 fMRI model estimation	65
9.1 Select SPM.mat	65
9.2 Method	65
9.2.1 Classical	65
9.2.2 Bayesian 1st-level	67
9.2.3 Bayesian 2nd-level	70
9.3 Output files	70
9.3.1 Classical 1st-level	70
9.3.2 Bayesian 1st-level	70
9.4 Model comparison	71
IV EEG/MEG	73
10 SPM for EEG/MEG overview	75
11 EEG/MEG preprocessing — Tutorial	77
11.1 ERP analysis	77
11.1.1 Convert	77
11.1.2 Epoch	77
11.1.3 Filter	79
11.1.4 Downsample	79
11.1.5 Artefacts	79
11.1.6 Averaging	79
11.2 Other useful functions	79
11.2.1 Time-Frequency	79
11.2.2 Average TF	80
12 EEG/MEG preprocessing — Reference	81
12.1 Conversion of data	81
12.1.1 Converting data yourself	81
12.2 Converting CNT or BDF files	82
12.2.1 Syntax	82
12.3 BDF data	82
12.3.1 Syntax	82
12.4 CNT data	83

12.4.1 Syntax	83
12.5 The MEEG SPM format	83
12.5.1 channels	84
12.5.2 events	84
12.5.3 filter	85
12.5.4 threshold	85
12.6 Reading of data	85
12.6.1 Syntax	85
12.7 The channel template file	86
12.7.1 Structure	86
12.7.2 Creating your own channel template file	86
12.8 Epoching the data	87
12.8.1 Syntax	87
12.9 Filtering the data	87
12.9.1 Syntax	87
12.10 Artefact detection and rejection	88
12.10.1 Syntax	88
12.11 Downsampling	89
12.11.1 Syntax	89
12.12 Rereferencing	89
12.12.1 Syntax	89
12.13 Grand mean	90
12.13.1 Syntax	90
12.14 Merge	90
12.14.1 Syntax	90
12.15 Time-frequency decomposition	90
12.15.1 Syntax	91
12.16 Averaging	91
12.16.1 Syntax	91
12.17 Linear combinations of epochs	91
12.17.1 Syntax	91
12.18 Mixing of channels	92
12.18.1 Syntax	92
12.19 Weighting of the time-series	92
12.19.1 Syntax	92
12.20 Displaying data	93
13 3D source reconstruction: Imaging approach	95
13.1 Introduction	95
13.2 Data structure	96
13.3 Source space modeling (<i>mesh</i>)	96
13.4 Data Registration (<i>datareg</i>)	97
13.5 Forward computation (<i>forward</i>)	97
13.6 Inverse reconstruction (<i>inverse</i>)	98
14 M/EEG modelling and statistics	99
14.1 Preliminary remarks	99
14.2 How epoched time-series are analysed in SPM	99
14.3 1st level	100
14.3.1 The aim	100
14.3.2 Start	100
14.3.3 All options	100
14.3.4 How many factors?	101
14.3.5 Factor names and # of levels for factors	101
14.3.6 Select design component	101
14.3.7 Data	101
14.3.8 'Estimation'	101

14.3.9 Results	101
14.3.10 Display	102
14.3.11 ERP/ERF	102
14.3.12 Multiple subjects	102
14.4 2nd level models	102
14.4.1 All options	102
14.4.2 Factors	102
14.4.3 Design partitions and design components	102
14.4.4 Covariance components	103
14.4.5 Data	103
14.4.6 Estimation and Results	103
15 Equivalent current Dipole fitting	105
15.1 Necessary data	105
15.2 Model building	105
15.3 Dipole fitting	106
15.4 Result display	107
15.5 Preparing the *.pol files	107
16 Dynamic Causal Modelling for evoked responses	109
16.1 Calling DCM for ERP/ERF	109
16.2 cd, load and save	110
16.3 Data selection	110
16.4 Spatial model specification	110
16.5 Connectivity model specification	111
16.6 Estimation	111
16.7 Results	111
V Utilities	113
17 Display Image	115
17.1 Image to Display	116
18 Check Registration	119
18.1 Images to Display	119
19 Image Calculator	121
19.1 Input Images	121
19.2 Output Filename	121
19.3 Expression	121
19.4 Options	122
19.4.1 Data Matrix	122
19.4.2 Masking	122
19.4.3 Interpolation	122
19.4.4 Data Type	122
20 Create Rendering/Surface	123
20.1 Grey+white matter image	123
20.2 Output	123
21 DICOM Import	125
21.1 DICOM files	125
21.2 Output directory	125

22 MINC Import	127
22.1 MINC files	127
22.2 Options	127
22.2.1 Data Type	127
22.2.2 NIFTI Type	127
23 ECAT Import	129
23.1 ECAT files	129
23.2 Options	129
23.2.1 NIFTI Type	129
24 Execute Batch Jobs	131
24.1 Batch Files	131
25 Change Directory	133
25.1 Select a directory	133
26 Make Directory	135
26.1 Select a base directory	135
26.2 Enter a directory name	135
27 Deformations	137
27.1 Composition	137
27.1.1 Imported _sn.mat	137
27.1.2 Deformation Field	138
27.1.3 Inverse	138
27.1.4 Composition	140
27.2 Save as	142
27.3 Apply to	143
28 User Interface	145
28.1 Appearance	145
28.1.1 Background Colour 1	145
28.1.2 Background Colour 2	145
28.1.3 Foreground Colour	145
28.1.4 Font Size	145
28.2 Printing	145
VI Tools	147
29 Toolboxes	149
30 3D to 4D	151
30.1 3D Volumes	151
VII Data sets	153
31 Auditory fMRI data	155
31.1 Spatial pre-processing	155
31.1.1 Realignment	155
31.1.2 Coregistration	157
31.1.3 Segmentation	160
31.1.4 Normalize	162
31.1.5 Smoothing	162
31.2 Model specification, review and estimation	164
31.2.1 Estimate	164

31.3 Inference	167
31.3.1 Contrast manager	167
31.3.2 Masking	168
31.3.3 Thresholds	168
31.3.4 Files	168
31.3.5 Maximum Intensity Projections	169
31.3.6 Design matrix	169
31.3.7 Statistical tables	169
31.3.8 Plotting responses at a voxel	171
31.3.9 Overlays	172
31.3.10 Miscellaneous	174
31.4 Bayesian analysis	174
31.4.1 Specification	174
31.4.2 Estimation	175
31.4.3 Inference	175
32 Face fMRI data	179
32.1 Spatial pre-processing	179
32.1.1 Display	179
32.1.2 Realignment	179
32.1.3 Slice timing correction	183
32.1.4 Coregistration	183
32.1.5 Segmentation	186
32.1.6 Normalize	186
32.1.7 Smoothing	187
32.2 Modelling categorical responses	189
32.2.1 Estimate	190
32.2.2 Inference for categorical design	190
32.2.3 Statistical tables	193
32.2.4 F-contrasts	195
32.2.5 Plotting parameter estimates at a voxel	197
32.2.6 F-contrasts for testing effects of movement	198
32.3 Modelling parametric responses	198
32.3.1 Estimate	200
32.3.2 Plotting parametric responses	203
32.4 Bayesian analysis	204
32.4.1 Specification	204
32.4.2 Estimation	206
32.4.3 Inference	208

The SPM5 User Interface

Top Left Panel

The current list of jobs, which is represented as a tree-structure. Double-clicking can expand/contract items of the tree (marked with +/-) for visualisation. Items marked with X still require some values to be set before the job can be run, although an incompletely specified job can still be saved and loaded.

Top Right Panel

These are the options available for the currently highlighted item. Changing the list of jobs is done by clicking on an option in the menu. Items can be created, replicated or removed, allowing the processing stream to be modified. Values are also modified or entered via this panel. This is either by specifying values as text, selecting a menu option, or by file selection.

Centre Right Panel

This panel shows the current value of the highlighted item (where relevant).

Save, Load & Run

Jobs can be saved and loaded at a later time, either as XML or Matlab .mat files. The format depends on the extension you give the filename. XML files can be loaded into Matlab via "load-xml", modified, and saved again by "savexml", whereas "load" and "save" can be used for Matlab .mat files. Incomplete jobs can be loaded or saved, but the specification needs to be complete for a job to be run.

Bottom Panel

This panel provides information about the meaning of the current item.

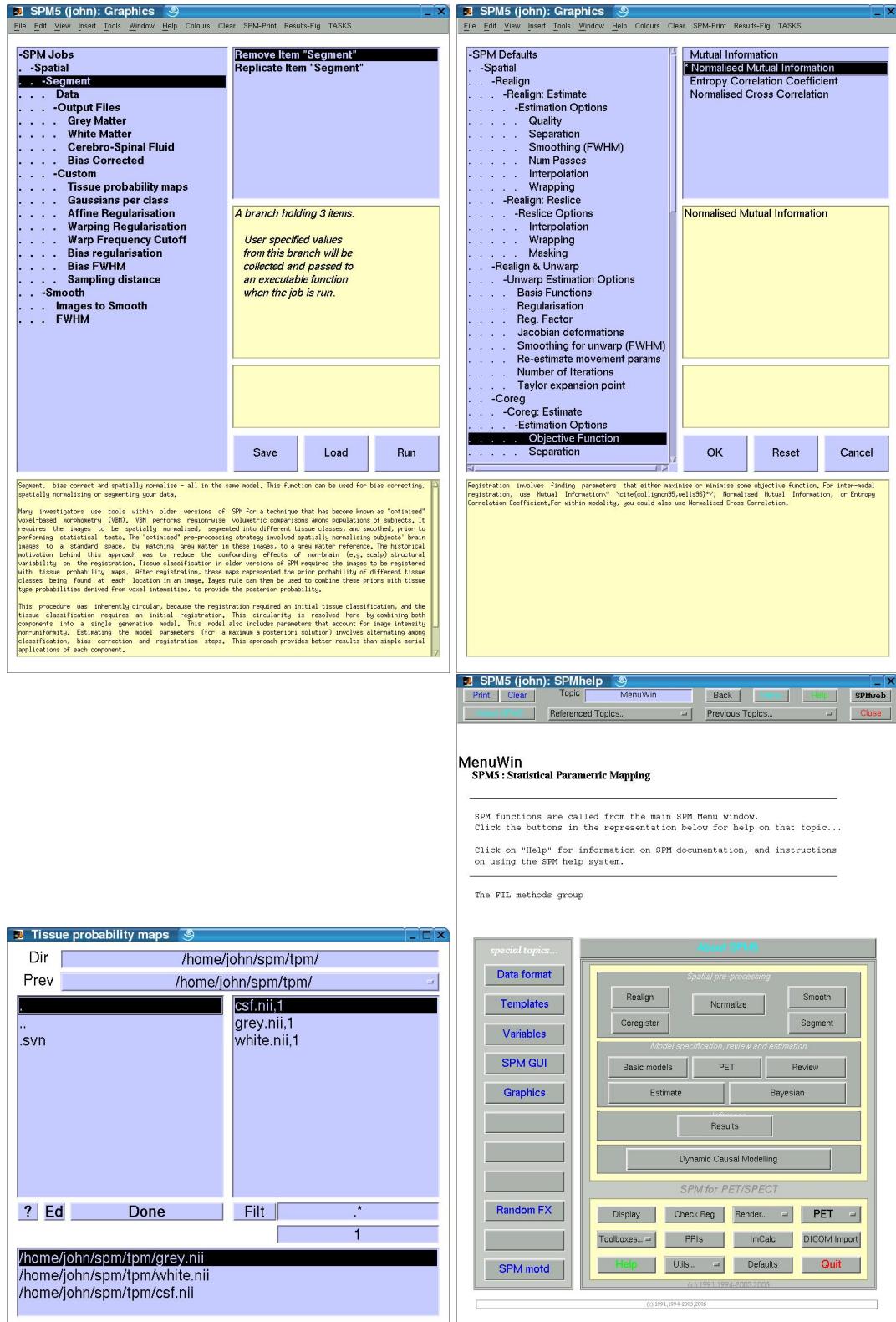


Figure 1: The SPM5 user interface. *Top left:* The usual user-interface. *Top right:* The Defaults user-interface. *Bottom left:* The file selector (click the (?) button for more information about filtering filenames, or selecting individual volumes within a 4D file). *Bottom right:* more online help can be obtained via the main help button.

Part I

Temporal processing

Chapter 1

Slice Timing

Contents

1.1	Data	16
1.1.1	Sessions	16
1.2	Number of Slices	16
1.3	TR	16
1.4	TA	16
1.5	Slice order	16
1.6	Reference Slice	16

Correct differences in image acquisition time between slices. Slice-time corrected files are prepended with an 'a'.

Note: The sliceorder arg that specifies slice acquisition order is a vector of N numbers, where N is the number of slices per volume. Each number refers to the position of a slice within the image file. The order of numbers within the vector is the temporal order in which those slices were acquired. To check the order of slices within an image file, use the SPM Display option and move the crosshairs to a voxel co-ordinate of z=1. This corresponds to a point in the first slice of the volume.

The function corrects differences in slice acquisition times. This routine is intended to correct for the staggered order of slice acquisition that is used during echoplanar scanning. The correction is necessary to make the data on each slice correspond to the same point in time. Without correction, the data on one slice will represent a point in time as far removed as 1/2 the TR from an adjacent slice (in the case of an interleaved sequence).

This routine "shifts" a signal in time to provide an output vector that represents the same (continuous) signal sampled starting either later or earlier. This is accomplished by a simple shift of the phase of the sines that make up the signal. Recall that a Fourier transform allows for a representation of any signal as the linear combination of sinusoids of different frequencies and phases. Effectively, we will add a constant to the phase of every frequency, shifting the data in time.

Shifter - This is the filter by which the signal will be convolved to introduce the phase shift. It is constructed explicitly in the Fourier domain. In the time domain, it may be described as an impulse (delta function) that has been shifted in time the amount described by TimeShift. The correction works by lagging (shifting forward) the time-series data on each slice using sinc-interpolation. This results in each time series having the values that would have been obtained had the slice been acquired at the same time as the reference slice. To make this clear, consider a neural event (and ensuing hemodynamic response) that occurs simultaneously on two adjacent slices. Values from slice "A" are acquired starting at time zero, simultaneous to the neural event, while values from slice "B" are acquired one second later. Without corection, the "B" values will

describe a hemodynamic response that will appear to have began one second EARLIER on the "B" slice than on slice "A". To correct for this, the "B" values need to be shifted towards the Right, i.e., towards the last value.

This correction assumes that the data are band-limited (i.e. there is no meaningful information present in the data at a frequency higher than that of the Nyquist). This assumption is supported by the study of Josephs et al (1997, NeuroImage) that obtained event-related data at an effective TR of 166 msec. No physio-logical signal change was present at frequencies higher than our typical Nyquist (0.25 HZ).

Written by Darren Gitelman at Northwestern U., 1998. Based (in large part) on ACQCOR-RECT.PRO from Geoff Aguirre and Eric Zarahn at U. Penn.

1.1 Data

Subjects or sessions. The same parameters specified below will be applied to all sessions.

1.1.1 Sessions

Select images to acquisition correct.

1.2 Number of Slices

Enter the number of slices

1.3 TR

Enter the TR in seconds

1.4 TA

The TA (in secs) must be entered by the user. It is usually calculated as TR-(TR/nslices). You can simply enter this equation with the variables replaced by appropriate numbers.

1.5 Slice order

Enter the slice order. Bottom slice = 1. Sequence types and examples of code to enter are given below.

```

ascending (first slice=bottom): [1:1:nslices]
descending (first slice=top): [nslices:-1:1]
interleaved (middle-top):
for k = 1:nslices,
round((nslices-k)/2 + (rem((nslices-k),2) * (nslices - 1)/2)) + 1,
end
interleaved (bottom -> up): [1:2:nslices 2:2:nslices]
interleaved (top -> down): [nslices:-2:1, nslices-1:-2:1]
```

1.6 Reference Slice

Enter the reference slice

Part II

Spatial processing

Chapter 2

Realign

Contents

2.1	Realign: Estimate	19
2.1.1	Data	19
2.1.2	Estimation Options	20
2.2	Realign: Reslice	21
2.2.1	Images	21
2.2.2	Reslice Options	21
2.3	Realign: Estimate & Reslice	21
2.3.1	Data	22
2.3.2	Estimation Options	22
2.3.3	Reslice Options	23

Within-subject registration of image time series.

2.1 Realign: Estimate

This routine realigns a time-series of images acquired from the same subject using a least squares approach and a 6 parameter (rigid body) spatial transformation [19]. The first image in the list specified by the user is used as a reference to which all subsequent scans are realigned. The reference scan does not have to be the first chronologically and it may be wise to chose a "representative scan" in this role.

The aim is primarily to remove movement artefact in fMRI and PET time-series (or more generally longitudinal studies). The headers are modified for each of the input images, such that. they reflect the relative orientations of the data. The details of the transformation are displayed in the results window as plots of translation and rotation. A set of realignment parameters are saved for each session, named rp_*.txt. These can be modelled as confounds within the general linear model [19].

2.1.1 Data

Add new sessions for this subject. In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

Session

Select scans for this session. In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

2.1.2 Estimation Options

Various registration options. If in doubt, simply keep the default values.

Quality

Quality versus speed trade-off. Highest quality (1) gives most precise results, whereas lower qualities give faster realignment. The idea is that some voxels contribute little to the estimation of the realignment parameters. This parameter is involved in selecting the number of voxels that are used.

Separation

The separation (in mm) between the points sampled in the reference image. Smaller sampling distances give more accurate results, but will be slower.

Smoothing (FWHM)

The FWHM of the Gaussian smoothing kernel (mm) applied to the images before estimating the realignment parameters.

- * PET images typically use a 7 mm kernel.
- * MRI images typically use a 5 mm kernel.

Num Passes

Register to first: Images are registered to the first image in the series. Register to mean: A two pass procedure is used in order to register the images to the mean of the images after the first realignment.

PET images are typically registered to the mean. This is because PET data are more noisy than fMRI and there are fewer of them, so time is less of an issue.

MRI images are typically registered to the first image. The more accurate way would be to use a two pass procedure, but this probably wouldn't improve the results so much and would take twice as long to run.

Interpolation

The method by which the images are sampled when estimating the optimum transformation. Higher degree interpolation methods provide better interpolation, but they are slower because they use more neighbouring voxels [44, 45, 46].

Wrapping

These are typically:

- No wrapping - for PET or images that have already been spatially transformed.
- Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Weighting

The option of providing a weighting image to weight each voxel of the reference image differently when estimating the realignment parameters. The weights are proportional to the inverses of the standard deviations. For example, when there is a lot of extra-brain motion - e.g., during speech, or when there are serious artifacts in a particular region of the images.

2.2 Realign: Reslice

This function reslices a series of registered images such that they match the first image selected voxel-for-voxel. The resliced images are named the same as the originals, except that they are prefixed by 'r'.

2.2.1 Images

Select scans to reslice to match the first.

2.2.2 Reslice Options

Various reslicing options. If in doubt, simply keep the default values.

Resliced images

All Images (1..n) : This reslices all the images - including the first image selected - which will remain in its original position.

Images 2..n : Reslices images 2..n only. Useful for if you wish to reslice (for example) a PET image to fit a structural MRI, without creating a second identical MRI volume.

All Images + Mean Image : In addition to reslicing the images, it also creates a mean of the resliced image.

Mean Image Only : Creates the mean resliced image only.

Interpolation

The method by which the images are sampled when being written in a different space. Nearest Neighbour is fastest, but not recommended for image realignment. Bilinear Interpolation is probably OK for PET, but not so suitable for fMRI because higher degree interpolation generally gives better results [44, 45, 46]. Although higher degree methods provide better interpolation, but they are slower because they use more neighbouring voxels. Fourier Interpolation [17, 14] is another option, but note that it is only implemented for purely rigid body transformations. Voxel sizes must all be identical and isotropic.

Wrapping

These are typically:

No wrapping - for PET or images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

2.3 Realign: Estimate & Reslice

This routine realigns a time-series of images acquired from the same subject using a least squares approach and a 6 parameter (rigid body)spatial transformation [19]. The first image in the list specified by the user is used as a reference to which all subsequent scans are realigned. The reference scan does not have to be the first chronologically and it may be wise to chose a "representative scan" in this role.

The aim is primarily to remove movement artefact in fMRI and PET time-series (or more generally longitudinal studies) [3]. The headers are modified for each of the input images, such that. they reflect the relative orientations of the data. The details of the transformation are displayed

in the results window as plots of translation and rotation. A set of realignment parameters are saved for each session, named rp_*.txt. After realignment, the images are resliced such that they match the first image selected voxel-for-voxel. The resliced images are named the same as the originals, except that they are prefixed by 'r'.

2.3.1 Data

Add new sessions for this subject. In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

Session

Select scans for this session. In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

2.3.2 Estimation Options

Various registration options. If in doubt, simply keep the default values.

Quality

Quality versus speed trade-off. Highest quality (1) gives most precise results, whereas lower qualities give faster realignment. The idea is that some voxels contribute little to the estimation of the realignment parameters. This parameter is involved in selecting the number of voxels that are used.

Separation

The separation (in mm) between the points sampled in the reference image. Smaller sampling distances give more accurate results, but will be slower.

Smoothing (FWHM)

The FWHM of the Gaussian smoothing kernel (mm) applied to the images before estimating the realignment parameters.

- * PET images typically use a 7 mm kernel.
- * MRI images typically use a 5 mm kernel.

Num Passes

Register to first: Images are registered to the first image in the series. Register to mean: A two pass procedure is used in order to register the images to the mean of the images after the first realignment.

PET images are typically registered to the mean. This is because PET data are more noisy than fMRI and there are fewer of them, so time is less of an issue.

MRI images are typically registered to the first image. The more accurate way would be to use a two pass procedure, but this probably wouldn't improve the results so much and would take twice as long to run.

Interpolation

The method by which the images are sampled when estimating the optimum transformation. Higher degree interpolation methods provide the better interpolation, but they are slower because they use more neighbouring voxels [44, 45, 46].

Wrapping

These are typically:

No wrapping - for PET or images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Weighting

The option of providing a weighting image to weight each voxel of the reference image differently when estimating the realignment parameters. The weights are proportional to the inverses of the standard deviations. For example, when there is a lot of extra-brain motion - e.g., during speech, or when there are serious artifacts in a particular region of the images.

2.3.3 Reslice Options

Various reslicing options. If in doubt, simply keep the default values.

Resliced images

All Images (1..n) : This reslices all the images - including the first image selected - which will remain in its original position.

Images 2..n : Reslices images 2..n only. Useful for if you wish to reslice (for example) a PET image to fit a structural MRI, without creating a second identical MRI volume.

All Images + Mean Image : In addition to reslicing the images, it also creates a mean of the resliced image.

Mean Image Only : Creates the mean resliced image only.

Interpolation

The method by which the images are sampled when being written in a different space. Nearest Neighbour is fastest, but not recommended for image realignment. Bilinear Interpolation is probably OK for PET, but not so suitable for fMRI because higher degree interpolation generally gives better results [44, 45, 46]. Although higher degree methods provide better interpolation, but they are slower because they use more neighbouring voxels. Fourier Interpolation [17, 14] is another option, but note that it is only implemented for purely rigid body transformations. Voxel sizes must all be identical and isotropic.

Wrapping

These are typically:

No wrapping - for PET or images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

Chapter 3

Realign & Unwarp

Contents

3.1 Data	27
3.1.1 Session	28
3.2 Estimation Options	28
3.2.1 Quality	28
3.2.2 Separation	28
3.2.3 Smoothing (FWHM)	28
3.2.4 Num Passes	28
3.2.5 Interpolation	29
3.2.6 Wrapping	29
3.2.7 Weighting	29
3.3 Unwarp Estimation Options	29
3.3.1 Basis Functions	29
3.3.2 Regularisation	29
3.3.3 Reg. Factor	29
3.3.4 Jacobian deformations	29
3.3.5 First-order effects	30
3.3.6 Second-order effects	30
3.3.7 Smoothing for unwarp (FWHM)	30
3.3.8 Re-estimate movement params	30
3.3.9 Number of Iterations	30
3.3.10 Taylor expansion point	30
3.4 Unwarp Reslicing Options	30
3.4.1 Reslices images (unwarp)?	30
3.4.2 Interpolation	31
3.4.3 Wrapping	31
3.4.4 Masking	31

Within-subject registration and unwarping of time series.

The realignment part of this routine realigns a time-series of images acquired from the same subject using a least squares approach and a 6 parameter (rigid body) spatial transformation. The first image in the list specified by the user is used as a reference to which all subsequent scans are realigned. The reference scan does not have to be the first chronologically and it may be wise to choose a "representative scan" in this role.

The aim is primarily to remove movement artefact in fMRI and PET time-series (or more generally longitudinal studies). ".mat" files are written for each of the input images. The details

of the transformation are displayed in the results window as plots of translation and rotation. A set of realignment parameters are saved for each session, named rp_*.txt.

In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

The paper [2] is unfortunately a bit old now and describes none of the newer features. Hopefully we'll have a second paper out any decade now.

See also spm_uw_estimate.m for a detailed description of the implementation. Even after realignment there is considerable variance in fMRI time series that covary with, and is most probably caused by, subject movements [2]. It is also the case that this variance is typically large compared to experimentally induced variance. Anyone interested can include the estimated movement parameters as covariates in the design matrix, and take a look at an F-contrast encompassing those columns. It is quite dramatic. The result is loss of sensitivity, and if movements are correlated to task specificity. I.e. we may mistake movement induced variance for true activations. The problem is well known, and several solutions have been suggested. A quite pragmatic (and conservative) solution is to include the estimated movement parameters (and possibly squared) as covariates in the design matrix. Since we typically have loads of degrees of freedom in fMRI we can usually afford this. The problems occur when movements are correlated with the task, since the strategy above will discard "good" and "bad" variance alike (i.e. remove also "true" activations).

The "covariate" strategy described above was predicated on a model where variance was assumed to be caused by "spin history" effects, but will work pretty much equally good/bad regardless of what the true underlying cause is. Others have assumed that the residual variance is caused mainly by errors introduced by the interpolation kernel in the resampling step of the realignment. One has tried to solve this through higher order resampling (huge Sinc kernels, or k-space resampling). Unwarp is based on a different hypothesis regarding the residual variance. EPI images are not particularly faithful reproductions of the object, and in particular there are severe geometric distortions in regions where there is an air-tissue interface (e.g. orbitofrontal cortex and the anterior medial temporal lobes). In these areas in particular the observed image is a severely warped version of reality, much like a funny mirror at a fair ground. When one moves in front of such a mirror ones image will distort in different ways and ones head may change from very elongated to seriously flattened. If we were to take digital snapshots of the reflection at these different positions it is rather obvious that realignment will not suffice to bring them into a common space.

The situation is similar with EPI images, and an image collected for a given subject position will not be identical to that collected at another. We call this effect susceptibility-by-movement interaction. Unwarp is predicated on the assumption that the susceptibility-by- movement interaction is responsible for a sizable part of residual movement related variance.

Assume that we know how the deformations change when the subject changes position (i.e. we know the derivatives of the deformations with respect to subject position). That means that for a given time series and a given set of subject movements we should be able to predict the "shape changes" in the object and the ensuing variance in the time series. It also means that, in principle, we should be able to formulate the inverse problem, i.e. given the observed variance (after realignment) and known (estimated) movements we should be able to estimate how deformations change with subject movement. We have made an attempt at formulating such an inverse model, and at solving for the "derivative fields". A deformation field can be thought of as little vectors at each position in space showing how that particular location has been deflected. A "derivative field" is then the rate of change of those vectors with respect to subject movement. Given these "derivative fields" we should be able to remove the variance caused by the susceptibility-by-movement interaction. Since the underlying model is so restricted we would also expect experimentally induced variance to be preserved. Our experiments have also shown this to be true.

In theory it should be possible to estimate also the "static" deformation field, yielding an unwarped (to some true geometry) version of the time series. In practise that doesn't really seem to work. Hence, the method deals only with residual movement related variance induced by the

susceptibility-by-movement interaction. This means that the time-series will be undistorted to some "average distortion" state rather than to the true geometry. If one wants additionally to address the issue of anatomical fidelity one should combine Unwarp with a measured fieldmap.

The description above can be thought of in terms of a Taylor expansion of the field as a function of subject movement. Unwarp alone will estimate the first (and optionally second, see below) order terms of this expansion. It cannot estimate the zeroth order term (the distortions common to all scans in the time series) since that doesn't introduce (almost) any variance in the time series. The measured fieldmap takes the role of the zeroth order term. Refer to the FieldMap toolbox and the documents FieldMap.man and FieldMap_principles.man for a description of how to obtain fieldmaps in the format expected by Unwarp.

If we think of the field as a function of subject movement it should in principle be a function of six variables since rigid body movement has six degrees of freedom. However, the physics of the problem tells us that the field should not depend on translations nor on rotation in a plane perpendicular to the magnetic flux. Hence it should in principle be sufficient to model the field as a function of out-of-plane rotations (i.e. pitch and roll). One can object to this in terms of the effects of shimming (object no longer immersed in a homogenous field) that introduces a dependence on all movement parameters. In addition SPM/Unwarp cannot really tell if the transversal slices it is being passed are really perpendicular to the flux or not. In practice it turns out thought that it is never (at least we haven't seen any case) necessary to include more than Pitch and Roll. This is probably because the individual movement parameters are typically highly correlated anyway, which in turn is probably because most heads that we scan are attached to a neck around which rotations occur. On the subject of Taylor expansion we should mention that there is the option to use a second-order expansion (through the defaults) interface. This implies estimating also the rate-of-change w.r.t. to some movement parameter of the rate-of-change of the field w.r.t. some movement parameter (colloquially known as a second derivative). It can be quite interesting to watch (and it is amazing that it is possible) but rarely helpful/necessary.

In the defaults there is also an option to include Jacobian intensity modulation when estimating the fields. "Jacobian intensity modulation" refers to the dilution/concentration of intensity that ensue as a consequence of the distortions. Think of a semi-transparent coloured rubber sheet that you hold against a white background. If you stretch a part of the sheet (induce distortions) you will see the colour fading in that particular area. In theory it is a brilliant idea to include also these effects when estimating the field (see e.g. Andersson et al, NeuroImage 20:870-888). In practice for this specific problem it is NOT a good idea.

It should be noted that this is a method intended to correct data afflicted by a particular problem. If there is little movement in your data to begin with this method will do you little good. If on the other hand there is appreciable movement in your data ($>1\text{deg}$) it will remove some of that unwanted variance. If, in addition, movements are task related it will do so without removing all your "true" activations. The method attempts to minimise total (across the image volume) variance in the data set. It should be realised that while (for small movements) a rather limited portion of the total variance is removed, the susceptibility-by-movement interaction effects are quite localised to "problem" areas. Hence, for a subset of voxels in e.g. frontal-medial and orbitofrontal cortices and parts of the temporal lobes the reduction can be quite dramatic (>90). The advantages of using Unwarp will also depend strongly on the specifics of the scanner and sequence by which your data has been acquired. When using the latest generation scanners distortions are typically quite small, and distortion-by-movement interactions consequently even smaller. A small check list in terms of distortions is

- a) Fast gradients->short read-out time->small distortions
- b) Low field (i.e. $<3\text{T}$)->small field changes->small distortions
- c) Low res (64x64)->short read-out time->small distortions
- d) SENSE/SMASH->short read-out time->small distortions

If you can tick off all points above chances are you have minimal distortions to begin with and you can say "sod Unwarp" (but not to our faces!).

3.1 Data

Data sessions to unwarp.

3.1.1 Session

Only add similar session data to a realign+unwarp branch, i.e., choose Data or Data+phase map for all sessions, but don't use them interchangably.

In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

Images

Select scans for this session.

In the coregistration step, the sessions are first realigned to each other, by aligning the first scan from each session to the first scan of the first session. Then the images within each session are aligned to the first image of the session. The parameter estimation is performed this way because it is assumed (rightly or not) that there may be systematic differences in the images between sessions.

Phase map (vdm* file)

Select precalculated phase map, or leave empty for no phase correction. The vdm* file is assumed to be already in alignment with the first scan of the first session.

3.2 Estimation Options

Various registration options that could be modified to improve the results. Whenever possible, the authors of SPM try to choose reasonable settings, but sometimes they can be improved.

3.2.1 Quality

Quality versus speed trade-off. Highest quality (1) gives most precise results, whereas lower qualities gives faster realignment. The idea is that some voxels contribute little to the estimation of the realignment parameters. This parameter is involved in selecting the number of voxels that are used.

3.2.2 Separation

The separation (in mm) between the points sampled in the reference image. Smaller sampling distances gives more accurate results, but will be slower.

3.2.3 Smoothing (FWHM)

The FWHM of the Gaussian smoothing kernel (mm) applied to the images before estimating the realignment parameters.

* PET images typically use a 7 mm kernel.

* MRI images typically use a 5 mm kernel.

3.2.4 Num Passes

Register to first: Images are registered to the first image in the series. Register to mean: A two pass procedure is used in order to register the images to the mean of the images after the first realignment.

* PET images are typically registered to the mean.

* MRI images are typically registered to the first image.

3.2.5 Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour - Fastest, but not normally recommended.

Bilinear Interpolation - OK for PET, or realigned fMRI.

B-spline Interpolation/* [44] - Better quality (but slower) interpolation, especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

3.2.6 Wrapping

These are typically:

No wrapping - for images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

3.2.7 Weighting

The option of providing a weighting image to weight each voxel of the reference image differently when estimating the realignment parameters. The weights are proportional to the inverses of the standard deviations. For example, when there is a lot of extra-brain motion - e.g., during speech, or when there are serious artifacts in a particular region of the images.

3.3 Unwarp Estimation Options

Various registration & unwarping estimation options.

3.3.1 Basis Functions

Number of basis functions to use for each dimension. If the third dimension is left out, the order for that dimension is calculated to yield a roughly equal spatial cut-off in all directions. Default: [12 12 *]

3.3.2 Regularisation

Unwarps looks for the solution that maximises the likelihood (minimises the variance) while simultaneously maximising the smoothness of the estimated field (c.f. Lagrange multipliers). This parameter determines how to balance the compromise between these (i.e. the value of the multiplier). Test it on your own data (if you can be bothered) or go with the defaults.

Regularisation of derivative fields is based on the regorder'th (spatial) derivative of the field. The choices are 0, 1, 2, or 3. Default: 1

3.3.3 Reg. Factor

Regularisation factor. Default: Medium.

3.3.4 Jacobian deformations

In the defaults there is also an option to include Jacobian intensity modulation when estimating the fields. "Jacobian intensity modulation" refers to the dilution/concentration of intensity that ensue as a consequence of the distortions. Think of a semi-transparent coloured rubber sheet that you hold against a white background. If you stretch a part of the sheet (induce distortions) you will see the colour fading in that particular area. In theory it is a brilliant idea to include also these effects when estimating the field (see e.g. Andersson et al, NeuroImage 20:870-888). In practice for this specific problem it is NOT a good idea. Default: No

3.3.5 First-order effects

Theoretically (ignoring effects of shimming) one would expect the field to depend only on subject out-of-plane rotations. Hence the default choice ("Pitch and Roll", i.e., [4 5]). Go with that unless you have very good reasons to do otherwise

Vector of first order effects to model. Movements to be modelled are referred to by number. 1= x translation; 2= y translation; 3= z translation 4 = x rotation, 5 = y rotation and 6 = z rotation.

To model pitch & roll enter: [4 5]

To model all movements enter: [1:6]

Otherwise enter a customized set of movements to model

3.3.6 Second-order effects

List of second order terms to model second derivatives of. This is entered as a vector of movement parameters similar to first order effects, or leave blank for NONE

Movements to be modelled are referred to by number:

1= x translation; 2= y translation; 3= z translation 4 = x rotation, 5 = y rotation and 6 = z rotation.

To model the interaction of pitch & roll enter: [4 5]

To model all movements enter: [1:6]

The vector will be expanded into an n x 2 matrix of effects. For example [4 5] will be expanded to:

$$\begin{bmatrix} 4 & 4 \\ 4 & 5 \\ 5 & 5 \end{bmatrix}$$

3.3.7 Smoothing for unwarp (FWHM)

FWHM (mm) of smoothing filter applied to images prior to estimation of deformation fields.

3.3.8 Re-estimate movement params

Re-estimation means that movement-parameters should be re-estimated at each unwarping iteration. Default: Yes.

3.3.9 Number of Iterations

Maximum number of iterations. Default: 5.

3.3.10 Taylor expansion point

Point in position space to perform Taylor-expansion around. Choices are ('First', 'Last' or 'Average'). 'Average' should (in principle) give the best variance reduction. If a field-map acquired before the time-series is supplied then expansion around the 'First' MIGHT give a slightly better average geometric fidelity.

3.4 Unwarp Reslicing Options

Various registration & unwarping estimation options.

3.4.1 Reslices images (unwarp)?

All Images (1..n)

This reslices and unwarps all the images.

All Images + Mean Image

In addition to reslicing the images, it also creates a mean of the resliced images.

3.4.2 Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour - Fastest, but not normally recommended.

Bilinear Interpolation - OK for PET, or realigned fMRI. B-spline Interpolation[44]

- Better quality (but slower) interpolation, especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

3.4.3 Wrapping

These are typically:

No wrapping - for PET or images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

3.4.4 Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

Chapter 4

Coreg

Contents

4.1	Coreg: Estimate	33
4.1.1	Reference Image	33
4.1.2	Source Image	34
4.1.3	Other Images	34
4.1.4	Estimation Options	34
4.2	Coreg: Reslice	34
4.2.1	Image Defining Space	34
4.2.2	Images to Reslice	34
4.2.3	Reslice Options	34
4.3	Coreg: Estimate & Reslice	35
4.3.1	Reference Image	35
4.3.2	Source Image	35
4.3.3	Other Images	35
4.3.4	Estimation Options	35
4.3.5	Reslice Options	36

Within-subject registration using a rigid-body model. A rigid-body transformation (in 3D) can be parameterised by three translations and three rotations about the different axes.

You get the options of estimating the transformation, reslicing images according to some rigid-body transformations, or estimating and applying rigid-body transformations.

4.1 Coreg: Estimate

The registration method used here is based on work by Collignon et al [13]. The original interpolation method described in this paper has been changed in order to give a smoother cost function. The images are also smoothed slightly, as is the histogram. This is all in order to make the cost function as smooth as possible, to give faster convergence and less chance of local minima.

At the end of coregistration, the voxel-to-voxel affine transformation matrix is displayed, along with the histograms for the images in the original orientations, and the final orientations. The registered images are displayed at the bottom.

Registration parameters are stored in the headers of the "source" and the "other" images.

4.1.1 Reference Image

This is the image that is assumed to remain stationary (sometimes known as the target or template image), while the source image is moved to match it.

4.1.2 Source Image

This is the image that is jiggled about to best match the reference.

4.1.3 Other Images

These are any images that need to remain in alignment with the source image.

4.1.4 Estimation Options

Various registration options, which are passed to the Powell optimisation algorithm [41].

Objective Function

Registration involves finding parameters that either maximise or minimise some objective function. For inter-modal registration, use Mutual Information [13, 47]*, Normalised Mutual Information [43], or Entropy Correlation Coefficient [34]. For within modality, you could also use Normalised Cross Correlation.

Separation

The average distance between sampled points (in mm). Can be a vector to allow a coarse registration followed by increasingly fine ones.

Tolerances

The accuracy for each parameter. Iterations stop when differences between successive estimates are less than the required tolerance.

Histogram Smoothing

Gaussian smoothing to apply to the 256x256 joint histogram. Other information theoretic coregistration methods use fewer bins, but Gaussian smoothing seems to be more elegant.

4.2 Coreg: Reslice

Reslice images to match voxel-for-voxel with an image defining some space. The resliced images are named the same as the originals except that they are prefixed by 'r'.

4.2.1 Image Defining Space

This is analogous to the reference image. Images are resliced to match this image (providing they have been coregistered first).

4.2.2 Images to Reslice

These images are resliced to the same dimensions, voxel sizes, orientation etc as the space defining image.

4.2.3 Reslice Options

Various reslicing options.

Interpolation

The method by which the images are sampled when being written in a different space. Nearest Neighbour is fastest, but not normally recommended. It can be useful for reorienting images while preserving the original intensities (e.g. an image consisting of labels). Bilinear Interpolation is OK for PET, or realigned and resliced fMRI. If subject movement (from an fMRI time series) is included in the transformations then it may be better to use a higher degree approach. Note that higher degree B-spline interpolation [44, 45, 46] is slower because it uses more neighbours.

Wrapping

These are typically:

No wrapping - for PET or images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

4.3 Coreg: Estimate & Reslice

The registration method used here is based on work by Collignon et al [13]. The original interpolation method described in this paper has been changed in order to give a smoother cost function. The images are also smoothed slightly, as is the histogram. This is all in order to make the cost function as smooth as possible, to give faster convergence and less chance of local minima.

At the end of coregistration, the voxel-to-voxel affine transformation matrix is displayed, along with the histograms for the images in the original orientations, and the final orientations. The registered images are displayed at the bottom.

Registration parameters are stored in the headers of the "source" and the "other" images. These images are also resliced to match the source image voxel-for-voxel. The resliced images are named the same as the originals except that they are prefixed by 'r'.

4.3.1 Reference Image

This is the image that is assumed to remain stationary (sometimes known as the target or template image), while the source image is moved to match it.

4.3.2 Source Image

This is the image that is jiggled about to best match the reference.

4.3.3 Other Images

These are any images that need to remain in alignment with the source image.

4.3.4 Estimation Options

Various registration options, which are passed to the Powell optimisation algorithm [41].

Objective Function

Registration involves finding parameters that either maximise or minimise some objective function. For inter-modal registration, use Mutual Information [13, 47]*/, Normalised Mutual Information [43], or Entropy Correlation Coefficient [34]. For within modality, you could also use Normalised Cross Correlation.

Separation

The average distance between sampled points (in mm). Can be a vector to allow a coarse registration followed by increasingly fine ones.

Tolerances

The accuracy for each parameter. Iterations stop when differences between successive estimates are less than the required tolerance.

Histogram Smoothing

Gaussian smoothing to apply to the 256x256 joint histogram. Other information theoretic coregistration methods use fewer bins, but Gaussian smoothing seems to be more elegant.

4.3.5 Reslice Options

Various reslicing options.

Interpolation

The method by which the images are sampled when being written in a different space. Nearest Neighbour is fastest, but not normally recommended. It can be useful for reorienting images while preserving the original intensities (e.g. an image consisting of labels). Bilinear Interpolation is OK for PET, or realigned and resliced fMRI. If subject movement (from an fMRI time series) is included in the transformations then it may be better to use a higher degree approach. Note that higher degree B-spline interpolation [44, 45, 46] is slower because it uses more neighbours.

Wrapping

These are typically:

No wrapping - for PET or images that have already been spatially transformed.

Wrap in Y - for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Masking

Because of subject motion, different images are likely to have different patterns of zeros from where it was not possible to sample data. With masking enabled, the program searches through the whole time series looking for voxels which need to be sampled from outside the original images. Where this occurs, that voxel is set to zero for the whole set of images (unless the image format can represent NaN, in which case NaNs are used where possible).

Chapter 5

Segment

Contents

5.1	Data	38
5.2	Output Files	38
5.2.1	Grey Matter	41
5.2.2	White Matter	41
5.2.3	Cerebro-Spinal Fluid	41
5.2.4	Bias Corrected	41
5.3	Custom	41
5.3.1	Tissue probability maps	41
5.3.2	Gaussians per class	42
5.3.3	Affine Regularisation	42
5.3.4	Warping Regularisation	42
5.3.5	Warp Frequency Cutoff	42
5.3.6	Bias regularisation	42
5.3.7	Bias FWHM	43
5.3.8	Sampling distance	43

Segment, bias correct and spatially normalise - all in the same model [7]. This function can be used for bias correcting, spatially normalising or segmenting your data.

Many investigators use tools within older versions of SPM for a technique that has become known as "optimised" voxel-based morphometry (VBM). VBM performs region-wise volumetric comparisons among populations of subjects. It requires the images to be spatially normalised, segmented into different tissue classes, and smoothed, prior to performing statistical tests [49, 36, 5, 6]. The "optimised" pre-processing strategy involved spatially normalising subjects' brain images to a standard space, by matching grey matter in these images, to a grey matter reference. The historical motivation behind this approach was to reduce the confounding effects of non-brain (e.g. scalp) structural variability on the registration. Tissue classification in older versions of SPM required the images to be registered with tissue probability maps. After registration, these maps represented the prior probability of different tissue classes being found at each location in an image. Bayes rule can then be used to combine these priors with tissue type probabilities derived from voxel intensities, to provide the posterior probability.

This procedure was inherently circular, because the registration required an initial tissue classification, and the tissue classification requires an initial registration. This circularity is resolved here by combining both components into a single generative model. This model also includes parameters that account for image intensity non-uniformity. Estimating the model parameters (for a maximum a posteriori solution) involves alternating among classification, bias correction

and registration steps. This approach provides better results than simple serial applications of each component.

Note that multi-spectral segmentation (e.g. from a registered T1 and T2 image) is not yet implemented, but is planned for a future SPM version.

5.1 Data

Select scans for processing. This assumes that there is one scan for each subject. Note that multi-spectral (when there are two or more registered images of different contrasts) processing is not yet implemented for this method.

5.2 Output Files

This routine produces spatial normalisation parameters (*.seg_sn.mat files) by default. These can be used for writing spatially normalised versions of your data, via the "Normalise: Write" option. This mechanism may produce superior results than the "Normalise: Estimate" option, although this may need some empirical evaluations.

In addition, it also produces files that can be used for doing inverse normalisation. If you have an image of regions defined in the standard space, then the inverse deformations can be used to warp these regions so that it approximately overlay your image. To use this facility, the bounding-box and voxel sizes should be set to non-finite values (e.g. [NaN NaN NaN] for the voxel sizes, and ones(2,3)*NaN for the bounding box. This would be done by the spatial normalisation module, which allows you to select a set of parameters that describe the nonlinear warps, and the images that they should be applied to.

There are a number of options about what data you would like the routine to produce. The routine can be used for producing images of tissue classes, as well as bias corrected images. The native space option will produce a tissue class image (c^*) that is in alignment with the original (see Figure 5.1). You can also produce spatially normalised versions - both with (mwc*) and without (wc*) modulation (see Figure 5.2). The bounding box and voxel sizes of the spatially normalised versions are the same as that of the tissue probability maps with which they are registered. These can be used for doing voxel-based morphometry with (both un-modulated and modulated). All you need to do is smooth them and do the stats (which means no more questions on the mailing list about how to do "optimized VBM").

Modulation is to compensate for the effect of spatial normalisation. When warping a series of images to match a template, it is inevitable that volumetric differences will be introduced into the warped images. For example, if one subject's temporal lobe has half the volume of that of the template, then its volume will be doubled during spatial normalisation. This will also result in a doubling of the voxels labeled grey matter. In order to remove this confound, the spatially normalised grey matter (or other tissue class) is adjusted by multiplying by its relative volume before and after warping. If warping results in a region doubling its volume, then the correction will halve the intensity of the tissue label. This whole procedure has the effect of preserving the total amount of grey matter signal in the normalised partitions.

A deformation field is a vector field, where three values are associated with each location in the field. The field maps from co-ordinates in the normalised image back to co-ordinates in the original image. The value of the field at co-ordinate $[x \ y \ z]$ in the normalised space will be the co-ordinate $[x' \ y' \ z']$ in the original volume. The gradient of the deformation field at a co-ordinate is its Jacobian matrix, and it consists of a 3x3 matrix:

$$\begin{pmatrix} \frac{dx'}{dx} & \frac{dx'}{dy} & \frac{dx'}{dz} \\ \frac{dy'}{dx} & \frac{dy'}{dy} & \frac{dy'}{dz} \\ \frac{dz'}{dx} & \frac{dz'}{dy} & \frac{dz'}{dz} \end{pmatrix}$$

The value of dx'/dy is a measure of how much x' changes if y is changed by a tiny amount. The determinant of the Jacobian is the measure of relative volumes of warped and unwarped

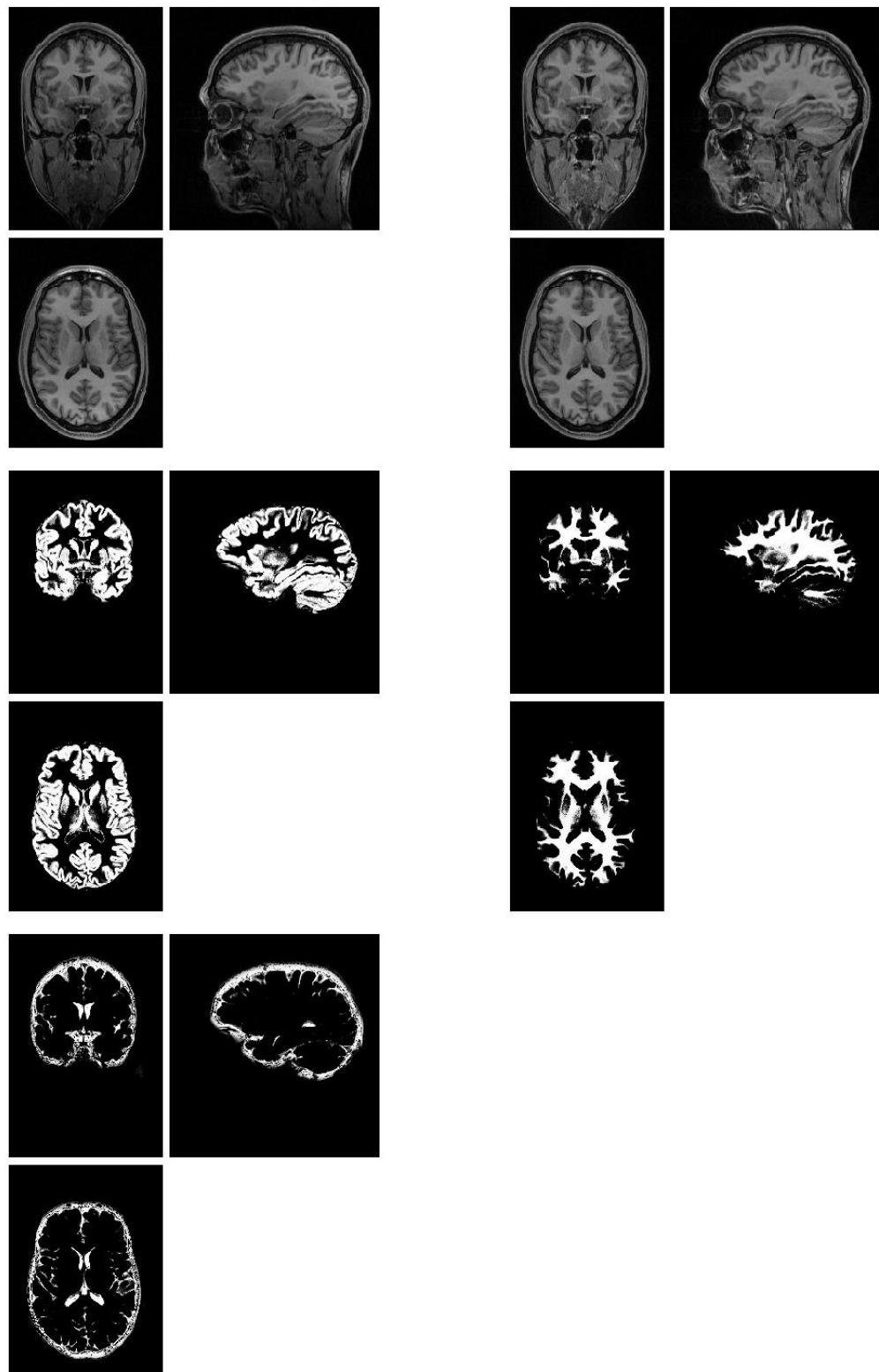


Figure 5.1: Segmentation results. These are the results that can be obtained in the original space of the image (i.e. the results that are not spatially normalised). Top left: original image (X.img). Top right: bias corrected image (mX.img). Middle and bottom rows: segmented grey matter (c1X.img), white matter (c2X.img) and CSF (c3X.img).

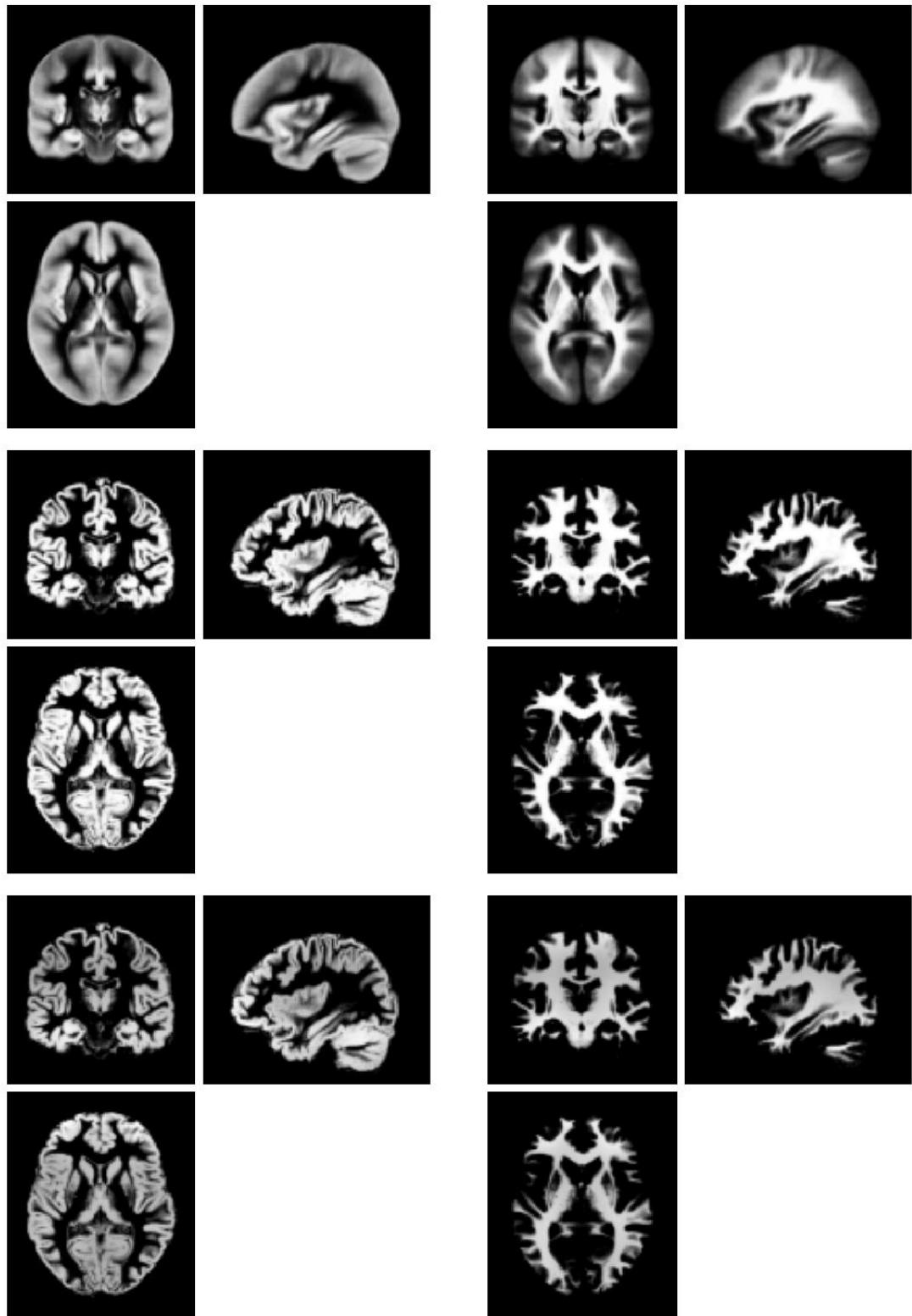


Figure 5.2: Segmentation results. These are the spatially normalised results that can be obtained (note that CSF data is not shown). Top row: The tissue probability maps used to guide the segmentation. Middle row: Spatially normalised tissue maps of grey and white matter (wc1X.img and wc2X.img). Bottom row: Modulated spatially normalised tissue maps of grey and white matter (mwc1X.img and mwc2X.img).

structures. The modulation step simply involves multiplying by the relative volumes (see Figure 5.2).

5.2.1 Grey Matter

Options to produce grey matter images: c1*.img, wc1*.img and mwc1*.img.

5.2.2 White Matter

Options to produce white matter images: c2*.img, wc2*.img and mwc2*.img.

5.2.3 Cerebro-Spinal Fluid

Options to produce CSF images: c3*.img, wc3*.img and mwc3*.img.

5.2.4 Bias Corrected

This is the option to produce a bias corrected version of your image. MR images are usually corrupted by a smooth, spatially varying artifact that modulates the intensity of the image (bias). These artifacts, although not usually a problem for visual inspection, can impede automated processing of the images. The bias corrected version should have more uniform intensities within the different types of tissues.

5.3 Custom

Various options can be adjusted in order to improve the performance of the algorithm with your data. Knowing what works best should be a matter of empirical exploration. For example, if your data has very little intensity nonuniformity artifact, then the bias regularisation should be increased. This effectively tells the algorithm that there is very little bias in your data, so it does not try to model it.

5.3.1 Tissue probability maps

Select the tissue probability images. These should be maps of grey matter, white matter and cerebro-spinal fluid probability. A nonlinear deformation field is estimated that best overlays the tissue probability maps on the individual subjects' image. The default tissue probability maps are modified versions of the ICBM Tissue Probabilistic Atlases. These tissue probability maps are kindly provided by the International Consortium for Brain Mapping, John C. Mazziotta and Arthur W. Toga. http://www.loni.ucla.edu/ICBM/ICBM_TissueProb.html. The original data are derived from 452 T1-weighted scans, which were aligned with an atlas space, corrected for scan inhomogeneities, and classified into grey matter, white matter and cerebrospinal fluid. These data were then affine registered to the MNI space and downsampled to 2mm resolution.

Rather than assuming stationary prior probabilities based upon mixing proportions, additional information is used, based on other subjects' brain images. Priors are usually generated by registering a large number of subjects together, assigning voxels to different tissue types and averaging tissue classes over subjects. Three tissue classes are used: grey matter, white matter and cerebro-spinal fluid. A fourth class is also used, which is simply one minus the sum of the first three. These maps give the prior probability of any voxel in a registered image being of any of the tissue classes - irrespective of its intensity.

The model is refined further by allowing the tissue probability maps to be deformed according to a set of estimated parameters. This allows spatial normalisation and segmentation to be combined into the same model. This implementation uses a low-dimensional approach, which parameterises the deformations by a linear combination of about a thousand cosine transform bases. This is not an especially precise way of encoding deformations, but it can model the variability of overall brain shape. Evaluations by Hellier et al have shown that this simple model can achieve a registration accuracy comparable to other fully automated methods with many more parameters.

5.3.2 Gaussians per class

The number of Gaussians used to represent the intensity distribution for each tissue class can be greater than one. In other words, a tissue probability map may be shared by several clusters. The assumption of a single Gaussian distribution for each class does not hold for a number of reasons. In particular, a voxel may not be purely of one tissue type, and instead contain signal from a number of different tissues (partial volume effects). Some partial volume voxels could fall at the interface between different classes, or they may fall in the middle of structures such as the thalamus, which may be considered as being either grey or white matter. Various other image segmentation approaches use additional clusters to model such partial volume effects. These generally assume that a pure tissue class has a Gaussian intensity distribution, whereas intensity distributions for partial volume voxels are broader, falling between the intensities of the pure classes. Unlike these partial volume segmentation approaches, the model adopted here simply assumes that the intensity distribution of each class may not be Gaussian, and assigns belonging probabilities according to these non-Gaussian distributions. Typical numbers of Gaussians could be two for grey matter, two for white matter, two for CSF, and four for everything else.

5.3.3 Affine Regularisation

The procedure is a local optimisation, so it needs reasonable initial starting estimates. Images should be placed in approximate alignment using the Display function of SPM before beginning. A Mutual Information affine registration with the tissue probability maps (D'Agostino et al, 2004) is used to achieve approximate alignment. Note that this step does not include any model for intensity nonuniformity. This means that if the procedure is to be initialised with the affine registration, then the data should not be too corrupted with this artifact. If there is a lot of intensity nonuniformity, then manually position your image in order to achieve closer starting estimates, and turn off the affine registration.

Affine registration into a standard space can be made more robust by regularisation (penalising excessive stretching or shrinking). The best solutions can be obtained by knowing the approximate amount of stretching that is needed (e.g. ICBM templates are slightly bigger than typical brains, so greater zooms are likely to be needed). For example, if registering to an image in ICBM/MNI space, then choose this option. If registering to a template that is close in size, then select the appropriate option for this.

5.3.4 Warping Regularisation

The objective function for registering the tissue probability maps to the image to process, involves minimising the sum of two terms. One term gives a function of how probable the data is given the warping parameters. The other is a function of how probable the parameters are, and provides a penalty for unlikely deformations. Smoother deformations are deemed to be more probable. The amount of regularisation determines the tradeoff between the terms. Pick a value around one. However, if your normalized images appear distorted, then it may be an idea to increase the amount of regularization (by an order of magnitude). More regularisation gives smoother deformations, where the smoothness measure is determined by the bending energy of the deformations.

5.3.5 Warp Frequency Cutoff

Cutoff of DCT bases. Only DCT bases of periods longer than the cutoff are used to describe the warps. The number actually used will depend on the cutoff and the field of view of your image. A smaller cutoff frequency will allow more detailed deformations to be modelled, but unfortunately comes at a cost of greatly increasing the amount of memory needed, and the time taken.

5.3.6 Bias regularisation

MR images are usually corrupted by a smooth, spatially varying artifact that modulates the intensity of the image (bias). These artifacts, although not usually a problem for visual inspection, can impede automated processing of the images.

An important issue relates to the distinction between intensity variations that arise because of bias artifact due to the physics of MR scanning, and those that arise due to different tissue properties. The objective is to model the latter by different tissue classes, while modelling the former with a bias field. We know a priori that intensity variations due to MR physics tend to be spatially smooth, whereas those due to different tissue types tend to contain more high frequency information. A more accurate estimate of a bias field can be obtained by including prior knowledge about the distribution of the fields likely to be encountered by the correction algorithm. For example, if it is known that there is little or no intensity non-uniformity, then it would be wise to penalise large values for the intensity nonuniformity parameters. This regularisation can be placed within a Bayesian context, whereby the penalty incurred is the negative logarithm of a prior probability for any particular pattern of nonuniformity.

5.3.7 Bias FWHM

FWHM of Gaussian smoothness of bias. If your intensity nonuniformity is very smooth, then choose a large FWHM. This will prevent the algorithm from trying to model out intensity variation due to different tissue types. The model for intensity nonuniformity is one of i.i.d. Gaussian noise that has been smoothed by some amount, before taking the exponential. Note also that smoother bias fields need fewer parameters to describe them. This means that the algorithm is faster for smoother intensity nonuniformities.

5.3.8 Sampling distance

The approximate distance between sampled points when estimating the model parameters. Smaller values use more of the data, but the procedure is slower.

Chapter 6

Normalise

Contents

6.1	Normalise: Estimate	46
6.1.1	Data	46
6.1.2	Estimation Options	46
6.2	Normalise: Write	47
6.2.1	Data	47
6.2.2	Writing Options	48
6.3	Normalise: Estimate & Write	48
6.3.1	Data	48
6.3.2	Estimation Options	49
6.3.3	Writing Options	50

This module spatially (stereotactically) normalizes MRI, PET or SPECT images into a standard space defined by some ideal model or template image[s]. The template images supplied with SPM conform to the space defined by the ICBM, NIH P-20 project, and approximate that of the the space described in the atlas of Talairach and Tournoux (1988). The transformation can also be applied to any other image that has been coregistered with these scans.

Generally, the algorithms work by minimising the sum of squares difference between the image which is to be normalised, and a linear combination of one or more template images. For the least squares registration to produce an unbiased estimate of the spatial transformation, the image contrast in the templates (or linear combination of templates) should be similar to that of the image from which the spatial normalization is derived. The registration simply searches for an optimum solution. If the starting estimates are not good, then the optimum it finds may not find the global optimum.

The first step of the normalization is to determine the optimum 12-parameter affine transformation. Initially, the registration is performed by matching the whole of the head (including the scalp) to the template. Following this, the registration proceeded by only matching the brains together, by appropriate weighting of the template voxels. This is a completely automated procedure (that does not require “scalp editing”) that discounts the confounding effects of skull and scalp differences. A Bayesian framework is used, such that the registration searches for the solution that maximizes the a posteriori probability of it being correct [8] . i.e., it maximizes the product of the likelihood function (derived from the residual squared difference) and the prior function (which is based on the probability of obtaining a particular set of zooms and shears).

The affine registration is followed by estimating nonlinear deformations, whereby the deformations are defined by a linear combination of three dimensional discrete cosine transform (DCT) basis functions [4] . The default options result in each of the deformation fields being described

by 1176parameters, where these represent the coefficients of the deformations in three orthogonal directions. The matching involved simultaneously minimizing the membrane energies of the deformation fields and the residual squared difference between the images and template(s).

The primarily use is for stereotactic normalization to facilitate inter-subject averaging and precise characterization of functional anatomy [3] . It is not necessary to spatially normalise the data (this is only a pre-requisite for intersubject averaging or reporting in the Talairach space). If you wish to circumnavigate this step (e.g. if you have single slice data or do not have an appropriate high resolution MRI scan) simply specify where you think the anterior commissure is with the ORIGIN in the header of the first scan (using the 'Display' facility) and proceed directly to 'Smoothing' or 'Statistics'.

All normalized *.img scans are written to the same subdirectory as the original *.img, prefixed with a 'w' (i.e. w*.img). The details of the transformations are displayed in the results window, and the parameters are saved in the " *_sn.mat" file.

6.1 Normalise: Estimate

Computes the warp that best registers a source image (or series of source images) to match a template, saving it to a file imagename'_sn.mat'.

6.1.1 Data

List of subjects. Images of each subject should be warped differently.

Subject

Data for this subject. The same parameters are used within subject.

Source Image The image that is warped to match the template(s). The result is a set of warps, which can be applied to this image, or any other image that is in register with it.

Source Weighting Image Optional weighting images (consisting of pixel values between the range of zero to one) to be used for registering abnormal or lesioned brains. These images should match the dimensions of the image from which the parameters are estimated, and should contain zeros corresponding to regions of abnormal tissue.

6.1.2 Estimation Options

Various settinds for estimating warps.

Template Image

Specify a template image to match the source image with. The contrast in the template must be similar to that of the source image in order to achieve a good registration. It is also possible to select more than one template, in which case the registration algorithm will try to find the best linear combination of these images in order to best model the intensities in the source image.

Template Weighting Image

Applies a weighting mask to the template(s) during the parameter estimation. With the default brain mask, weights in and around the brain have values of one whereas those clearly outside the brain are zero. This is an attempt to base the normalization purely upon the shape of the brain, rather than the shape of the head (since low frequency basis functions can not really cope with variations in skull thickness).

The option is now available for a user specified weighting image. This should have the same dimensions and mat file as the template images, with values in the range of zero to one.

Source Image Smoothing

Smoothing to apply to a copy of the source image. The template and source images should have approximately the same smoothness. Remember that the templates supplied with SPM have been smoothed by 8mm, and that smoothnesses combine by Pythagorus' rule.

Template Image Smoothing

Smoothing to apply to a copy of the template image. The template and source images should have approximately the same smoothness. Remember that the templates supplied with SPM have been smoothed by 8mm, and that smoothnesses combine by Pythagorus' rule.

Affine Regularisation

Affine registration into a standard space can be made more robust by regularisation (penalising excessive stretching or shrinking). The best solutions can be obtained by knowing the approximate amount of stretching that is needed (e.g. ICBM templates are slightly bigger than typical brains, so greater zooms are likely to be needed). If registering to an image in ICBM/MNI space, then choose the first option. If registering to a template that is close in size, then select the second option. If you do not want to regularise, then choose the third.

Nonlinear Frequency Cutoff

Cutoff of DCT bases. Only DCT bases of periods longer than the cutoff are used to describe the warps. The number used will depend on the cutoff and the field of view of the template image(s).

Nonlinear Iterations

Number of iterations of nonlinear warping performed.

Nonlinear Regularisation

The amount of regularisation for the nonlinear part of the spatial normalisation. Pick a value around one. However, if your normalized images appear distorted, then it may be an idea to increase the amount of regularization (by an order of magnitude) - or even just use an affine normalization. The regularization influences the smoothness of the deformation fields.

6.2 Normalise: Write

Allows previously estimated warps (stored in `imagename'_sn.mat'` files) to be applied to series of images.

6.2.1 Data

List of subjects. Images of each subject should be warped differently.

Subject

Data for this subject. The same parameters are used within subject.

Parameter File Select the '`_sn.mat`' file containing the spatial normalisation parameters for that subject. If nothing is selected, then the routine will assume that no more subjects need to be selected.

Images to Write These are the images for warping according to the estimated parameters. They can be any images that are in register with the "source" image used to generate the parameters.

6.2.2 Writing Options

Various options for writing normalised images.

Preserve

Preserve Concentrations: Spatially normalised images are not "modulated". The warped images preserve the intensities of the original images.

Preserve Total: Spatially normalised images are "modulated" in order to preserve the total amount of signal in the images. Areas that are expanded during warping are correspondingly reduced in intensity.

Bounding box

The bounding box (in mm) of the volume which is to be written (relative to the anterior commissure).

Voxel sizes

The voxel sizes (x, y & z, in mm) of the written normalised images.

Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour: - Fastest, but not normally recommended.

Bilinear Interpolation: - OK for PET, or realigned fMRI.

B-spline Interpolation: - Better quality (but slower) interpolation [44], especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

Wrapping

These are typically:

No wrapping: for PET or images that have already been spatially transformed.

Wrap in Y: for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

6.3 Normalise: Estimate & Write

Computes the warp that best registers a source image (or series of source images) to match a template, saving it to the file `imagename'_sn.mat'`. This option also allows the contents of the `imagename'_sn.mat'` files to be applied to a series of images.

6.3.1 Data

List of subjects. Images of each subject should be warped differently.

Subject

Data for this subject. The same parameters are used within subject.

Source Image The image that is warped to match the template(s). The result is a set of warps, which can be applied to this image, or any other image that is in register with it.

Source Weighting Image Optional weighting images (consisting of pixel values between the range of zero to one) to be used for registering abnormal or lesioned brains. These images should match the dimensions of the image from which the parameters are estimated, and should contain zeros corresponding to regions of abnormal tissue.

Images to Write These are the images for warping according to the estimated parameters. They can be any images that are in register with the "source" image used to generate the parameters.

6.3.2 Estimation Options

Various settings for estimating warps.

Template Image

Specify a template image to match the source image with. The contrast in the template must be similar to that of the source image in order to achieve a good registration. It is also possible to select more than one template, in which case the registration algorithm will try to find the best linear combination of these images in order to best model the intensities in the source image.

Template Weighting Image

Applies a weighting mask to the template(s) during the parameter estimation. With the default brain mask, weights in and around the brain have values of one whereas those clearly outside the brain are zero. This is an attempt to base the normalization purely upon the shape of the brain, rather than the shape of the head (since low frequency basis functions can not really cope with variations in skull thickness).

The option is now available for a user specified weighting image. This should have the same dimensions and mat file as the template images, with values in the range of zero to one.

Source Image Smoothing

Smoothing to apply to a copy of the source image. The template and source images should have approximately the same smoothness. Remember that the templates supplied with SPM have been smoothed by 8mm, and that smoothnesses combine by Pythagorus' rule.

Template Image Smoothing

Smoothing to apply to a copy of the template image. The template and source images should have approximately the same smoothness. Remember that the templates supplied with SPM have been smoothed by 8mm, and that smoothnesses combine by Pythagorus' rule.

Affine Regularisation

Affine registration into a standard space can be made more robust by regularisation (penalising excessive stretching or shrinking). The best solutions can be obtained by knowing the approximate amount of stretching that is needed (e.g. ICBM templates are slightly bigger than typical brains, so greater zooms are likely to be needed). If registering to an image in ICBM/MNI space, then choose the first option. If registering to a template that is close in size, then select the second option. If you do not want to regularise, then choose the third.

Nonlinear Frequency Cutoff

Cutoff of DCT bases. Only DCT bases of periods longer than the cutoff are used to describe the warps. The number used will depend on the cutoff and the field of view of the template image(s).

Nonlinear Iterations

Number of iterations of nonlinear warping performed.

Nonlinear Regularisation

The amount of regularisation for the nonlinear part of the spatial normalisation. Pick a value around one. However, if your normalized images appear distorted, then it may be an idea to increase the amount of regularization (by an order of magnitude) - or even just use an affine normalization. The regularization influences the smoothness of the deformation fields.

6.3.3 Writing Options

Various options for writing normalised images.

Preserve

Preserve Concentrations: Spatially normalised images are not "modulated". The warped images preserve the intensities of the original images.

Preserve Total: Spatially normalised images are "modulated" in order to preserve the total amount of signal in the images. Areas that are expanded during warping are correspondingly reduced in intensity.

Bounding box

The bounding box (in mm) of the volume which is to be written (relative to the anterior commissure).

Voxel sizes

The voxel sizes (x, y & z, in mm) of the written normalised images.

Interpolation

The method by which the images are sampled when being written in a different space.

Nearest Neighbour: - Fastest, but not normally recommended.

Bilinear Interpolation: - OK for PET, or realigned fMRI.

B-spline Interpolation: - Better quality (but slower) interpolation [44], especially with higher degree splines. Do not use B-splines when there is any region of NaN or Inf in the images.

Wrapping

These are typically:

No wrapping: for PET or images that have already been spatially transformed.

Wrap in Y: for (un-resliced) MRI where phase encoding is in the Y direction (voxel space).

Chapter 7

Smooth

Contents

7.1	Images to Smooth	51
7.2	FWHM	51

This is for smoothing (or convolving) image volumes with a Gaussian kernel of a specified width. It is used as a preprocessing step to suppress noise and effects due to residual differences in functional and gyral anatomy during inter-subject averaging.

7.1 Images to Smooth

Specify the images to smooth. The smoothed images are written to the same subdirectories as the original *.img and are prefixed with a 's' (i.e. s*.img).

7.2 FWHM

Specify the full-width at half maximum (FWHM) of the Gaussian smoothing kernel in mm. Three values should be entered, denoting the FWHM in the x, y and z directions.

Part III

fMRI Statistics

Chapter 8

fMRI model specification

Statistical analysis of fMRI data uses a mass-univariate approach based on General Linear Models (GLMs). It comprises the following steps (1) specification of the GLM design matrix, fMRI data files and filtering (2) estimation of GLM parameters using classical or Bayesian approaches and (3) interrogation of results using contrast vectors to produce Statistical Parametric Maps (SPMs) or Posterior Probability Maps (PPMs).

The design matrix defines the experimental design and the nature of hypothesis testing to be implemented. The design matrix has one row for each scan and one column for each effect or explanatory variable. (eg. regressor or stimulus function). You can build design matrices with separable session-specific partitions. Each partition may be the same (in which case it is only necessary to specify it once) or different.

Responses can be either event- or epoch related, the only distinction is the duration of the underlying input or stimulus function. Mathematically they are both modeled by convolving a series of delta (stick) or box functions (u), indicating the onset of an event or epoch with a set of basis functions. These basis functions model the hemodynamic convolution, applied by the brain, to the inputs. This convolution can be first-order or a generalized convolution modeled to second order (if you specify the Volterra option). The same inputs are used by the Hemodynamic model or Dynamic Causal Models which model the convolution explicitly in terms of hidden state variables.

Event-related designs may be stochastic or deterministic. Stochastic designs involve one of a number of trial-types occurring with a specified probability at successive intervals in time. These probabilities can be fixed (stationary designs) or time-dependent (modulated or non-stationary designs). The most efficient designs obtain when the probabilities of every trial type are equal. A critical issue in stochastic designs is whether to include null events. If you wish to estimate the evoked response to a specific event type (as opposed to differential responses) then a null event must be included (even if it is not modeled explicitly).

In SPM, analysis of data from multiple subjects typically proceeds in two stages using models at two ‘levels’. The ‘first level’ models are used to implement a within-subject analysis. Typically there will be as many first level models as there are subjects. Analysis proceeds as described using the ‘Specify first level’ and ‘Estimate’ options. The results of these analyses can then be presented as ‘case studies’. More often, however, one wishes to make inferences about the population from which the subjects were drawn. This is an example of a ‘Random-Effects (RFX) analysis’ (or, more properly, a mixed-effects analysis). In SPM, RFX analysis is implemented using the ‘summary-statistic’ approach where contrast images from each subject are used as summary measures of subject responses. These are then entered as data into a ‘second level’ model.

Figure 8.1 shows how the SPM graphics window appears during fMRI model specification.

8.1 Timing parameters

Specify various timing parameters needed to construct the design matrix. This includes the units of the design specification and the interscan interval.

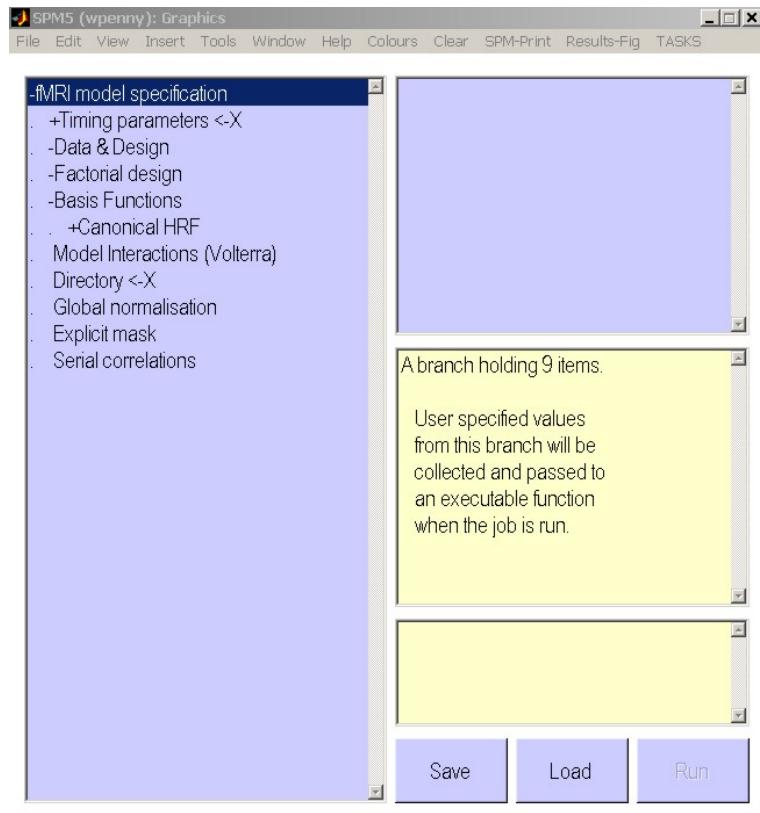


Figure 8.1: After starting SPM in fMRI mode, pressing the ‘Specify 1st-level’ button, and then double-clicking on the ‘+fMRI model specification’ text, the SPM graphics window should appear as above. The options under ‘fMRI model specification’ can be examined by clicking on them. A single click will bring up some help text in the lower subwindow (not shown in the above graphic). A double-click on options prefixed by a ‘+’ will allow you to specify options at a greater level of detail. Options highlighted with a ‘<-X’ are mandatory and must be filled in by the user. Each of the options shown above is described in this chapter.

Also, with long TRs you may want to shift the regressors so that they are aligned to a particular slice. This is effected by changing the microtime resolution and onset.

8.1.1 Units for design

The onsets of events or blocks can be specified in either scans or seconds.

8.1.2 Interscan interval

Interscan interval, TR, (specified in seconds). This is the time between acquiring a plane of one volume and the same plane in the next volume. It is assumed to be constant throughout.

8.1.3 Microtime resolution

In Echo-Planar Imaging (EPI), data is acquired a plane at a time. To acquire a whole volume of data takes at least a second or two.

It is possible, however, that experimental events may occur between scan (volume) acquisition times. This can be specified when building your design matrix either by (i) specifying your design in scans and using non-integer values or (ii) specifying your design in seconds at a resolution greater than the TR.

SPM takes these timing specifications and builds its regressors using a ‘microtime’ time-scale. The microtime resolution, t, is the number of time-bins per scan.

Do not change this parameter unless you have a long TR and wish to shift regressors so that they are aligned to a particular slice.

8.1.4 Microtime onset

The microtime onset, t0, is the first time-bin at which the regressors are resampled to coincide with data acquisition. If t0 = 1 then the regressors will be appropriate for the first slice. If you want to temporally realign the regressors so that they match responses in the middle slice then make t0 = t/2 (assuming there is a negligible gap between volume acquisitions).

Do not change the default setting unless you have a long TR.

A typical use of the t and t0 parameters is to set them to correspond to the results of any slice timing correction you have made eg. if you have 24 slices and have made slice 12 the reference slice you would set t=24, t0=12.

8.2 Data & Design

The design matrix defines the experimental design and the nature of hypothesis testing to be implemented. The design matrix has one row for each scan and one column for each effect or explanatory variable. (e.g. regressor or stimulus function). Figure 8.2 shows an example of a design matrix.

You can build design matrices with separable session-specific partitions. Each partition may be the same (in which case it is only necessary to specify it once) or different. Responses can be either event- or epoch related, where the latter model involves prolonged and possibly time-varying responses to state-related changes in experimental conditions. Event-related response are modelled in terms of responses to instantaneous events. Mathematically they are both modelled by convolving a series of delta (stick) or box-car functions, encoding the input or stimulus function, with a set of hemodynamic basis functions.

8.2.1 Subject/Session

The design matrix for fMRI data consists of one or more separable, session-specific partitions. These partitions are usually either one per subject, or one per fMRI scanning session for that subject.

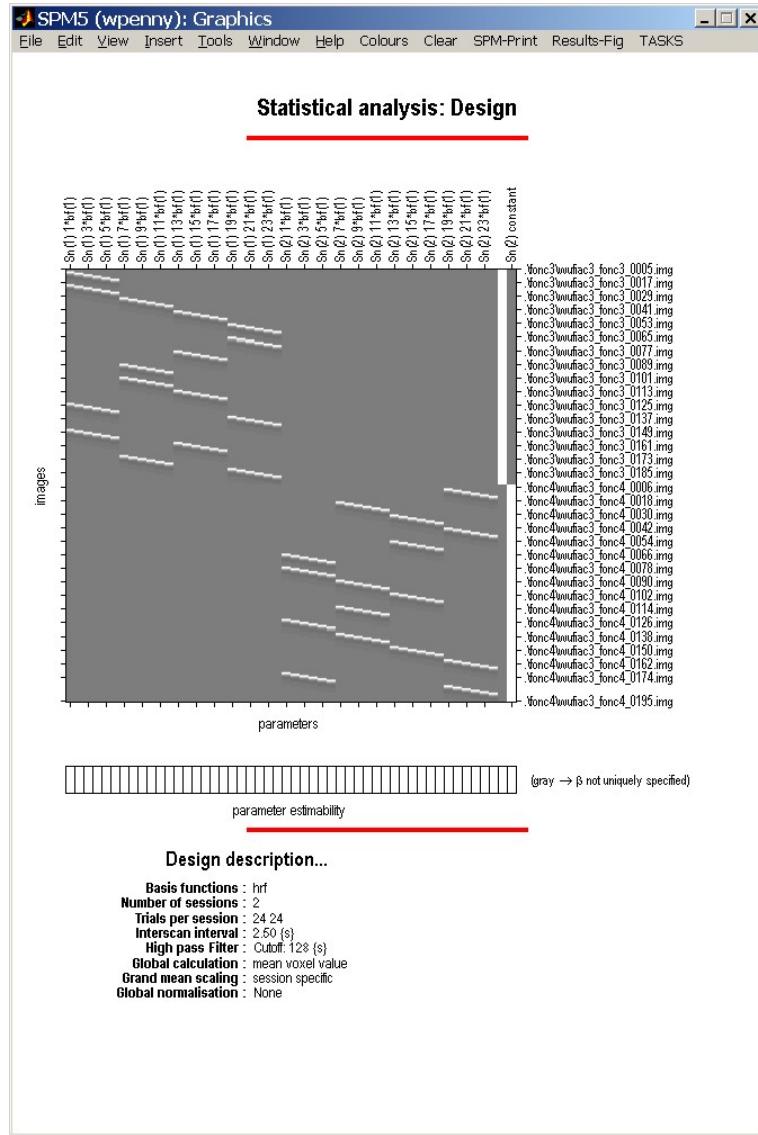


Figure 8.2: *Design matrix for fMRI data from two sessions. There are 24 experimental conditions for each session. The last two columns model the average activity in each session, giving a total of 50 regressors. There are 191 fMRI scans for each session. The overall design matrix therefore has 382 rows and 50 columns.*

Scans

Select the fMRI scans for this session. They must all have the same image dimensions, orientation, voxel size etc. This is implemented using SPM's file selector.

Conditions

You are allowed to combine both event- and epoch-related responses in the same model and/or regressor. Any number of condition (event or epoch) types can be specified. Epoch and event-related responses are modeled in exactly the same way by specifying their onsets [in terms of onset times] and their durations. Events are specified with a duration of 0. If you enter a single number for the durations it will be assumed that all trials conform to this duration. For factorial designs, one can later associate these experimental conditions with the appropriate levels of experimental factors.

Condition An array of input functions is constructed, specifying occurrence events or epochs (or both). These are convolved with a basis set at a later stage to give regressors that enter into the design matrix. Interactions of evoked responses with some parameter (time or a specified variate) enter at this stage as additional columns in the design matrix with each trial multiplied by the [expansion of the] trial-specific parameter. The 0th order expansion is simply the main effect in the first column.

Name Condition Name

Onsets Specify a vector of onset times for this condition type. This can be entered using the keyboard eg. typing in ‘100 300’ and then hitting return or ‘100;300’ or ‘[100,300]’ or ‘[100,300]’.

More usually, however, this specification takes place using variables that have been created before and loaded into matlab. For example, an `my_onsets` cell array¹ might exist in a file you created earlier called `my_design.mat`. You would then type `load my_design` at the matlab command prompt before pressing the ‘Specify 1st-level’ button.

You could then specify the onsets for condition 2 by typing in eg. `my_onsets{2}` instead of entering the numbers via the keyboard.

Durations Specify the event durations (in seconds). Epoch and event-related responses are modeled in exactly the same way but by specifying their different durations. Events are specified with a duration of 0. If you enter a single number for the durations it will be assumed that all trials conform to this duration. If you have multiple different durations, then the number must match the number of onset times.

Time Modulation This option allows for the characterisation of nonstationary responses. Specifically, you can model either linear or nonlinear time effects. For example, 1st order modulation would model the stick functions and a linear change of the stick function heights over time. Higher order modulation will introduce further columns that contain the stick functions scaled by time squared, time cubed etc.

Parametric Modulations The stick function itself can be modulated by some parametric variate (this can be time or some trial-specific variate like reaction time) modeling the interaction between the trial and the variate. The events can be modulated by zero or more parameters.

See [12, 11] for further details of parametric modulations.

¹Cell arrays are usually used in preference to matrices as different event types can then have different numbers of events.

Multiple conditions

If you have multiple conditions then entering the details a condition at a time is very inefficient. This option can be used to load all the required information in one go.

You will need to create a `*.mat` file containing the relevant information. This `*.mat` file must include the following cell arrays: names, onsets and durations eg. `names{2}='SSent-DSpeak'`, `onsets{2}=[3 5 19 222]`, `durations{2}=[0 0 0 0]` contain the required details of the second condition. These cell arrays may be made available by your stimulus delivery program eg. CO-GENT. The duration vectors can contain a single entry if the durations are identical for all events.

You then need to use SPM's file selector to select this `*.mat` file.

Regressors

Regressors are additional columns included in the design matrix, which may model effects that would not be convolved with the haemodynamic response. One such example would be the estimated movement parameters, which may confound the data.

Regressor

Name Enter name of regressor eg. First movement parameter

Value Enter the values that the regressor takes. This could also be, for example, the name of a variable in MATLAB's work space that you have previously loaded in from a file. This might be a subjects movement parameters or reaction times.

Multiple regressors

If you have mutliple regressors eg. realignment parameters, then entering the details a regressor at a time is very inefficient. This option can be used to load all the required information in one go.

You will first need to create a `*.mat` file containing a matrix R. Each column of R will contain a different regressor. When SPM creates the design matrix the regressors will be named R1, R2, R3, ..etc.

You then need to use SPM's file selector to select this `*.mat` file.

High-pass filter

The default high-pass filter cutoff is 128 seconds. Slow signal drifts with a period longer than this will be removed. Use 'Explore design' to ensure this cut-off is not removing too much experimental variance. This is described later in section 8.10. High-pass filtering is implemented using a residual forming matrix (i.e. it is not a convolution) and is simply a way to remove confounds without estimating their parameters explicitly. The constant term is also incorporated into this filter matrix.

8.3 Factorial design

If you have a factorial design then SPM can automatically generate the contrasts necessary to test for the main effects and interactions.

This includes the F-contrasts necessary to test for these effects at the within-subject level (first level) and the simple contrasts necessary to generate the contrast images for a between-subject (second-level) analysis.

To use this option, create as many factors as you need and provide a name and number of levels for each. SPM assumes that the condition numbers of the first factor change slowest, the second factor next slowest etc. It is best to write down the contingency table for your design to ensure this condition is met. This table relates the levels of each factor to the conditions.

For example, if you have 2-by-3 design your contingency table has two rows and three columns where the first factor spans the rows, and the second factor the columns. The numbers of the conditions are 1,2,3 for the first row and 4,5,6 for the second.

See [27] for more information on SPM and factorial designs.

8.3.1 Factor

Add a new factor to your experimental design

Name

Name of factor, eg. 'Repetition'

Levels

Enter number of levels for this factor, eg. 2

8.4 Basis Functions

SPM uses basis functions to model the hemodynamic response. This could be a single basis function or a set of functions. The most common choice is the 'Canonical HRF' with or without time and dispersion derivatives.

8.4.1 Canonical HRF

Canonical Hemodynamic Response Function (HRF). This is the default option. Contrasts of these effects have a physical interpretation and represent a parsimonious way of characterising event-related responses. This option is also useful if you wish to look separately at activations and deactivations. This is implemented using a t-contrast with a +1 or -1 entry over the canonical regressor.

Model derivatives

Model HRF Derivatives. The canonical HRF combined with time and dispersion derivatives comprise an 'informed' basis set, as the shape of the canonical response conforms to the hemodynamic response that is commonly observed. The incorporation of the derivative terms allow for variations in subject-to-subject and voxel-to-voxel responses. The time derivative allows the peak response to vary by plus or minus a second and the dispersion derivative allows the width of the response to vary by a similar amount.

A positive estimate of the time-derivative regression coefficient implies that the peak hemodynamic response occurs later than usual ie. than would be expected using just the canonical regressor. A positive estimate for the dispersion derivative implies a more dispersed response than usual.

The informed basis set requires an SPMF for inference. T-contrasts over just the canonical are perfectly valid but assume constant delay/dispersion. The informed basis set compares favourably with eg. FIR bases on many data sets [28].

8.4.2 Other basis sets

The other basis sets supported by SPM are

1. Fourier Set
2. Fourier Set (Hanning)
3. Gamma Functions
4. Finite Impulse Response (FIR)

For each of these options you must also specify the **window length** which is the length in seconds of the post-stimulus time window that the basis functions span. You must also specify the **order**, that is, how many basis functions to use.

Usually, an informed basis set should be sufficient for most data sets. If this does not provide a good fit to the data it may be worthwhile re-considering how the neuronal events are modelled ie. is the timing correct ? should events be split into subsets ?

Alternatively, the gamma basis functions are an interesting choice as a particular linear combination of them is actually used to specify the canonical HRF. The FIR approach is of interest as it is equivalent to the method of ‘selective averaging’. See [26] for further details.

8.5 Model Interactions (Volterra)

Generalized convolution of inputs, U , with basis set, bf .

For first order expansions the causes are simply convolved (e.g. stick functions) in U by the basis functions in bf to create a design matrix X . For second order expansions new entries appear that correspond to the interaction among the original causes. The basis functions for these effects are two dimensional and are used to assemble the second order kernel.

Interactions or response modulations can enter at two levels. Firstly the stick function itself can be modulated by some parametric variate. This can be time or some trial-specific variate like reaction time modeling the interaction between the trial and the variate. Secondly interactions among the trials themselves can be modeled using a Volterra series formulation that accommodates interactions over time (and therefore within and between trial types).

This last option is useful for accommodating nonlinearities in the hemodynamic response. For example, if two events occur within a second or so of each other then the hemodynamic response to the pair may be less than the sum of the responses to each event when occurring in isolation. This type of ‘sub-linear’ response can be modelled using Volterra kernels. See [22] for further details.

8.6 Directory

Select a directory where the SPM.mat file containing the specified design matrix will be written. If this directory already contains an SPM.mat file then SPM will warn you of this before overwriting it, when the specification job is run.

8.7 Global normalisation

SPM can normalise fMRI data in one of two ways. These are selected using the options ‘None’ (the default) and ‘Scaling’.

Both methods are based on first estimating the average within-brain fMRI signal, g_{ns} , where n denotes scan and s denotes session. If you select ‘Scaling’, SPM will multiply each fMRI value in scan n and session s by $100/g_{ns}$.

If you select ‘None’ then SPM computes the grand mean value, $g_s = \frac{\sum_{n=1}^N g_{ns}}{N}$ where N is the number of scans in that session. This is the fMRI signal averaged over all voxels within the brain and all time points within session s . SPM then implements ‘Session-specific grand mean scaling’ by multiplying each fMRI data point in session s by $100/g_s$.

See [1] for further discussion of this issue.

8.8 Explicit mask

Specify an image for explicitly masking the analysis. A sensible option here is to use a segmentation of structural images to specify a within-brain mask. If you select that image as an explicit mask then only those voxels in the brain will be analysed. This both speeds the estimation and restricts SPMs/PPMs to within-brain voxels. Alternatively, if such structural images are unavailable or no masking is required, then leave this field empty.

8.9 Serial correlations

Serial correlations in fMRI time series due to aliased biorhythms and unmodelled neuronal activity can be accounted for using an autoregressive AR(1) model during Classical (ReML) parameter estimation.

This estimate assumes the same correlation structure for each voxel, within each session. ReML estimates are then used to correct for non-sphericity during inference by adjusting the statistics and degrees of freedom appropriately. The discrepancy between estimated and actual correlations are greatest at low frequencies. Therefore specification of the high-pass filter is particularly important.

Serial correlation can be ignored if you choose the ‘none’ option. Note that the above options only apply if you later specify that your model will be estimated using the Classical (ReML) approach. If you choose Bayesian estimation these options will be ignored. For Bayesian estimation, the choice of noise model (AR model order) is made under the estimation options. See [24, 38] for further discussion of these issues.

8.10 Reviewing your design

After you have completed the SPM ‘job’ file for specifying your fMRI design, and have run it, you will then be able to review your design by pressing the ‘Review’ button in SPM’s button window (the top-left window). This is particularly useful, for example, for checking that your experimental variance has not been removed by high-pass filtering, as shown in Figure 8.3.

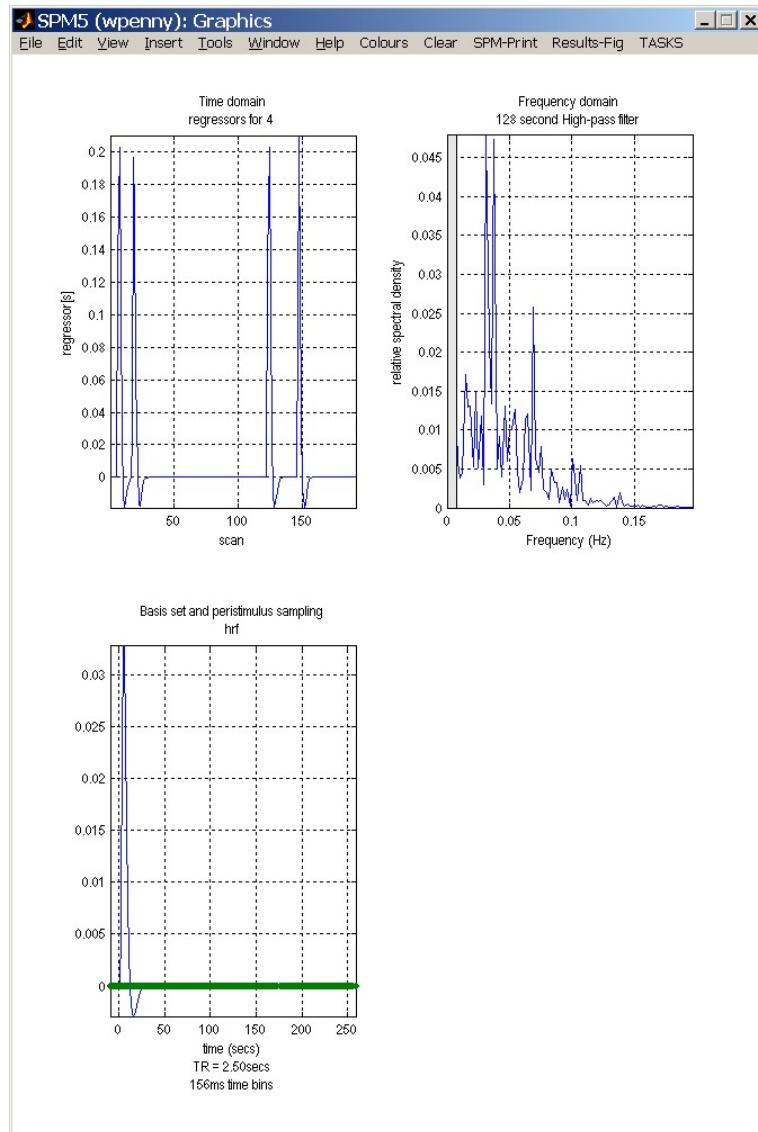


Figure 8.3: After pressing ‘Review’, selecting the pull-down ‘Design’ menu, Explore->Session, and selecting the regressor you wish to look at, you should get a plot similar to the one above. The top row shows time and frequency domain plots of the time-series corresponding to this regressor. In this particular case we have four events. Each event or ‘stick function’ has been convolved with the hemodynamic response function shown in the bottom panel. The frequency domain graph is useful for checking that experimental variance is not removed by high-pass filtering. The grayed out section of the frequency plot shows those frequencies which are removed. For this regressor we have plenty of remaining experimental variance (see the peak at about 0.04Hz).

Chapter 9

fMRI model estimation

Model parameters can be estimated using classical (ReML - Restricted Maximum Likelihood) or Bayesian algorithms. After parameter estimation, the RESULTS button can be used to specify contrasts that will produce Statistical Parametric Maps (SPMs), Effect Size Maps (ESMs) or Posterior Probability Maps (PPMs) and tables of statistics.

9.1 Select SPM.mat

Select the SPM.mat file that contains the design specification. SPM will output the results of its analysis into this directory. This includes overwriting the SPM.mat file. When the estimation job is run, no warning will be given that the SPM.mat file will be overwritten. A warning is given at the specification stage. When it comes to estimation, SPM assumes that you've now sorted out your directory structures.

9.2 Method

There are three possible estimation procedures for fMRI models (1) classical (ReML) estimation of first or second level models, (2) Bayesian estimation of first level models and (3) Bayesian estimation of second level models. Option (2) uses a Variational Bayes (VB) algorithm that is new to SPM5. Option (3) uses the Empirical Bayes algorithm with global shrinkage priors that was also in SPM2.

To use option (3) you must have already estimated the model using option (1). That is, for second-level models you must run a ReML estimation before running a Bayesian estimation. This is not necessary for option (2). Bayesian estimation of 1st-level models using VB does not require a prior ReML estimation.

9.2.1 Classical

Model parameters are estimated using Restricted Maximum Likelihood (ReML). This assumes the error correlation structure is the same at each voxel. This correlation can be specified using either an AR(1) or an Independent and Identically Distributed (IID) error model. These options are chosen at the model specification stage. ReML estimation should be applied to spatially smoothed functional images. See [24, 20] for further details of the ReML estimation scheme. After estimation, specific profiles of parameters are tested using a linear compound or contrast with the T or F statistic. The resulting statistical map constitutes an SPM. The SPMT/F is then characterised in terms of focal or regional differences by assuming that (under the null hypothesis) the components of the SPM (ie. residual fields) behave as smooth stationary Gaussian fields.

The rest of this chapter describes the Bayesian estimation options. So, please skip to the next chapter if you are interested only in classical estimation and inference.

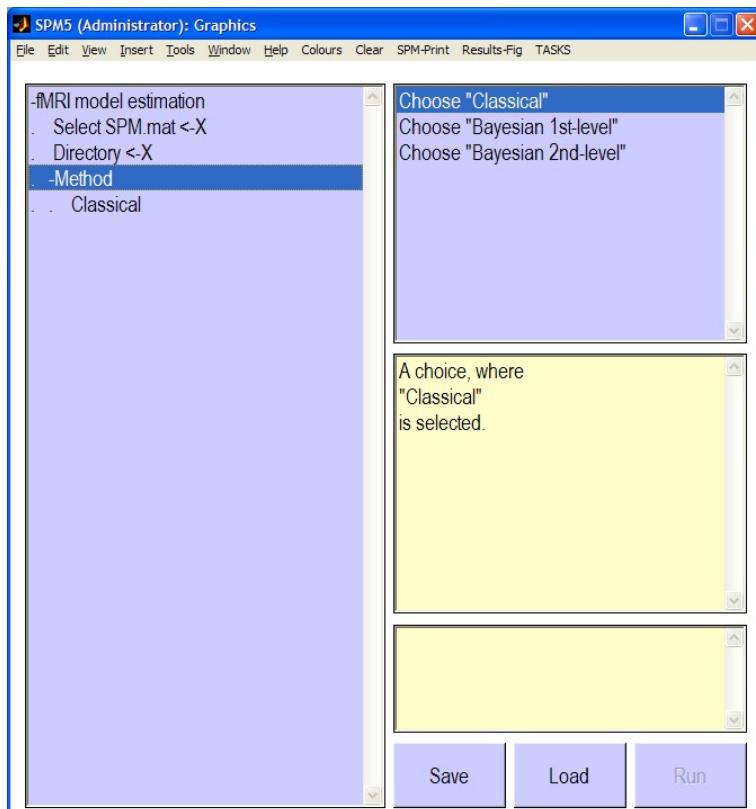


Figure 9.1: After starting SPM in fMRI mode, pressing the ‘Estimate’ button, and then double-clicking on the ‘+fMRI model estimation’ text, the SPM graphics window should appear as above. The options under ‘-fMRI model estimation’ can be examined by clicking on them. A single click will bring up some help text in the lower subwindow (not shown in the above graphic). A double-click on options prefixed by a ‘+’ will allow you to specify options at a greater level of detail. Options highlighted with a ‘<-X’ are mandatory and must be filled in by the user. Each of the options shown above is described in this chapter.

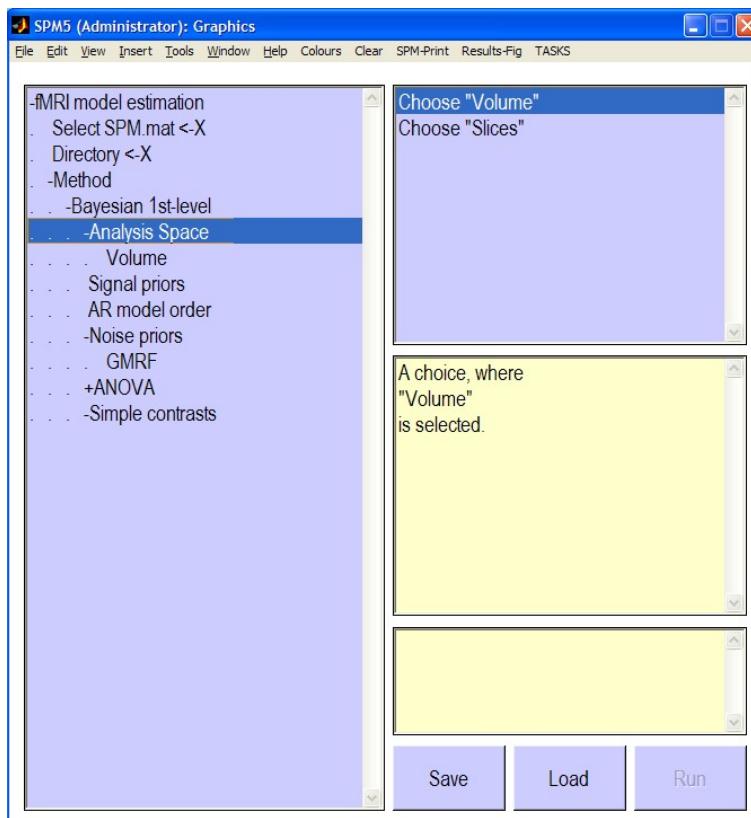


Figure 9.2: After choosing Bayesian 1st-level under ‘Method’ and then double-clicking on the ‘+Bayesian 1st-level’ text, the SPM graphics window should appear as above. Each of the options shown above is described in this chapter.

9.2.2 Bayesian 1st-level

Model parameters are estimated using Variational Bayes (VB). This allows you to specify spatial priors for regression coefficients and regularised voxel-wise AR(P) models for fMRI noise processes. The algorithm does not require functional images to be spatially smoothed. Estimation will take about 5 times longer than with the classical approach. This is why VB is not the default estimation option. The VB approach has been described in a number of papers [38, 40, 37, ?].

After estimation, contrasts are used to find regions with effects larger than a user-specified size eg. 1 per cent of the global mean signal. These effects are assessed statistically using a Posterior Probability Map (PPM) [23].

Analysis Space

Because estimation can be time consuming, an option is provided to analyse selected slices rather than the whole volume.

Volume You have selected the Volume option. SPM will analyse fMRI time series in all slices of each volume.

Slices Enter Slice Numbers. This can be a single slice or multiple slices. If you select a single slice or only a few slices you must be aware of the interpolation options when, after estimation, displaying the estimated images eg. images of contrasts or AR maps. The default interpolation option may need to be changed to nearest neighbour (NN) (see bottom right hand of graphics window) for your slice maps to be visible.

Signal priors

- [GMRF] Gaussian Markov Random Field. This spatial prior is the recommended option. Regression coefficients at a given voxel are (softly) constrained to be similar to those at nearby voxels. The strength of this constraint is determined by a spatial precision parameter that is estimated from the data. Different regression coefficients have different spatial precisions allowing each putative experimental effect to have its own spatial regularity.
- [LORETA] Low Resolution Tomography Prior. This spatial prior is very similar to the GMRF prior and is a standard choice for MEG/EEG source localisation algorithms. It does, however, have undesirable edge effects.
- [Global] Global Shrinkage prior. This is not a spatial prior in the sense that regression coefficients are constrained to be similar to neighboring voxels. Instead, the average effect over all voxels (global effect) is assumed to be zero and all regression coefficients are shrunk towards this value in proportion to the prior precision. This is the same prior that is used for Bayesian estimation at the second level (see also [23]), except that here the prior precision is estimated separately for each slice.
- [Uninformative] A flat prior. Essentially, no prior information is used. If you select this option then VB reduces to Maximum Likelihood (ML) estimation. This option is useful if, for example, you do not wish to use a spatial prior but wish to take advantage of the voxel-wise AR(P) modelling of noise processes. In this case, you would apply the algorithm to images that have been spatially smoothed. For P=0, ML estimation in turn reduces to Ordinary Least Squares (OLS) estimates, and for P>0, ML estimation is equivalent to a weighted least squares (WLS) algorithm but where the weights are different at each voxel. This reflects the different noise correlations at each voxel.

AR model order

An AR model order of 3 is the default. Cardiac and respiratory artifacts are periodic in nature and therefore require an AR order of at least 2. In previous work, voxel-wise selection of the optimal model order showed that a value of 3 was the highest order required [38].

Higher model orders have little effect on the estimation time. If you select a model order of zero this corresponds to the assumption that the errors are Independent and Identically Distributed (IID). This AR specification overrides any choices that were made in the model specification stage.

Voxel-wise AR models are fitted separately for each session of data. For each session this therefore produces maps of AR(1), AR(2) etc coefficients in the output directory.

Noise priors

There are three noise prior options.

- [GMRF] Gaussian Markov Random Field. This is the default option. This spatial prior is the same as that used for the regression coefficients. Spatial precisions are estimated separately for each AR coefficient eg. the AR(1) coefficient over space, AR(2) over space etc.
- [LORETA] Low Resolution Tomography Prior. See comments on LORETA priors for regression coefficients.
- [Tissue-type] This provides an estimation of AR coefficients at each voxel that are biased towards typical values for that tissue type (eg. gray, white, CSF). If you select this option you will need to then select files that contain tissue type maps (see below). These are typically chosen to be Grey Matter, White Matter and CSF images derived from segmentation of registered structural scans.

Previous work has shown that there is significant variation in AR values with tissue type. However, GMRF priors have previously been favoured by Bayesian model comparison [?].

ANOVA

Perform 1st or 2nd level Analysis of Variance.

First level This is implemented using Bayesian model comparison as described in [?]. For example, to test for the main effect of a factor two models are compared, one where the levels are represented using different regressors and one using the same regressor. This therefore requires explicit fitting of several models at each voxel and is computationally demanding (requiring several hours of computation). The recommended option is therefore NO.

To use this option you must have already specified your factorial design during the model specification stage.

Second level This option tells SPM to automatically generate the simple contrasts that are necessary to produce the contrast images for a second-level (between-subject) ANOVA. Naturally, these contrasts can also be used to characterise simple effects for each subject.

With the Bayesian estimation option it is recommended that contrasts are computed during the parameter estimation stage (see 'simple contrasts' below). The recommended option here is therefore YES.

To use this option you must have already specified your factorial design during the model specification stage.

If you wish to use these contrast images for a second-level analysis then you will need to spatially smooth them to take into account between-subject differences in functional anatomy ie. the fact that one persons V5 may be in a different position than another.

Simple contrasts

'Simple' contrasts refers to a contrast that spans one-dimension ie. to assess an effect that is increasing or decreasing.

If you have a factorial design then the contrasts needed to generate the contrast images for a 2nd-level ANOVA (or to assess these simple effects within-subject) can be specified automatically using the ANOVA->Second level option.

When using the Bayesian estimation option it is computationally more efficient to compute the contrasts when the parameters are estimated. This is because estimated parameter vectors have potentially different posterior covariance matrices at different voxels and these matrices are not stored. If you compute contrasts post-hoc these matrices must be recomputed. This uses an approximate reconstruction based on a Taylor series expansion described in [37]. It is therefore recommended to specify as many contrasts as possible prior to parameter estimation.

If you wish to use these contrast images for a second-level analysis then you will need to spatially smooth them to take into account between-subject differences in functional anatomy ie. the fact that one persons V5 may be in a different position than another.

Simple contrast

Name Name of contrast eg. 'Positive Effect'

Contrast vector These contrasts are used to generate PPMs which characterise effect sizes at each voxel. This is different to SPMs in which eg. maps of t-statistics show the ratio of the effect size to effect variability (standard deviation). SPMs are therefore a-dimensional. This is not the case for PPMs as the size of the effect is of primary interest. Some care is therefore needed about the scaling of contrast vectors. For example, if you are interested in the differential effect size averaged over conditions then the contrast $[0.5, 0.5, -0.5, -0.5]$ would be more suitable than the $[1, 1, -1, -1]$ contrast which looks at the differential effect size summed over conditions.

9.2.3 Bayesian 2nd-level

Bayesian estimation of 2nd level models. This option uses the Empirical Bayes algorithm with global shrinkage priors that was previously implemented in SPM2. It is described in detail in [23].

Use of the global shrinkage prior embodies a prior belief that, on average over all voxels, there is no net experimental effect. Some voxels will respond negatively and some positively with a variability determined by the prior precision. This prior precision can be estimated from the data using Empirical Bayes.

9.3 Output files

After estimation a number of files are written to the output directory. These are

- An `SPM.mat` file containing specification of the design and estimated model parameters

9.3.1 Classical 1st-level

For classical 1st-level models the following files are also produced

- Images of estimated regression coefficients `beta_000k.img` where k indexes the k th regression coefficient.
- An image of the variance of the error `ResMS.img`.
- An image `mask.img` indicating which voxels were included in the analysis.
- The image `RPV.img`, the estimated resels per voxel.
- If contrasts have been specified SPM also writes `con_000i.img` if the i th contrast is a t-contrast and the extra sum of squares image `ess_000i.img` if it is an F-contrast.

Type `help spm_spm` at the matlab command prompt for further information.

9.3.2 Bayesian 1st-level

For Bayesian 1st-level models the following files are also produced

- Images of estimated regression coefficients `Cbeta_000k.img` where k indexes the k th regression coefficient. These filenames are prefixed with a ‘C’ indicating that these are the mean values of the ‘Conditional’ or ‘Posterior’ density.
- Images of error bars/standard deviations on the regression coefficients `SDbeta_000k.img`.
- An image of the standard deviation of the error `Sess1_SDerror.img`.
- An image `mask.img` indicating which voxels were included in the analysis.
- If a non-zero AR model order is specified then SPM also writes images `Sess1_AR_000p.img` where p indexes the p th AR coefficient.
- If contrasts have been specified SPM also writes `con_000i.img` and `con_sd_000i.img` which are the mean and standard deviation of the i th pre-defined contrast.

Each of these images can be inspected using the ‘Display’ button. Type `help spm_spm_vb` at the matlab command prompt for further information.

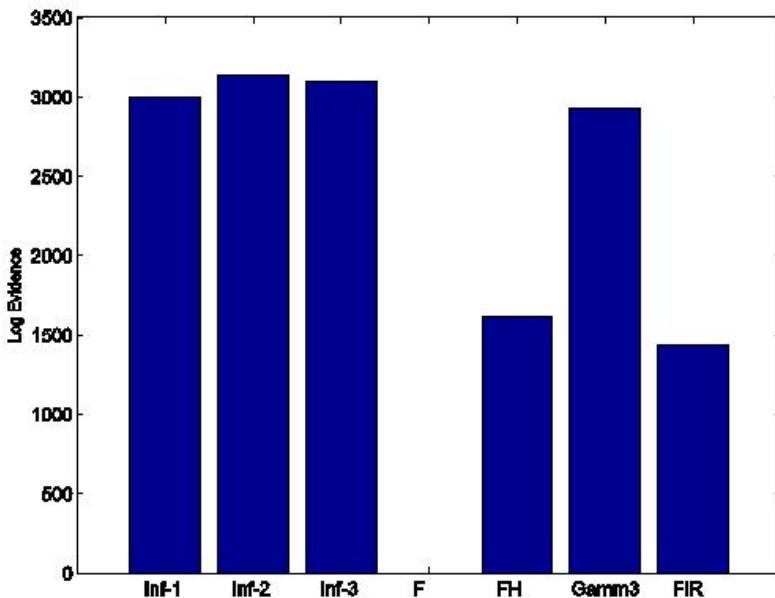


Figure 9.3: This plot shows the model evidence for a number of different hemodynamic basis sets: *Inf1* - Canonical HRF, *Inf2* - Canonical plus temporal derivative, *Inf3* - Canonical plus temporal and dispersion derivatives, *F* - Fourier, *FH* - Fourier with a Hanning Window, *Gamm3* - 3 Gamma basis functions and *FIR* - a Finite Impulse Response function. An informed basis set provides the best model of the data for the selected region.

9.4 Model comparison

Once you have estimated a model you can use SPM's results button to look at the results. You can also extract fMRI data from regions of interest using the ROI button. You can then compare GLMs based on different hemodynamic basis sets using the Bayesian model evidence.

This is described in [?] and implemented using the command line option 'spm_vb_roi_basis'. This requires a VOI filename (created using the ROI button) and an SPM data structure. Type 'help spm_vb_roi_basis' at the matlab command prompt for further information. Figure 9.3 shows an example output from the function indicating that, for the data in this brain region, an informed basis set has the highest model evidence.

Part IV

EEG/MEG

Chapter 10

SPM for EEG/MEG overview

Unlike previous versions, SPM5 provides for the analysis of EEG and MEG data. The initial main motivation for this big leap came from the insight that any integration of modalities like fMRI and EEG should be based on a common theoretical and practical basis. Historically, research into fMRI and EEG/MEG models and analysis has been quite divorced. The same is partially true for the EEG and MEG field. It is our hope that SPM5 provides a common analysis ground for modellers and experimentalists of both the PET/fMRI and EEG/MEG fields.

SPM for EEG/MEG was primarily developed for the analysis of epoched data. This is because we are mostly interested in experiments which perturb the system with a designed stimulus. The analysis of continuous data is typically performed for experiments without designed stimuli like in sleep or epilepsy research. Note that although SPM5 does not provide for an analysis of continuous data per se, many SPM5 routines can be used for these data.

SPM for EEG/MEG can be partitioned into four parts. These are (i) preprocessing, (ii) projection to voxel-space/source reconstruction, (iii) statistical analysis, and (iv) Dynamical Causal Modelling (DCM).

The preprocessing functions provide for simple operations that are standard in other software packages.

The projection to voxel-space is a critical step. When using a source reconstruction, it takes the analysis to brain space. But even when using a simple projection to some 2D-sensor plane, we can then use SPM functions and concepts that were developed for voxel-based data. The projection to a 2D-plane is performed using a simple interpolation and is mostly equivalent to widely used sensor-based analyses. The source reconstruction to brain space is based on models that assume many distributed dipoles in brain space. Solutions to these models typically show dispersed activity and are well-suited for SPM mass-univariate analysis approach.

The statistical analysis is one of the strong points of SPM. PET/fMRI users already familiar with the graphical representation of general linear models won't have difficulties to use SPM for the analysis of EEG/MEG data. SPM5 provides for a comprehensive range of classical linear models that can be used to model the data. These models are basically the same as used in the EEG/MEG field for random effects analyses of multiple subjects. In SPM, we assume that the data in voxel-space is a sampled version of some continuous Gaussian random field. This allows us to use Random field theory (GFT) for the correction of multiple comparisons to control family-wise error over voxels. The GFT approach has the advantage that super-threshold maxima are assessed for their significance, whereas conventional approaches have to specify a-priori the locations of expected activations.

Dynamical Causal Modelling (DCM) is a departure from the mass-univariate approach and is an extension of the SPM software package. DCM for ERP/ERFs is a generative model for evoked responses. The observed data is modelled as the spatiotemporal expression of a small hierarchical network that responds to a stimulus. Differences between evoked responses due to different stimuli are modelled as modulations of the coupling between specific areas. Importantly,

this approach is based on a neurobiologically grounded model. This allows us to obtain parameter estimates that have some physiological interpretation.

The following chapters will go through all the EEG/MEG related functionality of SPM5. All users will probably find the tutorial useful for a quick start. A further detailed description of the preprocessing functions is given in chapter 12. The 3D-source reconstruction is described in chapter 13 and some dipole fitting technique in chapter 15. In chapter 14, we guide you through the modelling of M/EEG data at the first and second level of a hierarchical linear model. Finally, in chapter 16, we describe the graphical user interface for dynamical causal modelling for evoked responses, i.e. event-related potentials (ERPs) and event-related fields (ERFs).

Chapter 11

EEG/MEG preprocessing — Tutorial

This tutorial will give a users guide to the pre-processing sections of SPM M/EEG. We will use an example data set collected on a 128 active electrode Biosemi EEG system. This data set is available from the SPM website. The data was recorded continuously and had three event types (event identifiers 1,2 and 4) These event types indicated the type of visual stimulus presented.

11.1 ERP analysis

11.1.1 Convert

Convert reads the EEG data file and writes the data in a format that is used for SPM analysis. The SPM format has two components a *.mat and *.dat file. The *.mat file contains the data structure D and the *.dat is the M/EEG data.

After clicking on Convert you will be asked to select the format of your EEG or MEG data. For the example data set we select BDF. Next select the data file. For the example data we select the EEGexample.bdf. Next select the data template file from the EEG template file. This file contains a template of electrode positions in 2D. For the example data set select the bdf_setup.mat.

NB the questions then asked depend upon the data format selected. Here we will only address the questions for the BDF format of the example data set.

For BDF files the HEOG, VEOG and any other additional recordings are saved in the EXG channels. Here we attribute labels to these recordings. For the example data set we recorded HEOG (EXG 3 4), VEOG (EXG 4 5) and recorded from the earlobes (EXG 1 2) to allow re-referencing offline. No other additional data were recorded. Therefore we enter the following:

The Convert function writes an EEGexample.mat and an EEGexample.dat file in the current directory.

11.1.2 Epoch

To epoch the data click on epoching. Select the EEG mat file to epoch. Choose the peri-stimulus time window. Choose the events to epoch. Possible events will be listed in the Matlab command window. If you do not have any events in you converted data file you can input them in by reading in a new event list.

For the example data set exampleEEG.mat was selected and the following parameters were used:

Having epoched the data one could filter and downsample the epoched data as above. Epoching writes a *.mat and *.dat file prefixed by a 'e_-'.

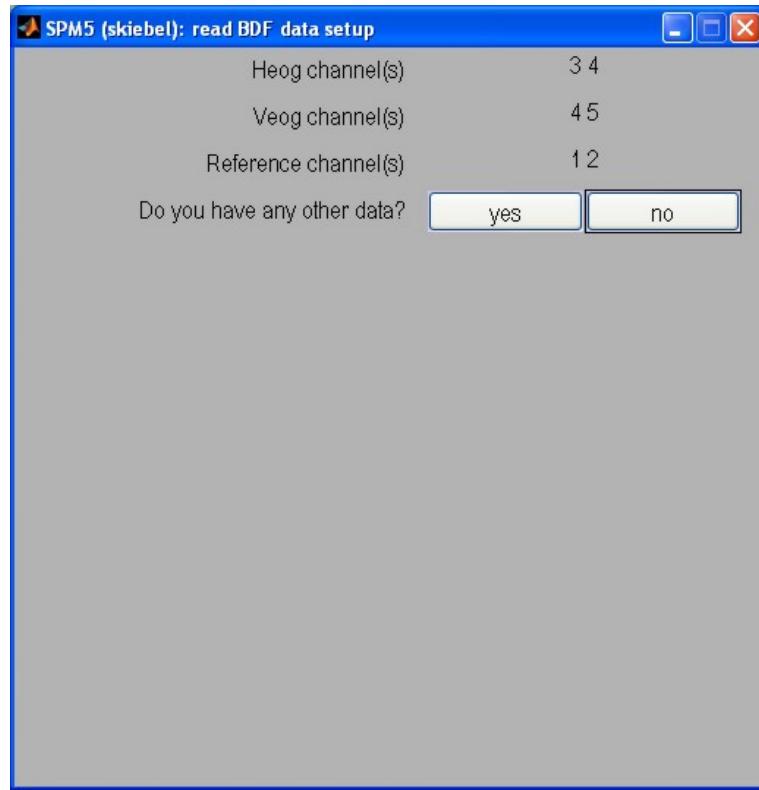


Figure 11.1: *Specifying the options for converting bdf-files*

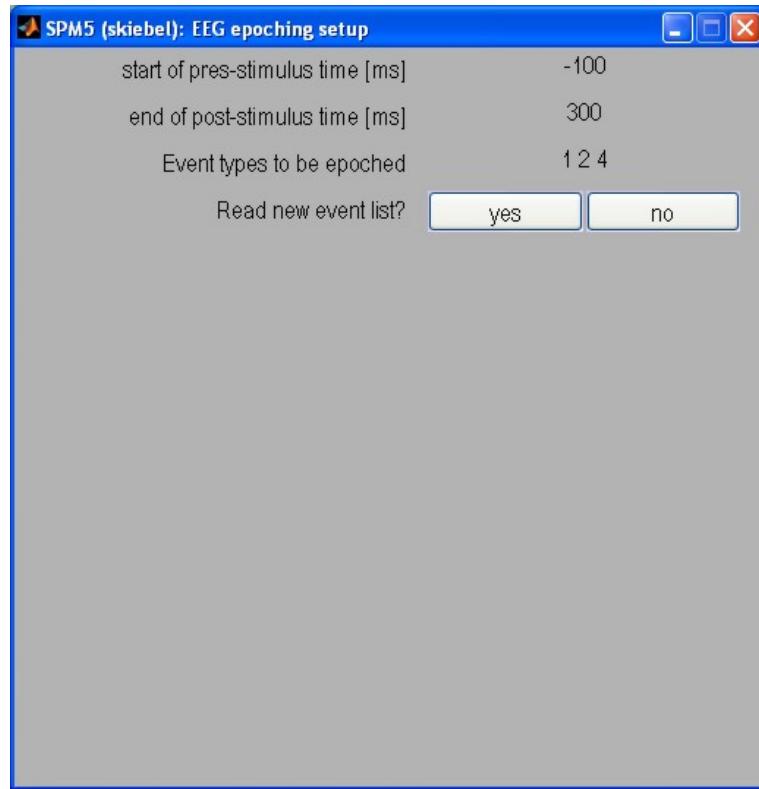


Figure 11.2: *Specifying the options for epoching the example data*

11.1.3 Filter

Filter filters the data using a Butterworth filter. This can be applied either before or after epoching (see 1.3). Here it is applied after.

After clicking Filter select the *.mat file produced by Convert. For the example data set select e_EEGexample.mat.

Next select the filter type, either lowpass or bandpass. If lowpass is selected enter the cut-off frequency. If bandpass is selected enter the two frequencies that specify the band of interest.

The Filter function writes a *.mat and *.dat file prefixed by a 'f'.

For the example data set e_exampleEEG.mat was selected and a bandpass filter was used from 0.1-45 Hz.

11.1.4 Downsample

To downsample the data select downsample from the 'Other' pull-down menu. Select the data file to downsample. Enter the new sampling rate. For the example dataset we downsampled to 100 Hz. Downsample writes a *.mat and *.dat file prefixed by a 'd'.

For the example data set fe_exampleEEG.mat was selected.

11.1.5 Artefacts

Two different methods of artefact removal are implemented in SPM5. One is a simple thresholding method. The other uses a robust averaging methodology to weight each time point by a function of the residuals.

To remove artefacts click on Artefacts. Select the epoched data file to analyse. If you know of bad trials or electrodes that you noted during acquisition or found using another methodology you can read them in using the 'read own artefact list'. If you want to use robust averaging click yes to the following question if not click no. To threshold channels click yes and then enter the threshold you wish to use. The thresholding has two passes. One to find bad electrodes and the second to find bad trials. If robust averaging was selected the second pass will apply the robust averaging approach but a first pass could use a thresholding method to find the bad electrodes prior to robust averaging.

Artefacts writes a *.mat and *.dat file prefixed by an 'a'.

For the example data set dfe_exampleEEG.mat was selected. No artefact list was used and robust averaging was selected with a threshold of 100 v. For the robust averaging the default parameters were used.

11.1.6 Averaging

To produce a mean ERP click on averaging and select the EEG mat file that you wish to average. This will automatically average either ignoring the bad channels and trials or it will apply the weighting matrix calculated from robust averaging. For the example data set adfe_exampleEEG.mat was selected.

Averaging writes a *.mat and *.dat file prefixed by an 'm'.

11.2 Other useful functions

11.2.1 Time-Frequency

In SPM5 it is possible to apply a time-frequency analysis to the data. This can either be applied to the averaged ERP to produce time-frequency plot of evoked oscillations. Or it can be applied to each trial and then averaged across trials to produce time-frequency plot of evoked oscillations

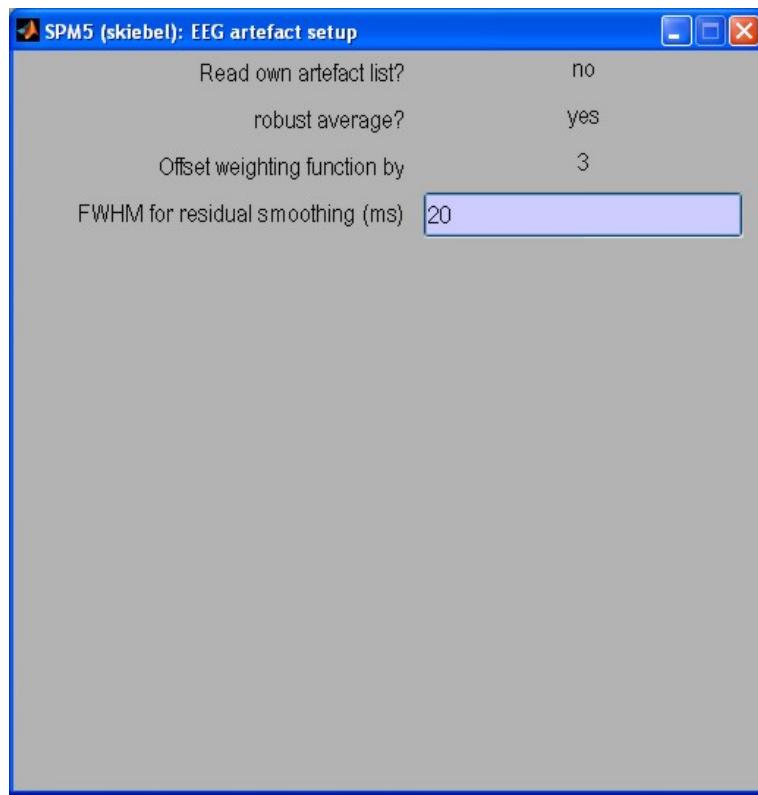


Figure 11.3: *Specifying the options for artefact detection in the example data*

and induced oscillations. To produce time-frequency plots select time-frequency from the other pull-down menu. Select the EEG mat file to analyse. Enter a vector containing the frequencies that you wish to analyse. You are then given the option to baseline correct the time-frequency data. Next enter the Morlet wavelet factor that you wish to use. The default value is 7. Next you can select the channels you wish to analyse. The default is all channels.

Time-Frequency writes two *.mat and *.dat files. The first, the power, is prefixed by 't1'. The second, the phase, is prefixed by 't2'.

11.2.2 Average TF

To average time-frequency data sets across trials select average TF from the other pull down menu. Select the T1*.mat EEG mat file to average. As with the Averaging function Average TF writes a *.mat and *.dat file prefixed by an 'm'.

Chapter 12

EEG/MEG preprocessing — Reference

In this chapter we will describe the purpose and syntax of all SPM/MEEG preprocessing functions. These functions can be called either from SPM’s graphical user interface (GUI) or from the matlab command line. For the command line, we follow the concept of providing only one input argument to each function. This input argument is usually a structure (struct) that contains all input arguments as fields. This has the advantage that the input does not need to follow a specific input argument order. If an obligatory input argument is missing, the function will invoke the GUI and ask the user for the missing argument. When using the GUI, a function is called without any input argument, i.e. SPM will ask for all input arguments. If using the command line, you can specify all arguments in advance and use SPM/MEEG functions in batch mode. We provided some sample batch script (*meeg_preprocess*) in the *man/example_scripts/* folder of the distribution.

12.1 Conversion of data

The conversion of data is necessary to read data from a native machine-dependent format to a matlab-based, common SPM format. This format stores the data in a *.dat file and all other information in a *.mat file. The contents of the *.mat file is a single struct with fields that contain all information about the data (described further below). Currently, SPM can deal with a few formats (s. below).

12.1.1 Converting data yourself

If your format is not one of these, you need to convert the data yourself. This might sound difficult, but it is actually easier than most people think. If things go wrong, the SPM developer team is usually quick to answer questions concerning help with the conversion of data. What we can’t do though is to provide a conversion routine for every M/EEG format. The reason is that there are many formats around, which also evolve over time. To support all these formats would be simply too much work for us.

To write a conversion routine, you (or a helpful colleague) need a minimum knowledge of MatLab. To make things easier and to provide you with a starting point, we wrote a generic conversion script that can be easily modified to work with your specific data. You find this script (*meeg_read_data*) in the *man/example_scripts/* folder of the distribution.

There are three parts to this script. In the first part you provide SPM with information about your data, e.g. sample rate, number of conditions, etc. The second part will read the actual data into the MatLab workspace. To do this you’ll need to write a few lines of matlab. This can be easy, if your data is in ASCII format. It’s more difficult, when the data is still in its native (binary) format. In that case you must know the file specification. Alternatively, you might be fortunate and find free third-party MatLab-software, somewhere on the internet, that does this job for you! In the third part, the data and all information is converted to the SPM format.

The final step is to generate a channel template file. This is necessary to determine the coordinates of the channels in 2D-space. These coordinates are needed for viewing data and projection to voxel-space¹. See below how such a channel template file can be generated. If you're using a standard setup, it's likely that we or someone else have already provided such a file.

12.2 Converting CNT or BDF files

The conversion routine can be started either by using the GUI (Convert) or by calling the function *spm_eeg_convert2mat*. This function is simply a wrapper function that calls the appropriate conversion function.

12.2.1 Syntax

```
D = spm_eeg_converteeg2mat(S)
```

Input

The input struct *S* is optional and has the following optional fields:

- | | |
|------------------|---|
| fmt | - string that determines type of input file. Currently, this string can be either 'CNT', 'BDF', 'EGI-txt' |
| Mname | - char matrix of input file name(s) |
| Fchannels | - String containing name of channel template file |

Output

The output struct *D* contains the header struct of the converted file. This struct has been written to a *.mat file of the same name as the converted file. The data has been written to a corresponding *.dat file.

12.3 BDF data

The Biosemi Data format (BDF) can be converted with the function *spm_eeg_rdata_bdf*. There is no explicit reference electrode, because the Biosemi system uses reference-free measurements. Nearly all information is contained in the raw *.bdf file. The only information that is not in the file is the actual usage of the 8 external channels. Typically these are used for EOG and some reference measurement. These information must be supplied to SPM. Keep in mind that a false declaration of external channels can severely degrade the quality of your data.

The conversion routine can be started by calling the function *spm_eeg_rdata_bdf*.

12.3.1 Syntax

```
D = spm_eeg_rdata_bdf(S)
```

Input

The input struct *S* is optional and has the following optional fields:

¹If you don't want to look at your data, or project to 2D voxel-space, you can actually proceed without this channel template file.

Fdata	-	filename of bdf-file
Fchannels	-	String containing name of channel template file
Cheog	-	One or two indices of external channels used for HEOG. Valid indices lie between 1 and 8.
Cveog	-	indices (1 - 8) of external channels used for VEOG. Valid indices lie between 1 and 8.
Creference	-	indices (1 - 8) of external channels used for reference. Valid indices lie between 1 and 8.

Output

The output struct D contains the header struct of the converted file. This struct has been written to a *.mat file of the same name as the converted file. The data has been written to a corresponding *.dat file.

12.4 CNT data

The neuroscan CNT format can be converted with the function *spm_eeg_rdata*. Nearly all information is contained in the raw *.cnt file. The only information that is not in there is about the reference used. SPM will ask you explicitly about the reference. However, if you don't want to re-reference your data at a later stage, you don't need to supply information about the reference electrode. The same is true, if you want to re-reference, but don't want to transform the reference channel again to an EEG channel. For example, this is the case when the reference were the earlobes.

The conversion routine can be started by calling the function *spm_eeg_rdata*.

12.4.1 Syntax

$D = \text{spm_eeg_rdata}(S)$

Input

The input struct S is optional and has the following optional fields:

Fdata	-	filename of CNT-file
Fchannels	-	String containing name of channel template file
reference	-	name of reference channel. If you want to make this channel an EEG channel at a later re-referencing, you need to supply the exact name of the channel. If you just want to store the reference name (e.g. earlobes), just enter any descriptive text.

Output

The output struct D contains the header struct of the converted file. This struct has been written to a *.mat file of the same name as the converted file. The data has been written to a corresponding *.dat file.

12.5 The MEEG SPM format

The SPM-format stores the binary data in a *.dat file. All header information are stored in a *.mat file. This *.mat file contains a single struct named D which contains several fields. Note that the data should always be read using the routine *spm_eeg_ldata*, see section 12.6. In the following, we will first describe all single-element entries, and then all entries that are itself structs.

Nchannels	- The number of channels. This number also includes channels like EOG or other external channels
Nevents	- The number of epochs
Nsamples	- The number of time bins in one epoch
Radc	- The sampling rate measured in Hertz
fnamedat	- The name of the *.dat file without leading path
fname	- The name of the *.mat file without leading path
path	- The path to the directory where the *.mat and *.dat file are stored
datatype	- The datatype with which the data in the *.dat file is stored. Possible datatypes are 'int16' and 'float'
data	- This is a <code>spm_file_array</code> struct that contains the memory mapped data. For epoch data this is effectively a three-dimensional array of the dimensions <i>Nchannels</i> × <i>Nsamples</i> × <i>Nevents</i> .
scale	- A matrix with internally used scaling values for the memory mapping of data. For documentation, see the directory <code>file_array</code> in the SPM main directory.
modality	- A string that is (currently) either 'EEG' or 'MEG' and describes the type of data in the file
units	- A string that determines the units of the data in tex-format, e.g. μV for micro V

12.5.1 channels

The substruct `channels` contains all channel-related information.

ctf	- The name of a channel template file (CTF) without leading path. It is assumed that the CTF is located in the <code>EEGtemplates</code> sub-directory of the SPM5 main directory. This file contains standard channel names for a given setup and their coordinates on a 2D plane. When converting a file to the SPM-format, a link is made to a CTF. Identification of channels in the data file is via the channel names. (See also sec. 12.7)
Bad	- An index vector of bad channels
name	- A cell vector of channel names
eeg	- The indices of actual EEG channels. For example, these exclude the EOG channels.
order	- An index vector that maps from the data order of channels to the corresponding channel in the CTF.
heog	- The channel index of the HEOG channel.
veog	- The channel index of the VEOG channel.
reference	- If available, this is an index of the reference channel in the order of the CTF. Otherwise this is 0.
ref_name	- If available, the name of the reference channel. This actually does not need to be a valid channel name, but is just used as a reminder for the user (e.g. 'earlobes').
thresholded	- A cell vector that contain channel indices (in data order) of epochs with data surpassing some threshold. This is usually generated by the <code>spm_eeg_artefact</code> function.

12.5.2 events

The substruct `events` contains information related to the epochs.

- | | |
|---------------|---|
| code | - A vector which contains event numbers for each event. These were read during the conversion from the event channel of the raw data. |
| time | - A vector which contains the timing of stimulus presentation for each event (measured in time bins). This is used for epoching the data. |
| start | - The number of time bins before onset of the stimulus |
| stop | - The number of time bins after onset of the stimulus |
| Ntypes | - The number of different event types |
| reject | - A vector which for each event a 0 or 1's indicating whether this trial was rejected or not. |
| repl | - A vector with the number of single trials which were used for each event-type by the averaging function. |

12.5.3 filter

The substruct *filter* contains information about filtering the data. This struct is usually filled by the function *spm_eeg_filter*.

- | | |
|-------------|---|
| type | - The name of the used filter |
| band | - 'lowpass' or 'bandpass' |
| PHz | - The cutoff of the filter in Hertz |
| para | - A cell vector with filter parameters. See the matlab function <i>filter</i> for a description of what these parameters are. |

12.5.4 threshold

The substruct *threshold* contains information about thresholding the data. This struct is usually filled by the function *spm_eeg_artefact*.

- | | |
|----------------------|---|
| External_list | - Indicator (0/1) whether external information was used which trials were artefactual or clean. |
| threshold | - The threshold used in microVolt |

12.6 Reading of data

Once the data is in SPM-format, it can be read into matlab. This should be done using the routine *spm_eeg_ldata*. (Note: If you only work with the GUI, you won't need to call this function.) The routine will mainly do two things. First, it will load the header struct in the *.mat-file. Secondly, it will memory map the data in the *.dat file to a field in this struct. The memory mapped data can be addressed like a matrix which is convenient for accessing the data in a random access way. However, a word of caution: If you write new values to the D.data-matrix, the matrix is not only changed in the matlab variable (in memory), but also physically on the hard disk.

This function can only be called via matlab command line.

12.6.1 Syntax

$$D = \text{spm_eeg_ldata}(P)$$

Input

The input string *P* is optional and contains the file name of the *.mat file.

Output

The output struct *D* contains all header information about the data. The data are memory mapped and can be accessed as the field *data*.

12.7 The channel template file

The channel template file is SPM's way of connecting acquired channel data to a spatial location. The locations of channels are typically not contained in the MEEG raw data file. A channel template file contains channel names for a given setup and their locations in some coordinate system. All channel template files are contained in the subdirectory *EEGtemplates* in the SPM5-directory. During the initial conversion of data each channel is identified by its name and mapped to the corresponding channel location contained in the channel template file. If a channel's name is not contained in the user-specified channel template file, a warning is issued. Many warnings usually mean that the wrong channel template file for a specific setup was selected. Note that even if the mapping from channels to their locations were not identified correctly, it is still possible to perform preprocessing operations (e.g. epoching, filtering, etc.) on the converted data. However, the channels' locations are needed for display and mapping to voxel space.

Currently, the channel template files in SPM5b all map into some standard 2D space on the scalp. This is useful for mapping multiple subjects' data to a standard space and performing SPM analyses in 2D scalp space. Future updates of SPM5b will supply channel template files that map to a 3D sensor space, which is critical for 3D source reconstruction. This 3D space can be some standard space which might be useful for MEG data and EEG data acquired with a cap. Alternatively, one can also use digitized sensor positions as locations, e.g. acquired with a Polymus system.

12.7.1 Structure

A channel template file (CTF) is a mat-file that contains four variables:

- | | |
|------------------|--|
| Nchannels | - The number of channels known to the CTF |
| Cnames | - A cell vector of channel names. Each cell can contain either a string or a cell vector of strings. The latter allows to have multiple versions of a given channel name. Case can be ignored, i.e. it doesn't matter whether channel names are in small or capital letters. |
| Cpos | - A $2 \times Nchannels$ -matrix of channel coordinates on a 2D plane. In x - and y -direction the minimum coordinate must be ≤ 0.05 and the maximum coordinate must be ≥ 0.95 . |
| Rxy | - A factor that determines the width of the display plots to their height when displaying the data. Standard is 1.5. |

12.7.2 Creating your own channel template file

The channel template file is important for using SPM's full functionality for MEEG data. The channel template files contained in the *EEGtemplates* directory are the ones that we or our collaborators found useful. Other groups will need different channel template files, because they might have different setups, i.e. they use different channel names and/or different channel coordinates. Note that if a specific setup is just a subset of channels of an existing setup, the CTF of the full setup can be used.

If a new channel template file is needed, this can be simply created by saving the variables *Nchannels*, *Cnames*, *Cpos* and *Rxy* to a new channel template file. Typically this would be done by running a script that creates these four variables and saves them to a file. The creative bit is to list the actual coordinates of the channels on a 2D plane. We found two feasible ways for doing this. The first is to note that many electrode setups consist of electrodes sitting on concentric rings equidistant to other electrodes on each ring. Such a setup can be programmed as a script which places electrodes on each of these rings. A second way is that at least some producers of EEG caps provide coordinates for specific setups in 3D space. For example, have a look at http://www.easycap.de/easycap/e/downloads/electrode_sites_coordinates.htm. The projection to 2D coordinates could be done by first using Matlab's *sph2cart* function to transform to Cartesian coordinates. This is followed by applying the subfunction *CartToFlat* of *spm_eeg_DrawSV* (SPM5 DipoleFit toolbox) to the Cartesian coordinates. We provided an ex-

ample script (*make_Easycap_montage1*) in the *man/example_scripts/* folder of the distribution to illustrate this process.

12.8 Epoching the data

Epoching cuts out little chunks of data and saves them as 'single trials'. For each stimulus onset, the epoched trial starts at some user-specified pre-stimulus time and end at some post-stimulus time, e.g. from -100 to 400 milliseconds in peri-stimulus time. The epoched data is also baseline-corrected, i.e. the mean of the pre-stimulus time is subtracted from the whole trial. The resulting event codes are the same as saved in the *.mat file. One can re-code events by supplying a vector of event codes.

The epoching routine can be started either by using the GUI (Epoching) or by calling the function *spm_eeg_epochs*.

12.8.1 Syntax

$D = \text{spm_eeg_epochs}(S)$

Input

The input struct S is optional and has the following optional fields:

D	- filename of MEEG mat file
events	- a struct containing the following fields
start	- pre-stimulus start of epoch[ms]
stop	- post-stimulus end of epoch[ms]
types	- vector of event types to extract
Inewlist	- indicate (0/1) to use new list of event codes
Ec	- vector of new event codes

Output

The output struct D contains the header struct of the epoched file. This struct has been written to a *.mat file of the same name as the input file, but prepended with *e_*. The data has been written to a corresponding *.dat file.

12.9 Filtering the data

Continuous or epoched data can be filtered with a low- or bandpass-filter. SPM uses a Butterworth filter to do this. Phase delays are minimised by using matlab's *filtfilt* function which filters the data twice, forwards and backwards. SPM's filter function *spm_eeg_filter* uses matlab's signal processing toolbox. If you don't have this toolbox, you cannot filter your data using SPM.

The filter routine can be started either by using the GUI (Filter) or by calling the function *spm_eeg_filter*.

12.9.1 Syntax

$D = \text{spm_eeg_filter}(S)$

Input

The input struct S is optional and has the following optional fields:

D	- filename of MEEG mat file
filter	- a struct containing the following fields
type	- type of filter, currently must be 'butterworth'
band	- a string, 'lowpass' or 'bandpass'
PHz	- one (lowpass) or two (bandpass) cut-offs [Hz]

Output

The output struct *D* contains the header struct of the filtered file. This struct has been written to a *.mat file of the same name as the input file, but prepended with *f*. The data has been written to a corresponding *.dat file.

12.10 Artefact detection and rejection

Some trials are likely to not only contain neuronal signals of interest, but also signal from other sources like eye movements or muscular activity. These signal components are referred to as artefacts. In SPM, we use only two simple automatic artefact detection schemes. The first is thresholding the data and the second is robust averaging. One can also choose to detect artefacts manually by visualizing each trial (see below). Another option is to use a more sophisticated artefact detection approach (implemented by some other software) and supply that information to SPM.

Thresholding the data is done in two passes. In the first pass, SPM detects all instances for which the threshold was passed by the absolute value for a channel and single trial. If a channel has more than a certain percentage of artefactual trials, it is defined as a bad channel. In a second pass the thresholding is repeated, but without taking into account bad channels. A trial for which the absolute data surpasses the threshold in some channel (excluding bad channels) is considered artefactual.

The function only indicates which trials are artefactual or clean and subsequent processing steps (e.g. averaging) will take this information into account. However, no data is actually removed from the *.dat file.

The artefact routine can be started either by using the GUI (Artefacts) or by calling the function *spm_eeg_artefact*.

12.10.1 Syntax

$$D = \text{spm_eeg_artefact}(S)$$

Input

The input struct *S* is optional and has the following optional fields:

D	- filename of MEEG mat file
thresholds	- a struct containing the following fields
External_list	- indicate (0/1) to use external artefact list
out_list	- index vector of artefactual trials
in_list	- index vector of clean trials
Check_Threshold	- indicate (0/1) whether to threshold channels
threshold	- threshold to use: can be either a scalar which is the threshold for all channels, or a vector of channel-wise thresholds
artefact	
in_list	- index vector of clean trials
weighted	- a struct containing the following fields
wtrials	- indicate (0/1) whether to use robust averaging
	- indicate (0/1) whether to use robust averaging across trials

Output

The output struct D contains the header struct of the artefact-detected file. This struct has been written to a *.mat file of the same name as the input file, but prepended with a . The data has been written to a corresponding *.dat file.

12.11 Downsampling

The data can be downsampled to any sample rate. This is useful if the data was acquired at a higher sampling rate than one needs for making inferences about low-frequency components. SPM's downsampling routine uses the matlab function *resample*, which is part of matlab's signal processing toolbox. If you don't have this toolbox, you cannot downsample your data using SPM.

The downsampling routine can be started either by using the GUI (Other/downsample) or by calling the function *spm_eeg_downsample*.

12.11.1 Syntax

$D = \text{spm_eeg_downsample}(S)$

Input

The input struct S is optional and has the following optional fields:

- | | |
|-----------------|----------------------------------|
| D | - filename of MEEG mat file |
| Radc_new | - the new sampling rate in Hertz |

Output

The output struct D contains the header struct of the downsampled file. This struct has been written to a *.mat file of the same name as the input file, but prepended with d . The data has been written to a corresponding *.dat file.

12.12 Rereferencing

When you acquired data to a certain reference, you can simply re-reference the data to another channel or to the average over a set of channels. Bad channels are excluded from an average reference. If there was only a single reference channel before, one can add it again to the data. The rereferencing routine displays the indices of all channels of the data as a help to decide which indices to select as a new reference.

The rereferencing routine can be started either by using the GUI (Other/rerefence) or by calling the function *spm_eeg_rerefence*.

12.12.1 Syntax

$D = \text{spm_eeg_rerefence}(S)$

Input

The input struct S is optional and has the following optional fields:

- | | |
|---------------|--|
| D | - filename of MEEG mat file |
| newref | - a struct containing the following fields |

Output

The output struct D contains the header struct of the rereferenced file. This struct has been written to a *.mat file of the same name as the input file, but prepended with R . The data has been written to a corresponding *.dat file.

12.13 Grand mean

The grand mean is usually understood as the average of ERPs over subjects. The grand mean function in SPM is typically used to do exactly this, but can also be used to average over multiple EEG files, e.g. multiple sessions of a single subject. The averaged file will be written into the same directory as the first selected file.

The grand mean routine can be started either by using the GUI (Other/grand mean) or by calling the function *spm_eeg_grandmean*.

12.13.1 Syntax

$$D = \text{spm_eeg_grandmean}(S)$$

Input

The input struct *S* is optional and has the following optional fields:

P - filenames of M/EEG mat files (char matrix)

Output

The output struct *D* contains the header struct of the averaged file. This struct has been written to a *.mat file of the same name as the input file, but prepended with *g*. The data has been written to a corresponding *.dat file.

12.14 Merge

Merging several MEEG files can be useful for concatenating multiple sessions of a single subject. Another use is to merge files and then use the display tool on the concatenated file. This is the preferred way in SPM to display data together that is split up into several files. The merged file will be written into the same directory as the first selected file.

The merge routine can be started either by using the GUI (Other/merge) or by calling the function *spm_eeg_merge*.

12.14.1 Syntax

$$D = \text{spm_eeg_merge}(S)$$

Input

The input struct *S* is optional and has the following optional fields:

P - filenames of MEEG mat files (char matrix)

Output

The output struct *D* contains the header struct of the merged file. This struct has been written to a *.mat file of the same name as the input file, but prepended with *c*. The data has been written to a corresponding *.dat file.

12.15 Time-frequency decomposition

The time-frequency decomposition is performed by using a continuous Morlet wavelet transform. The result is written as two result files, one contains the instantaneous power and the other the phase estimates. One can select the channels and frequencies for which power and phase should be estimated. Optionally, one can apply a baseline correction to the power estimates, i.e. the mean power of the pre-stimulus time is subtracted from the power estimates.

The time-frequency decomposition routine can be started either by using the GUI (Other/time-frequency) or by calling the function *spm_eeg_tf*.

12.15.1 Syntax

$D = \text{spm_eeg_tf}(S)$

Input

The input struct S is optional and has the following optional fields:

D	- filename of MEEG mat file
frequencies	- vector of frequencies [Hertz] at which decomposition is performed
rm_baseline	- indicate (0/1) whether baseline should be subtracted
Sbaseline	- start and stop of baseline (in time bins)
channels	- indices of channels for which to perform time-frequency decomposition
Mfactor	- the so called Morlet wavelet factor, defaults to 7.

Output

The output struct D contains the header struct of the phase information. This struct has been written to a *.mat file of the same name as the input file, but prepended with *t2*_. The data has been written to a corresponding *.dat file. The power data has been written to a file prepended with *t1*_.

12.16 Averaging

Averaging of the single trial data is the crucial step to obtain the ERP. By default, when averaging single trial data, single trials are averaged within trial type. Power data of single trials (see sec. 12.15) can also be averaged.

The averaging routine can be started either by using the GUI (average) or by calling the function *spm_eeg_average*.

12.16.1 Syntax

$D = \text{spm_eeg_average}(S)$

Input

The input struct S is optional and has the following optional fields:

D	- filename of MEEG mat file
----------	-----------------------------

12.17 Linear combinations of epochs

As an extension to the averaging functionality, SPM can also be used to compute linear combinations of single trials or epochs. For example, you might be interested in computing the difference between two ERPs. This can be done by calling the function *spm_eeg_weight_epochs*.

12.17.1 Syntax

$D = \text{spm_eeg_weight_epochs}(S)$

Input

The input struct S is optional and has the following optional fields:

- D** - filename of MEEG mat file
- c** - a weight (contrast) matrix with dimensions $N_{contrasts} \times N_{epochs}$. Each row of c contains one contrast vector. For a simple difference between two ERPs use $[-1 1]$.

Output

The output struct D contains the header struct of the averaged file. This struct has been written to a *.mat file of the same name as the input file, but prepended with m . The data has been written to a corresponding *.dat file.

12.18 Mixing of channels

SPM can also be used to compute the mixing of channels by a square matrix. For example, we found this useful for computing a weighting of the data with an independent component analysis (ICA) mixing matrix. You can do this by calling the function `spm_eeg_weight_channels`.

12.18.1 Syntax

$D = \text{spm_eeg_weight_channels}(S)$

Input

The input struct S is optional and has the following optional fields:

- D** - filename of MEEG mat file
- W** - a mixing matrix with dimensions $N_{channels} \times N_{channels}$.
Hint: If you call the function without arguments, prepare a variable that contains this matrix.

Output

The output struct D contains the header struct of the averaged file. This struct has been written to a *.mat file of the same name as the input file, but prepended with w . The data has been written to a corresponding *.dat file.

12.19 Weighting of the time-series

You can use SPM to multiply your data, over peri-stimulus time, by some weighting function. For example, we found this useful for removing stimulus-related artefacts due to an electrical impulse at peri-stimulus time 0. The weighting would be a function over peri-stimulus time consisting of 1s everywhere, except for time 0, where you would remove data by putting in a 0. You can do this by calling the function `spm_eeg_weight_time`.

12.19.1 Syntax

$D = \text{spm_eeg_weight_time}(S)$

Input

The input struct S is optional and has the following optional fields:

- | | |
|---------------|--|
| D | - filename of MEEG mat file |
| weight | - a weighting function (vector) with length peri-stimulus time points. Hint: If you call the function without arguments, prepare a variable that contains this vector. |

Output

The output struct *D* contains the header struct of the averaged file. This struct has been written to a *.mat file of the same name as the input file, but prepended with *w*. The data has been written to a corresponding *.dat file.

12.20 Displaying data

SPM can be used to display epoched data. The viewer is called by choosing the EEG/MEG modality and clicking on the *M/EEG* entry in the *Display* menu. After selecting an epoched M/EEG file in SPM format, the viewer displays all channels of the first trial or trial type. The position of the channels is taken from the channel template file (see above).

Navigation through trials or trial types is either by the trial slider or by the trial listbox. The scaling of the displayed data (for EEG: μ Volt, for MEG: in 10^2 femto Tesla) can be changed by using the scaling slider. Up to four trials or trial types can be plotted at the same time by using the shift or ctrl-button while selecting files with the left mouse button in the listbox.

Single trials can be classified as either artefactual or clean by pressing the *Reject* button. This information can be saved to the *.mat file by pressing the *Save* button.

A left-clicking on a single channel plot will plot the time-series of this channel in much more detail in a pop-up figure. Another left-click on the (small) channel plot will close the pop-up figure again.

The topography at a specific time point can be displayed either in 2D or in 3D by clicking on the *Topography* button and selecting a peri-stimulus time and choosing between 2D/3D. In this display, bad channels will not be interpolated, but no data is plotted in the location of the bad channel.

The set of displayed channels can be changed by clicking on the *channel* button. This is useful for (i) faster plotting of single trial data by choosing less channels to display and (ii) having larger plots in the display tool. In the channel select tool, you can click on the channel to select or deselect this channel. Alternatively, you can also use the listbox with shift/ctrl to select or deselect channels. Channel selection can be saved and loaded to mat-files. Pressing *Ok* will confirm your selected channels and update your display.

Chapter 13

3D source reconstruction: Imaging approach

Here is a brief help to the 3D reconstruction based on the Imaging approach. In the near future, this will be improved by including more theoretical details upon the different procedures as well as a practical tutorial that will guide the user through the SPM interface via the analysis of a sample dataset.

13.1 Introduction

This chapter focuses on the imaging (or distributed) method for doing EEG/MEG source reconstruction in SPM. Such an approach to spatial projection onto (3D) brain space consists in considering a large amount of dipolar sources all over the cortical sheet, with fixed locations and orientations. This renders the observation model linear, the unknown variables being the source amplitudes or power.

Given epoched and preprocessed data (see chapter ...), the evoked and/or induced activity for each dipolar source can be estimated, for a single time-sample or a wider peristimulus time window.

The obtained reconstructed activity is in 3D voxel space and enables mass-univariate analysis in SPM (see chapter...).

Contrary to PET/fMRI data reconstruction, EEG/MEG source reconstruction is a non trivial operation. Often compared to estimating a body shape from its shadow, inferring brain activity from scalp data is mathematically ill-posed and requires prior information such as anatomical, functional or mathematical constraints to isolate a unique and most probable solution [9].

Distributed linear models have been around for more than a decade now [15] and the proposed pipeline in SPM for 'Imaging' solution is classical and very similar to common approaches in the field. However, at least two aspects are quite original and should be emphasized here:

- Based on an empirical Bayesian formalism, the inversion is meant to be generic in the sense it can incorporate and estimate the relevance of multiple constraints of various nature; data-driven relevance estimation being made possible through Bayesian model comparison [24, ?, ?, 21].
- The subject's specific anatomy is incorporated in the generative model of the data, in a fashion that eschews individual cortical surface extraction. The individual cortical mesh is obtained automatically from a canonical mesh in MNI space, providing a simple and efficient way of reporting results in stereotactic coordinates.

The EEG/MEG imaging pipeline is divided into four consecutive steps which characterize any inverse procedure. In this chapter, we go through each of those steps that all need to be completed when proceeding with a full inverse analysis:

1. Source space modeling,

2. Data co-registration,
3. Forward computation,
4. Inverse reconstruction.

Whereas the three first steps are part of the whole generative model, the last step consists in the Bayesian inversion and is the only one involving the actual EEG/MEG data.

Everything which is described hereafter is a new feature in SPM and is accessible from SPM5 user-interface by choosing the 'EEG/MEG' application, '3D source reconstruction' and 'Imaging'.

13.2 Data structure

The Matlab structure describing a given EEG/MEG dataset in SPM is denoted as D . Within that structure, each new inverse analysis will be described by a new cell of sub-structure field $D.inv$ and will be made of the following fields:

- *method*: character string indicating the method, either 'ECD' or 'Imaging' in present case;
- *mesh*: sub-structure with relevant variables and filenames for source space and head modeling;
- *datareg*: sub-structure with relevant variables and filenames for EEG/MEG data registration into MRI space;
- *forward*: sub-structure with relevant variables and filenames for forward computation;
- *inverse*: sub-structure with relevant variable, filenames as well as results files;
- *comment*: character string provided by the user to characterize the present analysis;
- *date*: date of the last modification made to this analysis.

13.3 Source space modeling (*mesh*)

The individual cortical mesh is obtained from a template mesh. Four Mesh sizes are available (3004, 4004, 5004 and 7204 vertices). If not yet obtained, the spatial normalization of the subject's T1 MRI into MNI space is performed (see *spm_preproc.m* based on tissue probability maps). The inverse of that transformation is computed and applied to the template mesh to furnish the individual cortical mesh.

Individual meshes for the inner-skull and scalp surfaces are also computed from the individual T1 MRI. They are obtained by performing a binary mask of the the volumes delimited by the inner-skull and scalp surface respectively. Then, using an initial spherical mesh, a realistic-shaped mesh is obtained for each of the two tissues and further regularized via an erosion and growing procedure.

The meshing module includes the following functions:

- *spm_eeg_inv_mesh_ui.m*: run the user interface for this module,
- *spm_eeg_inv_spatnorm.m*: normalize the T1 image if needed,
- *spm_eeg_inv_meshing.m*: main function to produce Cortex, Inner-skull and Scalp meshes,
- *spm_eeg_inv_getmasks.m*: produce masks of Inner-skull and Scalp,
- *spm_eeg_inv_ErodeGrow.m*: erosion and growing procedure,
- *spm_eeg_inv_getmeshes.m*: obtains the inner-skull and scalp meshes from correpsonding binary masks,

- *spm_eeg_inv_CtrBin.m*
- *spm_eeg_inv_TesBin.m*
- *spm_eeg_inv_ElastM.m*
- *spm_eeg_inv_checkmeshes.m*: displays the computed three meshes in the SPM main figure

13.4 Data Registration (*datareg*)

There are two possible ways of coregistering the EEG/MEG data into the structural MRI space.

1. A Landmark based coregistration (using fiducials only).

The rigid transformation matrices (Rotation and Translation) are computed such that they match each fiducial in the EEG/MEG space into the corresponding one in sMRI space. The same transformation is then applied to the sensor positions.

2. Surface matching (between some headshape in MEG/EEG space and some sMRI derived scalp tesselation). For EEG, the sensor locations can be used instead of the headshape. For MEG, the headshape is first coregistered into sMRI space; the same transformation is then applied to the sensors.

Surface matching is performed using an Iterative Closest Point algorithm (ICP). The ICP algorithm [10] is an iterative alignment algorithm that works in three phases:

- Establish correspondence between pairs of features in the two structures that are to be aligned based on proximity;
- Estimate the rigid transformation that best maps the first member of the pair onto the second;
- Apply that transformation to all features in the first structure. These three steps are then reapplied until convergence is concluded. Although simple, the algorithm works quite effectively when given a good initial estimate.

The data-registration module includes the following functions:

- *spm_eeg_inv_datareg_ui.m*: run the user interface for this module,
- *spm_eeg_inv_datareg.m*: main co-registration function,
- *spm_eeg_inv_checkdatareg.m*: display meshes, sensor locations and fiducials in native MRI space to enable one checking the co-registration by eye.

13.5 Forward computation (*forward*)

Several methods are proposed, depending on the modality (EEG or MEG). All these approaches/functions are identical to the one initially developed and provided by the BrainSTorm package (Matlab open-source and free software: <http://neuroimage.usc.edu/brainstorm/>).

For EEG [18]:

1. single sphere (scalp surface),
2. three spheres (inner, outer skull and scalp surfaces),
3. three spheres (+ Berg correction),
4. overlapping spheres (one fitted sphere per sensor).

For MEG [29]:

1. single sphere,
2. overlapping spheres

The forward module includes the following functions:

1. *spm_eeg_inv_forward_ui.m*: run the user interface for this module,
2. *spm_eeg_inv_BSTcreatefiles.m*: create the structure and required files and parameters to interface SPM and BrainSTorm,
3. *spm_eeg_inv_BSTfwdsol.m*: compute the BrainSTorm forward solution, calling function *bst_headmodeler.m*,
4. *spm_eeg_inv_PCAgain*: compute the svd of the gain matrix.

13.6 Inverse reconstruction (*inverse*)

The reconstruction is based on an empirical Bayesian approach to localize either the evoked response, the evoked power or the induced power, as measured by EEG or MEG.

The inverse module includes the following functions:

- *spm_eeg_inv_inverse_ui.m*: run the user interface for this module,
- *spm_eeg_inv_inverse.m*: main function,
- *spm_eeg_inv_evoked.m*: compute the evoked response,
- *spm_eeg_inv_induced.m*: compute the evoked and/or induced power,
- *spm_eeg_inv_msp.m*: Multivariate Source Prelocalisation [35].

Chapter 14

M/EEG modelling and statistics

After projection to 2D- or a 3D-space (source reconstruction), the data is in voxel-space and ready to be analysed. There are several ways how one can proceed. In this chapter, we will focus on analyzing epoched time-series data. These can be event-related responses (ERPs), event-related fields (ERFs) or single trials (M/EEG).

In the following, we will go through the various stages of modelling using typical examples to illustrate the procedures.

14.1 Preliminary remarks

All analyses can be done using either the graphical user interface (GUI) or a batch system (i.e. using scripts in the SPM2-fashion, s. below). The GUI has the advantage that one doesn't need matlab-knowledge to analyse data. The batch system has the advantage that it is a fast and efficient way of entering model and data. Its disadvantage is that some Matlab-Knowledge is required. However, with this distribution, we provide some template scripts to analyse (typical) data in batch mode. We assume that with slight modifications these scripts can be used for most analyses.

14.2 How epoched time-series are analysed in SPM

After preprocessing the data (i.e. epoching, filtering, etc...) and projection to voxel-space, we have discretely sampled versions of continuous fields [31]. These data can be analysed with a mass-univariate approach using results from Random Field theory (RFT) to adjust p-values for multiple comparisons [48]. The model used at each voxel is a general linear model [32]. Typically one wants to analyse multiple subjects' data acquired under multiple conditions. Given that each evoked-response has up to hundreds of time points, this is an awful lot of data at each voxel. The ideal way to analyse these data would be to specify a single hierarchical model (1st level: within-subject, 2nd level: over subjects) and estimate its parameters. However, this is computationally not feasible because of the length of the data vector at each voxel. Fortunately, such a 2-level model can usually be split up into two models: The 1st level and the 2nd level model. The input data to the 2nd model are contrasts of the 1st level model [32]. In all cases considered in this chapter, this 2-stage procedure gives exactly the same results as the 2-level model. The reason for this is that we are not really *modelling* the data at the 1st level, but simply forming weighted sums of the data, over time. For example, if we are interested in the N170 component, one could average the data from 150 to 190 milliseconds. This is exactly the approach used in conventional ERP analysis. This approach is not a model, because simply taking sums corresponds to using an identity matrix as design matrix. This procedure leaves no degrees of freedom for error estimation.

In summary, the SPM-approach is to form, at each voxel, weighted sums of the data, over time, at the 1st level. We refer to these weighted sums as contrast images. These form the input to the 2nd level, where one usually tests for differences between conditions or between groups (s. below). The second level models are usually the same as the ones one would use for functional

magnetic resonance imaging (fMRI). Importantly, these 2nd level models have enough degrees of freedom to estimate the error, i.e. statistics can be computed.

The output of such a 2nd level analysis is a voxel-volume (or map), where each voxel contains one statistical value. The associated p-value is adjusted for multiple comparisons [48]. This adjustment is important, because there are many other voxels or channels. One (disadvantageous) alternative to adjustment is to consider only pre-selected channels or averages over channels. This is why the adjustment is especially important for high-density measurements, because there are many channels to select from. We believe that it is generally too subjective to select channels for analysis a-priori. We see the GFT-adjustment as a good way of looking at the whole data without any prior selection. This has been already demonstrated for EEG data (in another context) by [33].

14.3 1st level

At the 1st level, we select periods or time points in peri-stimulus time that we would like to analyse. Critically, this choice must be made a-priori by you. The alternative would be to not treat peri-stimulus time as a factor, but as a dimension of a Random Field approach. This alternative approach is often used in time-frequency analysis of induced and evoked oscillations, where it seems sometimes difficult to specify areas of interest on the time-frequency plane a-priori [33].

In the present approach, time is a factor, and you have to form weighted-sums over peri-stimulus time to provide input to the 2nd level. Of course, you don't need to constrain yourself to a single contrast around a specific peri-stimulus time, but you can compute as many as you like. For example, to analyse multiple aspects of an ERP, it is not uncommon to form averages around several time-points of an ERP. At the 2nd level, these can be either analysed independently or within one model to make inferences about interactions between conditions and peri-stimulus time.

In the following, we will go through model specification and computation of contrast images. This guide is not written as a tutorial (i.e. detailed instructions for a given data set), but describes each design option and hopefully provides deeper background knowledge.

14.3.1 The aim

The aim of the 1st level is to compute contrast images that provide the input to the 2nd level. We will describe this using the example of 2D-data, i.e. data that has not been source reconstructed but, for each peri-stimulus time point, has been projected to a 2D-plane (s. chapter ??).

14.3.2 Start

Start SPM by the command '*spm eeg*' from the matlab command line. Press the *EEG/MEG* button. Your first choice is to either specify the model design or the input data. One always starts with the design. Currently, there are two design options: (i) *all options* and (ii) *ERP/ERF*. The latter option is a shortcut to quickly input an evoked responses study. We will first describe *all options* and then treat the *ERP/ERF* option as a special case.

14.3.3 All options

You first have to answer the question whether this is a 1st level design. This determines whether SPM expects to model peri-stimulus time as a factor. Also, if one models first-level data, SPM will ask next for **one** M/EEG-matfile before the data was projected to voxel-space. The reason for this is that the voxel-images lost important information during the conversion. For example, all timing information were lost. With the nifti-images only, SPM doesn't know the peri-stimulus time of each data point. However, this information is critical as soon as you try to specify (later on) linear weights in terms of peri-stimulus time. So, when you select an M/EEG file, SPM will read timing information from this file. For an ERP-study, the M/EEG-file of the average (ERP) is a good choice.

14.3.4 How many factors?

This question starts off the design specification proper. SPM needs to know the number of factors which you want to model. At the 1st level, there are typically only factors *peri-stimulus time* and *condition*. If you like, you can further subdivide the condition-factor in its components. For instance, if you have a 2x2 factorial design, you may want to specify 3 factors: *factor1*, *factor2* and *peri-stimulus time*.

14.3.5 Factor names and # of levels for factors

For each factor, you now input its name, e.g. condition, and enter the number of levels. For instance, if you have 2 conditions, you enter 2. For peri-stimulus time, you enter the number of time points in your evoked responses. Important: You should call the peri-stimulus time factor 'time'. For the number of levels for this special factor, SPM defaults to the correct number of peri-stimulus time points. (Note that it is currently not possible to model only a subset of time points.)

14.3.6 Select design component

You have the choice between *Identity* and *Constant*. Your selected design components are combined (by Kronecker tensor product) to form the 1st level design matrix. This has also been described in [32]. For the 1st level, you simply choose for all factors *identity*. This completes model specification.

14.3.7 Data

For selecting data, press the *EEG/MEG* button again. After selecting the *SPM.mat* file, you are asked to select data for each factor. The order in which you input data depends on the order of how you named the individual factors. We recommend that you make the *peri-stimulus time* factor the last factor. After projection to voxel-space, the data are stored as 4-dimensional files with the third dimension $z = 1$. If you want to input all peri-stimulus time points for a given file, you have to select all volumes along the 4th dimension. This is done by setting the number '1' in the SPM-file selector (below the 'Filt' line) to '1:101', where '101' is the total number of peri-stimulus time points. Of course, you have to replace '101' by the number of time points of your data (or by any natural number bigger than that). This choice will make all time points selectable. Then right-click over the file names and *Select all*. Press *done* to confirm your choice. This completes data selection.

14.3.8 'Estimation'

Although there is actually nothing to estimate, clicking the *Estimation* button will prepare some internal structure for the results section. We kept this (otherwise redundant) estimation step to provide for greater similarity with other analyses using SPM.

14.3.9 Results

After clicking on *Results*, choose the appropriate *SPM* and the contrast manager will pop up. In contrast to a usual SPM study, we don't use the contrast manager to compute statistics, but contrasts only!

Click *Define new contrast...* and enter a name for your contrast. Then note a (new) button called *components* which is only visible for M/EEG models. Clicking this button opens the contrast components manager. This is simply a tool that exploits the knowledge about the factors which you have specified earlier. Knowing the factors and their levels makes it easy to split up a (long) contrast weight vector into a few components. For each contrast weight vector, each factor contributes one component. By using the Kronecker tensor product, these components can be combined into the resulting contrast weight vector. This is not only time-saving, but many people tend to find this approach more intuitive than the usual approach of figuring out the contrasts yourself. For instance, if you have specified two conditions, you might be interested in

their difference. Enter a $[-1 \ 1]$ as contrast component. For the *time* factor, instead of entering one number for each time point, better click on the *Generate* button. Click on the 'Time' button and specify a rectangular averaging window by providing the start and end of this window (in milliseconds). Press *Compute*. You can see now in the contrast manager window that your contrast weights have been computed and are displayed above the identity (design) matrix. You can also specify the contrast weights as usual in the contrast box, but this would require to enter several hundreds to thousands of numbers. Press *ok* to proceed and compute the contrast.

14.3.10 Display

You can display the resulting contrast image by using the *Display* button.

14.3.11 ERP/ERF

You can shortcut some of the question and especially the data selection by choosing the *ERP/ERF* option (instead of *all options* when specifying a design. This option assumes that you have two factors, *condition* and *time*. There are less questions during design specification. When selecting data, you don't need to select all time point, but only the first! SPM will assume that you want to select all time points of the selected file. Using this option will otherwise result in the same model as described above.

14.3.12 Multiple subjects

For each of your subjects, you perform these operations in a separate 1st-level analysis. For each subject, you want to compute the same contrasts and use them as input to a model, where *subjects* is the repetition factor.

14.4 2nd level models

For 2nd level modelling, you can use different ways to specify a model. There is *Basic models* which was primarily developed for PET/fMRI but is equally appropriate for EEG/MEG data. These are suited best when the model is simple (like a 1-sample or 2-sample t-test). In our experience, most EEG/MEG models fall into this category of simple models. If models are more complicated, like, e.g., two groups with multiple subjects/conditions, we recommend using the *EEG/MEG* models.

14.4.1 All options

As above, go for *All options*. This time, press 'no' for the question 'Is this a first-level design'.

14.4.2 Factors

This includes all factors, even repetition factors. For example, at the 2nd level a 2x2 factorial design has 3 factors: *subject*, *factor1* and *factor2*.

14.4.3 Design partitions and design components

The way this modelling device constructs a design matrix is by using the Kronecker tensor product on the hierarchy of specified design components. However, some/many designs can't be constructed in this way. For example, the design matrix of a paired two sample-test consists of two merged partitions, each of which is a Kronecker tensor product of design components. For each partition, the factors and the levels are the same. The difference is in the choice of the design components for each factor under each partition. For example, for a paired two-sample-test, one has 2 factors (subjects and conditions) and 2 design partitions. For the 1st partition, choose *Constant* for subjects and *Identity* for conditions. For the 2nd partition, it's the other way around, i.e. *Identity* for subjects and *Constant* for conditions.

14.4.4 Covariance components

Specification of the covariance components determines the error model [25]. For each factor, there are two questions: (i) Identical variance for factor *xxx*, and (ii) Independence for factor *xxx*. SPM constructs all the variance components from your answers. The first question pertains to the assumption whether each level of this factor has identical variance. The second questions asks whether the different levels for a given factor are correlated. Some examples: For a repetition factor like subjects, you should always answer both questions with yes. For a group factor, one would assume that the levels of this factor (the groups) have unequal variance structures, but are uncorrelated (i.e., (i) no, (ii) yes). For a condition factor, the choice is up to you. A very restrained model would follow from using (i) yes (ii) yes, whereas the most liberal model is given by (i) no (ii) no.

14.4.5 Data

For each combination of factors, SPM asks you for the filenames of the data. Sometimes, this process can be more convient for you, when you have specified the factors in a specific order. For example, if you have two factors *subjects* and *condition*, the order (i) subjects, (ii) condition will ask for all images for each subject. This is convienient if you have stored the contrast images in their individual subject folder. This is the case, if you have computed 1st level contrasts following the approach described above. However, if, in an intermediate step, you have saved contrasts in condition-specific folders, the alternative order ((i) condition, (ii) subjects) is more appropriate.

14.4.6 Estimation and Results

The estimation follows the usual scheme, i.e. for a classical estimation procedure we use exactly the same routine as for PET/fMRI data (i.e. maximum-likelihood estimators for the parameters and Restricted Maximum Likelihood for estimation of the variance parameters).

For specification of contrasts, you have the option to specify contrasts component-wise. This can be useful for complex designs, when it's no longer easy to work out the interaction contrasts.

For 2D data the statistical map is displayed instead of the usual glass brain. You can invoke all the usual functions that are also available for fMRI/PET data. An additional option is *channels* which let you visualise to which voxel each channel maps. You can select this option by right-clicking the button on the statistical map background. SPM asks you then for one of the original M/EEG-mat files to read the channel mapping.

Chapter 15

Equivalent current Dipole fitting

This little chapter demonstrates how to use the ECD (Equivalent Current Dipole) routines with the multimodal dataset available on the FIL website. The aim is to fit a single dipole on the N170 wave visible in the 3 conditions. I will briefly describe how to analyse the dataset. For more details about the implementation, please refer to the help bit of and comments in the routines themselves.

15.1 Necessary data

Before proceeding any further, we have to make sure that we have all the necessary data in the right format. We need

- the *amri.img/hdr* structural MRI of the subject. It will be used to build the head model and display the results in the subject's anatomical space.
- the *mae_eeg.dat/mat* EEG data files. These are the fully processed data with one ERP per condition.
- the coordinates of the sensors, fiducial markers and scalp points (headshape) in 3 distinct **.mat* files.

In the dataset provided on the web, the raw **.pol* files are available. It is necessary to prepare these files to use them with the source reconstruction routines. This is a crucial step as the registration between the "EEG space" and "patient/image space" relies entirely on these files! To prepare these files, use the little script *create_fid_files.m* distributed with SPM5. A copy is also available at the end of this chapter.

Once we have all the files ready, we can proceed with the 3 main steps: building the model, fitting the dipole and displaying the results. To launch the GUI, press "3D source reconstruction" in the main window of SPM.

15.2 Model building

After selecting the data file *mae_eeg.mat* and the method "ECD", the first step is building the meshes for the scalp and inner skull volume. This is done automatically through the "Meshes" button. Select the structural MRI to use (*amri.img* here) and wait...

This step takes some time as the MRI is normalised and segmented. The normalisation parameters are saved in the *amri_vbm_inv_sn.mat* file and will be used later to map coordinates between the template and subject spaces. With the segmentation, the brain and scalp binary volumes are built (*amri_iskull.img* and *amri_oscaldp.img*). These are used to build the outer scalp and inner skull surface meshes. These are saved in the *model_head_amri.mat* file with other information. The scalp mesh is also saved in the file *amri_scVert.mat*.

Once the head model is ready, we can co-register the EEG space with subject/image space. Use the "Data Reg." button and decide if the registration should be based on the fiducials only (which is quite approximate) or the fiducials and the scalp surface (which should be more precise). Then select the appropriate files: *fid_eeg.mat*, *fid_MRI.mat*, *headshape_orig.mat*, *amri_scVert.mat* and let the routine work.

To prepare the model for the forward solution, simply press "ForwardComp." and "individual" to use the subject's own MRI. The forward model uses a spherical approximation. The best fitting sphere are adjusted on the scalp surface and 2 other spheres are added to model the scalp and skull outer surfaces. Obviously the head is not spherical and there will be a mismatch between the scalp/brain surfaces and their respective spheres. We have used the idea proposed by Spinelli et al., 2000 [42], where the brain volume is warped into a sphere. This allows us to use an analytical formula to calculate the forward solution for each dipole location while preserving some anatomical characteristics: superficial (resp. deep) sources remain superficial (resp. deep) in the spherical head model.

At this last step, the electrodes are also introduced in the head model and positioned relative to the subject head, as in the MRI. The *model_head_amri.m* at contains the information about the fitted spheres and electrodes. Dipole fitting of the data is now possible.

15.3 Dipole fitting

By pressing the "Inverse Sol." you launch the dipole fitting procedure. A number of questions have to be answered in order to specify the kind of solution you want:

- "Condition to use", select which condition is used to fit the dipole(s). So far, it is not possible to fit multiple conditions (or linear combinations of them) at the same time. For example, for differences between conditions, you should pre-calculate this difference before trying to fit ECDs.
- "Time window", define the time window in ms on which the ECDs should be fitted. With the N170 demo data, a good window is 150 to 180.
- "Number of dipoles", this is the crucial question. How many dipoles should be used? It's up to you to decide... With the demo data, from the look of the EEG scalp map, 1 ECD should be enough.
- "Number of random seeds". In order to avoid being trapped in a local minimum during the optimisation process because of a peculiar starting point. The algorithm can be launched from multiple random starting 'seeds'. If they all converge to approximately the same solution, then we'll have most surely reached the local optimum.
- "Orientation of the dipoles". The location of the ECD will be constant throughout the time window but its orientation can be left free or be fixed as well. Leaving the orientation free allows the dipoles to rotate over time. To fix the orientation, we can use the (weighted according to the EEG power)) mean over the time window or use the orientation of the ECD fitting the time instant with maximum EEG power.
- "File name". File names are suggested but feel free to change it!

After fitting the N random seeds, the routine tries to group them in clusters of similar ECDs according to their location and signal variance explained. Eventually, these 'grouped' ECDs are displayed on the subject anatomy. The result of this clustering is saved in a mat file starting by *res_* and finishing with the name you entered.

15.4 Result display

Results can be redisplayed with the routine *spm_eeg_inv_ecd_DrawDip.m*. The routine asks you to select the solution file you want to display and the MR image to be used.

15.5 Preparing the *.pol files

```
fid_eeg = ([-0.0587687 6.79448 -0.00636311 ; ... 0.0352661 -6.78906 -0.00369206 ; ...
9.3675 0.0260009 0.00481311] + ... [-0.0328487 6.78991 0.00636288 ; ... 0.0563513
-6.79533 0.00369206 ; ... 9.45206 -0.0260009 -0.00481297])/2 ... * 10 ;
fid_mri = [-71.8 3.5 -58.8 ; ... 71.3 -6 -62.5 ; ... 0 90.6 -28.4] ;
sensors = load('sensors_noFid.pol','ASCII')*10; headshape = load('headshape_noFid.pol','-
ASCII')*10;
Rot = spm_matrix([0 0 0 0 0 -pi/2]); Rot = Rot(1:3,1:3); fid_eeg = (Rot*fid_eeg)';
sensors = (Rot*sensors)'; headshape = (Rot*headshape)';
save fid_eeg fid_eeg save fid_mri fid_mri save sensors_orig sensors save headshape_orig
headshape
```


Chapter 16

Dynamic Causal Modelling for evoked responses

Dynamic Causal Modelling for ERP/ERFs is described in [16] and [30], see also <http://www.fil.ion.ucl.ac.uk/spm/doc/biblio/>. We recommend reading these two communications as a starter, because it will enable you to better understand all the modelling options presented in this chapter.

In summary, the goal of DCM is to explain evoked responses as the output of an interacting network consisting of a few areas that receive an input stimulus. The difference between two evoked responses that receive comparable stimuli under two conditions is modelled as a modulation of some of the inter-areal connections [16]. This interpretation of the ERP makes hypotheses about connectivity directly testable. For example, one can ask, whether the difference between two evoked responses can be explained by some top-down modulation of early areas (Garrido et al., in preparation). Additionally, because DCM is framed in a Bayesian way, one can also compute model evidences. These can be used to compare alternative, equally plausible, models and decide which model is the better one [39].

DCM for ERP/ERFs takes the spatial forward model into account. To do this, we parameterise the lead field, i.e., the spatial projection of source activity to the sensors. In the present version, this is done by assuming that each area is modelled by one equivalent current dipole (ECD) [30]. In other words, DCM is used not only to solve for the connectivity but, simultaneously, also for the spatial parameters.

In the following, we will describe the graphical user interface (GUI). The GUI allows to comfortably specify all parameters of a model. If you want to specify lots of models, we recommend using a batch script. An example of such a script (*DCM-ERP_example*), which can be adapted to your own data, can be found in the *man/example_scripts/* folder of the distribution.

16.1 Calling DCM for ERP/ERF

Currently, the GUI is hidden away as a toolbox. You can find it under *Toolboxes* → *apierp*. The GUI is partitioned into 5 parts, going from the top to the bottom. The first part is about loading and saving DCMs. The second part is about selecting some data, the third is used to specify a spatial forward model, the fourth is for specifying the connectivity model, and the last allows you to estimate parameters and view results.

In general, you have to specify the data and model in a specific order. The first stage is data selection, the second the spatial model, followed by the connectivity model specification. This order is necessary, because there are dependencies among the three parts that would be hard to resolve without this order. However, at any time, you can switch forth and back between model parts. Also, within each part, you can specify information in any order you like.

16.2 cd, load and save

At the top of the GUI, you can *cd* to a new working directory, load existing DCMs or save a new one. In general, you can *save* and *load* during model specification at any time.

16.3 Data selection

In this part, you select the data. These can be either event-related potentials or fields (i.e. a data matrix (channels \times peri-stimulus time), averaged over single trials). Currently, you can only analyse one or two evoked responses in the same model¹. To select data, click on the button *choose data*. Select an M/EEG SPM-matfile which is the output of the SPM preprocessing. Alternatively, if your data are in another format, you have to convert this file to the SPM-format (see the preprocessing chapter of this manual to see how this can be done). Below the *choose data* button, you can choose under *nr* which of the evoked responses in the SPM-matfile you want to model. For example, if you want to model the second and third evoked response within a SPM-matfile, specify indices 2 and 3. If your two evoked responses are in different files, you have to merge these files first. You can do this with the SPM preprocessing function *merge (spm-eeg-merge)*, s. chapter 12. Under *ms* you can specify the period of peri-stimulus time which you want to model. After your data are loaded, the time-series of all sensors are displayed in the SPM Graphics window.

In DCM, we use a projection of the data to some subspace to reduce the amount of data. Additionally, this data selection also serves as a tool to model only the salient features of the data. Currently, we are using a simple singular value decomposition (SVD) to decompose the data. You can select the number of (first) modes you want to keep. The default is 3, which we experimentally found to be a good value for our own data.

Furthermore, you can choose whether you want to model the mean or drifts of the data at sensor level. If you don't want any such terms, select 0 for detrending. Otherwise, select the number of discrete cosine transform terms you want to use to model the mean (1) or low-frequency drifts (> 1).

If you are happy with your data, the projection and the detrending terms, you can click on the $>$ (forward) button, which will bring you to the next stage *Spatial model specification*. Additionally, when pressing the forward button, the reduced data is displayed in the Graphics window. If you want to try other choices, you can press the $<$ button (backward) button in the spatial modelling part. This will take you back to the data selection part, where you can change parameters and hit the forward button again.

16.4 Spatial model specification

With the present version of DCM, you have two options how to spatially model your data. Either you compute the leadfield, for each area, yourself, or you parameterise the leadfield using an equivalent current dipole model.

For the first option, you need to choose *fixed* in the pull-down menu. Then click on *load lead field* and specify a matfile. This matfile must contain a matrix with one column for each area. Each column must have one entry for each channel. For an example, see [16]. Of course, you must have some means of computing the leadfield for your experiment. Alternatively, you can opt for the alternative and make the leadfield a function of equivalent current dipoles (ECDs). The parameters of the ECDs (location and orientation) can be estimated by DCM. To do this, you need to either select *ECD EEG* or *ECD MEG*. For *ECD EEG*, you also need to select a sensor location file. This can be either a Polehmuus file or a matfile with a coordinate matrix. Under *names* specify the names of all areas, one name per row. Under *locations*, specify the locations of these areas in MNI-space, again one location (three coordinates — x y z) per row. You can check the locations of dipoles by clicking on the *plot* button. This will visualize the dipole locations overlaid on an MRI template in MNI space. Note that DCM uses by default uninformative priors

¹Note however, that the theoretical framework allows analysis of any number of evoked responses

on dipole orientations, but rather tight priors on locations [30]. One reason for the tight priors on locations is to ensure that each dipole stays in its designated area and retains its meaning. The prior location for each dipole can be found either by using available anatomical knowledge or by relying on source reconstruction of comparable studies. Also note that the prior location doesn't need to be overly exact, because the spatial resolution of M/EEG is on a scale of several millimeters. When you want to proceed to the next model specification stage, hit the > (forward) button and proceed to the *connectivity model specification*.

16.5 Connectivity model specification

Press *specify connections* to get access to selecting your model's connections between areas. There are 5 elements which you need to go through. The first three are the intrinsic connectivities. In DCM for ERP/ERF there are three types of connections, *forward*, *backward* and *lateral*. In each of these matrices you specify a connection *from* a source area *to* a target area. For example, switching on the element (2, 1) in the intrinsic forward connectivity matrix means that you specify a forward connection from area 1 to 2. Some people find the meaning of each element slightly counter-intuitive, because the column index corresponds to the source area, and the row index to the target area².

In the present implementation, there is only one input allowed. This input can go to any area, where it only goes to early areas typically. You can select these receiving areas by selecting area indices in the C vector.

The B matrix contains all gain modulations of intrinsic connections. These modulations model the difference between the first and second evoked response. In other words, the DCM explains two evoked responses by explaining the first response by using the intrinsic connections only. The 2nd response is modelled by modulating these intrinsic connections by the weights in matrix B . For instance, if you want to allow modulations of forward connections only, you switch on those connections in B which are also selected in the intrinsic forward connectivity matrix.

16.6 Estimation

When you are done with model specification, you can hit the *estimate* button in the lower left corner. DCM will first try to save your DCM. Select a file to save to. After saving, DCM will estimate model parameters. You can follow the estimation process by observing the model fit in the output window. In the matlab command window, you will see each iteration commented by iteration number, free energy F , and the change of F with respect to the updated variance parameters.

Note that a DCM for evoked responses is more complex than a DCM for fMRI. This means that more model parameters are used and, consequently, the estimation process takes longer. Expect something like 15 - 60 minutes, depending on model/data/computer specification.

16.7 Results

After estimation finished, you can assess the results by choosing from the pull-down menu at the bottom (middle).

With *ERPs channel* you can plot, for each mode, the data for both evoked responses, and its fit by the model.

When you select *ERPs sources*, the dynamics of each area are plotted. These corresponds to the (output) states of the dynamic system [16].

The option *coupling (A)* will take you to a summary about the posterior distributions of the intrinsic connectivities. In the upper row, you see the posterior means for all intrinsic connectivities. As above, element (i, j) corresponds to a connection from area j to i . In the lower row, you'll find, for each connection, the probability that its posterior mean is different from the prior mean, taking into account the posterior variance.

²Currently, you can't model self-connections, which may be introduced with a later SPM update.

With the option *coupling(B)* you can access the equivalent posterior means for the gain modulations of the intrinsic connectivities and their probability that they are unequal the prior means.

With *coupling(C)* you see the same summary of the posterior distribution for the strength of the input into the input receiving area. On the left hand side, DCM plots the posterior means for each area. On the right hand side, you can see the corresponding probabilities (s. above).

The option *Input* shows you the estimated input function. As described by [16], this is a gamma function with an addition of some low-frequency terms.

With *Response*, you can plot the selected data, i.e. the modes you have selected for DCM analysis.

And finally, with the option *Dipoles*, DCM displays an overlay of each dipole on an MRI template using the posterior means of its 3 orientation and 3 location parameters.

Part V

Utilities

Chapter 17

Display Image

Contents

17.1 Image to Display	116
---------------------------------	-----

This is an interactive facility that allows orthogonal sections from an image volume to be displayed. Clicking the cursor on either of the three images moves the point around which the orthogonal sections are viewed. The co-ordinates of the cursor are shown both in voxel co-ordinates and millimeters within some fixed framework. The intensity at that point in the image (sampled using the current interpolation scheme) is also given. The position of the crosshairs can also be moved by specifying the co-ordinates in millimeters to which they should be moved. Clicking on the horizontal bar above these boxes will move the cursor back to the origin (analogous to setting the crosshair position (in mm) to [0 0 0]).

The images can be re-oriented by entering appropriate translations, rotations and zooms into the panel on the left. The transformations can then be saved by hitting the "Reorient images..." button. The transformations that were applied to the image are saved to the header information of the selected images. The transformations are considered to be relative to any existing transformations that may be stored. Note that the order that the transformations are applied in is the same as in spm_matrix.m.

The "Reset..." button next to it is for setting the orientation of images back to transverse. It retains the current voxel sizes, but sets the origin of the images to be the centre of the volumes and all rotations back to zero.

The right panel shows miscellaneous information about the image. This includes:

Dimensions - the x, y and z dimensions of the image.

Datatype - the computer representation of each voxel.

Intensity - scalefactors and possibly a DC offset.

Miscellaneous other information about the image.

Vox size - the distance (in mm) between the centres of neighbouring voxels.

Origin - the voxel at the origin of the co-ordinate system

DIr Cos - Direction cosines. This is a widely used representation of the orientation of an image.

There are also a few options for different resampling modes, zooms etc. You can also flip between voxel space (as would be displayed by Analyze) or world space (the orientation that SPM considers the image to be in). If you are re-orienting the images, make sure that world space is specified. Blobs (from activation studies) can be superimposed on the images and the intensity windowing can also be changed.

If you have put your images in the correct file format, then (possibly after specifying some rigid-body rotations):

The top-left image is coronal with the top (superior) of the head displayed at the top and the left shown on the left. This is as if the subject is viewed from behind.

The bottom-left image is axial with the front (anterior) of the head at the top and the left shown on the left. This is as if the subject is viewed from above.

The top-right image is sagittal with the front (anterior) of the head at the left and the top of the head shown at the top. This is as if the subject is viewed from the left.

17.1 Image to Display

Image to display.

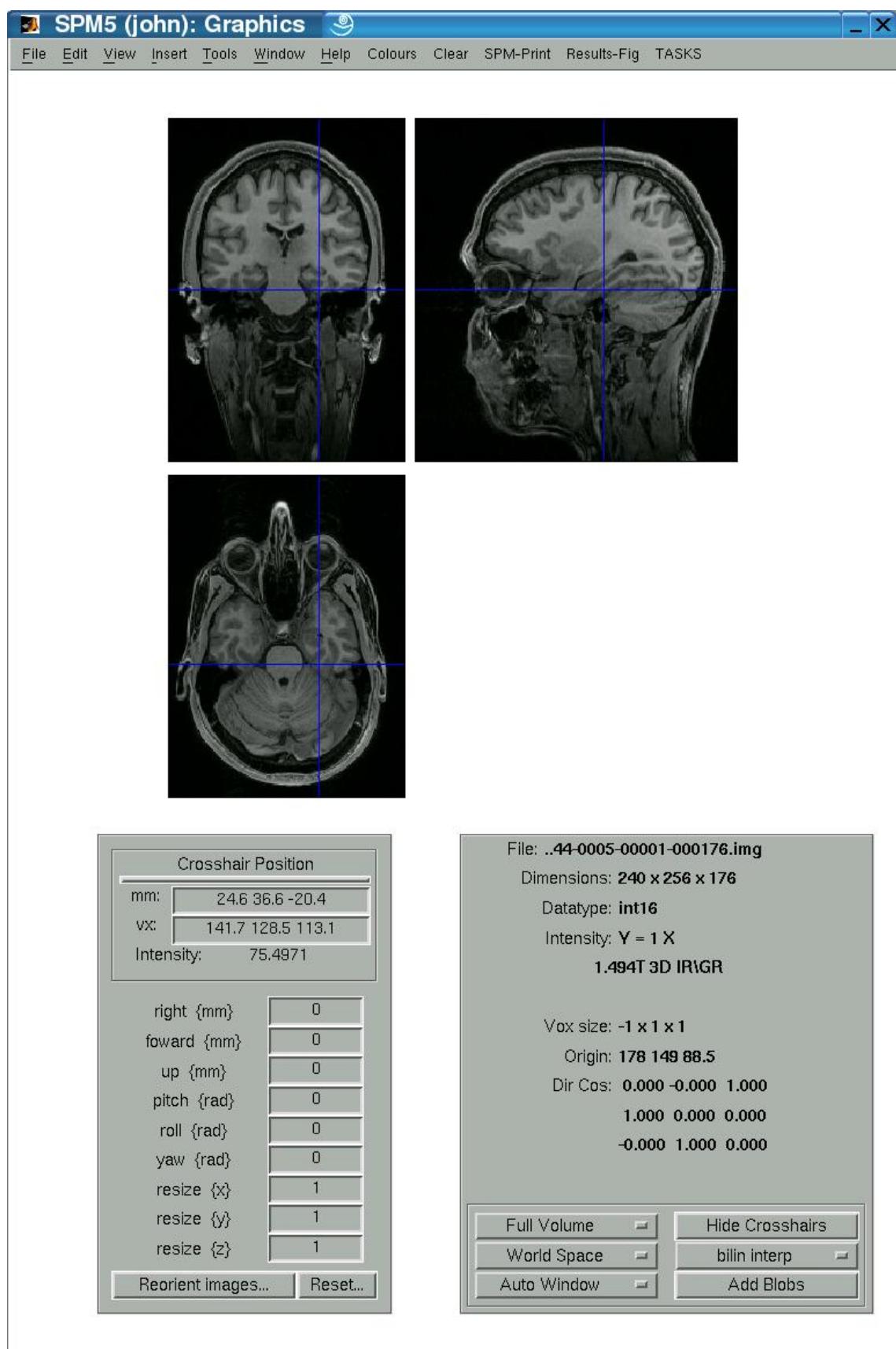


Figure 17.1: The Display routine.

Chapter 18

Check Registration

Contents

18.1 Images to Display	119
-------------------------------	-------	-----

Orthogonal views of one or more images are displayed. Clicking in any image moves the centre of the orthogonal views. Images are shown in orientations relative to that of the first selected image. The first specified image is shown at the top-left, and the last at the bottom right. The fastest increment is in the left-to-right direction (the same as you are reading this).

If you have put your images in the correct file format, then (possibly after specifying some rigid-body rotations):

The top-left image is coronal with the top (superior) of the head displayed at the top and the left shown on the left. This is as if the subject is viewed from behind.

The bottom-left image is axial with the front (anterior) of the head at the top and the left shown on the left. This is as if the subject is viewed from above.

The top-right image is sagittal with the front (anterior) of the head at the left and the top of the head shown at the top. This is as if the subject is viewed from the left.

18.1 Images to Display

Images to display.

Chapter 19

Image Calculator

Contents

19.1 Input Images	121
19.2 Output Filename	121
19.3 Expression	121
19.4 Options	122
19.4.1 Data Matrix	122
19.4.2 Masking	122
19.4.3 Interpolation	122
19.4.4 Data Type	122

The image calculator is for performing user-specified algebraic manipulations on a set of images, with the result being written out as an image. The user is prompted to supply images to work on, a filename for the output image, and the expression to evaluate. The expression should be a standard matlab expression, within which the images should be referred to as i1, i2, i3,... etc.

19.1 Input Images

These are the images that are used by the calculator. They are referred to as i1, i2, i3, etc in the order that they are specified.

19.2 Output Filename

The output image is written to current working directory unless a valid full pathname is given

19.3 Expression

Example expressions (f):

- * Mean of six images (select six images)
 $f = '(i1+i2+i3+i4+i5+i6)/6'$
- * Make a binary mask image at threshold of 100
 $f = 'i1>100'$
- * Make a mask from one image and apply to another
 $f = 'i2.*(i1>100)'$
- here the first image is used to make the mask, which is applied to the second image

```

* Sum of n images
f = 'i1 + i2 + i3 + i4 + i5 + ...'
* Sum of n images (when reading data into a data-matrix - use dmtx arg)
f = 'sum(X)'

```

19.4 Options

Options for image calculator

19.4.1 Data Matrix

If the dmtx flag is set, then images are read into a data matrix X (rather than into separate variables i1, i2, i3,...). The data matrix should be referred to as X, and contains images in rows. Computation is plane by plane, so in data-matrix mode, X is a NxK matrix, where N is the number of input images [prod(size(Vi))], and K is the number of voxels per plane [prod(Vi(1).dim(1:2))].

19.4.2 Masking

For data types without a representation of NaN, implicit zero masking assumes that all zero voxels are to be treated as missing, and treats them as NaN. NaN's are written as zero (by spm_write_plane), for data types without a representation of NaN.

19.4.3 Interpolation

With images of different sizes and orientations, the size and orientation of the first is used for the output image. A warning is given in this situation. Images are sampled into this orientation using the interpolation specified by the hold parameter.

- The method by which the images are sampled when being written in a different space.
 - Nearest Neighbour
 - Fastest, but not normally recommended.
 - Bilinear Interpolation
 - OK for PET, or realigned fMRI.
 - Sinc Interpolation
 - Better quality (but slower) interpolation, especially with higher degrees.

19.4.4 Data Type

Data-type of output image

Chapter 20

Create Rendering/Surface

Contents

20.1 Grey+white matter image	123
20.2 Output	123

20.1 Grey+white matter image

Images to create rendering/surface from (grey and white matter segments).

20.2 Output

Chapter 21

DICOM Import

Contents

21.1 DICOM files	125
21.2 Output directory	125

DICOM Conversion. Most scanners produce data in DICOM format. This routine attempts to convert DICOM files into SPM compatible image volumes, which are written into the current directory by default. Note that not all flavours of DICOM can be handled, as DICOM is a very complicated format, and some scanner manufacturers use their own fields, which are not in the official documentation at <http://medical.nema.org/>

21.1 DICOM files

Select the DICOM files to convert.

21.2 Output directory

Select a directory where files are written. Default is current directory.

Chapter 22

MINC Import

Contents

22.1 MINC files	127
22.2 Options	127
22.2.1 Data Type	127
22.2.2 NIFTI Type	127

MINC Conversion. MINC is the image data format used for exchanging data within the ICBM community, and the format used by the MNI software tools. It is based on NetCDF, but due to be superceded by a new version relatively soon. MINC is no longer supported for reading images into SPM, so MINC files need to be converted to NIFTI format in order to use them. See <http://www.bic.mni.mcgill.ca/software/> for more information.

22.1 MINC files

Select the MINC files to convert.

22.2 Options

Conversion options

22.2.1 Data Type

Data-type of output images. Note that the number of bits used determines the accuracy, and the amount of disk space needed.

22.2.2 NIFTI Type

Output files can be written as .img + .hdr, or the two can be combined into a .nii file.

Chapter 23

ECAT Import

Contents

23.1 ECAT files	129
23.2 Options	129
23.2.1 NIFTI Type	129

ECAT 7 Conversion. ECAT 7 is the image data format used by the more recent CTI PET scanners.

23.1 ECAT files

Select the ECAT files to convert.

23.2 Options

Conversion options

23.2.1 NIFTI Type

Output files can be written as .img + .hdr, or the two can be combined into a .nii file.

Chapter 24

Execute Batch Jobs

Contents

24.1 Batch Files	131
-------------------------	-----

This facility allows previously created batch jobs to be run. These are simply created by the batch user interface (which you are currently using).

24.1 Batch Files

Select the batch job files to be run.

Chapter 25

Change Directory

Contents

25.1 Select a directory	133
--------------------------------	-----

This facility allows programming a directory change. Directories are selected in the right listbox.

25.1 Select a directory

Select a directory to change to.

Chapter 26

Make Directory

Contents

26.1 Select a base directory	135
26.2 Enter a directory name	135

This facility allows programming a directory change. Directories are selected in the right listbox.

26.1 Select a base directory

Select a base directory.

26.2 Enter a directory name

Enter a directory name

Chapter 27

Deformations

Contents

27.1 Composition	137
27.1.1 Imported _sn.mat	137
27.1.2 Deformation Field	138
27.1.3 Inverse	138
27.1.4 Composition	140
27.2 Save as	142
27.3 Apply to	143

This is a utility for working with deformation fields. They can be loaded, inverted, combined etc, and the results either saved to disk, or applied to some image.

Note that ideal deformations can be treated as members of a Lie group. Future versions of SPM may base its warping on such principles.

27.1 Composition

Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $yox:A \rightarrow C$. Compositions can be combined in an associative way, such that $zo(yox) = (zoy)ox$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

27.1.1 Imported _sn.mat

Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *_sn.mat files, which can be converted to deformation fields.

Parameter File

Specify the _sn.mat to be used.

Voxel sizes

Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box

Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

27.1.2 Deformation Field

Deformations can be thought of as vector fields. These can be represented by three-volume images.

27.1.3 Inverse

Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If $y':A \rightarrow B$ is the inverse of $y:B \rightarrow A$, then $y' \circ y = y \circ y' = Id$, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition

Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle " \circ ". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $y \circ x:A \rightarrow C$. Compositions can be combined in an associative way, such that $z \circ (y \circ x) = (z \circ y) \circ x$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first \circ second) \circ third)... \circ last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported .sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *.sn.mat files, which can be converted to deformation fields.

Parameter File Specify the .sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If $y':A \rightarrow B$ is the inverse of $y:B \rightarrow A$, then $y' \circ y = y \circ y' = \text{Id}$, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $y \circ x:A \rightarrow C$. Compositions can be combined in an associative way, such that $z \circ (y \circ x) = (z \circ y) \circ x$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Image to base inverse on Specify the image file on which to base the dimensions, orientation etc of the inverse.

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $y \circ x:A \rightarrow C$. Compositions can be combined in an associative way, such that $z \circ (y \circ x) = (z \circ y) \circ x$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported .sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *.sn.mat files, which can be converted to deformation fields.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If $y':A \rightarrow B$ is the inverse of $y:B \rightarrow A$, then $y' \circ y = y \circ y' = \text{Id}$, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $y \circ x:A \rightarrow C$. Compositions can be combined in an associative way, such that $z \circ (y \circ x) = (z \circ y) \circ x$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last.

The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Image to base inverse on

Specify the image file on which to base the dimensions, orientation etc of the inverse.

27.1.4 Composition

Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $yox:A \rightarrow C$. Compositions can be combined in an associative way, such that $zo(yox) = (zoy)ox$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported .sn.mat

Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *.sn.mat files, which can be converted to deformation fields.

Parameter File Specify the .sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template image that was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template image that was originally used to estimate the deformation.

Deformation Field

Deformations can be thought of as vector fields. These can be represented by three-volume images.

Inverse

Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If $y':A \rightarrow B$ is the inverse of $y:B \rightarrow A$, then $y' \circ y = y \circ y' = Id$, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $yox:A \rightarrow C$. Compositions can be combined in an associative way, such that $zo(yox) = (zoy)ox$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported .sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *.sn.mat files, which can be converted to deformation fields.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If $y':A \rightarrow B$ is the inverse of $y:B \rightarrow A$, then $y' \circ y = y \circ y' = \text{Id}$, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $y \circ x:A \rightarrow C$. Compositions can be combined in an associative way, such that $z \circ (y \circ x) = (z \circ y) \circ x$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Image to base inverse on Specify the image file on which to base the dimensions, orientation etc of the inverse.

Composition

Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $y \circ x:A \rightarrow C$. Compositions can be combined in an associative way, such that $z \circ (y \circ x) = (z \circ y) \circ x$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported .sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *.sn.mat files, which can be converted to deformation fields.

Parameter File Specify the .sn.mat to be used.

Voxel sizes Specify the voxel sizes of the deformation field to be produced. Non-finite values will default to the voxel sizes of the template imagethat was originally used to estimate the deformation.

Bounding box Specify the bounding box of the deformation field to be produced. Non-finite values will default to the bounding box of the template imagethat was originally used to estimate the deformation.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

Inverse Creates the inverse of a deformation field. Deformations are assumed to be one-to-one, in which case they have a unique inverse. If $y':A \rightarrow B$ is the inverse of $y:B \rightarrow A$, then $y' \circ y = y \circ y' = \text{Id}$, where Id is the identity transform.

Deformations are inverted using the method described in the appendix of:

* Ashburner J, Andersson JLR & Friston KJ (2000) "Image Registration using a Symmetric Prior - in Three-Dimensions." Human Brain Mapping 9(4):212-225

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $y \circ x:A \rightarrow C$. Compositions can be combined in an associative way, such that $z \circ (y \circ x) = (z \circ y) \circ x$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Image to base inverse on Specify the image file on which to base the dimensions, orientation etc of the inverse.

Composition Deformation fields can be thought of as mappings. These can be combined by the operation of "composition", which is usually denoted by a circle "o". Suppose $x:A \rightarrow B$ and $y:B \rightarrow C$ are two mappings, where A, B and C refer to domains in 3 dimensions. Each element a in A points to element $x(a)$ in B. This in turn points to element $y(x(a))$ in C, so we have a mapping from A to C. The composition of these mappings is denoted by $y \circ x:A \rightarrow C$. Compositions can be combined in an associative way, such that $z \circ (y \circ x) = (z \circ y) \circ x$.

In this utility, the left-to-right order of the compositions is from top to bottom (note that the rightmost deformation would actually be applied first). i.e. ...((first o second) o third)...o last. The resulting deformation field will have the same domain as the first deformation specified, and will map to voxels in the codomain of the last specified deformation field.

Imported .sn.mat Spatial normalisation, and the unified segmentation model of SPM5 save a parameterisation of deformation fields. These consist of a combination of an affine transform, and nonlinear warps that are parameterised by a linear combination of cosine transform basis functions. These are saved in *.sn.mat files, which can be converted to deformation fields.

Deformation Field Deformations can be thought of as vector fields. These can be represented by three-volume images.

27.2 Save as

Save the result as a three-volume image. "y_" will be prepended to the filename. The result will be written to the current directory.

27.3 Apply to

Apply the resulting deformation field to some images. The warped images will be written to the current directory, and the filenames prepended by "w". Note that trilinear interpolation is used to resample the data, so the original values in the images will not be preserved.

Chapter 28

User Interface

Contents

28.1 Appearance	145
28.1.1 Background Colour 1	145
28.1.2 Background Colour 2	145
28.1.3 Foreground Colour	145
28.1.4 Font Size	145
28.2 Printing	145

28.1 Appearance

Appearance of user interface

- 28.1.1 Background Colour 1**
- 28.1.2 Background Colour 2**
- 28.1.3 Foreground Colour**
- 28.1.4 Font Size**

28.2 Printing

Select the printing option you want. The figure will be printed to a file named spm5*.* , in the current directory. PostScript files will be appended to, but other files will have "page numbers" appended to them.

Part VI

Tools

Chapter 29

Toolboxes

Toolbox configuration files should be placed in the toolbox directory, with their own `spm_config_*.m` files. If you write a toolbox, then you can include it in this directory - but remember to try to keep the function names unique (to reduce clashes with other toolboxes). See `spm_config.m` for information about the form of SPM's configuration files.

Chapter 30

3D to 4D

Contents

30.1 3D Volumes	151
----------------------------------	------------

Concatenate a number of 3D volumes into a single 4D file.
Note that output time series are stored as big-endian int16.

30.1 3D Volumes

Select the volumes to concatenate

Part VII

Data sets

Chapter 31

Auditory fMRI data

This data set comprises whole brain BOLD/EPI images acquired on a modified 2T Siemens MAGNETOM Vision system. Each acquisition consisted of 64 contiguous slices (64x64x64 3mm x 3mm x 3mm voxels). Acquisition took 6.05s, with the scan to scan repeat time (TR) set arbitrarily to 7s.

96 acquisitions were made (TR=7s) from a single subject, in blocks of 6, giving 16 42s blocks. The condition for successive blocks alternated between rest and auditory stimulation, starting with rest. Auditory stimulation was bi-syllabic words presented binaurally at a rate of 60 per minute. The functional data starts at acquisition 4, image `fM00223_004`. Due to T1 effects it is advisable to discard the first few scans (there were no "dummy" lead-in scans). A structural image was also acquired: `sM00223_002`. These images are stored in Analyse format and are available from the SPM site <http://www.fil.ion.ucl.ac.uk/spm/data/>. This data set was the first ever collected and analysed in the Functional Imaging Laboratory (FIL) and is known locally as the mother of all experiments (MoAE).

To analyse the data, first create a new directory DIR

eg. `c:\home\wpenny\fmri_analysis\auditory`, in which to place the results of your analysis. Then create 3 subdirectories (i) `jobs`, (ii) `classical` and (iii) `bayesian`. As the analysis proceeds these directories will be filled with job-specification files, design matrices and models estimated using classical or Bayesian methods.

Start up matlab, enter your jobs directory and type `spm fmri` at the matlab prompt. SPM will then open in fMRI mode with three windows (1) the top-left or 'command' window, (2) the bottom-left or 'interactive' window and (3) the right-hand or 'graphics' window. Analysis then takes place in three major stages (i) spatial pre-processing, (ii) model specification, review and estimation and (iii) inference. These stages organise the buttons in SPM's base window.

31.1 Spatial pre-processing

31.1.1 Realignment

Under the spatial pre-processing section of the SPM base window select 'Realign' from the 'Realign' pulldown menu. This will call up a realignment job specification in the graphics window. Then

- Select 'New Realign:Estimate and Reslice'
- Open the newly created 'Realign:Estimate and Reslice' option.
- Highlight data, select 'New Session', then highlight the newly created 'Session' option.
- Select 'Specify Files' and use the SPM file selector to choose all of your functional images eg. '`fM000*.img`'.
- Save the job file as eg. `DIR/jobs/realign.mat`.
- Press the RUN button in the graphics window.

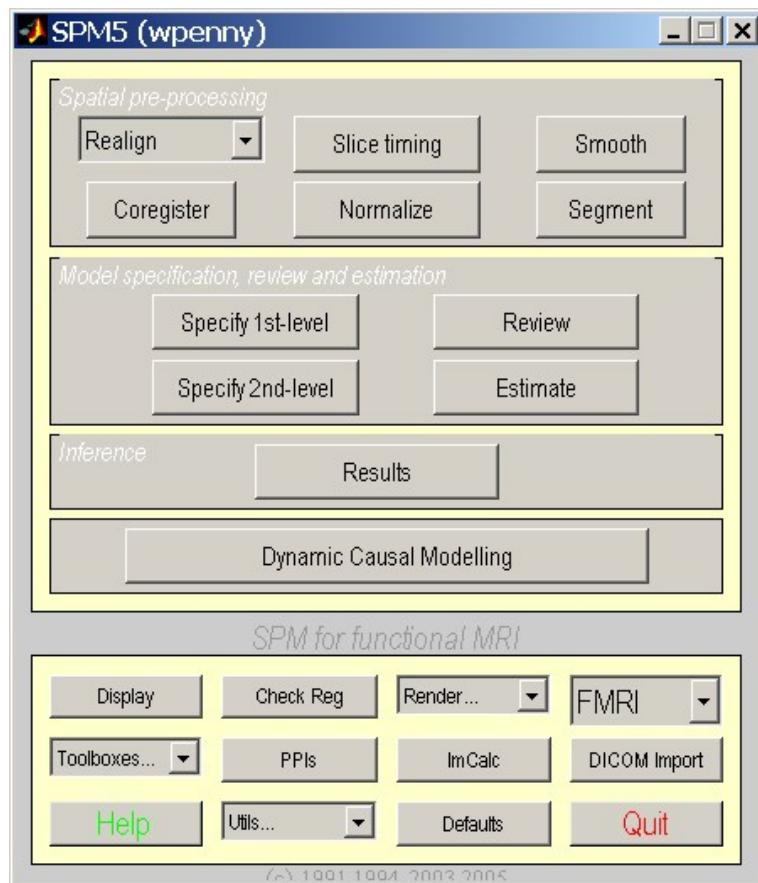


Figure 31.1: The SPM base window comprises three sections i) spatial pre-processing, (ii) model specification, review and estimation and (iii) inference.

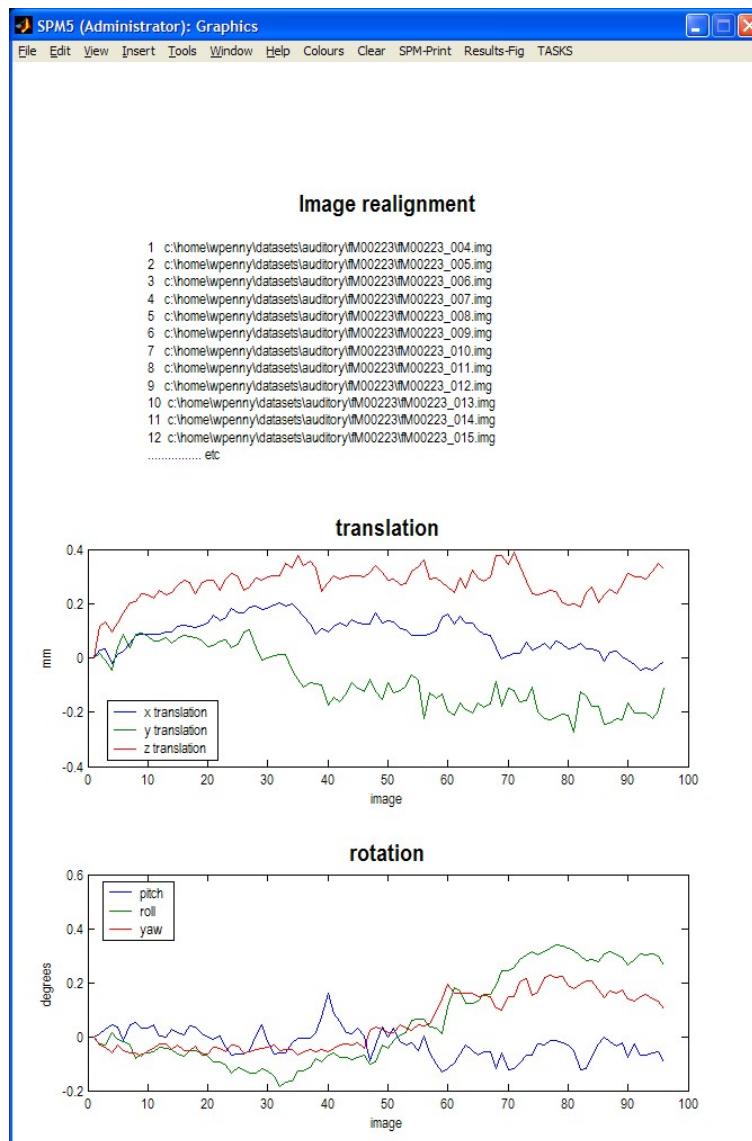


Figure 31.2: Realignment of auditory data.

This will run the realign job which will write realigned images into the directory where the functional images are. These new images will be prefixed with the letter ‘r’. SPM will then plot the estimated time series of translations and rotations shown in Figure 31.2. These data are also saved to a file eg. `rp_fM00223_004.txt`, so that these variables can be used as regressors when fitting GLMs. This allows movements effects to be discounted when looking for brain activations.

SPM will also create a mean image eg. `meanfM00223_004.img` which will be used in the next step of spatial processing - coregistration.

31.1.2 Coregistration

Press the ‘Coreg’ button. This will call up the specification of a coregistration job in the graphics window.

- Select New ”Coreg;Estimate”
- Double-click on the newly created Coreg;Estimate

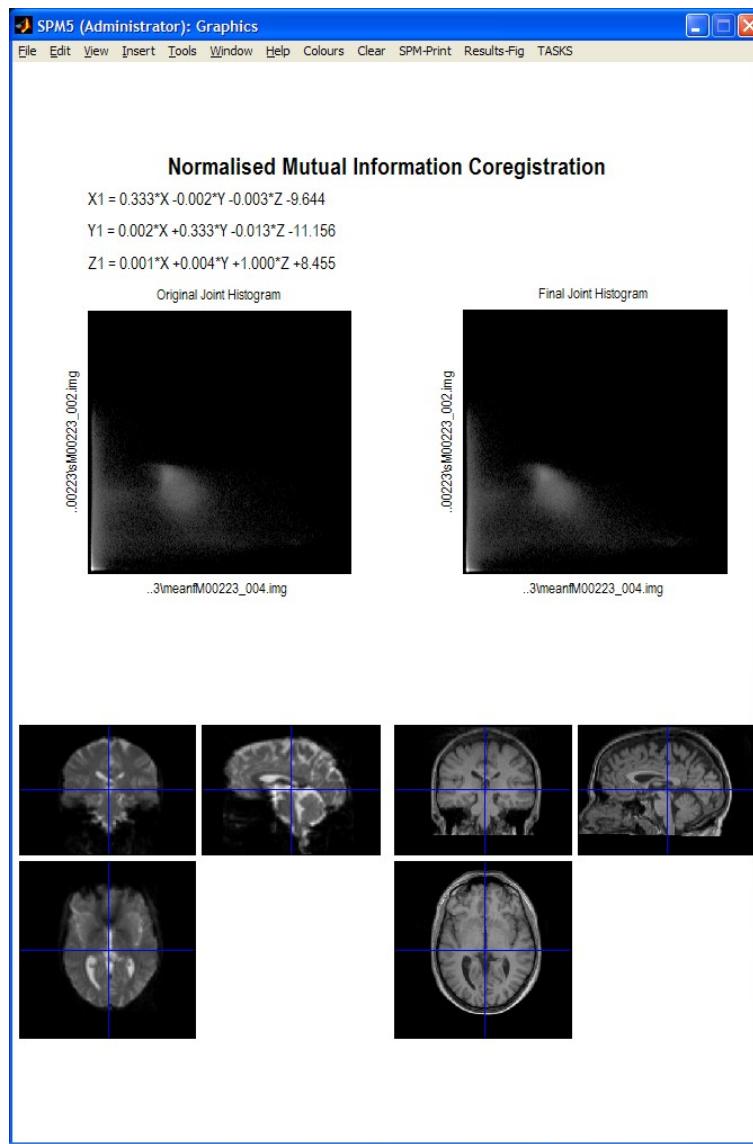


Figure 31.3: *Mutual Information Coregistration of Auditory data.*

- Highlight ‘Reference Image’ and then select the mean fMRI scan from realignment eg. `meanfM00223_004.img`
- Highlight ‘Source Image’ and then select the structural image eg. `sM00223_002.img`.
- Press the Save button and save the job as `coreg.job`
- Then press RUN

SPM will then implement a coregistration between the structural and functional data that maximises the mutual information. The image in figure 31.3 should then appear in the graphics window. SPM will have changed the header of the source file which in this case is the structural image `sM00223_002.hdr`.

The ‘Check Reg’ facility is useful here, to check the results of coregistration. Press the ‘Check Reg’ button in the lower section of the base window and then the select the Reference and Source Images specified above ie `meanfM00223_004.img` and `sM00223_002.img`. SPM will then produce an image like that shown in Figure 31.4 in the graphics window. You can then use your mouse to navigate these images to confirm that there is an anatomical correspondence.

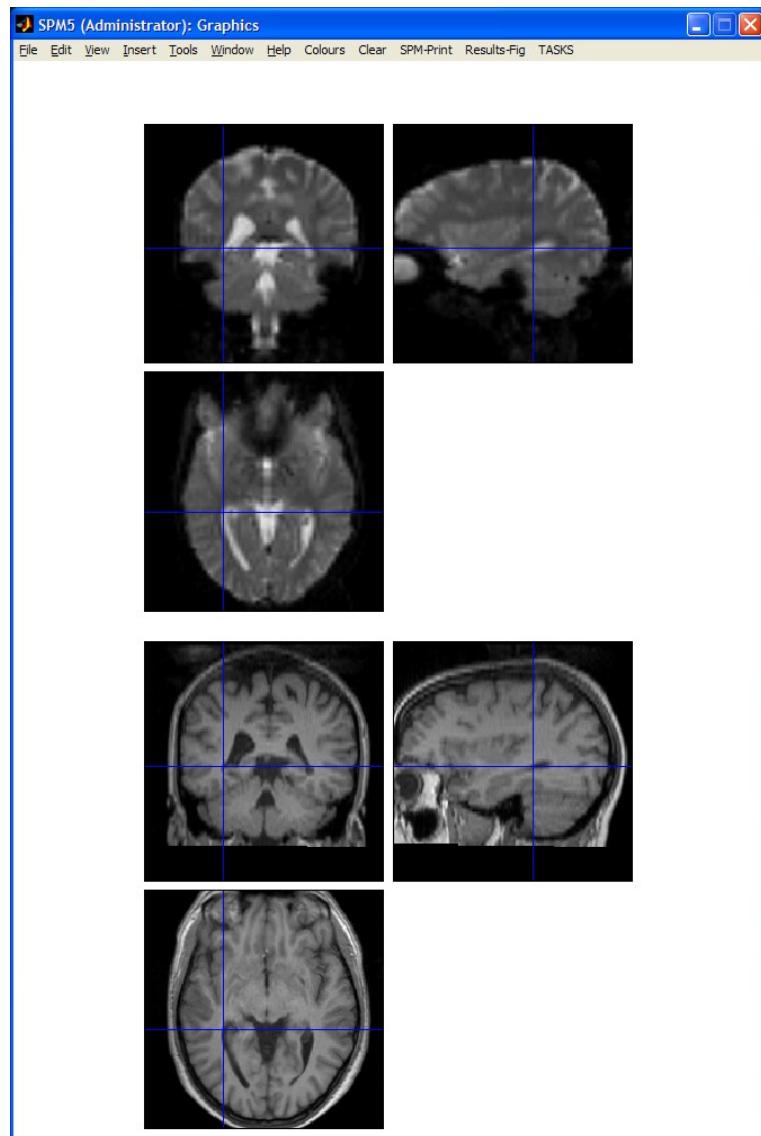


Figure 31.4: *Checking registration of functional and ‘registered’ structural data.*

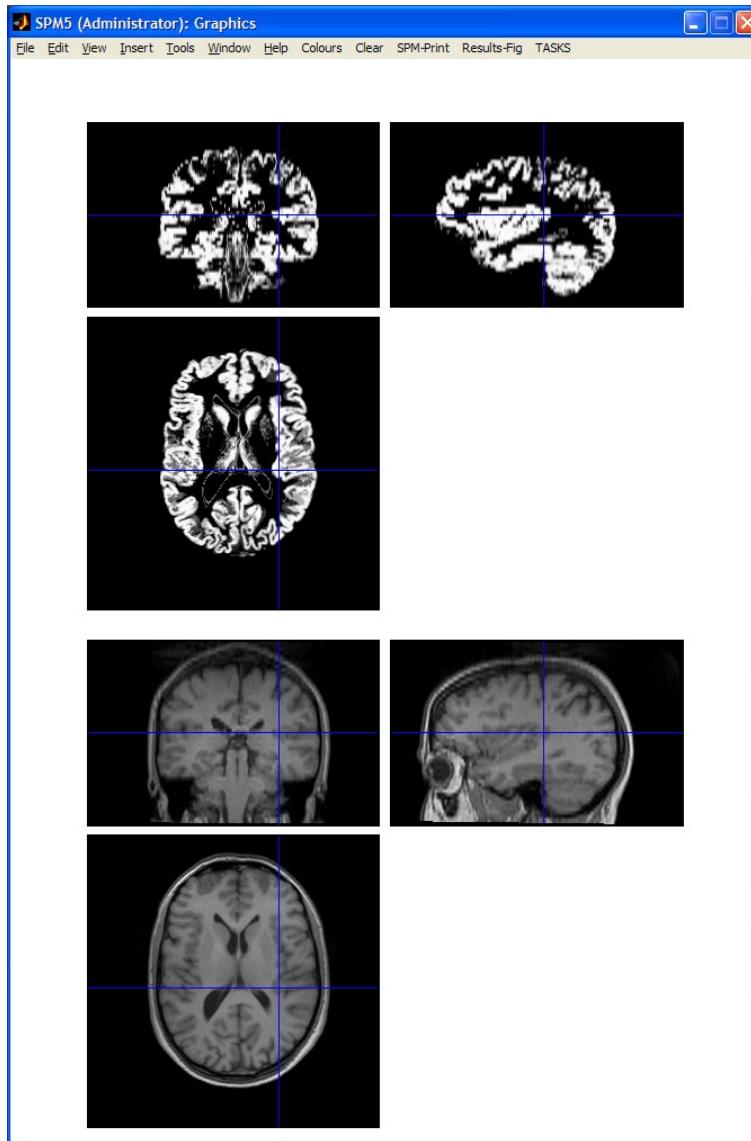


Figure 31.5: *Gray matter image and ‘registered’ structural image.*

31.1.3 Segmentation

Press the ‘Segment’ button. This will call up the specification of a segmentation job in the graphics window. Highlight the Data field and then select the subjects registered anatomical image eg. `sM00223_002.img`. Save the job file as `segment.mat` and then press RUN. SPM will segment the structural image using the default tissue probability maps as priors.

Faster, though perhaps less optimal results can be obtained by eg. reducing the number of Gaussians per class from [2 2 2 4] to eg. [1 1 1 4], increasing the sampling distance from eg. 3 to 4mm. These options can be edited under the ‘Custom’ sub-menu and saved before the job is run. The results obtained in figure 31.5 were obtained using the default values.

SPM will create, by default, gray and white matter images and bias-field corrected structural image. These can be viewed using the CheckReg facility as described in the previous section (press segment and select . Figure 31.5 shows the gray matter image, `c1sM0023_002.img` along with the original structural.

SPM will also write a spatial normalisation eg. `sM00223_0020_seg_sn.mat` and inverse spatial normalisation parameters `sM00223_0020_seg_inv_sn.mat` to files in the original structural directory. These can be used to normalise the functional data.

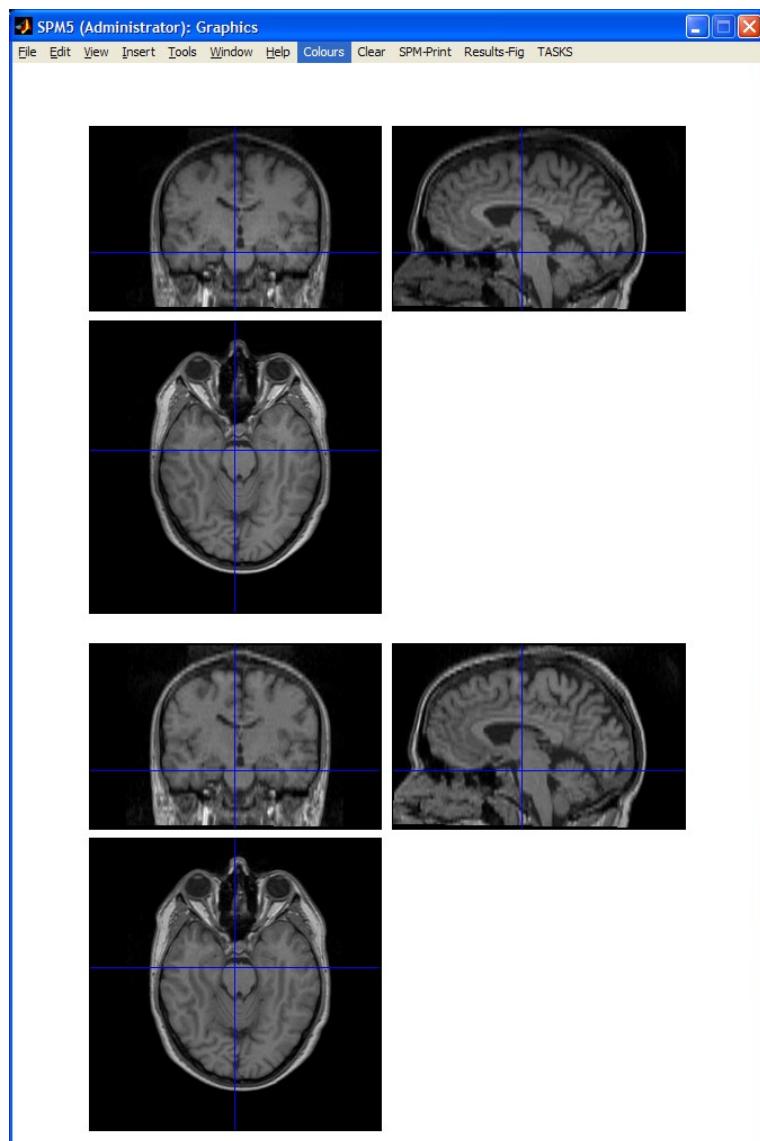


Figure 31.6: Structural image (top) and bias-corrected structural image (bottom). Notice that the original structural is darker at the top than at the bottom. This non-uniformity has been removed in the bias-corrected image.

31.1.4 Normalize

Press the ‘Normalize’ button. This will call up the specification of a normalise job in the graphics window.

- Select a New ”Normalise:Write” field. This allows previously estimated warps to be applied to a series of images.
- Highlight ‘Data’, select New ”Subject”
- Open ‘Subject’, highlight ‘Parameter File’ and select the `sM00223_0020_seg_sn.mat` file that you created in the previous section
- Highlight images to write and select all of the realigned functional images ‘`rfM000*.img`’. Note: This can be done efficiently by changing the filter in the SPM file selector to `^r.*`. SPM will then only list those files beginning with the letter *r* ie. those that have been realigned. You can then right click over the listed files, choose ‘Select all’ and press ‘Done’.
- Open ‘Writing Options’, and change ‘Voxel sizes’ from [2,2,2] to [3,3,3].¹
- Press ‘Save’, save the job as `normalise.mat` and then press ‘Run’.

SPM will then write spatially normalised files to the functional data directory. These files have the prefix ‘w’.

If you wish to superimpose a subject’s functional activations on their own anatomy² you will also need to apply the spatial normalisation parameters to their (bias-corrected) anatomical image. To do this

- Press ‘Normalise’, select New ”Normalise:Write”
- Open ‘Normalise: Write’, highlight ‘Data’, select New ”Subject”
- Open ‘Subject’
- Highlight ‘Parameter File’, select the `sM00223_0020_seg_sn.mat` file that you created in the previous section, press ‘Done’.
- Highlight ‘Images to Write’, select the bias-corrected structural eg. `msM00223_002.img`, press ‘Done’.
- Open ‘Writing Options’, select voxel sizes and change the default [2 2 2] to [1 1 3] which corresponds to the original resolution of the images.
- Save the job as `norm_struct.mat` and press ‘Run’.

31.1.5 Smoothing

Press the ‘Smooth’ button³. This will call up the specification of a smooth job in the graphics window.

- Open ‘Smooth’, select ‘Images to Smooth’ and then select the spatially normalised files created in the last section eg. `wrfM000*.img`.
- Highlight, ‘FWHM’ and change [8 8 8] to [6 6 6]. This will smooth the data by 6mm in each direction.
- Save the job as `smooth.mat` and press ‘Run’.

¹This step is not strictly necessary. It will write images out at a resolution closer to that at which they were acquired. This will speed up subsequent analysis and is necessary, for example, to make Bayesian fMRI analysis computationally efficient.

²Beginners may wish to skip this step, and instead just superimpose functional activations on an ‘average structural image’.

³The smoothing step is unnecessary if you are only interested in Bayesian analysis of your functional data.

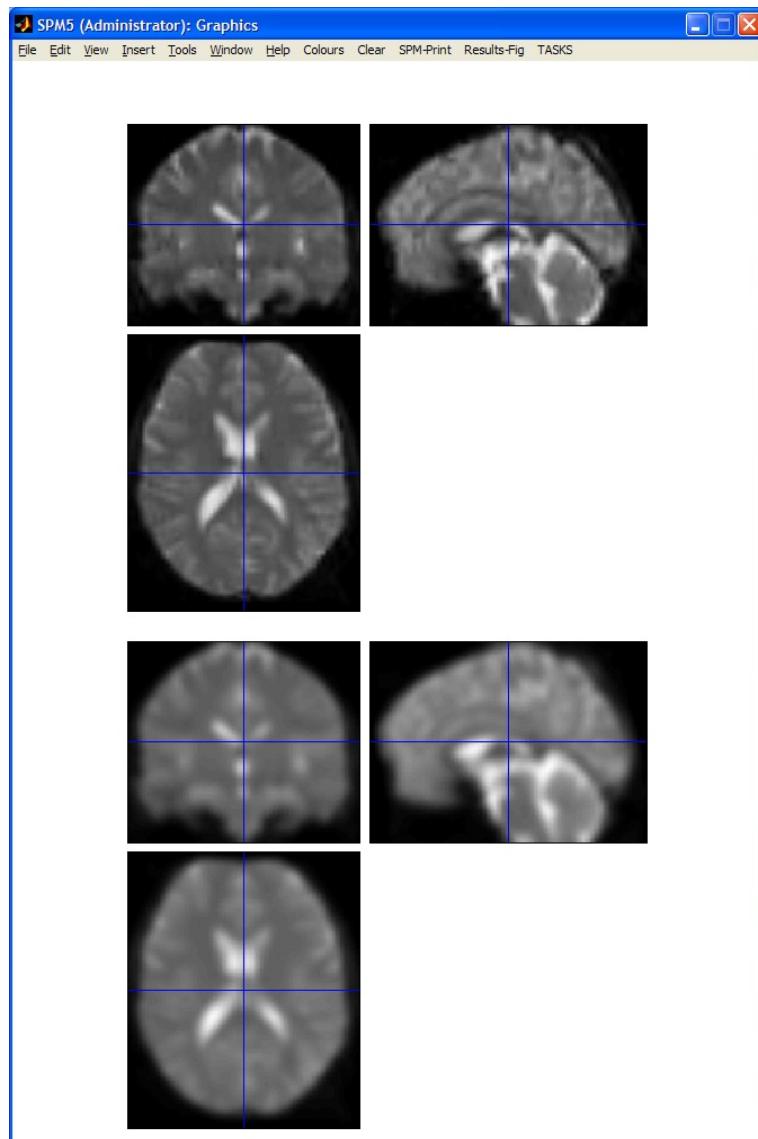


Figure 31.7: Functional image (top) and 6mm-smoothed functional image (bottom). These images were obtained using SPM's 'CheckReg' facility.

31.2 Model specification, review and estimation

To avoid T1 effects in the initial scans of an fMRI time series we recommend discarding the first few scans. To make this example simple, we'll discard the first complete cycle (12 scans, 04-15), leaving 84 scans, image files 16-99. This is best done by moving these files to a different directory.

Press the ‘Specify 1st-level’ button. This will call up the specification of an fMRI specification job in the graphics window. Then

- Open the fMRI model specification option
- Open the ‘Timing parameters’ option
- Highlight ‘Units for design’ and select ‘Scans’
- Highlight ‘Interscan interval’ and enter 7
- Highlight ‘Data and Design’ and select ‘New Subject/Session’. Then open the newly created ‘Subject/Session’ option.
- Highlight ‘Scans’ and use SPM’s file selector to choose the 84 smoothed, normalised functional images ie `swrfM00223_016.img - *_099.img`. These can be selected easily using the `^s.*` filter, and select all (provided you have moved the scans 4 to 15 into a different directory). Then press ‘Done’.
- Highlight ‘Condition’ and select ‘New condition’
- Open the newly created ‘Condition’ option. Highlight ‘Name’ and enter ‘active’. Highlight ‘Onsets’ and enter ‘6:12:84’. Highlight ‘Durations’ and enter ‘6’.
- Highlight ‘Directory’ and select the `DIR/classical` directory you created earlier.
- Save the job as `specify.mat` and press ‘RUN’

SPM will then write an `SPM.mat` file to the `DIR/classical` directory. It will also plot the design matrix, as shown in Figure 31.8.

At this stage it is advisable to check your model specification using SPM’s review facility which is accessed via the ‘Review’ button. This brings up a ‘design’ tab on the interactive window clicking on which produces a pulldown menu. If you select the first item ‘Design Matrix’ SPM will produce the image shown in Figure 31.8. If you select ‘Explore’ then ‘Session 1’ then ‘active’, SPM will produce the plots shown in Figure 31.9.

If you select the second item on the ‘Design’ tab, ‘Design Orthogonality’, SPM will produce the plot shown in Figure 31.10. Columns x_1 and x_2 are orthogonal if the inner product $x_1^T x_2 = 0$. The inner product can also be written $x_1^T x_2 = |x_1||x_2|\cos\theta$ where $|x|$ denotes the length of x and θ is the angle between the two vectors. So, the vectors will be orthogonal if $\cos\theta = 0$. The upper-diagonal elements in the matrix at the bottom of figure 31.10 plot $\cos\theta$ for each pair of columns in the design matrix. Here we have a single entry. A degree of non-orthogonality or collinearity is indicated by the gray shading.

31.2.1 Estimate

Press the ‘Estimate’ button. This will call up the specification of an fMRI estimation job in the graphics window. Then

- Open the ‘fMRI model estimation’ option
- Highlight the ‘Select SPM.mat’ option and then choose the `SPM.mat` file saved in the classical subdirectory
- Save the job as `estimate.job` and press Run

SPM will write a number of files into the selected directory including an `SPM.mat` file.

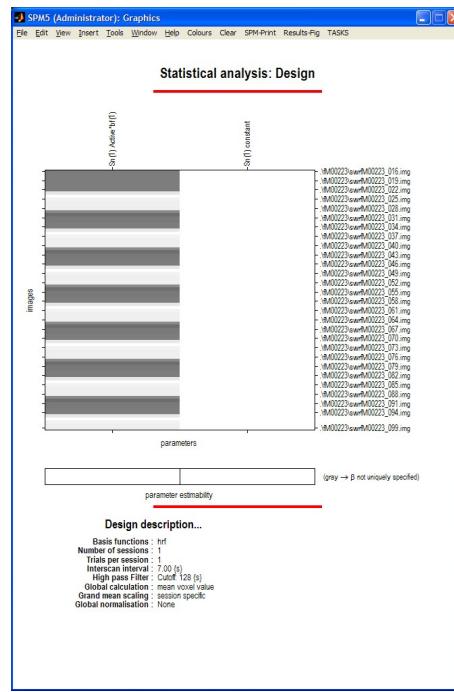


Figure 31.8: *Design matrix*. The filenames on the right-hand side of the design matrix indicate the scan associated with each row.

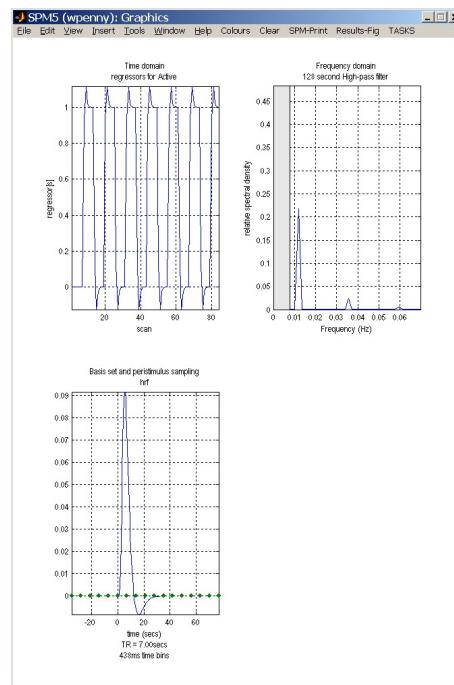


Figure 31.9: *Exploring the design matrix in Figure 31.8*. This shows the time series of the 'active' regressor (top left), a frequency domain plot of the active regressor (top right) and the basis function used to convert assumed neuronal activity into hemodynamic activity. In this model we used the default option - the canonical basis function. The frequency domain plot shows that the frequency content of the 'active' regressor is above the set frequencies that are removed by the High Pass Filter (HPF) (these are shown in gray - in this model we accepted the default HPF cut-off of 128s or 0.008Hz).

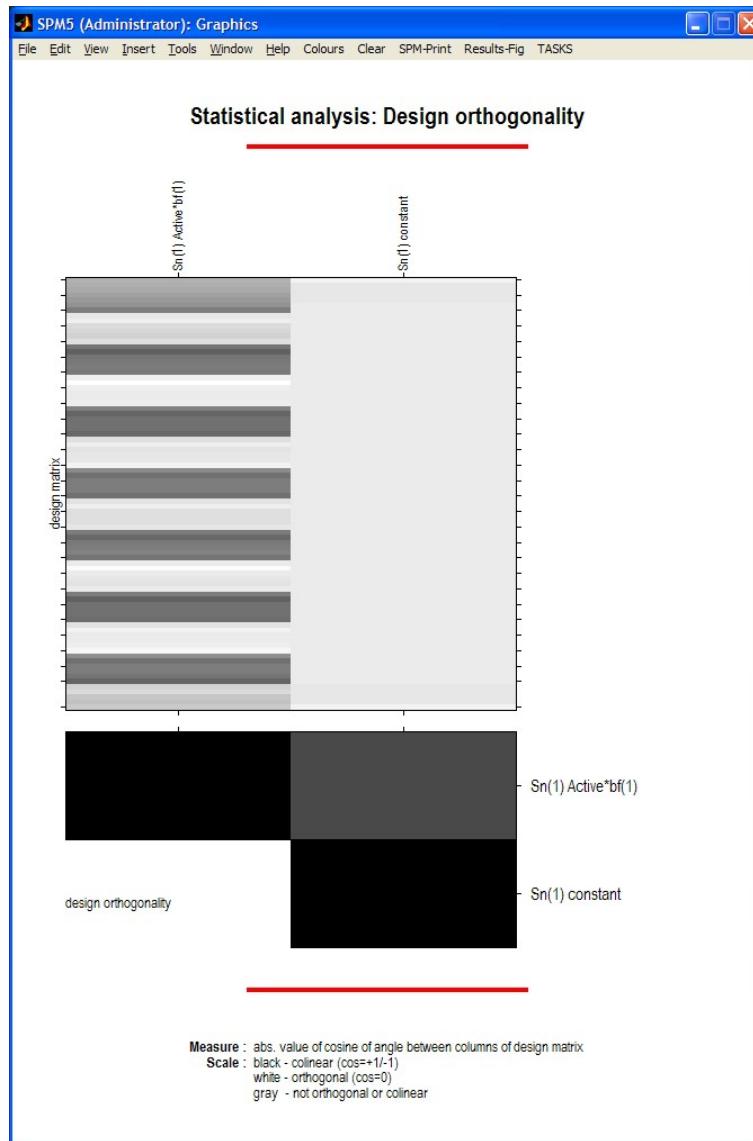


Figure 31.10: *Design Orthogonality*. The description above the first column in the design matrix $Sn(1)Active*bf(1)$ means that this column refers to the first session of data (in this analysis there is only 1 session), the name of this condition/trial is ‘Active’ and the trial information has been convolved with the first basis function (the canonical hemodynamic response). The constant regressor for session 1 is referred to as $Sn(1)Constant$. The orthogonality matrix at the bottom indicates a degree of collinearity between regressors.

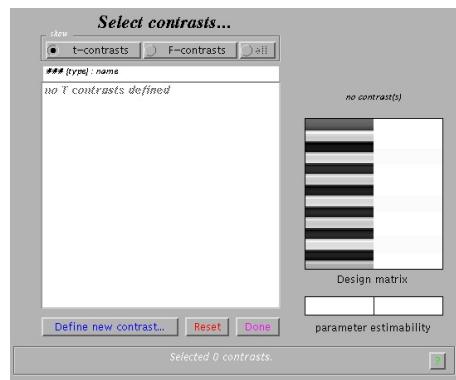


Figure 31.11: The contrast manager

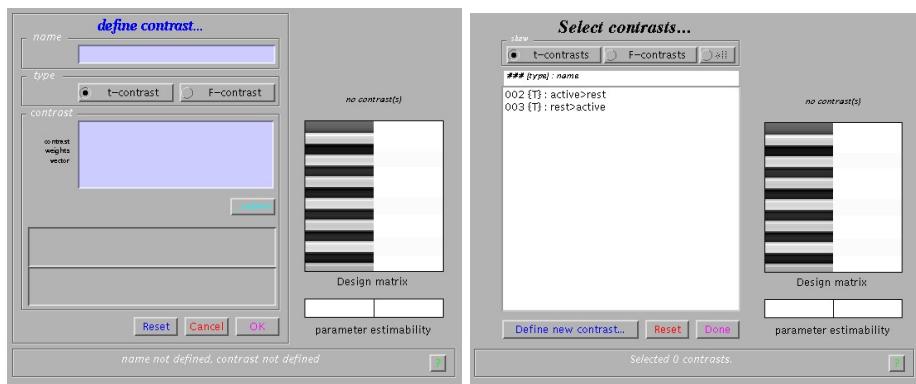


Figure 31.12: Left: A contrast is entered by specifying the numeric values in the lower window and the name in the upper window. Right: After contrasts have been specified they can be selected.

31.3 Inference

After estimation:

- Press ‘Results’
- Select the **SPM.mat** file created in the last section

This will invoke the contrast manager.

31.3.1 Contrast manager

The contrast manager displays the design matrix (surfable) in the right panel and lists specified contrasts in the left panel. Either ‘t-contrast’ or ‘F-contrast’ can be selected. To examine statistical results for condition effects

- Select ‘Define new contrast’

One sided main effects for the active condition (i.e., a one-sided t-test) can be specified (in this example) as ‘1’ (active > rest) and ‘-1’ (rest > active). SPM will accept correct contrasts only. Accepted contrasts are displayed at the bottom of the contrast manager window in green, incorrect ones are displayed in red. To view a contrast

- Select the contrast name e.g., ‘active > rest’
- Press ‘Done’

31.3.2 Masking

You will then be prompted with

- *Mask with other contrast ? [Yes/No]*
- Specify No.

Masking implies selecting voxels specified by other contrasts. If 'yes', SPM will prompt for (one or more) masking contrasts, the significance level of the mask (default $p = 0.05$ uncorrected), and will ask whether an inclusive or exclusive mask should be used. Exclusive will remove all voxels which reach the default level of significance in the masking contrast, inclusive will remove all voxels which do not reach the default level of significance in the masking contrast. Masking does not affect p-values of the 'target' contrast, it only includes or excludes voxels.

31.3.3 Thresholds

You will then be prompted with

- *Title for comparison ?*
- Enter eg. 'active > rest'
- *Corrected height threshold ? [Yes/No]*
- Enter Yes.
- *p value adjustment to control: [FWE/FDR/none]*
- Select FWE
- *p value(family-wise error)*
- Accept the default value, 0.05

A Family Wise Error (FWE) is a false positive anywhere in the SPM. Now, imagine repeating your experiment many times and producing SPMs. The proportion of SPMs containing FWEs is the FWE rate. A value of 0.05 implies that 1 in 20 SPMs contains a false positive somewhere in the image.

If you choose the 'none' option above this corresponds to making statistical inferences at the 'voxel level'. These use 'uncorrected' p values, whereas FWE thresholds are said to use 'corrected' p values. SPM's default uncorrected p value is $p=0.001$. This means that the probability of a false positive at each voxel is 0.001. So if, you have 50,000 voxels you can expect $50,000 \times 0.001 = 50$ false positives in each SPM.

The final option here is False Discovery Rate (FDR). If you set this at 0.1, this means that of all the discoveries you make (ie. above threshold voxels that appear in the SPM) 10% of them are likely to be false.

You will then be prompted with

- *Extent Threshold {voxels} [0]*
- Accept the default value, 0

Entering a value v here will produce SPMs with clusters containing at least v voxels. SPM will then produce the SPM shown in Figure 31.13.

- Select 'Define new contrast'

31.3.4 Files

A number of files are written to the working directory at this time. Images containing weighted parameter estimates are saved as `con-0002.hdr/img`, `con-0003.hdr/img`, etc. in the working directory. Images of T-statistics are saved as `spmT-0002.hdr/img`, `spmT-0003.hdr/img` etc., also in the working directory.

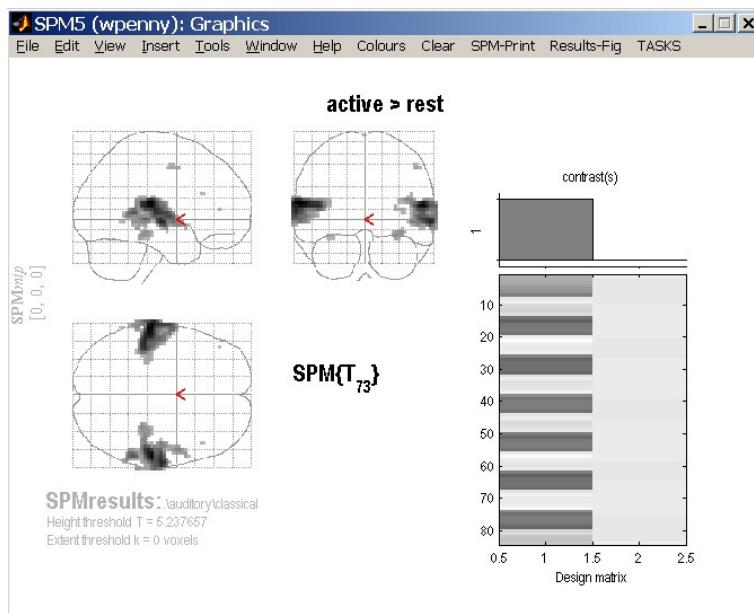


Figure 31.13: *SPM showing bilateral activation of auditory cortex.*

31.3.5 Maximum Intensity Projections

SPM displays a Maximum Intensity Projection (MIP) of the statistical map in the graphics window. The MIP is projected on a glass brain in three orthogonal planes. The MIP is surfable: R-clicking in the MIP will activate a pulldown menu, L-clicking on the red cursor will allow it to be dragged to a new position.

31.3.6 Design matrix

SPM also displays the design matrix with the selected contrast. The design matrix is also surfable: R-clicking will show parameter names, L-clicking will show design matrix values for each scan.

In the SPM Interactive window (lower left panel) a button box appears with various options for displaying statistical results (p-values panel) and creating plots/overlays (visualisation panel). Clicking 'Design' (upper left) will activate a pulldown menu as in the 'Explore design' option.

31.3.7 Statistical tables

To get a summary of local maxima, press the 'volume' button in the p-values section of the interactive window. This will list all clusters above the chosen level of significance as well as separate (>8mm apart) maxima within a cluster, with details of significance thresholds and search volume underneath, as shown in Figure 31.15

The columns in volume table show, from right to left:

- x, y, z (mm): coordinates in Talairach space for each maximum
- voxel-level: the chance (p) of finding (under the null hypothesis) a voxel with this or a greater height (T- or Z-statistic), corrected (FWE or FDR)/ uncorrected for search volume.
- cluster-level: the chance (p) of finding a cluster with this many(ke) or a greater number of voxels, corrected / uncorrected for search volume
- set-level: the chance (p) of finding this (c) or a greater number of clusters in the search volume

It is also worth noting that

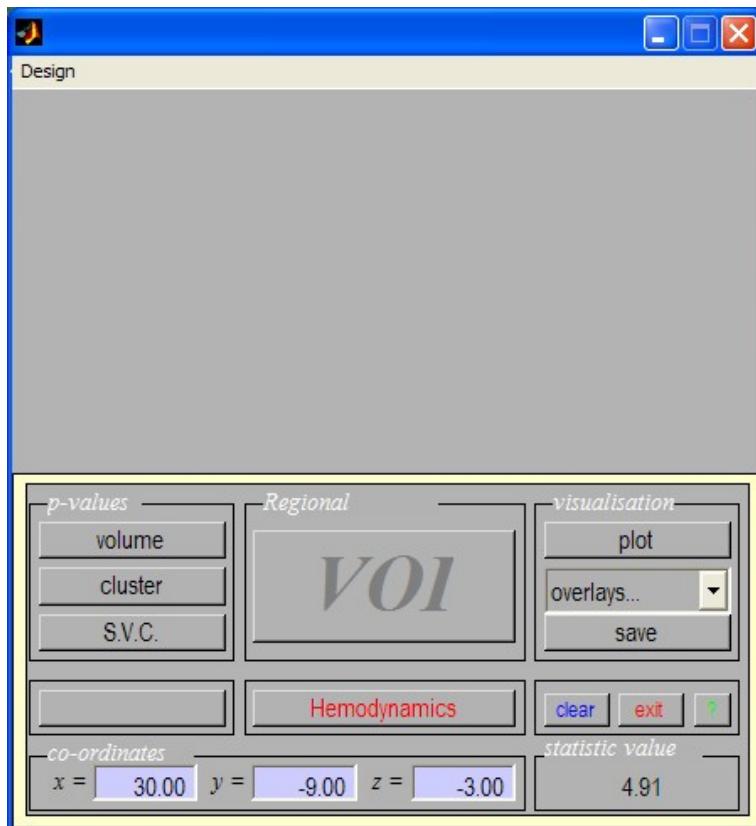


Figure 31.14: *SPM's interactive window during results assessment. The 'p-values' section is used to produce tables of statistical information. The visualisation section is used to plot responses at a voxel or to visualise activations overlaid on anatomical images. The 'Regional' section, i.e. the VOI button, is used to extract data for subsequent analyses such as assessment of PsychoPhysiological Interactions (PPIs) or Dynamic Causal Models (DCMs).*

Statistics: <i>p</i> -values adjusted for search volume									
set-level		cluster-level		voxel-level				x,y,z [mm]	
<i>p</i>	c	<i>p</i> connected	<i>k</i> _E	<i>p</i> uncorrected	<i>p</i> _{FWE-corr}	<i>p</i> _{FDR-corr}	T	$\langle Z \rangle$	<i>p</i> uncorrected
0.000	9	0.000	514	0.000	0.000	0.000	14.19	Inf	0.000
					0.000	0.000	11.86	Inf	0.000
					0.000	0.000	9.54	7.66	0.000
					0.000	0.000	13.62	Inf	0.000
					0.000	0.000	12.24	Inf	0.000
					0.000	0.000	9.82	7.80	0.000
					0.000	0.000	7.33	6.32	0.000
					0.001	0.000	6.32	5.63	0.000
					0.001	0.000	6.22	5.55	0.000
					0.002	0.000	6.07	5.44	0.000
					0.002	0.000	6.02	5.41	0.000
					0.006	0.000	5.76	5.22	0.000
					0.022	0.000	5.45	4.97	0.000
					0.047	0.000	5.25	4.82	0.000
0.005	2	0.058							
0.005	2	0.058							
0.015	1	0.166							

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 5.24, p = 0.000 (0.050)
 Extent threshold: k = 0 voxels, p = 1.000 (0.050)
 Expected voxels per cluster, $\langle k \rangle = 0.553$
 Expected number of clusters, $\langle c \rangle = 0.09$
 Expected false discovery rate, $\langle FDR \rangle = 0.00$

Degrees of freedom = [1.0, 73.0]
 FWHM = 8.9 8.9 7.9 mm; 3.0 3.0 2.6 (voxels);
 Volume: 1787508; 66204 voxels; 2573 resels
 Voxel size: 3.0 3.0 3.0 mm; (resel = 22.95 voxels)

Figure 31.15: Volume table for ‘active > rest’ effect. This table of values was created by pressing the ‘Results-Fig’ tab at the top of the graphics window and then pressing the ‘Volume’ button. This displays the table of results in a separate window.

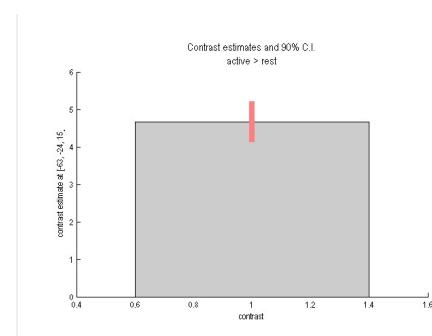
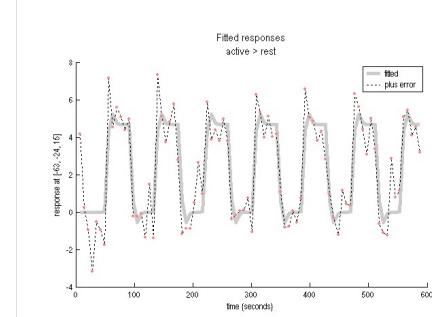
- The table is surfable: clicking a row of cluster coordinates will move the pointer in the MIP to that cluster, clicking other numbers will display the exact value in the Matlab window (e.g. 0.000 = 6.1971e-07).
- To inspect a specific cluster (e.g., in this example data set, the R auditory cortex), either move the cursor in the MIP (by L-clicking and dragging the cursor, or R-clicking the MIP background which will activate a pulldown menu).
- Alternatively, click the cluster coordinates in the volume table, or type the coordinates in the co-ordinates section of the interactive window.

It is also possible to produce tables of statistical information for a single cluster of interest rather than for the whole volume. Firstly, elect the relevant cluster in the MIP and then press the ‘cluster’ button in the p-values section of the interactive window. This will show coordinates and voxel-level statistics for local maxima (>4mm apart) in the selected cluster. This table is also surfable.

31.3.8 Plotting responses at a voxel

A voxel can be chosen with co-ordinates corresponding to those in the interactive window. The responses at this voxel can then be plotted using the ‘Plot’ button in the visualisation section of the interactive window. This will provide you with five further options:

- Contrast estimates and 90% CI: SPM will prompt for a specific contrast (e.g., active>rest). The plot will show effect size and 90% confidence intervals. See eg. Figure 31.16
- Fitted responses: Plots adjusted data and fitted response across session/subject. SPM will prompt for a specific contrast and provides the option to choose different ordinates (‘an explanatory variable’, ‘scan or time’, or ‘user specified’). If ‘scan or time’, the plot will show adjusted or fitted data with errors added as shown in Figure 31.17

Figure 31.16: *Estimated effect size.*Figure 31.17: *Fitted responses.*

3. Event-related responses: Plots adjusted data and fitted response across peri-stimulus time.
4. Parametric responses
5. Volterra kernels

For plotting event-related responses SPM provides three options

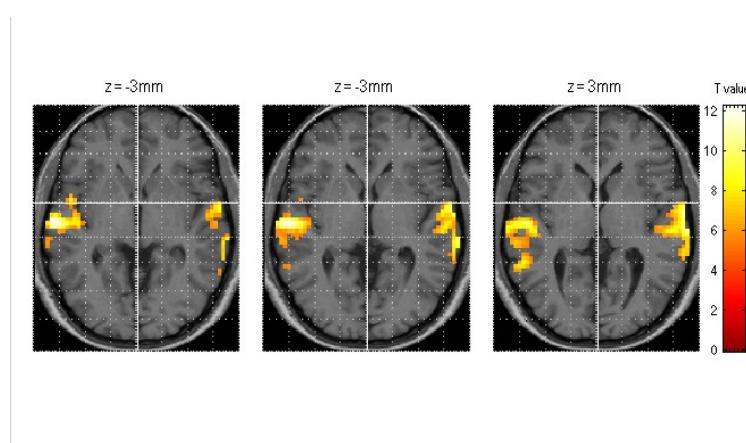
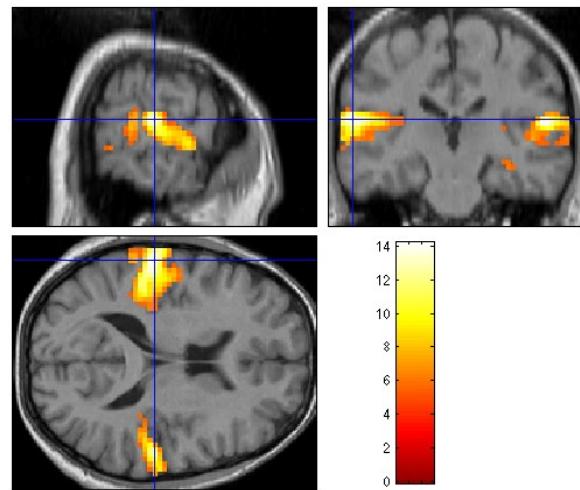
1. Fitted response and PSTH (peri-stimulus time histogram): plots mean regressor(s) (ie. averaged over session) and mean signal +/- SE for each peri-stimulus time bin.
2. Fitted response and 90% CI: plots mean regressor(s) along with a 90% confidence interval.
3. Fitted response and adjusted data: plots regressor(s) and individual data (note that in this example the data are shown in columns due to the fixed TR/ISI relationship).

Its worth noting that

- The values for the fitted response across session/subject for the selected plot can be displayed and accessed in the Matlab window by typing 'Y'. Typing 'y' will display the adjusted data.
- 'Adjusted' data = adjusted for confounds (e.g., global flow) and high- and low pass filtering.

31.3.9 Overlays

The visualisation section of the interactive window also provides an overlay facility for anatomical visualisation of clusters of activation. Pressing 'Overlays' will activate a pulldown menu with three options

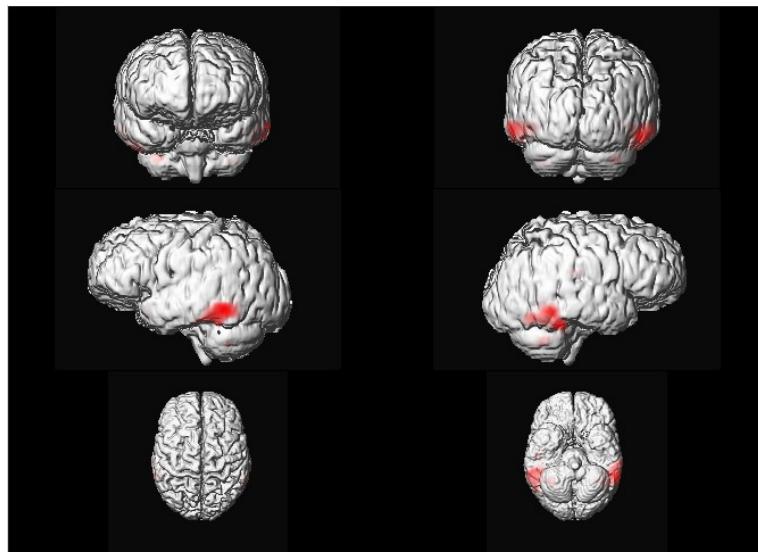
Figure 31.18: *Slices*.Figure 31.19: *Sections*.

1. Slices: overlay on three adjacent (2mm) transaxial slices. SPM will prompt for an image for rendering. This could be a canonical image (see `spm-template.man`) or an individual T1/mean EPI image for single-subject analyses.
2. Sections: overlay on three intersecting (sagittal, coronal, transaxial) slices. These renderings are surfable: clicking the images will move the crosshair.
3. Render: overlay on a volume rendered brain, with options for using a smoothed brain, and old (left) and new (right) style rendering.

Renderings can be saved as `filename.img` and `filename.hdr` in the working directory by using the *write filtered* option. In Figures 31.18, 31.19 and 31.20 the ‘active > rest’ activation has been superimposed on the spatially normalised, bias-corrected anatomical image `wmsM00223_002.img` created earlier.

For the ‘Render’ option we first created a rendering for this subject. This was implemented by

- Selecting ‘Xtract Surface’ from the ‘Render’ pulldown menu

Figure 31.20: *Render*.

- Selecting the gray and white matter images `c1sM00223_002.img` and `c2sM00223_002.img` created earlier.
- Saving the results using the default options (Rendering and Surface)

SPM plots the rendered anatomical image in the graphics window and saves it as `render_c1sM00223_002.img`. The surface image is saved as `surf_c1sM00223_002.img`.

31.3.10 Miscellaneous

Other options (in the results controls panel):

- clear: clears lower subpanel of Graphics window
- exit: exits the results section
- ? : launches spm-results-ui help

31.4 Bayesian analysis

31.4.1 Specification

Press the ‘Specify 1st-level’ button. This will call up an fMRI specification job in the graphics window. Then

- Open the fMRI model specification option
- Load the ‘specify.mat’ job file created for the classical analysis
- Open ‘Subject/Session’, highlight ‘Scans’
- Deselect the smoothed functional images using the ‘unselect all’ option available from a right mouse click in the SPM file selector (bottom window)
- Select the unsmoothed functional images using the `~w.*` filter and ‘select all’ option available from a right mouse click in the SPM file selector (top right window)⁴. The Bayesian analysis

⁴Remember not to select the first 12 scans, scans 4 to 15, as these may contain T1 effects. This can be done during selection or by first moving the files to a different directory.

uses a spatial prior where the spatial regularity in the signal is estimated from the data. It is therefore not necessary to create smoothed images if you are only going to do a Bayesian analysis.

- Press ‘Done’
- Highlight ‘Directory’ and select the DIR/bayesian directory you created earlier (you will first need to deselect the DIR/classical directory).
- Save the job as `specify_bayesian.mat` and press ‘Run’

31.4.2 Estimation

Press the ‘Estimate’ button. This will call up the specification of an fMRI estimation job in the graphics window. Then

- Open the ‘fMRI model estimation’ option
- Highlight the ‘Select SPM.mat’ option and then choose the SPM.mat file saved in the DIR/bayesian directory
- Highlight ‘Method’ and select the ‘Choose Bayesian 1st-level’ option.
- Open the newly created ‘Bayesian 1st-level’ option, highlight ‘AR model order’ and select 0. This data set has a TR=7s, so is unlikely to have temporally autocorrelated errors.
- Save the job as `estimate_bayesian.job` and press ‘Run’.

SPM will write a number of files into the output directory including

- An `SPM.mat` file.
- Images of estimated regression coefficients `Cbeta_0001.img` and `Cbeta_0002.img`. These filenames are prefixed with a ‘C’ indicating that these are the mean values of the ‘Conditional’ or ‘Posterior’ density.
- Images of error bars/standard deviations on the regression coefficients `SDbeta_0001.img` and `SDbeta_0002.img`.
- An image of the standard deviation of the error `Sess1_SDerror.img`.
- An image `mask.img` indicating which voxels were included in the analysis.

31.4.3 Inference

After estimation:

- Press ‘Results’
- Select the `SPM.mat` file created in the last section
- Select ‘Define new contrast’
- Enter the name ‘active > rest’
- Enter the value ‘1’, press ‘Submit’, ‘OK’, ‘Done’
- *Mask with other contrast ? [Yes/No]*
- Specify No
- Title for comparison, accept the default
- *Effect size threshold for PPM*

- Enter the value 2
- *Posterior probability threshold for PPM*
- Enter the value 0.99
- *Extent threshold [0]*
- Accept the default value
- *Plot effect size [Yes/No]*
- Select the default ‘Yes’

SPM will then plot a map of effect sizes at voxels where it is 99% sure that the effect size is greater than 2% of the global mean. This is a large activation. Then use overlays, sections, select the normalised structural image created earlier and move the cursor to the activation in the left hemisphere. This should create the plot shown in Figure 31.21

It is also possible to look for regions where responses in the active condition are different to those at rest. Active responses could be greater or smaller.

- Press ‘Results’
- Select the **SPM.mat** file created in the last section
- Select ‘Define new contrast’ and highlight the ‘F’ radio button
- Enter the name ‘active != rest’
- Enter the value ‘1’, press ‘Submit’, ‘OK’, ‘Done’
- *Mask with other contrast ? [Yes/No]*
- Specify No
- Title for comparison, accept the default
- *Posterior probability threshold for PPM*
- Accept the default value⁵
- *Extent threshold [0]*
- Accept the default value,0
- *Plot effect size [Yes/No]*
- Select the default ‘Yes’.

SPM will then plot a map of χ^2 statistic values at above threshold voxels. Then use overlays, sections, select the normalised structural image created earlier and move the cursor to the activation in the left hemisphere. This should create the plot shown in Figure 31.22

When you revisit the contrast manager this contrast will be referred to as a ‘P’ contrast, rather than an ‘F’ contrast. This indicates that Bayes rule is used to make the inference. To indicate that we are testing a two-sided effect it is advisable to make this clear when naming the contrast (as we have done with the label ‘active != rest’).

⁵The default PPM threshold is set to $1 - 1/S$ where S is the number of voxels in the volume being analysed. The rationale for this is that inference is based on an approximate posterior distribution, Q , which factorises across voxels. The approximate posterior is chosen to best match the true posterior in the sense of KL-divergence. Given the factorisation in Q , the expected number of false positives in the PPM is 1.

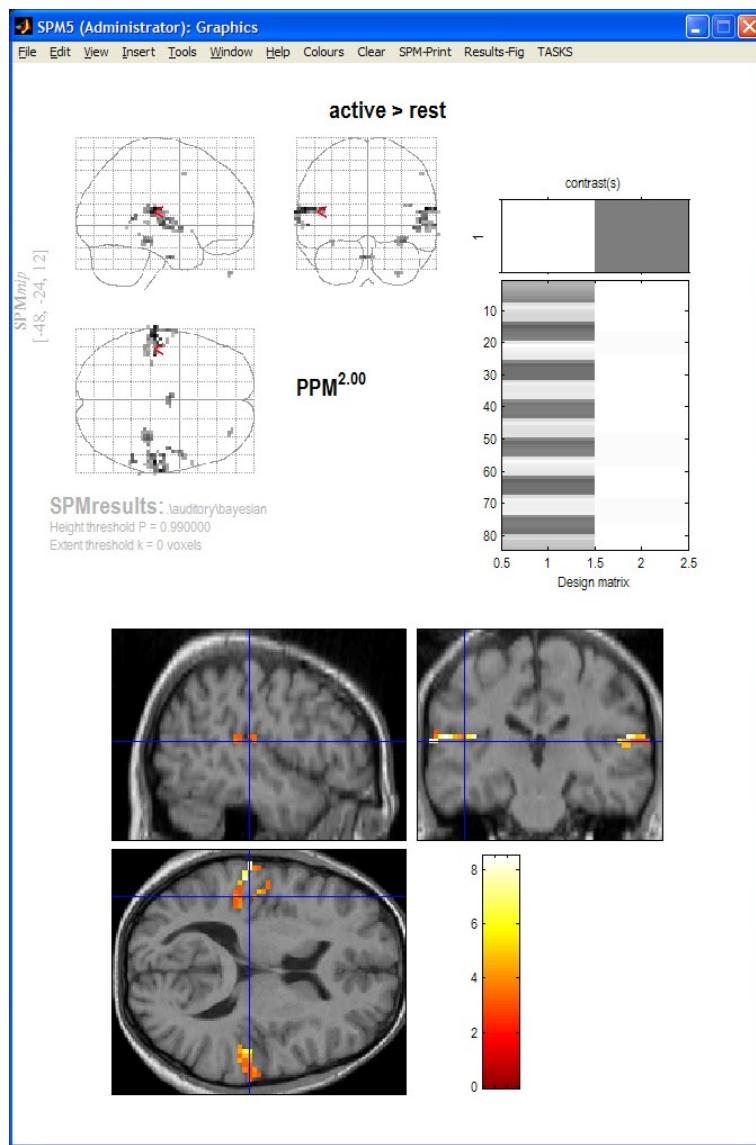


Figure 31.21: **Bayesian analysis:** MIP and overlay of effect sizes at voxels where SPM is 99% sure that the effect size is greater than 2% of the global mean.

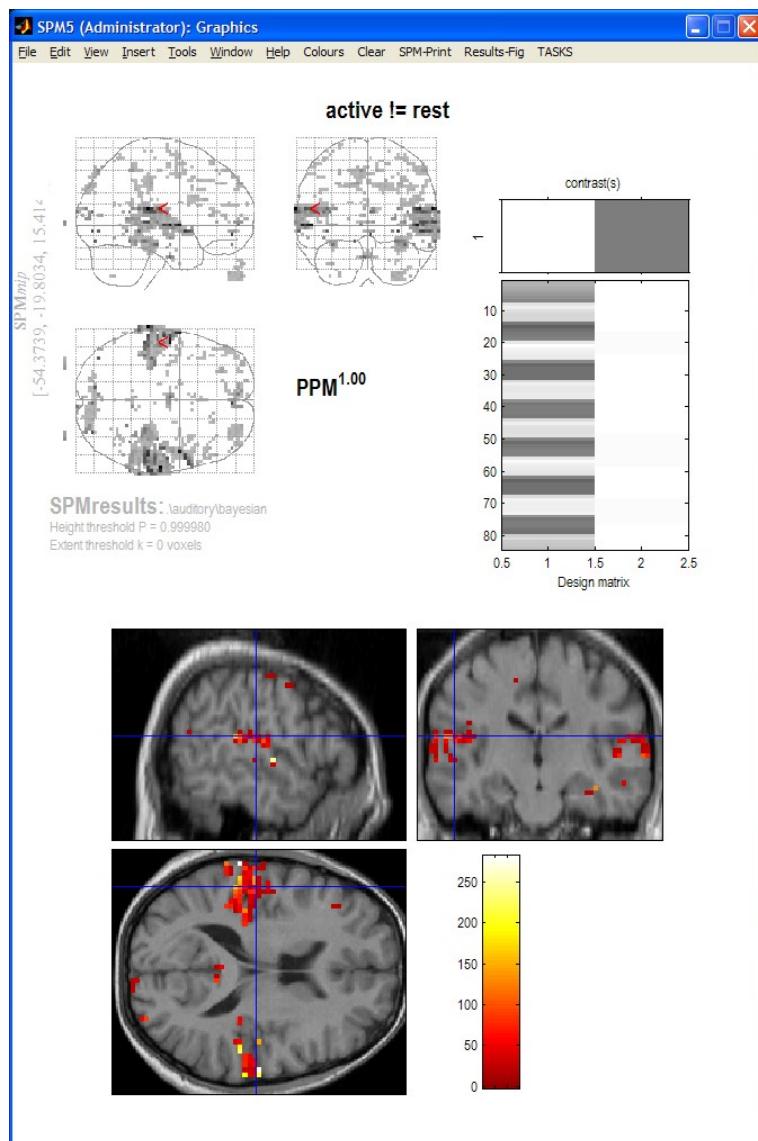


Figure 31.22: **Two-sided Bayesian analysis:** MIP and overlay of χ^2 statistic values at above threshold voxels. This shows regions where activity is different between active and rest conditions, whether positive or negative.

Chapter 32

Face fMRI data

As a third and more sophisticated example, consider the data from a repetition priming experiment performed using event-related fMRI. Briefly, this is a 2x2 factorial study with factors ‘fame’ and ‘repetition’ where famous and non-famous faces were presented twice against a checkerboard baseline. The subject was asked to make fame judgements by making key presses. There are thus four event-types of interest; first and second presentations of famous and non-famous faces, which we denote N1, N2, F1 and F2. The experimental stimuli and timings of events are shown in Figures 32.1 and 32.2.

Images were acquired using continuous Echo-Planar Imaging (EPI) with TE=40ms, TR=2s and 24 descending slices (64x64 3x3mm²), 3mm thick with a 1.5mm gap. The data archive is available from <http://www.fil.ion.ucl.ac.uk/spm/data/#event>. This contains 351 Analyse format functional images `sM03953_0005_*.img` of dimension 64x64x24 with 3mmx3mmx4.5mm voxels. A structural image is also provided `sM03953_0007.img` also in Analyse format.

To analyse the data, first create a new directory DIR

eg. `c:\home\wpenny\fmri_analysis\face-rep\all`, in which to place the results of your analysis. Then create 4 subdirectories (i) `jobs`, (ii) `categorical`, (iii) `parametric` and (iv) `bayesian`. As the analysis proceeds these directories will be filled with job-specification files, design matrices and models estimated using classical or Bayesian methods.

As well as the classical/Bayesian distinction we will show how this data can be analysed from a parametric as well as a categorical perspective. We will look at the main effects of fame and repetition and in the parametric analysis we will look at responses as a function of ‘lag’, that is, the number of faces intervening between repetition of a specific face.

Start up matlab, enter your `jobs` directory and type `spm fmri` at the matlab prompt. SPM will then open in fMRI mode with three windows (1) the top-left or ‘command’ window, (2) the bottom-left or ‘interactive’ window and (3) the right-hand or ‘graphics’ window. Analysis then takes place in three major stages (i) spatial pre-processing, (ii) model specification, review and estimation and (iii) inference. These stages organise the buttons in SPM’s base window.

32.1 Spatial pre-processing

32.1.1 Display

Display eg. the first functional image using the ‘Display’ button. Note orbitofrontal and inferior temporal drop-out and ghosting. This can be seen more clearly by selecting ‘brighten if necessary’ from the ‘Effects’ tab at the top of the graphics window.

32.1.2 Realignment

Under the spatial pre-processing section of the SPM base window select ‘Realign’ from the ‘Realign’ pulldown menu. This will call up a realignment job specification in the graphics window. Then

- Select ‘New Realign:Estimate and Reslice’

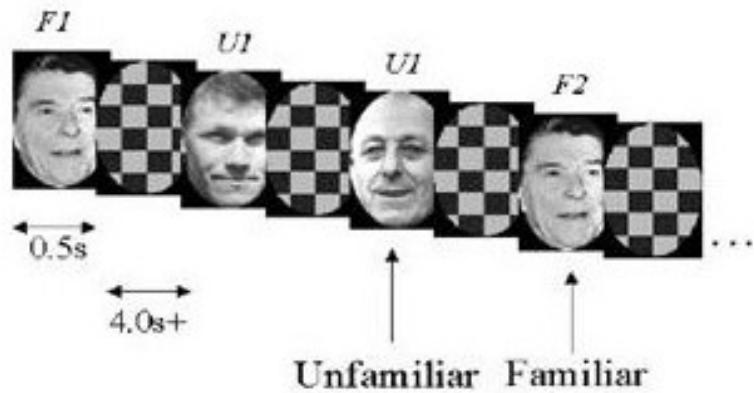


Figure 32.1: *Face repetition paradigm*. There were 2 presentations of 26 Famous and 26 Nonfamous Greyscale photographs, for 0.5s each, randomly intermixed. The minimal Stimulus Onset Asynchrony (SOA)=4.5s, with probability 2/3 (ie 1/3 null events). The subject made one of two right finger key presses denoting whether or not the subject thought the face was famous.

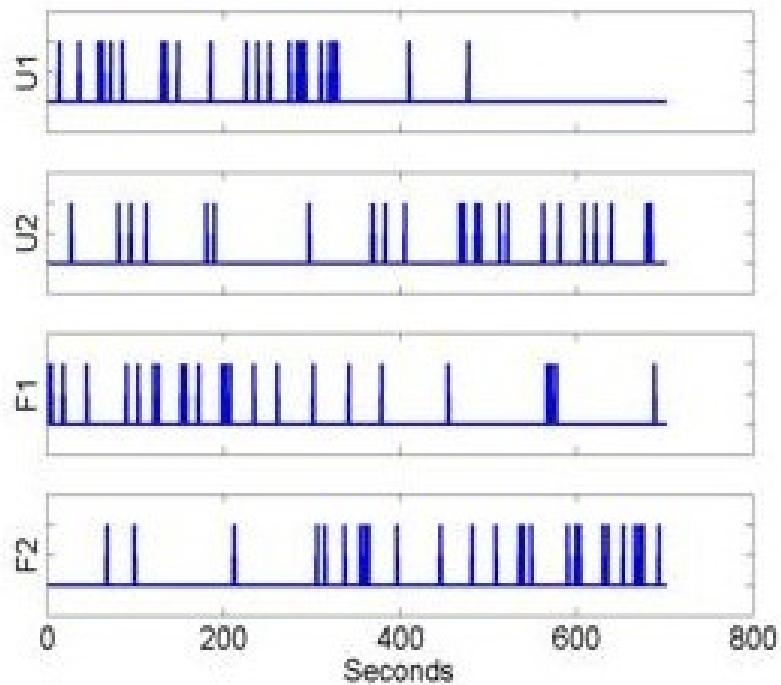


Figure 32.2: *Time series of events*.

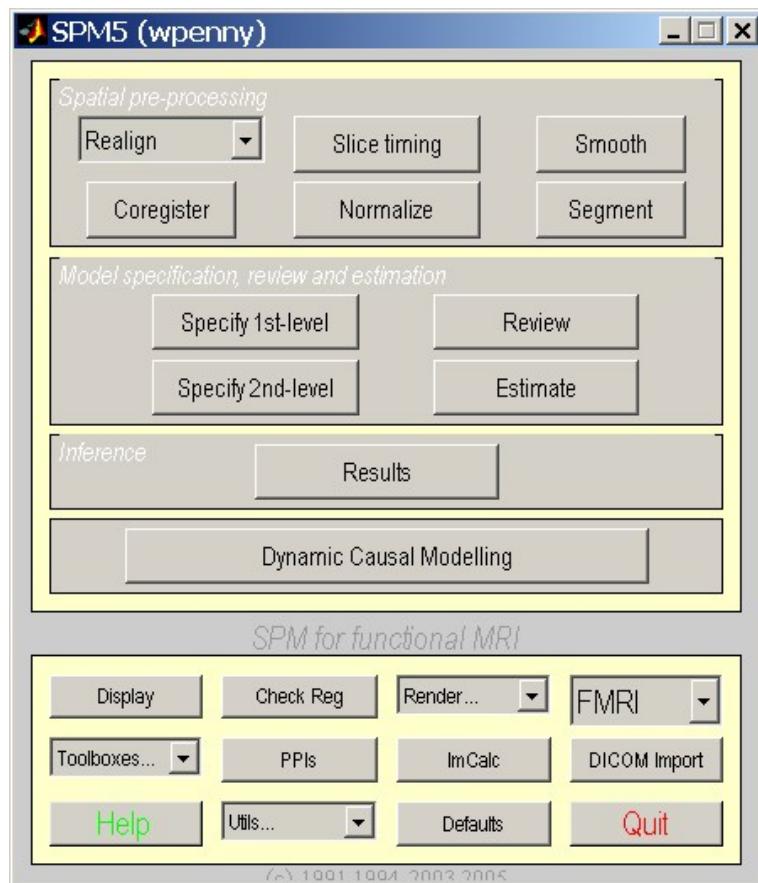


Figure 32.3: The SPM base window comprises three sections i) spatial pre-processing, (ii) model specification, review and estimation and (iii) inference.

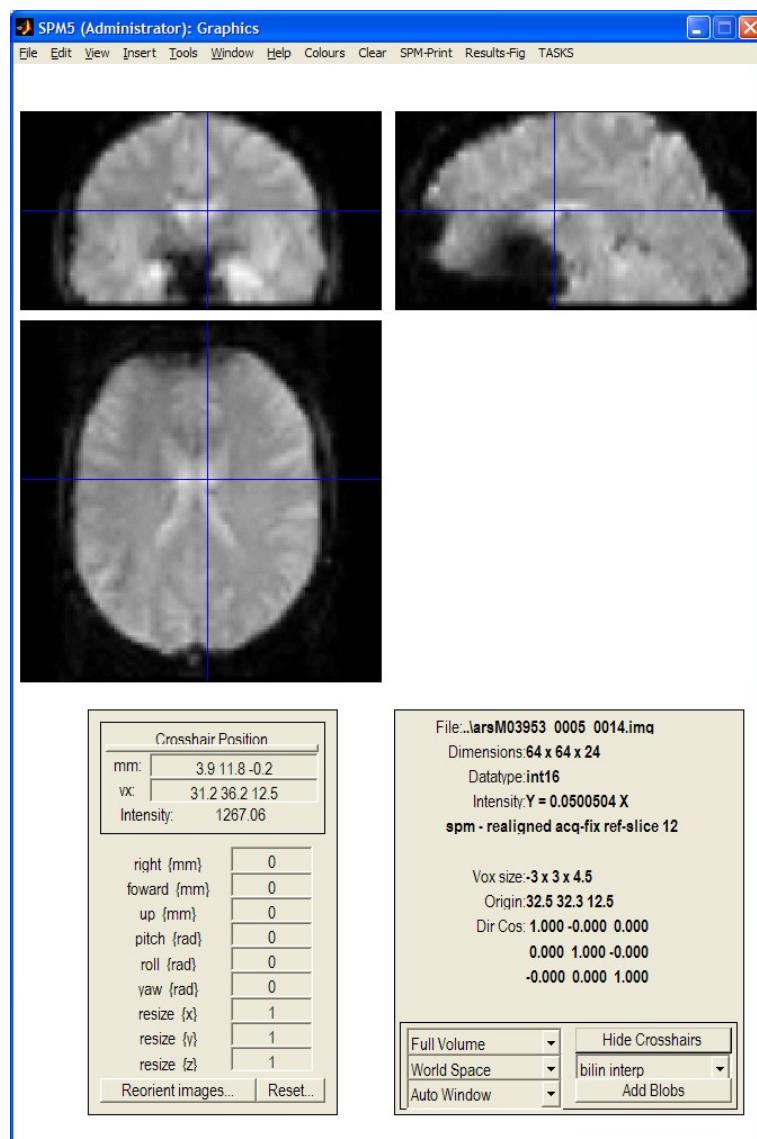


Figure 32.4: Signal dropout in EPI images.

- Open the newly created ‘Realign:Estimate and Reslice’ option.
- Highlight data, select ‘New Session’, then highlight the newly created ‘Session’ option.
- Select ‘Specify Files’ and use the SPM file selector to choose all of your functional images eg. `sM03953_0005_*.img`.
- Save the job file as eg. `DIR\jobs\realign.mat`.
- Press the RUN button in the graphics window.

This will run the realign job which will write realigned images into the directory where the functional images are. These new images will be prefixed with the letter ‘r’. SPM will then plot the estimated time series of translations and rotations shown in Figure 32.5. These data, the realignment parameters, are also saved to a file eg. `rp_sM03953_0005_0006.txt`, so that these variables can be used as regressors when fitting GLMs. To prepare for this copy the file into the `DIR\jobs\` directory and rename it `movepars.txt`. This allows movements effects to be discounted when looking for brain activations.

SPM will also create a mean image eg. `meansM03953_0005_0006.img` which will be used in the next step of spatial processing - coregistration.

32.1.3 Slice timing correction

Press the ‘Slice timing’ button. This will call up the specification of a slice timing job in the graphics window.

- Open the ‘Slice Timing’ option
- Highlight ‘Data’ and select ‘New Sessions’
- Highlight the newly create ‘Sessions’ option, ‘Specify Files’ and select the 351 realigned functional images using the filter `^r.*`.
- Select ‘Number of Slices’ and enter 24
- Select TR and enter 2
- Select TA and enter 2
- Select ‘Slice order’ and enter 24:-1:1
- Select ‘Reference Slice’, and enter 12
- Save the job as `slice_timing.mat` and press ‘Run’

SPM will write slice-time corrected files with the prefix ‘a’ in the functional data directory.

32.1.4 Coregistration

Press the ‘Coreg’ button. This will call up the specification of a coregistration job in the graphics window.

- Select New ”Coreg:Estimate”
- Double-click on the newly created Coreg:Estimate
- Highlight ‘Reference Image’ and then select the mean functional image `meansM03953_0005_0006.img`
- Highlight ‘Source Image’ and then select the structural image eg. `sM03953_0007.img`.
- Press the Save button and save the job as `coreg.job`
- Then press RUN

SPM will then implement a coregistration between the structural and functional data that maximises the mutual information. The image in figure 32.6 should then appear in the graphics window. SPM will have changed the header of the source file which in this case is the structural image `sM03953_0007.img`.

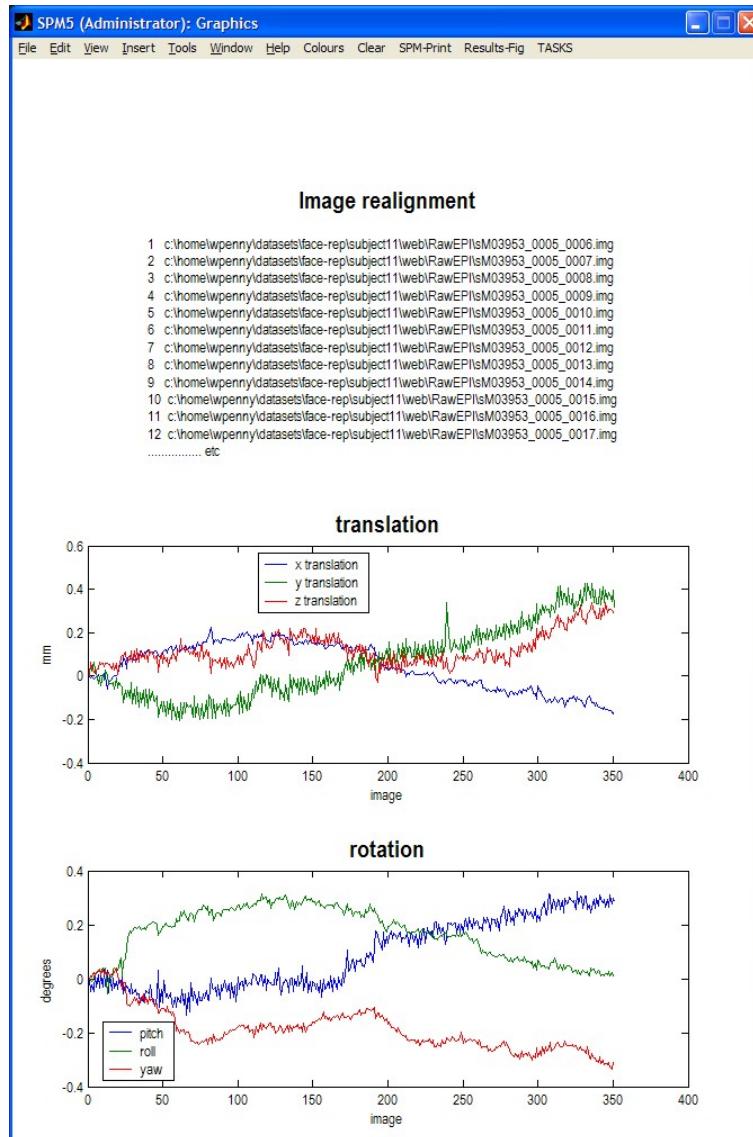


Figure 32.5: *Realignment of face data. Movement less than the size of a voxel, which for this data set is 3mm, is not considered problematic.*

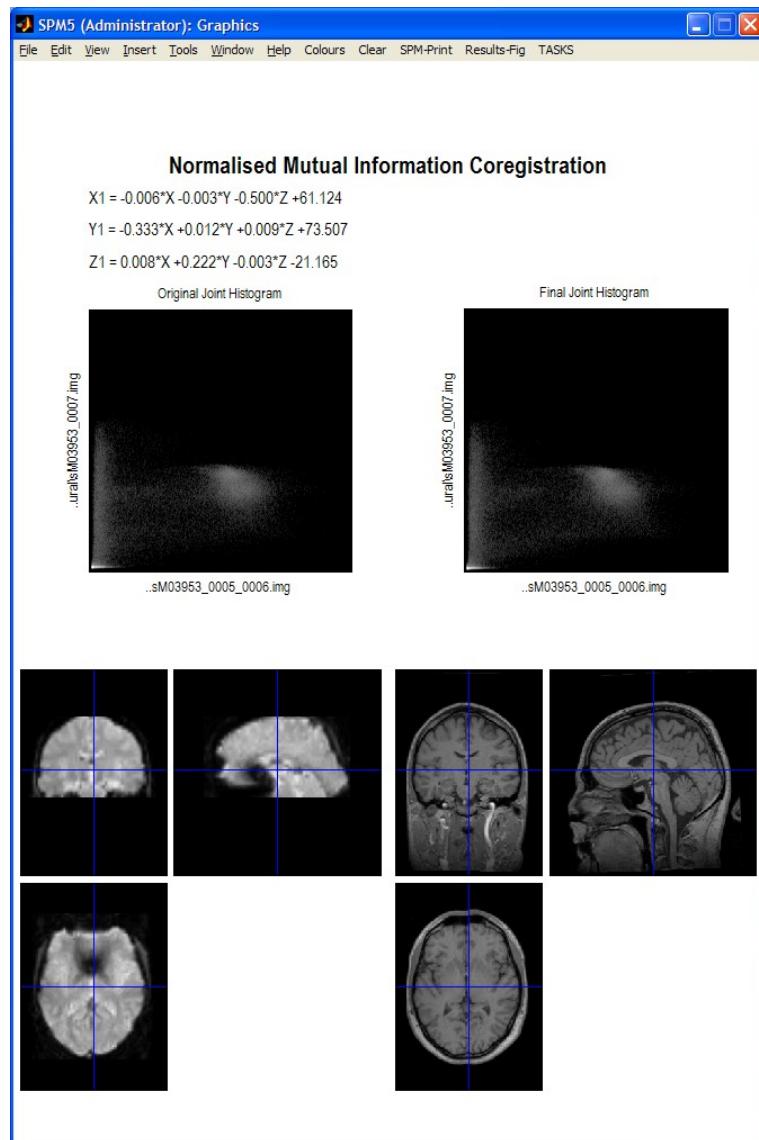


Figure 32.6: *Mutual Information Coregistration of Face data.*

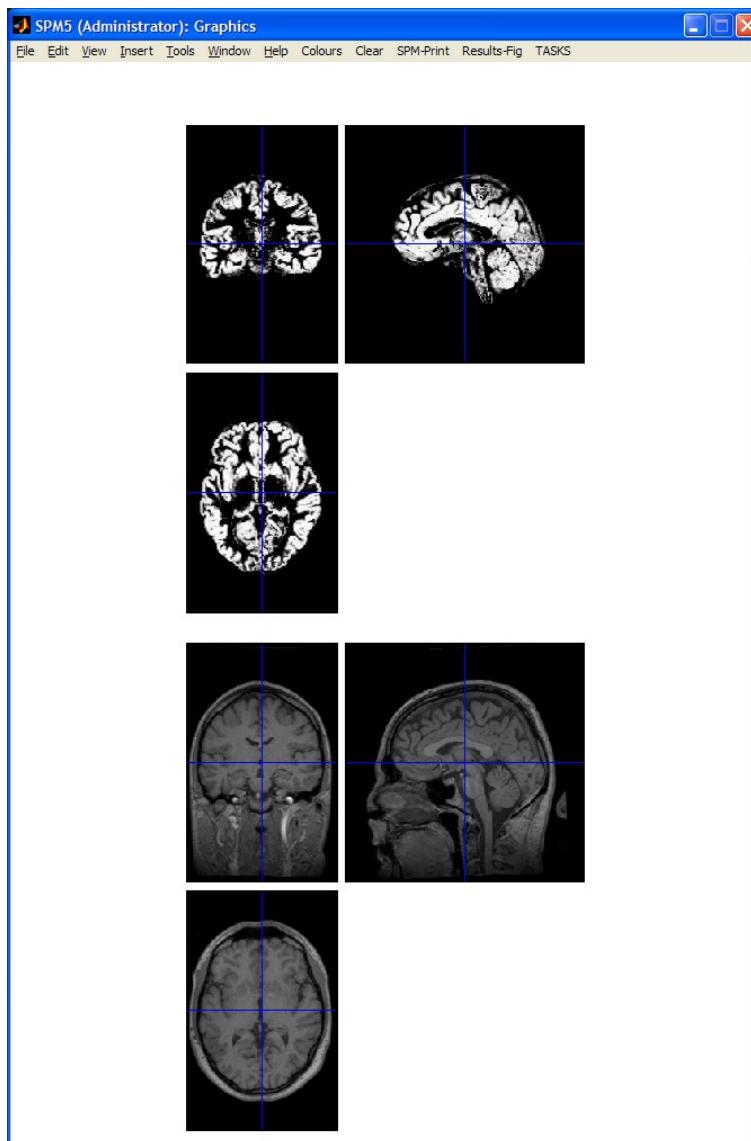


Figure 32.7: Gray matter (top) produced by segmentation of structural image (below).

32.1.5 Segmentation

Press the ‘Segment’ button. This will call up the specification of a segmentation job in the graphics window. Highlight the Data field and then select the subjects coregistered anatomical image eg. `sM03953_0007.img`. Save the job file as `segment.mat` and then press RUN. SPM will segment the structural image using the default tissue probability maps as priors. SPM will create, by default, gray and white matter images and bias-field corrected structural image. These can be viewed using the CheckReg facility as described in the previous section (press segment and select . Figure 32.7 shows the gray matter image, `c1sM03953_0007.img` along with the original structural.

SPM will also write a spatial normalisation eg. `sM03952_0007_seg_sn.mat` file in the original structural directory. This will be used in the next section to normalise the functional data.

32.1.6 Normalize

Press the ‘Normalize’ button. This will call up the specification of a normalise job in the graphics window.

- Select a New "Normalise:Write" field. This allows previously estimated warps to be applied to a series of images.
- Highlight 'Data', select New "Subject"
- Open 'Subject', highlight 'Parameter File' and select the `sM03952_0007_seg_sn.mat` file that you created in the previous section
- Highlight images to write and select all of the slice-time corrected, realigned functional images '`arsM*.img`'. Note: This can be done efficiently by changing the filter in the SPM file selector to `^ar.*`. SPM will then only list those files beginning with the letter *r* ie. those that have been realigned. You can then right click over the listed files, choose 'Select all' and press 'Done'.
- Open 'Writing Options', and change 'Voxel sizes' from [2,2,2] to [3,3,3].¹
- Press 'Save', save the job as `normalise.mat` and then press 'Run'.

SPM will then write spatially normalised files to the functional data directory. These files have the prefix 'w'.

If you wish to superimpose a subject's functional activations on their own anatomy² you will also need to apply the spatial normalisation parameters to their (bias-corrected) anatomical image. To do this

- Press 'Normalise', select New "Normalise:Write"
- Open 'Normalise: Write', highlight 'Data', select New "Subject"
- Open 'Subject'
- Highlight 'Parameter File', select the `sM03952_0007_seg_sn.mat` file that you created in the previous section, press 'Done'.
- Highlight 'Images to Write', select the bias-corrected structural eg. `msM03952_0007.img`, press 'Done'.
- Open 'Writing Options', select voxel sizes and change the default [2 2 2] to [1 1 1] which better matches the original resolution of the images [1 1 1.5].
- Save the job as `norm_struct.mat` and press 'Run'.

32.1.7 Smoothing

Press the 'Smooth' button³. This will call up the specification of a smooth job in the graphics window.

- Open 'Smooth', select 'Images to Smooth' and then select the spatially normalised files created in the last section eg. `war*.img`.
- Save the job as `smooth.mat` and press 'Run'.

This will smooth the data by 8mm in each direction, the default smoothing kernel width.

¹This step is not strictly necessary. It will write images out at a resolution closer to that at which they were acquired. This will speed up subsequent analysis and is necessary, for example, to make Bayesian fMRI analysis computationally efficient.

²Beginners may wish to skip this step, and instead just superimpose functional activations on an 'average structural image'.

³The smoothing step is unnecessary if you are only interested in Bayesian analysis of your functional data.

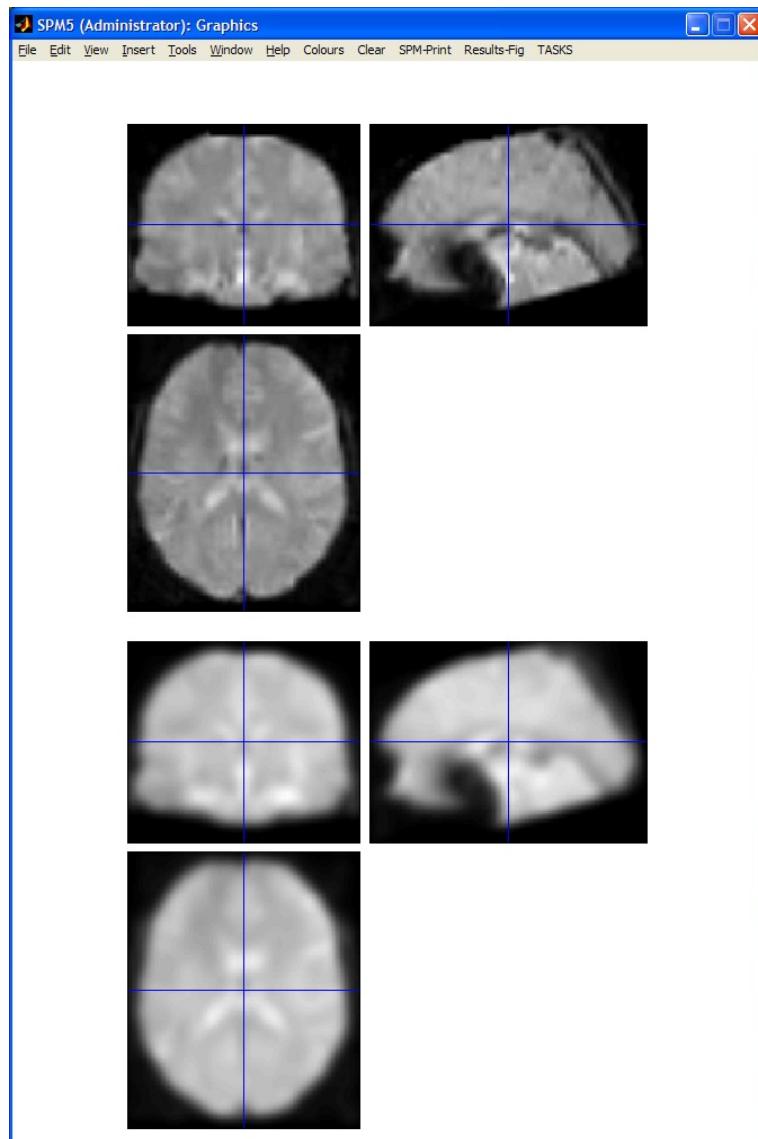


Figure 32.8: Functional image (top) and 8mm-smoothed functional image (bottom). These images were plotted using SPM's 'CheckReg' facility.

32.2 Modelling categorical responses

Before setting up the design matrix we must first load the Stimulus Onsets Times (SOTs) and movement parameters into matlab. SOTs are stored in the `sots.mat` file in a cell array such that eg. `sot{1}` contains stimulus onset times in TRs for event type 1, which is N1. Event-types 2,3 and 4 are N2, F1 and F2⁴.

- At the matlab command prompt type ‘load sots’
- Then type ‘load movepars.txt’

Now press the ‘Specify 1st-level’ button. This will call up the specification of a fMRI specification job in the graphics window. Then

- Open the fMRI model specification option
- Open the ‘Timing parameters’ option
- Highlight ‘Units for design’ and select ‘Scans’
- Highlight ‘Interscan interval’ and enter 2
- Highlight ‘Microtime resolution’ and enter 24
- Highlight ‘Microtime onset’ and enter 12. These last two options make the creating of regressors commensurate with the slice-time correction we have applied to the data.
- Highlight ‘Data and Design’ and select ‘New Subject/Session’. Then open the newly created ‘Subject/Session’ option.
- Highlight ‘Scans’ and use SPM’s file selector to choose the 351 smoothed, normalised, slice-time corrected, realigned functional images ie `swarsM.img`. These can be selected easily using the `^swar.*` filter, and select all. Then press ‘Done’.
- Highlight ‘Conditions’ and select ‘New condition’⁵
- Open the newly created ‘Condition’ option. Highlight ‘Name’ and enter ‘N1’. Highlight ‘Onsets’ and enter ‘`sot{1}`’. Highlight ‘Durations’ and enter 0.
- Highlight ‘Conditions’ and select ‘New condition’
- Open the newly created ‘Condition’ option. Highlight ‘Name’ and enter ‘N2’. Highlight ‘Onsets’ and enter ‘`sot{2}`’. Highlight ‘Durations’ and enter 0.
- Highlight ‘Conditions’ and select ‘New condition’
- Open the newly created ‘Condition’ option. Highlight ‘Name’ and enter ‘F1’. Highlight ‘Onsets’ and enter ‘`sot{3}`’. Highlight ‘Durations’ and enter 0.
- Highlight ‘Conditions’ and select ‘New condition’
- Open the newly created ‘Condition’ option. Highlight ‘Name’ and enter ‘F2’. Highlight ‘Onsets’ and enter ‘`sot{4}`’. Highlight ‘Durations’ and enter 0.
- Highlight ‘Regressors’⁶, select ‘New Regressor’, open the newly created ‘Regressor’ option, highlight ‘Name’ enter ‘`tx`’, highlight ‘Value’, enter `movepars(:,1)`

⁴Also included with this data set are variables describing two event-types of no interest - the (rare) errors made by this subject on the fame-judgement task. We will not, however, be using these in our analyses.

⁵It is also possible to enter information about all of the conditions in one go. This requires much less button pressing and can be implemented by highlighting the ‘Multiple conditions’ option and then selecting the `all-conditions.mat` file.

⁶It is also possible to enter information about all of the regressors in one go. This requires much less button pressing and can be implemented by highlighting the ‘Multiple regressors’ option and then selecting the `movement-parameters.mat` file.

- Highlight ‘Regressors’, select ‘New Regressor’, open the newly created ‘Regressor’ option, highlight ‘Name’ enter ‘ty’, highlight ‘Value’, enter `movepars(:,2)`
- Highlight ‘Regressors’, select ‘New Regressor’, open the newly created ‘Regressor’ option, highlight ‘Name’ enter ‘tz’, highlight ‘Value’, enter `movepars(:,3)`
- Highlight ‘Regressors’, select ‘New Regressor’, open the newly created ‘Regressor’ option, highlight ‘Name’ enter ‘pitch’, highlight ‘Value’, enter `movepars(:,4)`
- Highlight ‘Regressors’, select ‘New Regressor’, open the newly created ‘Regressor’ option, highlight ‘Name’ enter ‘roll’, highlight ‘Value’, enter `movepars(:,5)`
- Highlight ‘Regressors’, select ‘New Regressor’, open the newly created ‘Regressor’ option, highlight ‘Name’ enter ‘yaw’, highlight ‘Value’, enter `movepars(:,6)`
- Highlight ‘Factorial Design’, select ‘New Factor’, open the newly created ‘Factor’ option, highlight ‘Name’ and enter ‘Fam’, highlight ‘Levels’ and enter 2.
- Highlight ‘Factorial Design’, select ‘New Factor’, open the newly created ‘Factor’ option, highlight ‘Name’ and enter ‘Rep’, highlight ‘Levels’ and enter 2⁷.
- Open ‘Canonical HRF’ under ‘Basis Functions’. Select ‘Model derivatives’ and select ‘Time derivatives’.
- Highlight ‘Directory’ and select the `DIR/categorical` directory you created earlier.
- Save the job as `categorical_spec.mat` and press ‘RUN’

SPM will then write an `SPM.mat` file to the `DIR/categorical` directory. It will also plot the design matrix, as shown in Figure 32.9.

At this stage it is advisable to check your model specification using SPM’s review facility which is accessed via the ‘Review’ button. This brings up a ‘design’ tab on the interactive window clicking on which produces a pulldown menu. If you select the first item ‘Design Matrix’ SPM will produce the image shown in Figure 32.9. If you select ‘Explore’ then ‘Session 1’ then ‘N1’, SPM will produce the plots shown in Figure 32.10.

32.2.1 Estimate

Press the ‘Estimate’ button. This will call up the specification of an fMRI estimation job in the graphics window. Then

- Open the ‘fMRI model estimation’ option
- Highlight the ‘Select SPM.mat’ option and then choose the `SPM.mat` file saved in the `DIR/categorical` directory
- Save the job as `categorical_est.job` and press Run

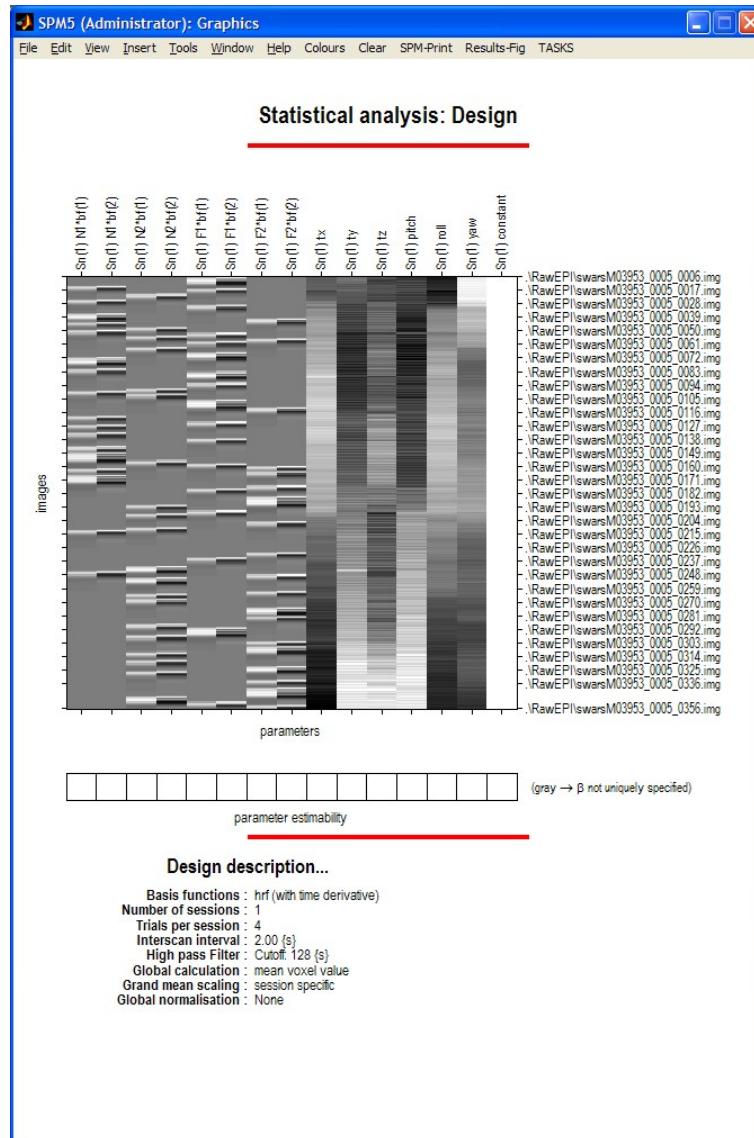
SPM will write a number of files into the selected directory including an `SPM.mat` file.

32.2.2 Inference for categorical design

Press ‘Results’ and select the `SPM.mat` file from `DIR\categorical`. This will again invoke the contrast manager. Because we specified that our model was using a ‘Factorial design’ a number of contrasts have been specified automatically, as shown in Figure 32.11.

- Select contrast number 5. This is a t-contrast `Positive effect of condition_1` This will show regions where the average effect of presenting faces is significantly positive, as modelled by the first regressor (hence the `_1`), the canonical HRF. Press ‘Done’.

⁷The order of naming these factors is important - the factor to be specified first is the one that ‘changes slowest’ ie. as we go through the list of conditions N1,N2,F1,F2 the factor ‘repetition’ changes every condition and the factor ‘fame’ changes every other condition. So ‘Fam’ changes slowest and is entered first.

Figure 32.9: *Design matrix.*

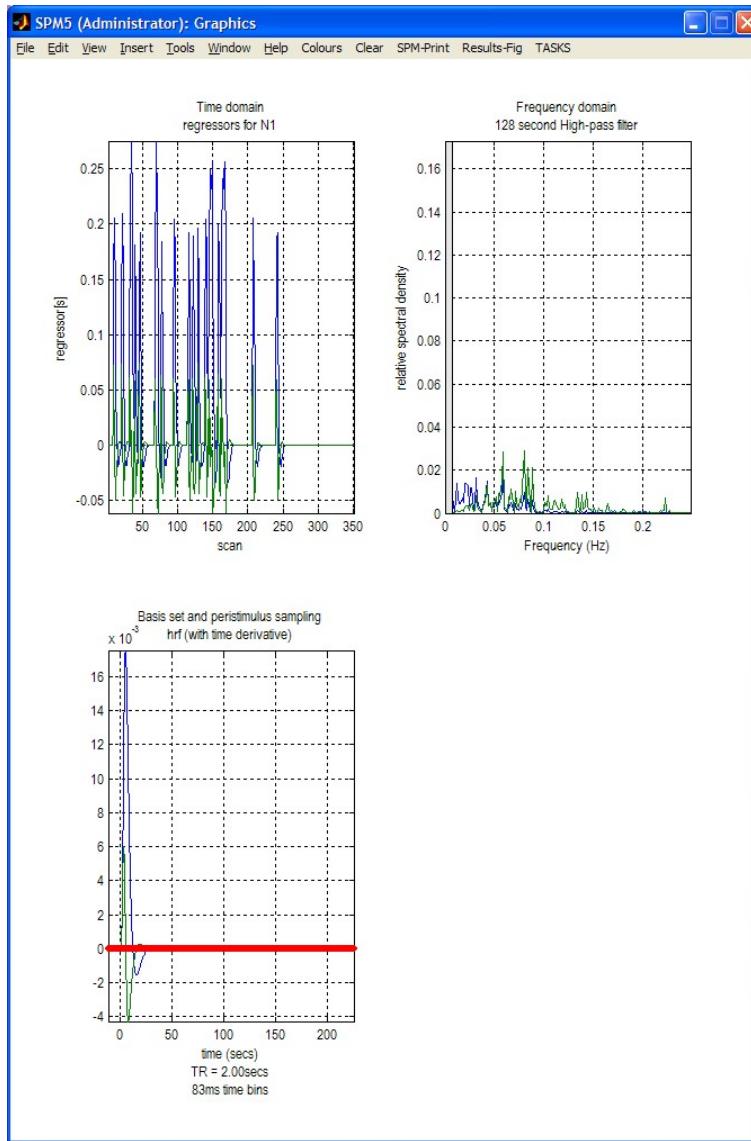


Figure 32.10: Exploring the design matrix in Figure 32.9. This shows the time series of the ‘active’ regressor (top left), a frequency domain plot of the active regressor (top right) and the basis function used to convert assumed neuronal activity into hemodynamic activity. In this model we used the default option - the canonical basis function. The frequency domain plot shows that the frequency content of the ‘N1’ regressor is above the set frequencies that are removed by the High Pass Filter (HPF) (these are shown in gray - in this model we accepted the default HPF cut-off of 128s or 0.008Hz).

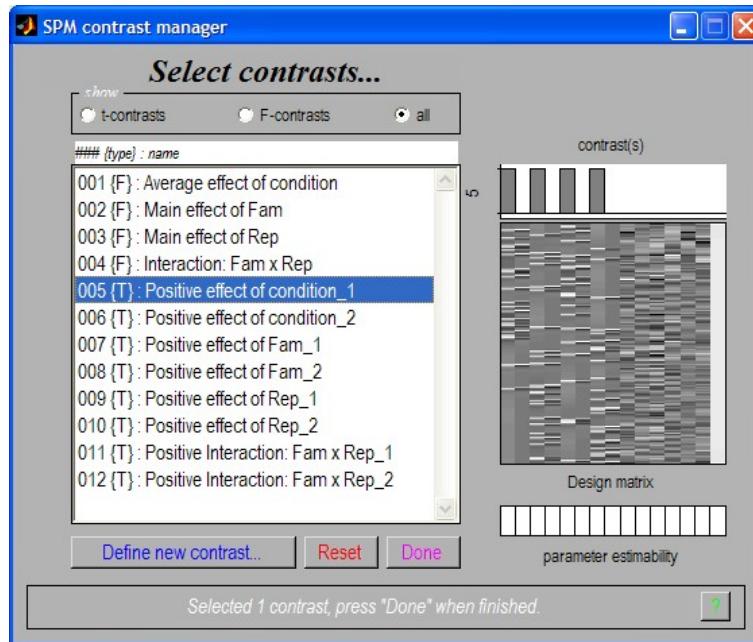


Figure 32.11: *Contrast Manager* containing default contrasts for categorical design.

- *Mask with other contrast ? [Yes/No]*
- Specify No.
- *Title for comparison ?*
- Enter ‘Canonical HRF: Faces > Baseline’
- *p value adjustment to control: [FWE/FDR/none]*
- Select FWE
- *Corrected p value(family-wise error)*
- Accept the default value, 0.05
- *Extent threshold {voxels} [0]*
- Accept the default value, 0

SPM will then produce the MIP shown in Figure 32.12.

32.2.3 Statistical tables

To get a summary of local maxima, press the ‘Volume’ button in the p-values section of the interactive window. This will list all clusters above the chosen level of significance as well as separate (>8mm apart) maxima within a cluster, with details of significance thresholds and search volume underneath, as shown in Figure 32.12

The columns in volume table show, from right to left:

- x, y, z (mm): coordinates in Talairach space for each maximum
- voxel-level: the chance (p) of finding (under the null hypothesis) a voxel with this or a greater height (T- or Z-statistic), corrected (FWE or FDR)/ uncorrected for search volume.
- cluster-level: the chance (p) of finding a cluster with this many(ke) or a greater number of voxels, corrected / uncorrected for search volume

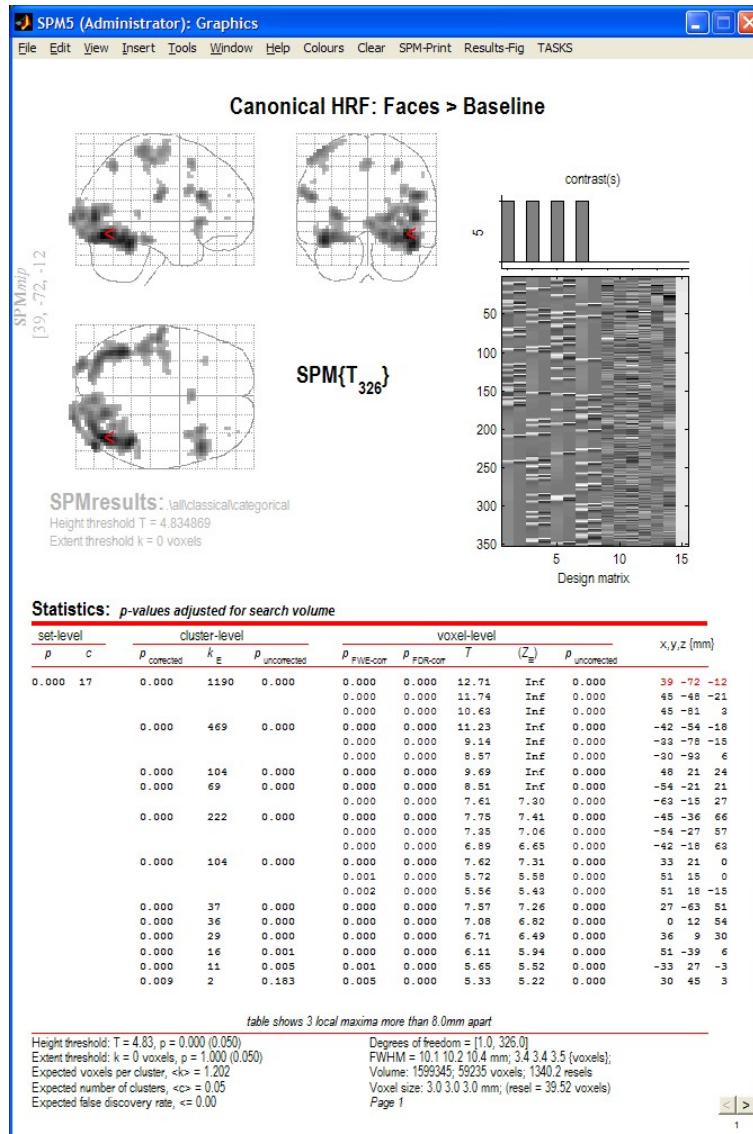


Figure 32.12: MIP and Volume table for Canonical HRF: Faces > Baseline.

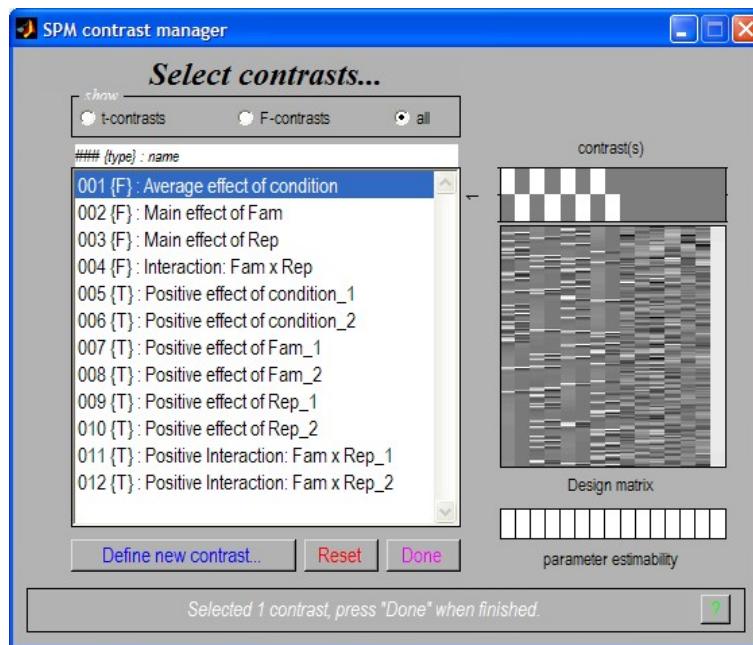


Figure 32.13: *Contrast manager showing selection of the first contrast 'Average effect of condition', also known as 'HRF + deriv: Faces > Baseline'.*

- set-level: the chance (p) of finding this (c) or a greater number of clusters in the search volume

32.2.4 F-contrasts

To assess the effect of presenting faces, as characterised by both the hrf *and* its temporal derivative, an F-contrast is required. Again, because we have told SPM that we have a factorial design, this required contrast will have been created automatically.

- Press 'Results' and select the SPM.mat file in the DIR/categorical directory
- Select contrast number 1, as shown in Figure 32.13. Press 'Done'.
- *Mask with other contrast ? [Yes/No]*
- Specify No.
- *Title for comparison ?*
- Enter 'HRF + deriv: Faces > Baseline'
- *p value adjustment to control: [FWE/FDR/none]*
- Select FWE
- *Corrected p value(family-wise error)*
- Accept the default value, 0.05
- *Extent threshold {voxels} [0]*
- Accept the default value, 0

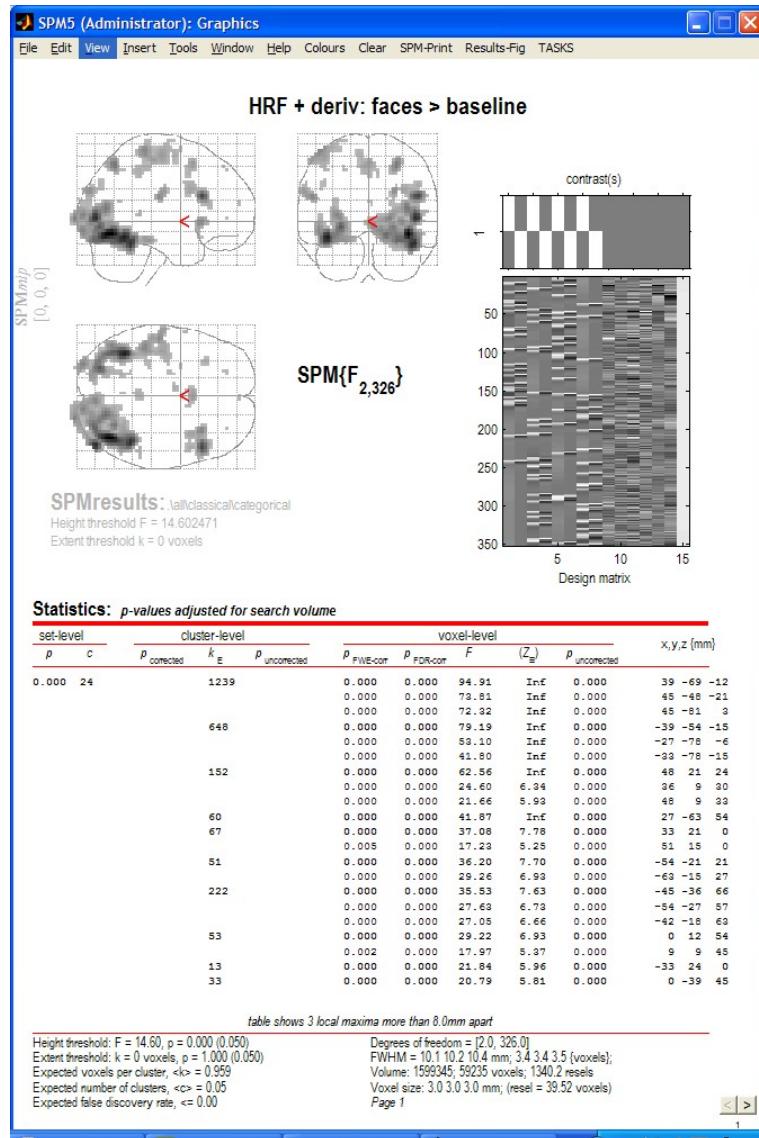


Figure 32.14: MIP and Volume table for Canonical HRF: Faces > Baseline.

Again, when the MIP appears, press 'Volume'. This should result in the display shown in Figure 32.14.

Although the MIP of this F-contrast looks similar to the previous t-contrast, note that the present contrast shows the areas for which the mean of the parameter estimates for the canonical hrf *or* its temporal derivative for the four conditions are different from zero (baseline). In other words, the F-contrast is two-sided, and tests two t-contrasts simultaneously. Also note that an F- (or t-) contrast such as [1 1 1 1 1 1 1], which tests whether the mean of the canonical hrf AND its temporal derivative for all conditions are different from (larger than) zero is not sensible. This is because the canonical hrf and its temporal derivative may cancel each other out while being significant in their own right. Finally, note that single F-contrasts can be regarded as t^2 -contrasts, so that the F-contrasts [1 0 -1 0 1 0 -1 0] and [-1 0 1 0 -1 0 1 0] are equivalent.

The relative contributions of the canonical hrf and its temporal derivative may also be assessed using the following F-contrast approach. This will allow us to plot the estimated effect sizes at a given voxel.

- Press 'Results' and select the SPM.mat file in the DIR/categorical directory
- Press 'Define new contrast', enter the name 'All effects of interest', press the 'F-contrast' radio button, enter [9:15] in the 'columns in reduced design' window, press 'OK', press 'Done'.
- *Mask with other contrast ? [Yes/No]*
- Specify No.
- *Title for comparison ?*
- Accept what is offered
- *p value adjustment to control: [FWE/FDR/none]*
- Select FWE
- *Corrected p value(family-wise error)*
- Accept the default value, 0.05
- *Extent threshold {voxels} [0]*
- Accept the default value, 0

Again, when the MIP appears, press 'Volume'. This should result in the display shown in Figure 32.16.

32.2.5 Plotting parameter estimates at a voxel

Move the cursor to e.g. the anterior right fusiform blob (the R fusiform face area) by clicking the second set of coordinates in the cluster (45 -48 -21) and press the 'plot' button in SPM's interactive window. Select 'Contrast of estimates and 90% CI' and select 'All effects of interest'. This will produce the plot in Figure 32.17.

A positive coefficient for the canonical indicates a positive response to faces, whereas a positive coefficient for the derivative indicates a later peak response to faces. Note, however, the large amplitudes and smaller error bars of the canonical hrfs compared to the temporal derivatives. This suggests that, for this region at least, the timing of the canonical model is adequate. Also note the repetition x stimulus type interaction; F1 seems larger than F2, whereas there is no difference between N1 and N2.

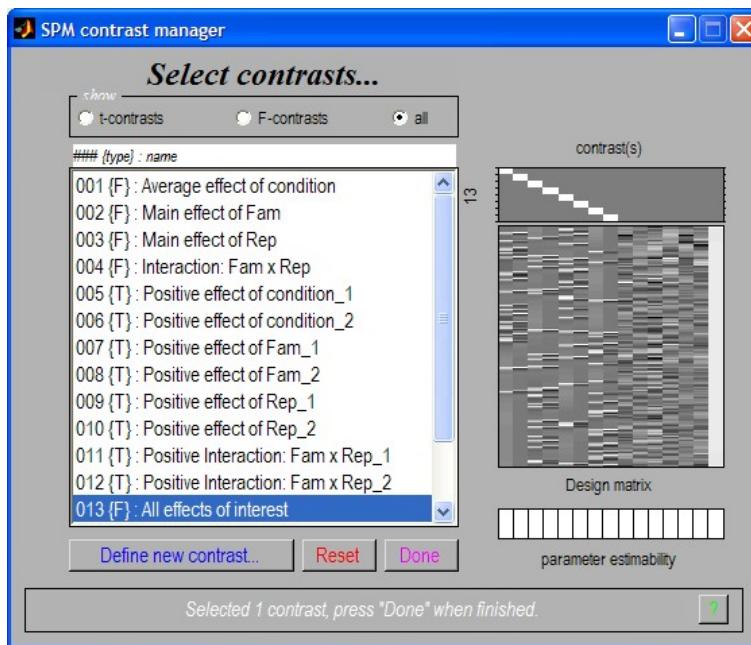


Figure 32.15: *Contrast manager showing specification of ‘All effects of interest’.*

32.2.6 F-contrasts for testing effects of movement

To assess movement-related activation

- Press ‘Results’, select the SPM.mat file, select ‘F-contrast’ in the Contrast Manager. Specify e.g. ‘Rotation-related activation’ (name) and in the ‘contrasts weights matrix’ window, or ‘1:11 15’ in the ‘columns for reduced design’ window.
- Submit and select the contrast, specify ‘mask with other contrasts?’ (no), ‘title for comparison’ (accept default), ‘corrected height threshold’ (FWE), and ‘corrected p-value’ (accept default).
- When the MIP appears, select ‘sections’ from the ‘overlays’ pulldown menu, and select the normalised structural image `wmsM03953_0007.img`

32.3 Modelling parametric responses

Before setting up the design matrix we must first load the Stimulus Onsets Times (SOTs), lag times, and movement parameters into matlab. SOTs are stored in the `sots.mat` file in a cell array such that eg. `sot{1}` contains stimulus onset times in TRs for event type 1, which is N1. Event-types 2,3 and 4 are N2, F1 and F2⁸.

- At the matlab command prompt type `load sots`. This loads the stimulus onset times (and the lag times).

Now press the ‘Specify 1st-level’ button. This will call up the specification of a fMRI specification job in the graphics window. Then

- Open the fMRI model specification option
- Pres, ‘Load’ and select the `categorical_spec.mat` job file you created earlier

⁸Also included with this data set are variables describing two event-types of no interest - the (rare) errors made by this subject on the fame-judgement task. We will not, however, be using these in our analyses.

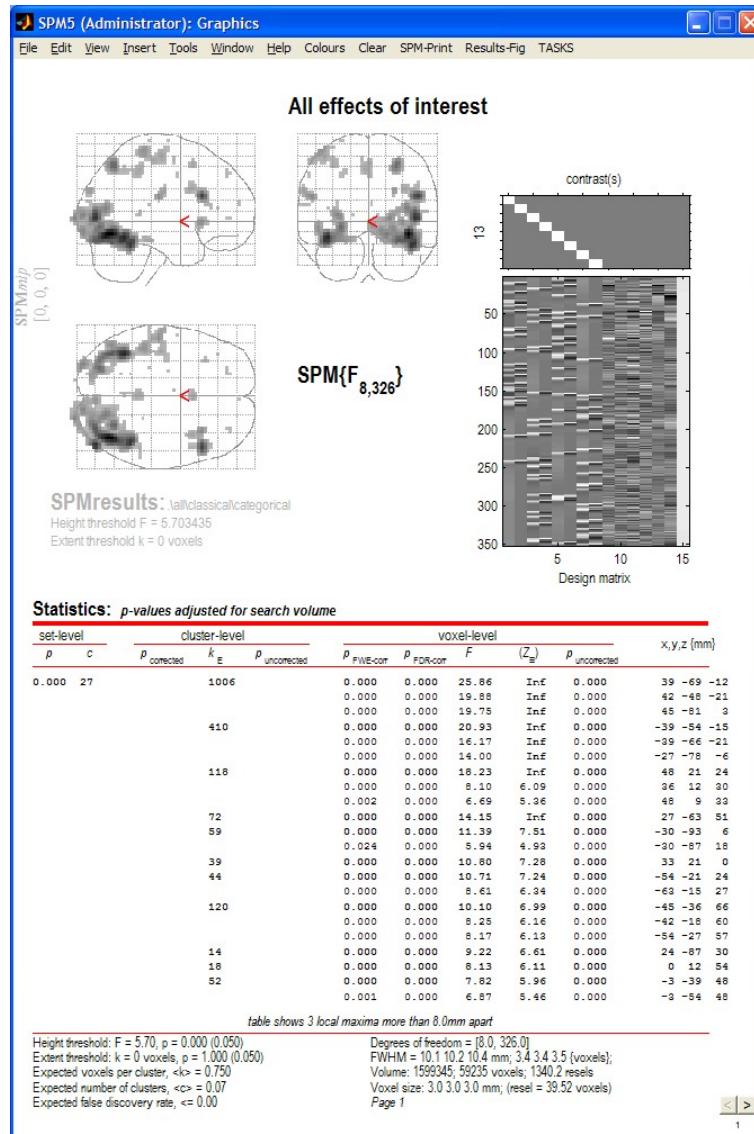


Figure 32.16: MIP and Volume table for 'All effects of interest'.

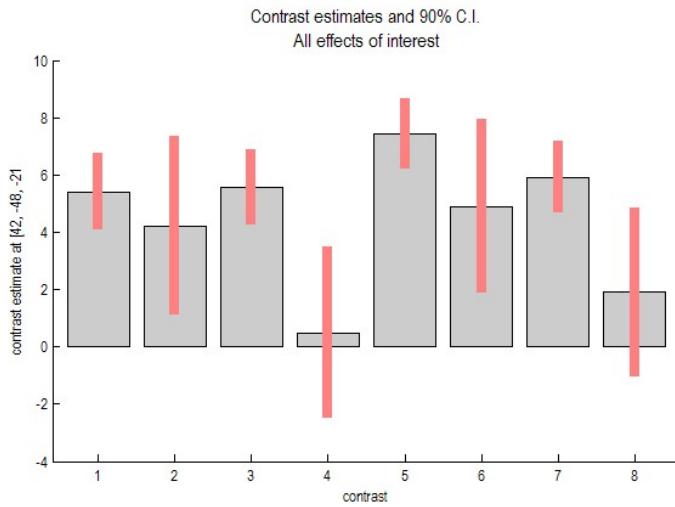


Figure 32.17: Estimated effect sizes and 90% confidence intervals for all effects of interest. Columns 1,3,5,7 are the canonical hrf's for N1 (non-famous faces, first presentation), N2 (non-famous faces, second presentation), F1 (famous faces, 1st presentation) and F2 (famous faces, 2nd presentation), whereas columns 2,4,6,8 are the temporal derivatives.

- Open ‘Conditions’ and then open the second ‘Condition’
- Highlight ‘Parametric Modulations’, select ‘New Parameter’ and open the newly created ‘Parameter’ option.
- Highlight ‘Name’ and enter ‘Lag’, highlight values and enter `itemlag{2}`, highlight polynomial expansion and ‘2nd order’.
- Now open the fourth ‘Condition’ under ‘Conditions’
- Highlight ‘Parametric Modulations’, select ‘New Parameter’ and open the newly created ‘Parameter’ option.
- Highlight ‘Name’ and enter ‘Lag’, highlight values and enter `itemlag{4}`, highlight polynomial expansion and ‘2nd order’.
- Open ‘Canonical HRF’ under ‘Basis Functions’, highlight ‘Model derivatives’ and select ‘No derivatives’.
- Highlight ‘Directory’ and select `DIR/parametric`
- Save the job as `parametric_spec` and press ‘Run’

This should produce the design matrix shown in Figure 32.19.

32.3.1 Estimate

Press the ‘Estimate’ button. This will call up the specification of an fMRI estimation job in the graphics window. Then

- Open the ‘fMRI model estimation’ option
- Highlight the ‘Select SPM.mat’ option and then choose the SPM.mat file saved in the `DIR/parametric` directory
- Save the job as `parametric_est.job` and press Run

SPM will write a number of files into the selected directory including an `SPM.mat` file.

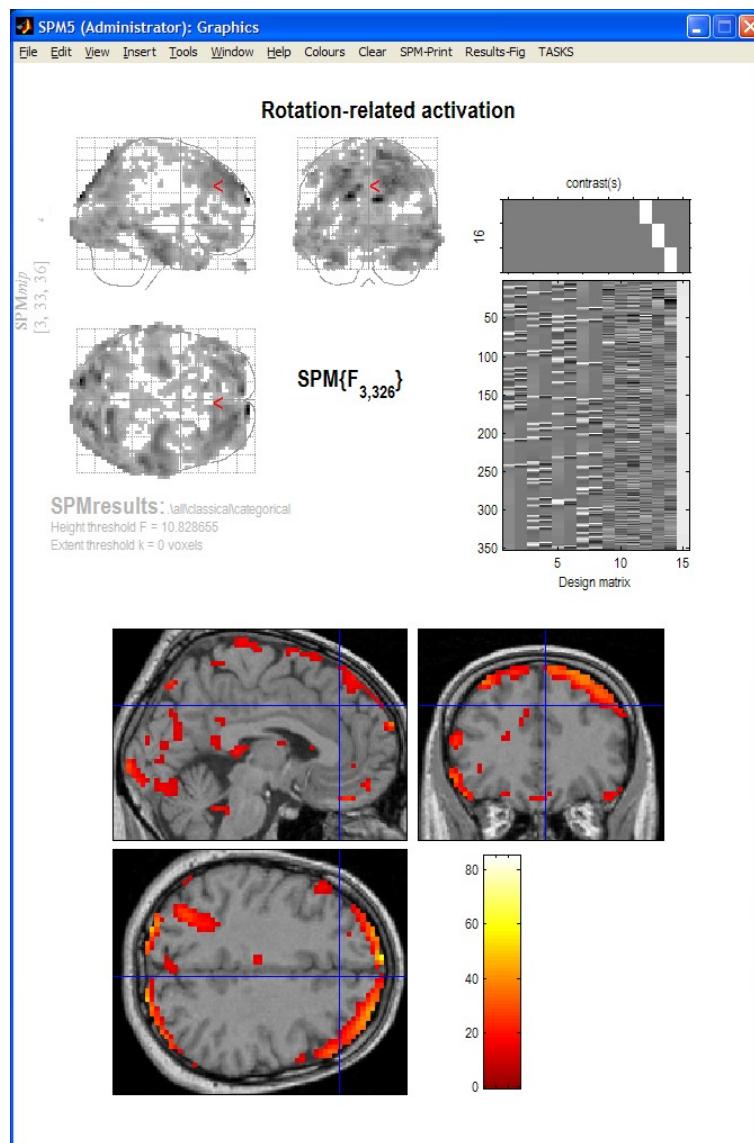


Figure 32.18: *Rotation-related activations*. These spurious ‘activations’ are due to rotation of the head during scanning. These effects occur at tissue boundaries and boundaries between brain and non-brain, as this is where contrast differences are greatest. Including these regressors in the design matrix means these effects cannot be falsely attributed to neuronal activity.

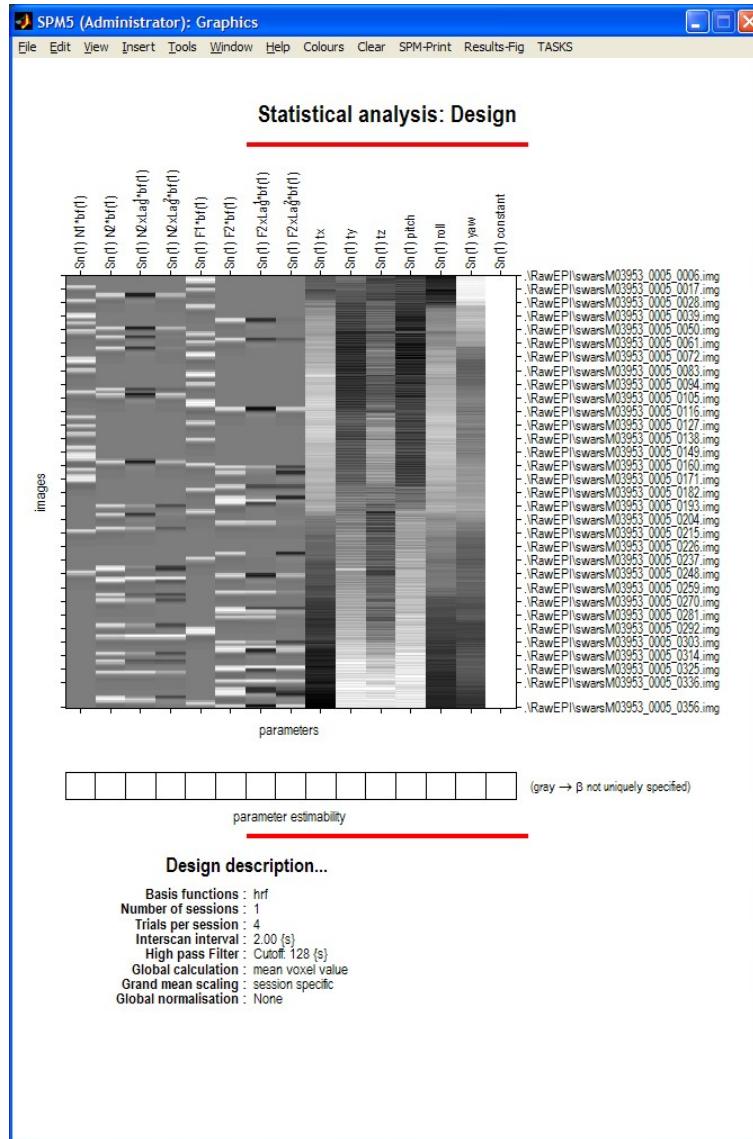


Figure 32.19: *Design matrix for testing repetition effects parametrically. Regressor 2 indicates the second occurrence of a nonfamous face. Regressor 3 modulates this linearly as a function of lag (ie. how many faces have been shown since that face was first presented), and regressor 4 modulates this quadratically as a function of lag. Regressors 6,7 and 8 play the same roles, but for famous faces.*

32.3.2 Plotting parametric responses

To plot parametric responses, we first set up a contrast that captures positive responses to F2.

- Press ‘Results’ and select the SPM.mat file in the DIR/parametric directory
- Press ‘Define new contrast’, enter the name ‘Pos F2’, press the ‘t-contrast’ radio button, enter the contrast [0 0 0 0 0 1], press ‘submit’, press ‘Done’.
- *Mask with other contrast ? [Yes/No]*
- Specify No.
- *Title for comparison ?*
- Accept what is offered
- *p value adjustment to control: [FWE/FDR/none]*
- Select FWE
- *Corrected p value(family-wise error)*
- Accept the default value, 0.05
- *Extent threshold {voxels} [0]*
- Accept the default value, 0

We then set up a contrast that captures some parametric modulation of F2.

- Press ‘Results’ and select the SPM.mat file in the DIR/parametric directory
- Press ‘Define new contrast’, enter the name ‘Famous Lag’, press the ‘F-contrast’ radio button, enter ‘1:6 9:15’ in the ‘columns in reduced design’ window, press ‘submit’, ‘OK’ and ‘Done’.
- *Mask with other contrast ? [Yes/No]*
- Specify No.
- *Title for comparison ?*
- Accept what is offered
- *p value adjustment to control: [FWE/FDR/none]*
- Select None
- *Threshold {F or p value}*
- Accept the default value, 0.001
- *Extent threshold {voxels} [0]*
- Accept the default value, 0

Now we will look at the modulation but masking it to restrict our attention to positive F2 effects. To do this

- Press ‘Results’ and select the SPM.mat file in the DIR/parametric directory
- Select the ‘Famous Lag’ contrast.
- *Mask with other contrast ? [Yes/No]*
- Specify Yes.

- Select the ‘Pos F2’ contrast
- Accept the 0.05 uncorrected mask p-value
- Nature of Mask ? inclusive
- *Title for comparison ?*
- Accept what is offered
- *p value adjustment to control: [FWE/FDR/none]*
- Select None
- *Threshold {F or p value}*
- Accept the default value, 0.001
- *Extent threshold {voxels} [0]*
- Accept the default value, 0

Figure 32.20 shows the MIP and an overlay of this parametric effect using overlays, sections and selecting the `wmsM03953_0007.img` image.

The effect is plotted in the time domain in figure 32.21. This was obtained by

- Right clicking on the MIP and selecting ‘global maxima’
- Pressing Plot, and selecting ‘parametric responses’ from the pull-down menu
- Which effect ? select ‘F2’

32.4 Bayesian analysis

32.4.1 Specification

Press the ‘Specify 1st-level’ button. This will call up an fMRI specification job in the graphics window. Then

- Open the fMRI model specification option
- Load the `categorical_spec.mat` job file created for the classical analysis
- Open ‘Subject/Session’, highlight ‘Scans’
- Deselect the smoothed functional images using the ‘unselect all’ option available from a right mouse click in the SPM file selector (bottom window)
- Select the unsmoothed functional images using the `~w.*` filter and ‘select all’ option available from a right mouse click in the SPM file selector (top right window)⁹. The Bayesian analysis uses a spatial prior where the spatial regularity in the signal is estimated from the data. It is therefore not necessary to create smoothed images if you are only going to do a Bayesian analysis.
- Press ‘Done’
- Highlight ‘Directory’ and select the `DIR/bayesian` directory you created earlier (you will first need to deselect the `DIR/categorical` directory).
- Save the job as `specify_bayesian.mat` and press ‘Run’

⁹Remember not to select the first 12 scans, scans 4 to 15, as these may contain T1 effects. This can be done during selection or by first moving the files to a different directory.

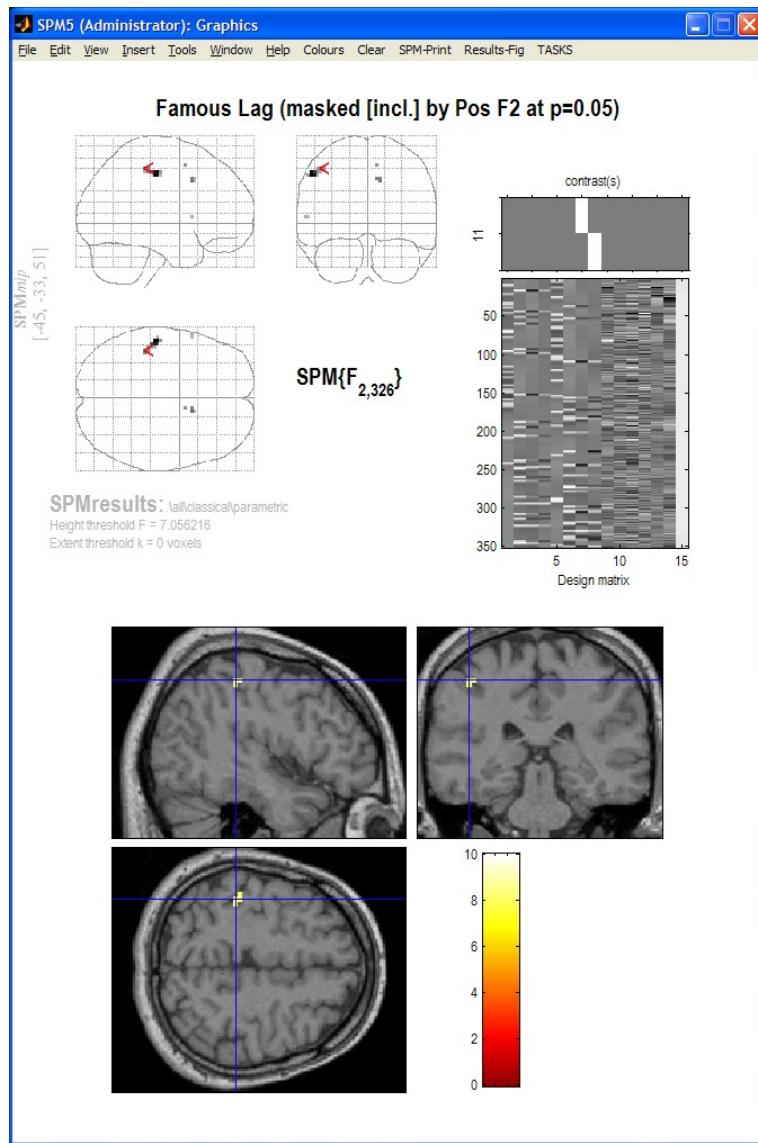


Figure 32.20: MIP and overlay of parametric lag effect in parietal cortex.

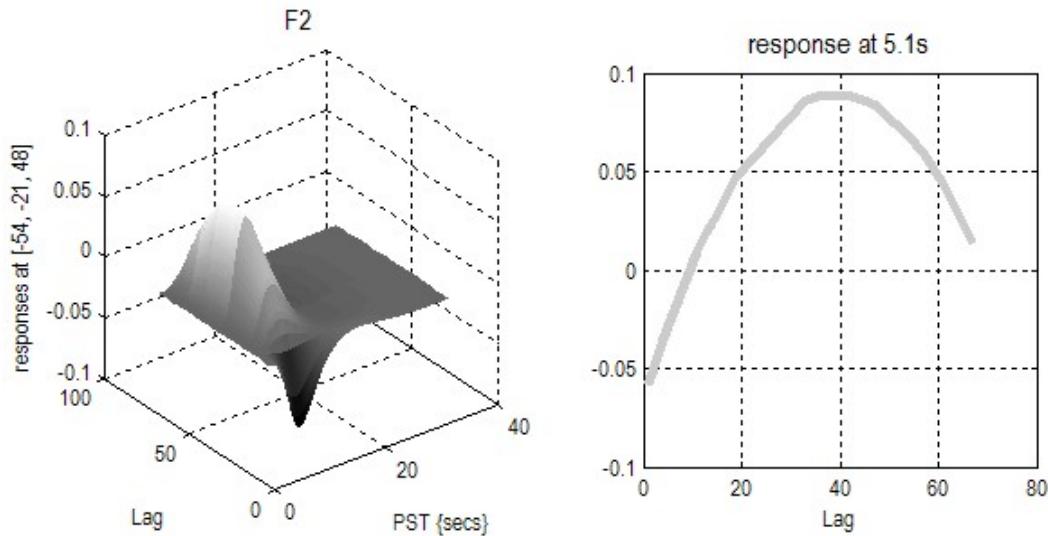


Figure 32.21: *Response as a function of lag.*

32.4.2 Estimation

Press the ‘Estimate’ button. This will call up the specification of an fMRI estimation job in the graphics window. Then

- Open the ‘fMRI model estimation’ option
- Highlight the ‘Select SPM.mat’ option and then choose the SPM.mat file saved in the DIR\bayesian subdirectory
- Highlight ‘Method’ and select the ‘Choose Bayesian 1st-level’ option.
- Save the job as `estimate_bayesian.job` and press Run

SPM will write a number of files into the output directory including

- An `SPM.mat` file.
- Images `Cbeta_000k.img` where k indexes the k th estimated regression coefficient. These filenames are prefixed with a ‘C’ indicating that these are the mean values of the ‘Conditional’ or ‘Posterior’ density.
- Images of error bars/standard deviations on the regression coefficients `SDbeta_000k.img`.
- An image of the standard deviation of the error `Sess1_SDerror.img`.
- An image `mask.img` indicating which voxels were included in the analysis.
- Images `Sess1_AR_000p.img` where p indexes the p th AR coefficient. See eg. Figure 32.22.
- Images `con_000i.img` and `con_sd_000i.img` which are the mean and standard deviation of the i th pre-defined contrast.

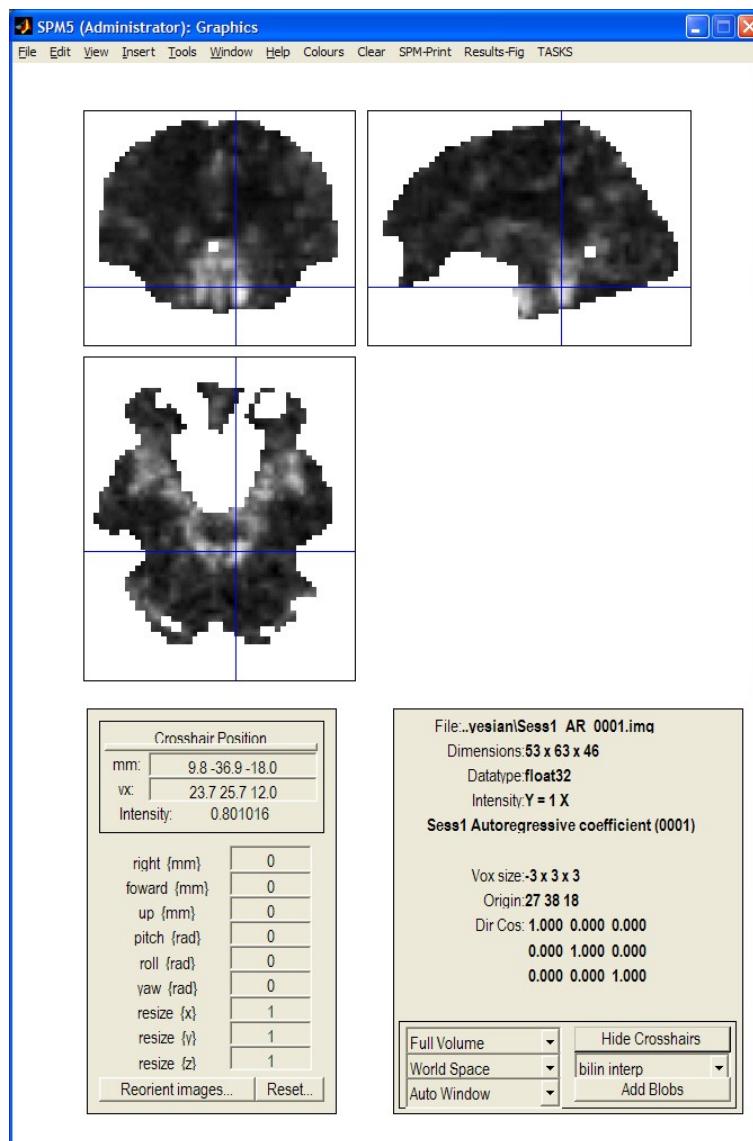


Figure 32.22: Bayesian analysis: Estimated AR(1) coefficient image indicating heterogeneity near the circle of Willis

32.4.3 Inference

After estimation:

- Press ‘Results’
- Select the `SPM.mat` file created in the last section
- Select contrast number 1: `Positive effect of condition_1`, type ‘Done’.
- *Mask with other contrast ? [Yes/No]*
- Specify No
- *Title for comparison*
- Enter ‘Canonical HRF: Faces > Baseline’
- *Effect size threshold for PPM*
- Enter the value 2
- *Posterior probability threshold for PPM*
- Enter the value 0.99
- *Extent threshold [0]*
- Accept the default value
- *Plot effect size [Yes/No]*
- Select the default ‘Yes’

SPM will then plot a map of effect sizes at voxels where it is 99% sure that the effect size is greater than 2% of the global mean. This is a large activation. Then use overlays, sections, select the normalised structural image created earlier and move the cursor to the activation in the left hemisphere. This should create the plot shown in Figure 32.23

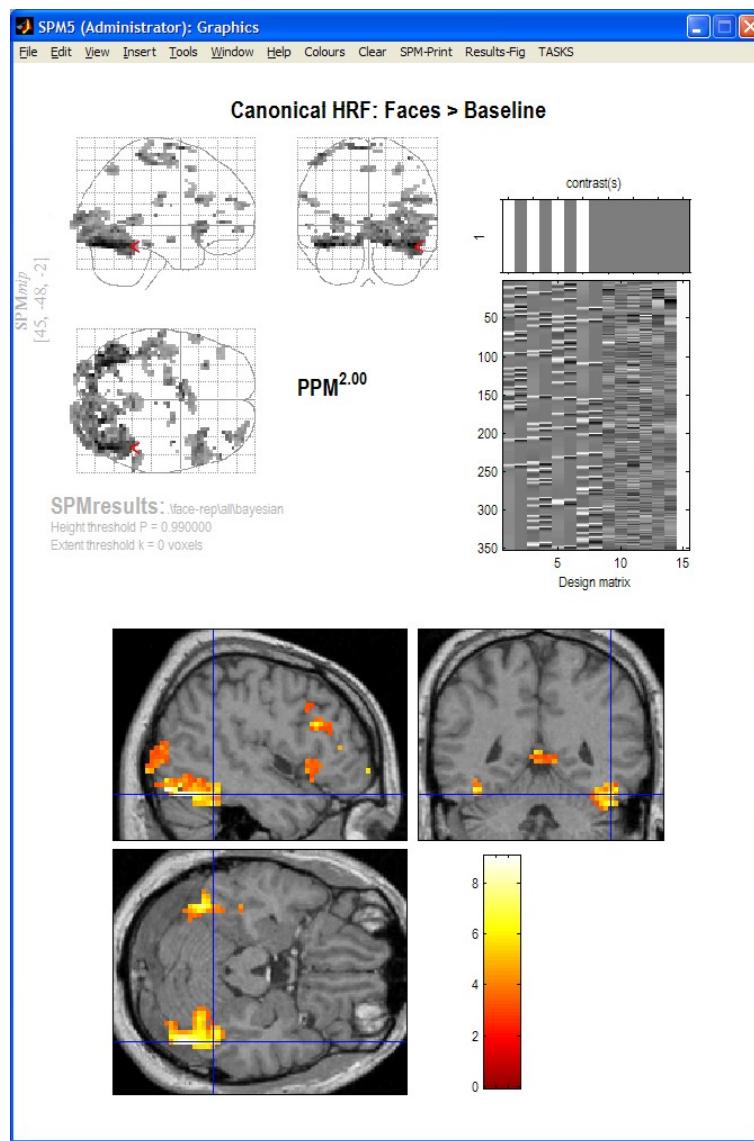


Figure 32.23: **Bayesian analysis:** MIP and overlay of effect sizes at voxels where SPM is 99% sure that the effect size is greater than 2% of the global mean. The cursor is at the location $x = 45, y = -48, z = -21$ mm

Bibliography

- [1] J. Andersson, J. Ashburner, and K.J. Friston. A global estimator unbiased by local changes. *NeuroImage*, 13(6):1193–1206, 2001.
- [2] J. Andersson, C. Hutton, J. Ashburner, R. Turner, and K.J. Friston. Modelling geometric deformations in epi time series. *NeuroImage*, 13(5):903–919, 2001.
- [3] J. Ashburner and K.J. Friston. The role of registration and spatial normalization in detecting activations in functional imaging. *Clinical MRI/Developments in MR*, 7(1):26–28, 1997.
- [4] J. Ashburner and K.J. Friston. Nonlinear spatial normalization using basis functions. *Human Brain Mapping*, 7(4):254–266, 1999.
- [5] J. Ashburner and K.J. Friston. Voxel-based morphometry – the methods. *NeuroImage*, 11:805–821, 2000.
- [6] J. Ashburner and K.J. Friston. Why voxel-based morphometry should be used. *NeuroImage*, 14(6):1238–1243, 2001.
- [7] J. Ashburner and K.J. Friston. Unified segmentation. *NeuroImage*, 26:839–851, 2005.
- [8] J. Ashburner, P. Neelin, D. L. Collins, A. C. Evans, and K.J. Friston. Incorporating prior knowledge into image registration. *NeuroImage*, 6:344–352, 1997.
- [9] S. Baillet, J.C. Mosher, and R.M. Leahy. Electromagnetic brain mapping. *IEEE Sign. Proc. Mag.*, 18:14–30, 2001.
- [10] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pat. Anal. and Mach. Intel.*, 14:239–256, 1992.
- [11] C. Buchel, A.P. Holmes, G. Rees, and K.J. Friston. Characterizing stimulus-response functions using nonlinear regressors in parametric fmri experiments. *NeuroImage*, 8:140–148, 1998.
- [12] C. Buchel, R.J.S. Wise, C.J. Mummary, J.B. Poline, and K.J. Friston. Nonlinear regression in parametric activation studies. *NeuroImage*, 4:60–66, 1996.
- [13] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal. Automated multi-modality image registration based on information theory. In Y. Bizais, C. Barillot, and R. Di Paola, editors, *Proc. Information Processing in Medical Imaging*, pages 263–274, Dordrecht, The Netherlands, 1995. Kluwer Academic Publishers.
- [14] R. W. Cox and A. Jesmanowicz. Real-time 3D image registration for functional MRI. *Magnetic Resonance In Medicine*, 42:1014–1018, 1999.
- [15] A.M. Dale and M. Sereno. Improved localization of cortical activity by combining EEG and MEG with MRI surface reconstruction: a linear approach. *J. Cognit. Neurosci.*, 5:162–176, 1993.
- [16] O. David, S.J. Kiebel, L. Harrison, J. Mattout, J. Kilner, and K.J. Friston. Dynamic causal modelling of evoked responses in eeg and meg. In press, NeuroImage, 2005.

- [17] W. F. Eddy, M. Fitzgerald, and D. C. Noll. Improved image registration by using Fourier interpolation. *Magnetic Resonance in Medicine*, 36:923–931, 1996.
- [18] J.J. Ermer, J.C. Mosher, S. Baillet, and R.M. Leahy. Rapidly recomputable EEG forward models for realistic head shapes. *Phys. Med. Biol.*, 46(4):1265–1281, 2001.
- [19] K.J. Friston, C. Frith, R.S.J. Frackowiak, and R. Turner. Characterizing dynamic brain responses with fmri. *NeuroImage*, 2:166–172, 1995.
- [20] K.J. Friston, D.E. Glaser, R.N.A. Henson, S.J. Kiebel, C. Phillips, and J. Ashburner. Classical and bayesian inference in neuroimaging: Applications. *NeuroImage*, 16:484–512, 2002.
- [21] K.J. Friston, R.N.A. Henson, C. Phillips, and J. Mattout. Bayesian estimation of evoked and induced responses. Submitted, 2005.
- [22] K.J. Friston, A. Mechelli, R. Turner, and C.J. Price. Nonlinear responses in fmri: The balloon model, volterra kernels and other hemodynamics. *NeuroImage*, 12:466–477, 2000.
- [23] K.J. Friston and W.D. Penny. Posterior probability maps and spms. *NeuroImage*, 19(3):1240–1249, 2003.
- [24] K.J. Friston, W.D. Penny, C. Phillips, S.J. Kiebel, G. Hinton, and J. Ashburner. Classical and bayesian inference in neuroimaging: Theory. *NeuroImage*, 16:465–483, 2002.
- [25] D.E. Glaser. Variance components. In R.S.J. Frackowiak, K.J. Friston, C. Frith, R. Dolan, K.J. Friston, C.J. Price, S. Zeki, J. Ashburner, and W.D. Penny, editors, *Human Brain Function*. Academic Press, 2nd edition, 2003.
- [26] R.N.A. Henson. Analysis of fmri time series. In R.S.J. Frackowiak, K.J. Friston, C. Frith, R. Dolan, K.J. Friston, C.J. Price, S. Zeki, J. Ashburner, and W.D. Penny, editors, *Human Brain Function*. Academic Press, 2nd edition, 2003.
- [27] R.N.A. Henson and W.D. Penny. Anovas and spm. Technical report, Wellcome Department of Imaging Neuroscience, 2003.
- [28] R.N.A. Henson, M.D. Rugg, and K.J. Friston. The choice of basis functions in event-related fmri. *NeuroImage*, 13(6):127, June 2001. Supplement 1.
- [29] M.X. Huang, J.C. Mosher, and R.M. Leahy. A sensor-weighted overlapping-sphere head model and exhaustive head model comparison for MEG. *Physics in Medicine and Biology*, 44(2):423–440, 1999.
- [30] S.J. Kiebel, O. David, and K.J. Friston. Dynamic causal modelling of evoked responses in eeg/meg with lead-field parameterization. Under revision, 2005.
- [31] S.J. Kiebel and K.J. Friston. Statistical parametric mapping for event-related potentials i: Generic considerations. *NeuroImage*, 22(2):492–502, 2004.
- [32] S.J. Kiebel and K.J. Friston. Statistical parametric mapping for event-related potentials ii: A hierarchical temporal model. *NeuroImage*, 22(2):503–520, 2004.
- [33] J. Kilner, S.J. Kiebel, and K.J. Friston. Applications of random field theory to electrophysiology. *Neuroscience Letters*, 374:174–178, 2005.
- [34] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Seutens. Multimodality image registration by maximisation of mutual information. *IEEE Transactions on Medical Imaging*, 16:187–197, 1997.
- [35] J. Mattout, M. Péligrini-Issac, L. Garnero, and H. Benali. Multivariate Source Prelocalization (MSP): use of functionally informed basis functions for better conditioning the MEG inverse problem. *NeuroImage*, 26(2):356–373, 2005.

- [36] A. Mechelli, C.J. Price, K.J. Friston, and J. Ashburner. Voxel-based morphometry of the human brain: Methods and applications. *Current Medical Imaging Reviews*, pages 105–113, 2005.
- [37] W.D. Penny and G. Flandin. Bayesian analysis of single-subject fmri: Spm implementation. Technical report, Wellcome Department of Imaging Neuroscience, 2005.
- [38] W.D. Penny, S.J. Kiebel, and K.J. Friston. Variational bayesian inference for fmri time series. *NeuroImage*, 19(3):727–741, 2003.
- [39] W.D. Penny, K.E. Stephan, A. Mechelli, and K.J. Friston. Comparing dynamic causal models. *NeuroImage*, 22(3):1157–1172, 2004.
- [40] W.D. Penny, N. Trujillo-Barreto, and K.J. Friston. Bayesian fmri time series analysis with spatial priors. *NeuroImage*, 24(2):350–362, 2005.
- [41] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C (Second Edition)*. Cambridge, Cambridge, 1992.
- [42] L. Spinelli, S.G. Andino, G. Lantz, M. Seeck, and C.M. Michel. Electromagnetic inverse solutions in anatomically constrained spherical head models. *Brain Topography*, 13(2):115–125, 2000.
- [43] C. Studholme, D. L. G. Hill, and D. J. Hawkes. An overlap invariant entropy measure of 3D medical image alignment. *Pattern Recognition*, 32:71–86, 1999.
- [44] P. Thévenaz, T. Blu, and M. Unser. Interpolation revisited. *IEEE Transactions on Medical Imaging*, 19(7):739–758, 2000.
- [45] M. Unser, A. Aldroubi, and M. Eden. B-spline signal processing: Part I – theory. *IEEE Transactions on Signal Processing*, 41(2):821–833, 1993.
- [46] M. Unser, A. Aldroubi, and M. Eden. B-spline signal processing: Part II – efficient design and applications. *IEEE Transactions on Signal Processing*, 41(2):834–848, 1993.
- [47] W. M. Wells III, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multi-modal volume registration by maximisation of mutual information. *Medical Image Analysis*, 1(1):35–51, 1996.
- [48] K.J. Worsley. Developments in random field theory. In R.S.J. Frackowiak, K.J. Friston, C. Frith, R. Dolan, K.J. Friston, C.J. Price, S. Zeki, J. Ashburner, and W.D. Penny, editors, *Human Brain Function*. Academic Press, 2nd edition, 2003.
- [49] I.C. Wright, P.K. McGuire, J.B. Poline, J.M. Travere, R.M. Murray, C. Frith, R.S.J. Frackowiak, and K.J. Friston. A voxel-based method for the statistical analysis of gray and white matter density applied to schizophrenia. *NeuroImage*, 2:244–252, 1995.