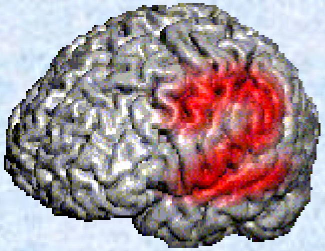
Spatial Preprocessing

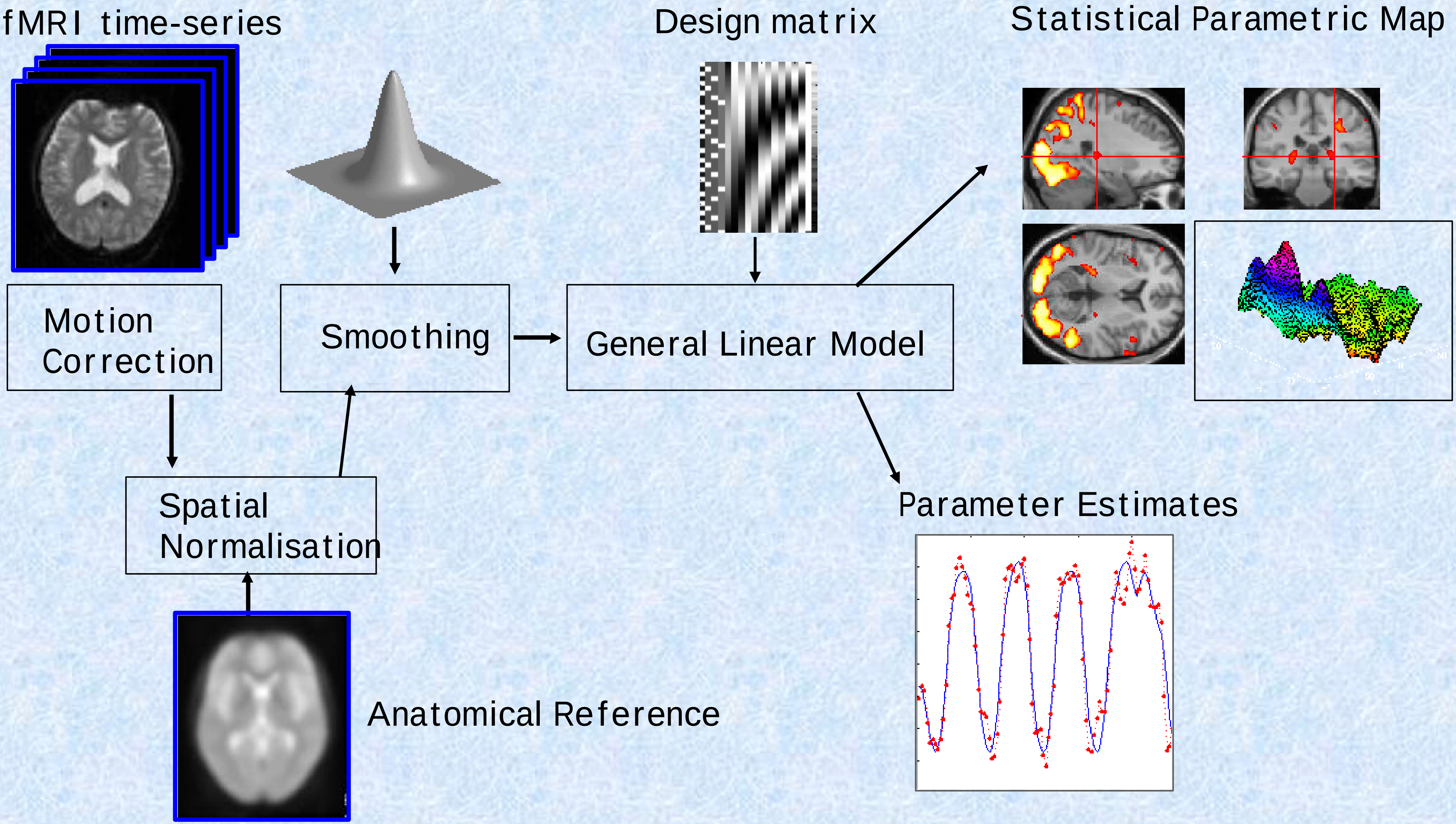
John Ashburner
john@fil.ion.ucl.ac.uk

- Smoothing
- Rigid registration
- Spatial normalisation

With slides by Chloe Hutton and Jesper Andersson



Overview of SPM Analysis



fMRI time-series

Motion Correction

Smoothing

Spatial Normalisation

Anatomical Reference

Design matrix

General Linear Model

Statistical Parametric Map

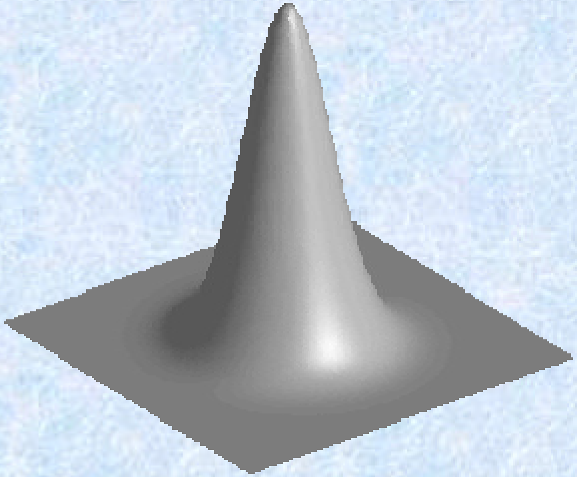
Parameter Estimates

Contents

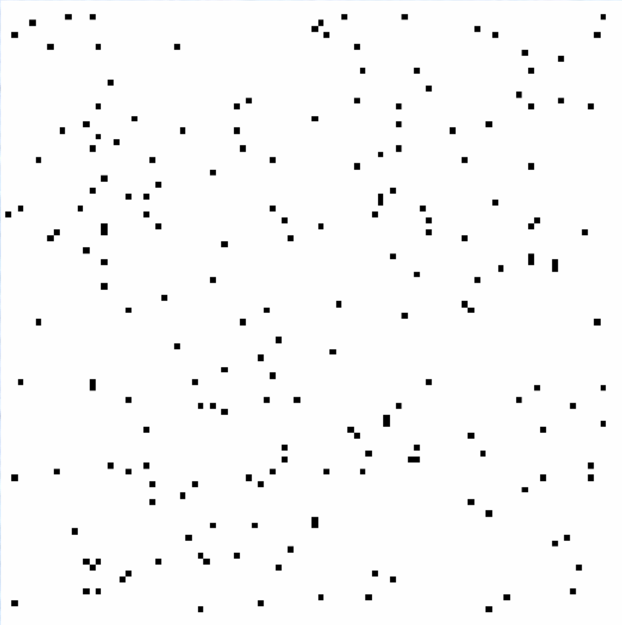
- Smoothing
- Rigid registration
- Spatial normalisation

Smoothing

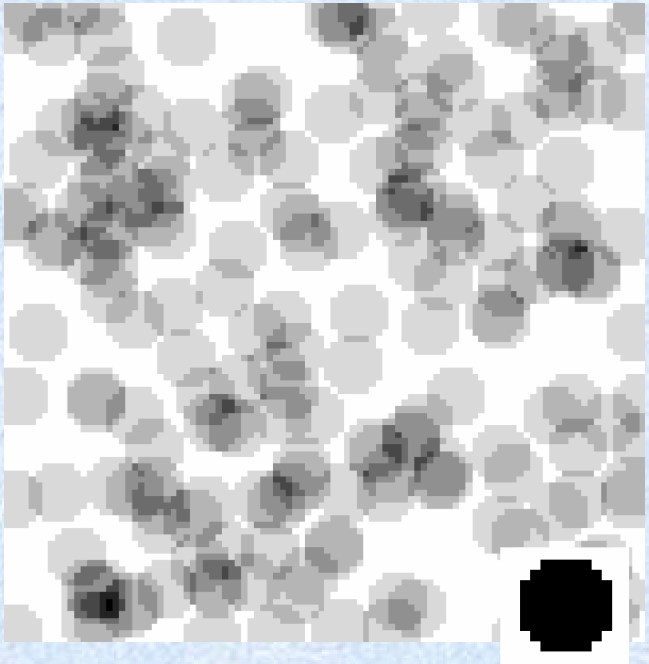
Each voxel after smoothing effectively becomes the result of applying a weighted region of interest (ROI).



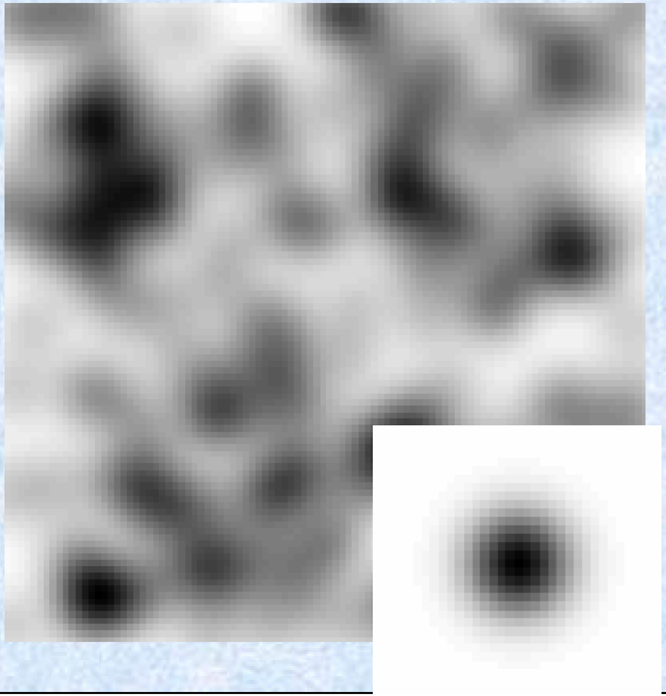
Before convolution



Convolved with a circle

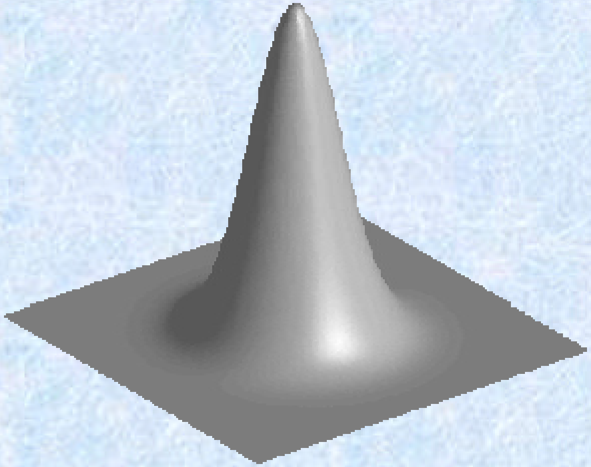


Convolved with a Gaussian

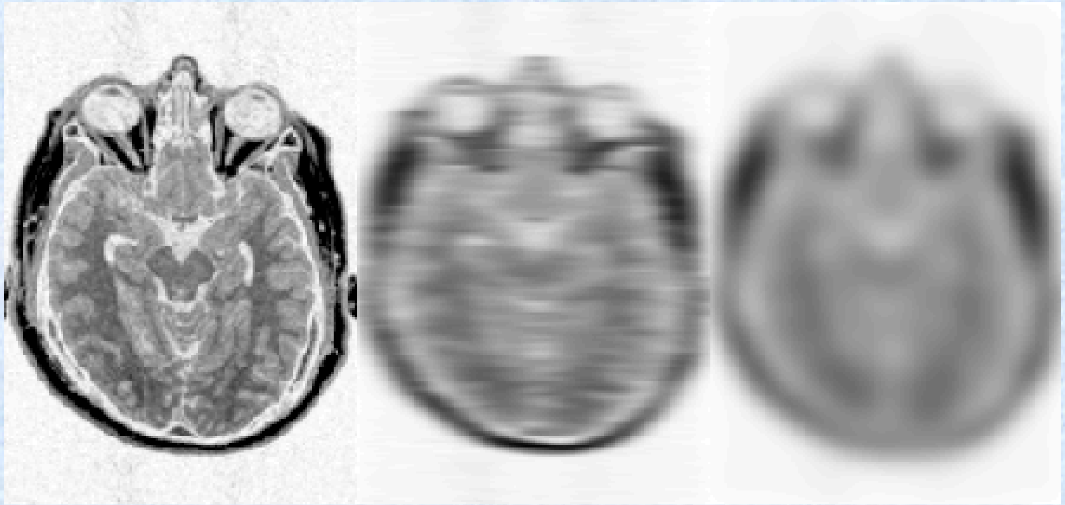


Smoothing

- Why smooth?
  - Potentially increase sensitivity
  - Inter-subject averaging
  - Increase validity of SPM
- Smoothing is a convolution with a Gaussian kernel



Gaussian convolution is separable



Contents

- Smoothing
- Rigid registration
  - Rigid-body transforms
  - Optimisation & objective functions
  - Interpolation
- Spatial normalisation

Within-subject Registration

- Assumes there is no shape change, and motion is rigid-body
- Used by [realign] and [coregister] functions
- The steps are:
  - **Registration** - i.e. Optimising the parameters that describe a rigid body transformation between the source and reference images
  - **Transformation** - i.e. Re-sampling according to the determined transformation

Affine Transforms

- Rigid-body transformations are a subset
- Parallel lines remain parallel
- Operations can be represented by:

$$x_1 = m_{11}x_0 + m_{12}y_0 + m_{13}z_0 + m_{14}$$

$$y_1 = m_{21}x_0 + m_{22}y_0 + m_{23}z_0 + m_{24}$$

$$z_1 = m_{31}x_0 + m_{32}y_0 + m_{33}z_0 + m_{34}$$

- Or as matrices:

$$\mathbf{Y}=\mathbf{Mx}$$

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

2D Affine Transforms

- Translations by t_x and t_y
  - $x_1 = x_0 + t_x$
  - $y_1 = y_0 + t_y$
- Rotation around the origin by Θ radians
  - $x_1 = \cos(\Theta)\ x_0 + \sin(\Theta)\ y_0$
  - $y_1 = -\sin(\Theta)\ x_0 + \cos(\Theta)\ y_0$
- Zooms by s_x and s_y
  - $x_1 = s_x\ x_0$
  - $y_1 = s_y\ y_0$
- Shear
  - $x_1 = x_0 + h\ y_0$
  - $y_1 = y_0$

2D Affine Transforms

- Translations by t_x and t_y
  - $x_1 = 1\ x_0 + 0\ y_0 + t_x$
  - $y_1 = 0\ x_0 + 1\ y_0 + t_y$
- Rotation around the origin by Θ radians
  - $x_1 = \cos(\Theta)\ x_0 + \sin(\Theta)\ y_0 + 0$
  - $y_1 = -\sin(\Theta)\ x_0 + \cos(\Theta)\ y_0 + 0$
- Zooms by s_x and s_y:
  - $x_1 = s_x\ x_0 + 0\ y_0 + 0$
  - $y_1 = 0\ x_0 + s_y\ y_0 + 0$
- Shear
  - $x_1 = 1\ x_0 + h\ y_0 + 0$
  - $y_1 = 0\ x_0 + 1\ y_0 + 0$

3D Rigid-body Transformations

- ⌘A 3D rigid body transform is defined by:
  - 3 translations - in X, Y & Z directions
  - 3 rotations - about X, Y & Z axes
- ⌘The order of the operations matters

$$\begin{pmatrix} 1 & 0 & 0 & X\,\mathrm{trans} \\ 0 & 1 & 0 & Y\,\mathrm{trans} \\ 0 & 0 & 1 & Z\,\mathrm{trans} \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos F & \sin F & 0 \\ 0 & -\sin F & \cos F & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \cos T & 0 & \sin T & 0 \\ 0 & 1 & 0 & 0 \\ -\sin T & 0 & \cos T & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \cos O & \sin O & 0 & 0 \\ -\sin O & \cos O & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Translations

Pitch
about x axis

Roll
about y axis

Yaw
about z axis

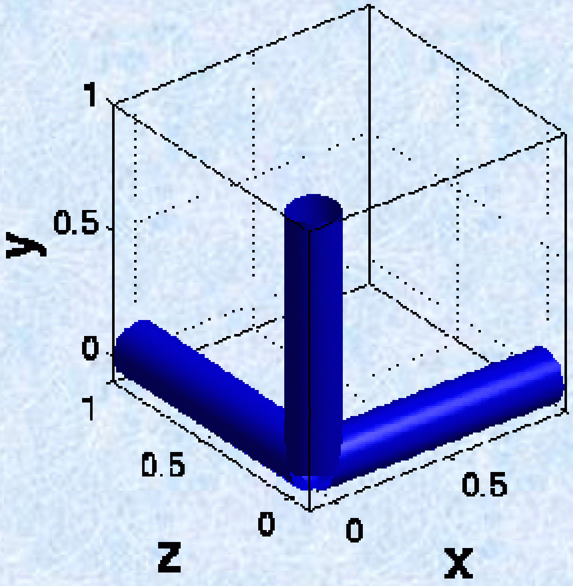
Voxel-to-world Transforms

- Affine transform associated with each image
  - Maps from voxels ($x=1..n_x$, $y=1..n_y$, $z=1..n_z$) to some world co-ordinate system. e.g.,
    - Scanner co-ordinates - images from DICOM toolbox
    - T&T/MNI coordinates - spatially normalised
- Registering image B (source) to image A (target) will update B’s vox-to-world mapping
  - Mapping from voxels in A to voxels in B is by
    - A-to-world using M_A, then world-to-B using M_B^{-1}
    - $M_B^{-1} M_A$

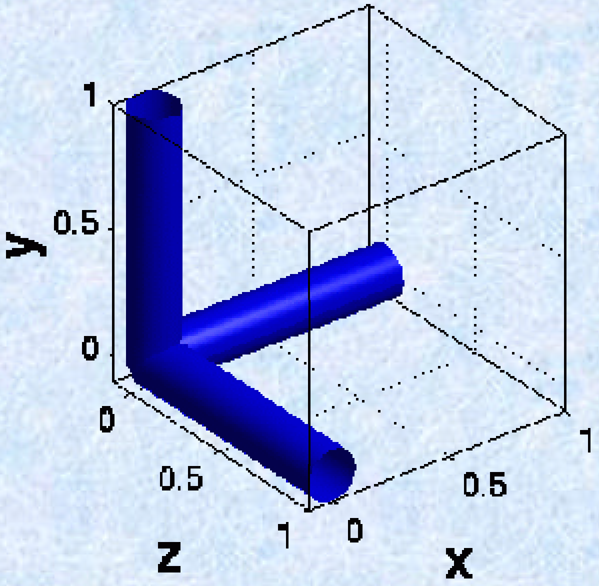
Left- and Right-handed Coordinate Systems

- Analyze™ files are stored in a left-handed system
- Talairach & Tournoux uses a right-handed system
- Mapping between them requires a flip
  - Affine transform with a negative determinant

Left–Handed

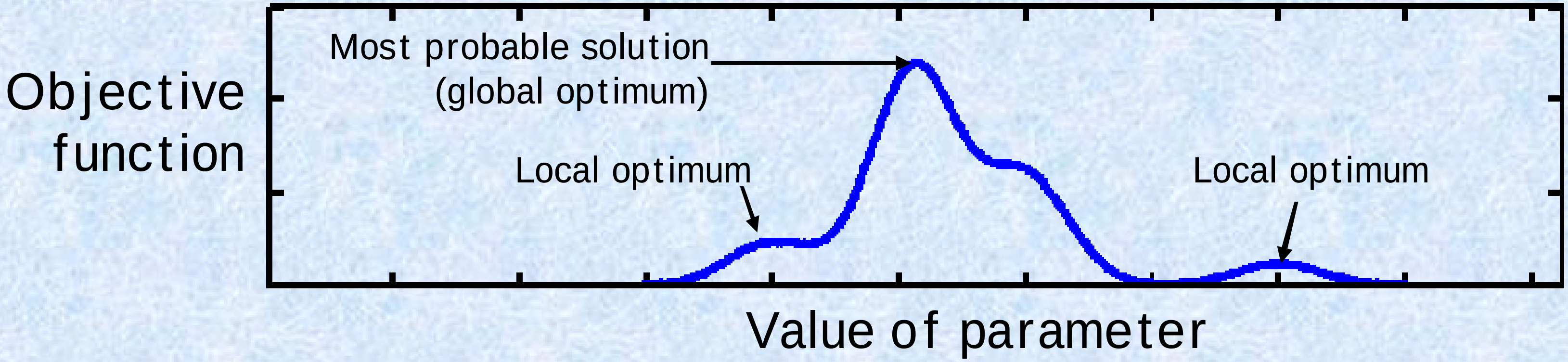


Right–Handed



Optimisation

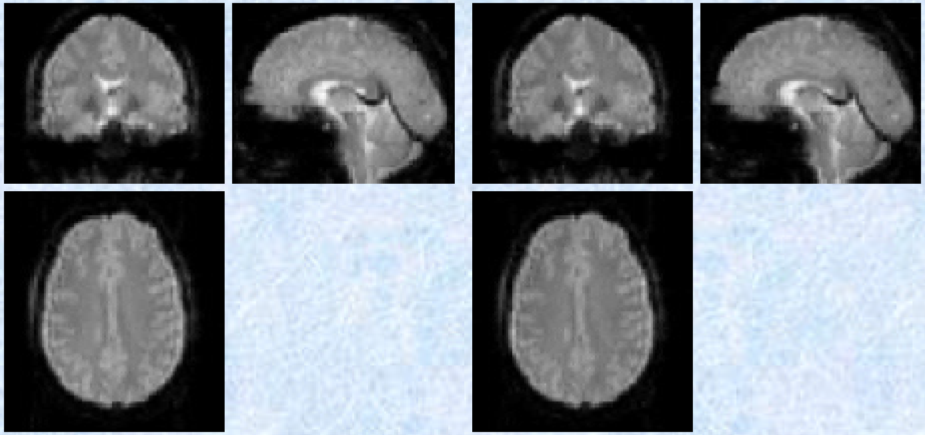
- ⌘Optimisation involves finding some "best" parameters according to an "objective function", which is either minimised or maximised
- ⌘The "objective function" is often related to a probability based on some model

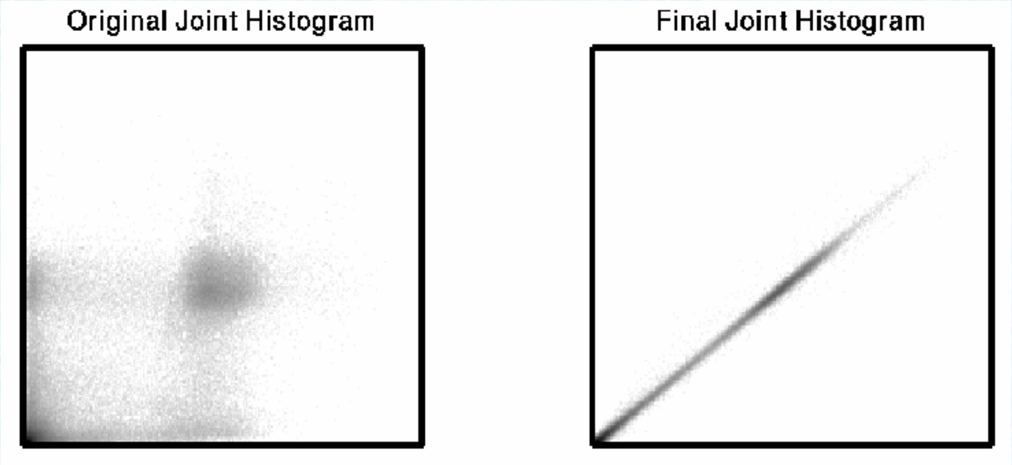


Objective Functions for Image Registration

- Intra-modal
  - Mean squared difference (minimise)
  - Normalised cross correlation (maximise)
  - Entropy of difference (minimise)
- Inter-modal (or intra-modal)
  - Mutual information (maximise)
  - Normalised mutual information (maximise)
  - Entropy correlation coefficient (maximise)
  - AIR cost function (minimise)

Mean-squared Difference





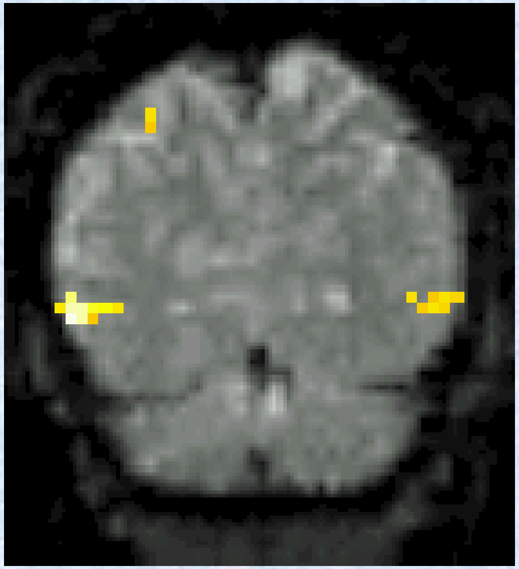
- Minimising mean-squared difference works for intra-modal registration (realignment)
- Simple relationship between intensities in one image, versus those in the other
  - Assumes normally distributed differences

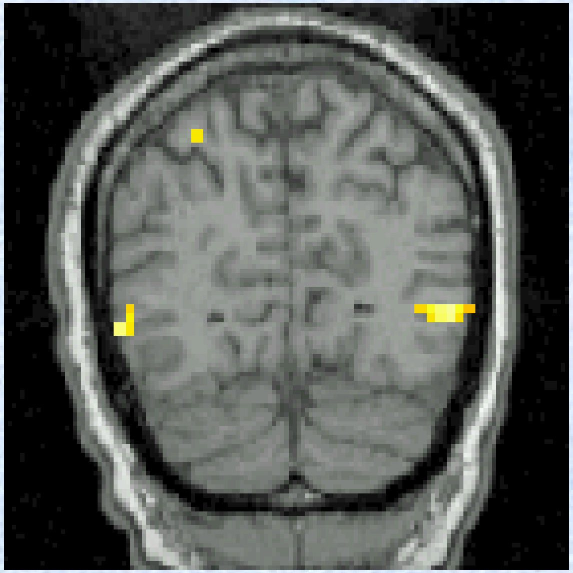
Gauss-newton Optimisation

- Works best for least-squares
- Minimum is estimated by fitting a quadratic at each iteration

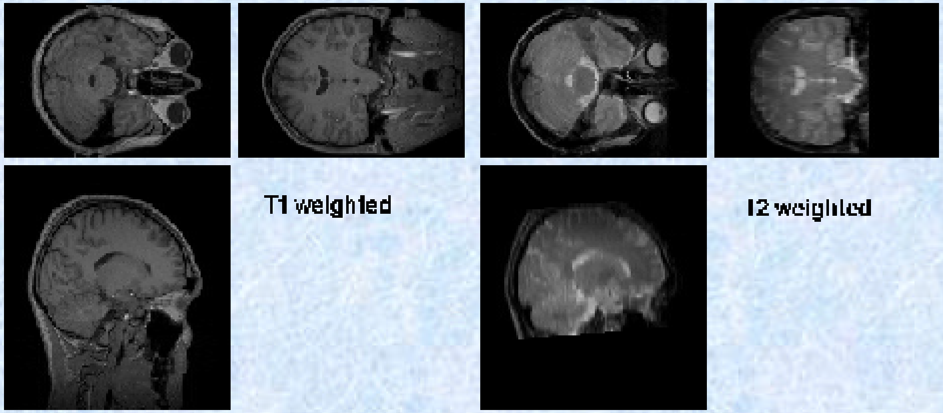
Inter-modal registration

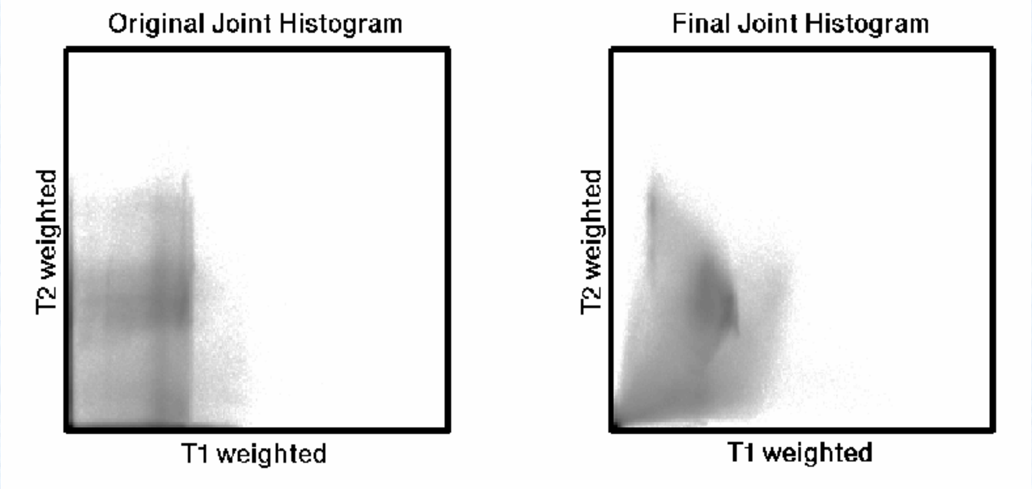
- Match images from same subject but different modalities:
  - anatomical localisation of single subject activations
  - achieve more precise spatial normalisation of functional image using anatomical image.





Mutual Information





- Used for between-modality registration
- Derived from joint histograms
- MI= $\int_{ab}$ P(a,b) $\log_2$ [P(a,b)/(P(a) P(b))]
  - Related to entropy: MI = -H(a,b) + H(a) + H(b)
    - Where H(a) = -$\int_a$ P(a) $\log_2$P(a) and H(a,b) = -$\int_a$ P(a,b) $\log_2$P(a,b)

Image Transformations

- Images are re-sampled. An example in 2D:

```
for $y_0$=1..$n_{y0}$  % loop over rows
    for $x_0$=1..$n_{x0}$ % loop over pixels in row
        $x_1 = t_x(x_0,y_0,\mathbf{q})$ % transform according to $\mathbf{q}$
        $y_1 = t_y(x_0,y_0,\mathbf{q})$
        if 1£$x_1$£ $n_{x1}$ & 1£$y_1$£$n_{y1}$ then  % voxel in range
            $f_1(x_0,y_0) = f_0(x_1,y_1)$ % assign re-sampled value
        end % voxel in range
    end % loop over pixels in row
end % loop over rows
```

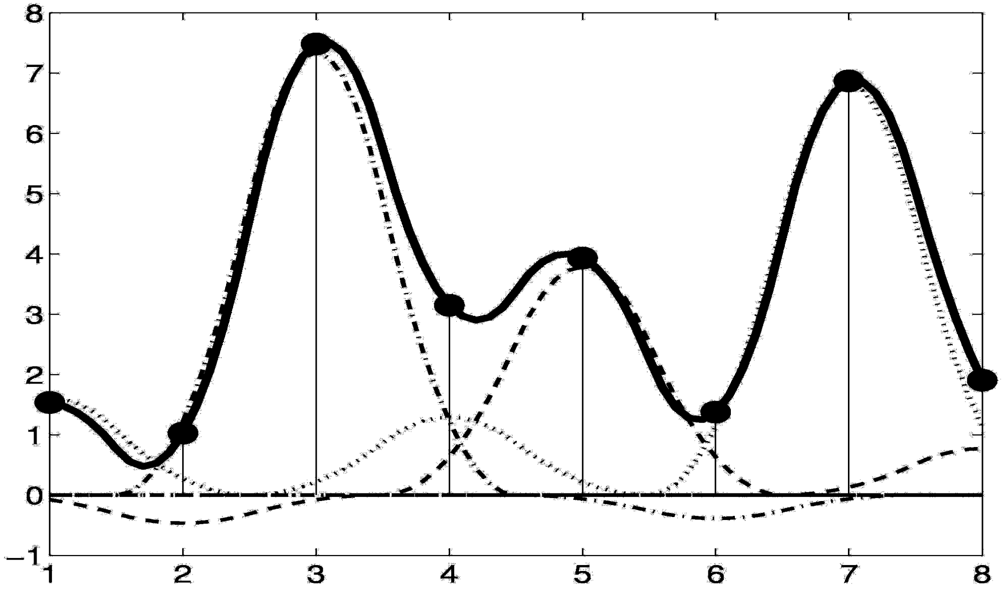
- What happens if x_1 and y_1 are not integers?

Simple Interpolation

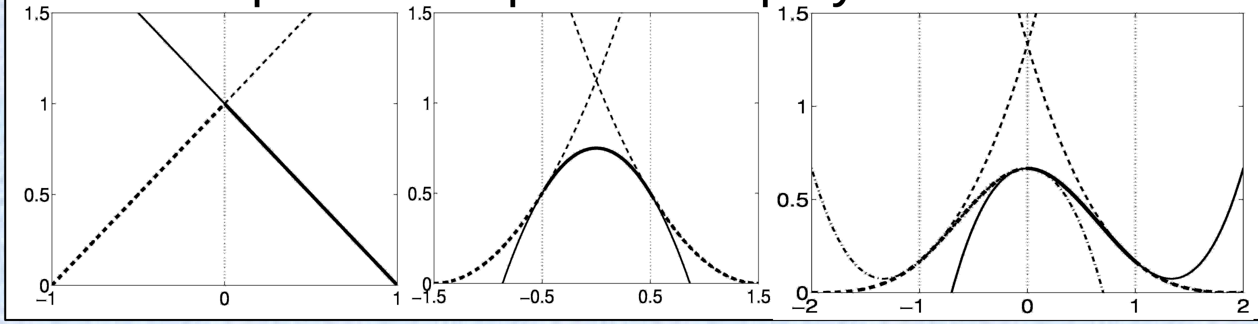
- Nearest neighbour
  - Take the value of the closest voxel
- Tri-linear
  - Just a weighted average of the neighbouring voxels
  - $f_5 = f_1 x_2 + f_2 x_1$
  - $f_6 = f_3 x_2 + f_4 x_1$
  - $f_7 = f_5 y_2 + f_6 y_1$

B-spline Interpolation

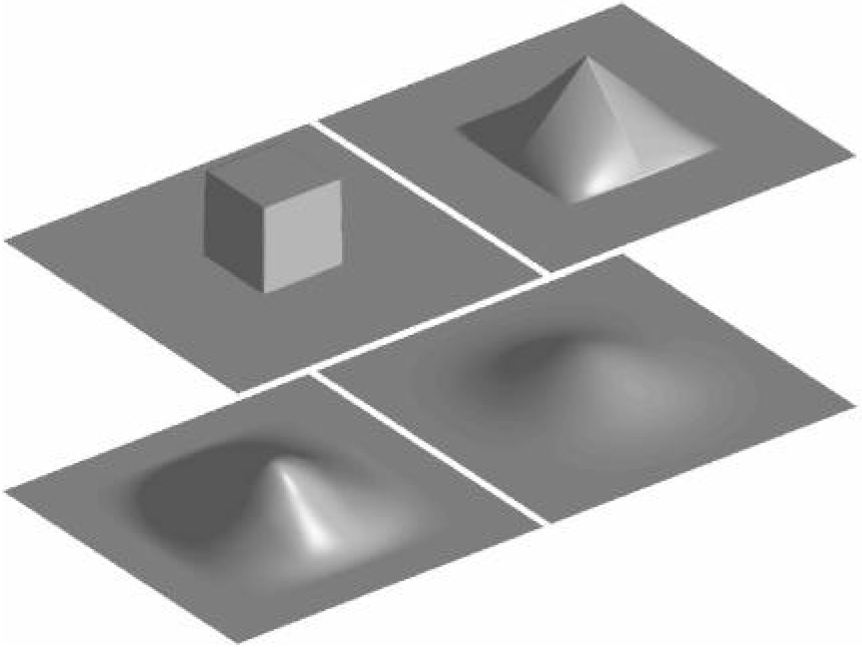
A continuous function is represented by a linear combination of basis functions



B-splines are piecewise polynomials



2D B-spline basis functions of degrees 0, 1, 2 and 3



Nearest neighbour and trilinear interpolation are the same as B-spline interpolation with degrees 0 and 1.

Residual Errors from aligned fMRI

- Re-sampling can introduce interpolation errors
  - especially tri-linear interpolation
- Gaps between slices can cause aliasing artefacts
- Slices are not acquired simultaneously
  - rapid movements not accounted for by rigid body model
- Image artefacts may not move according to a rigid body model
  - image distortion
  - image dropout
  - Nyquist ghost
- Functions of the estimated motion parameters can be modelled as confounds in subsequent analyses

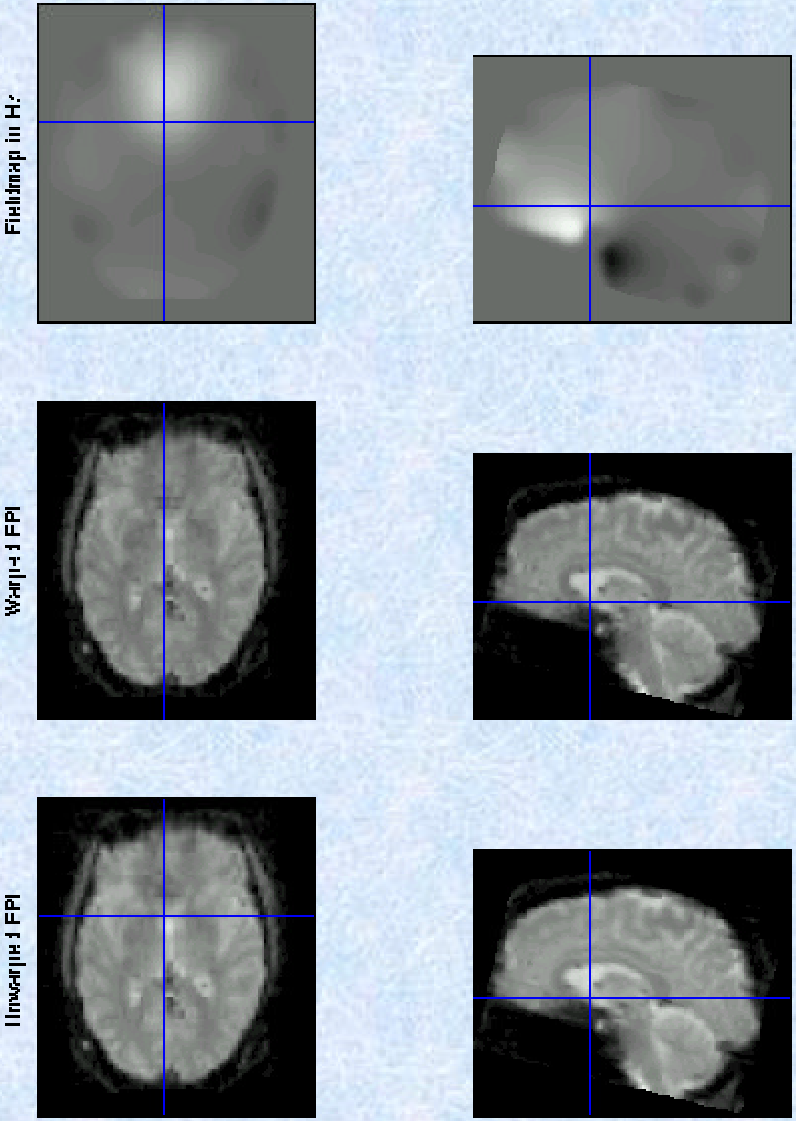
Movement by Distortion Interaction of fMRI

- Subject disrupts B_0 field, rendering it inhomogeneous

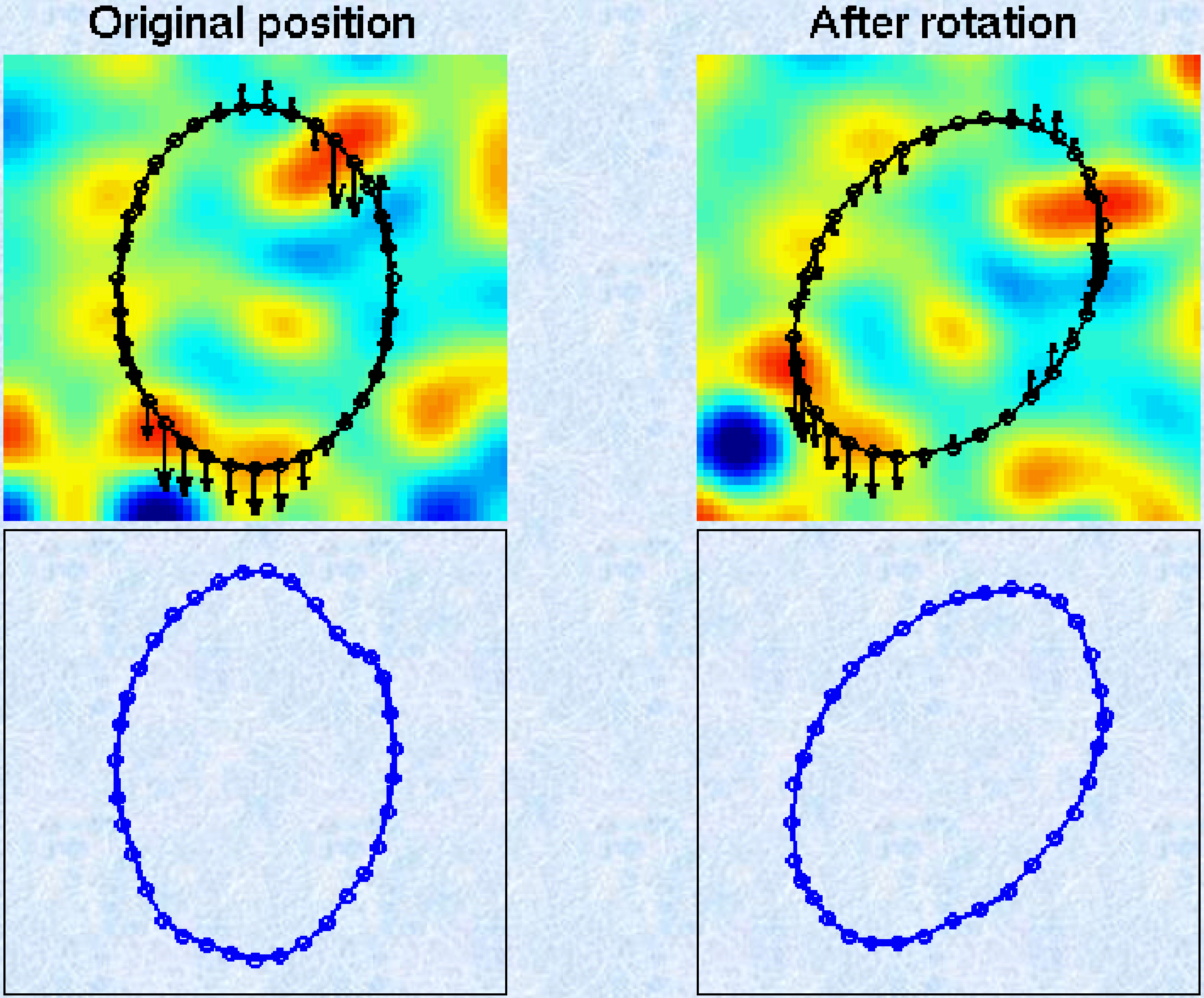
  => distortions in phase-encode direction

- Subject moves during EPI time series
- Distortions vary with subject orientation

  => shape varies



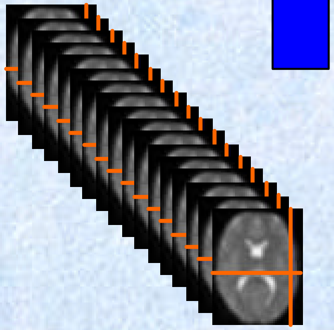
Movement by distortion interaction



Original position

After rotation

Correcting for distortion changes using Unwarp

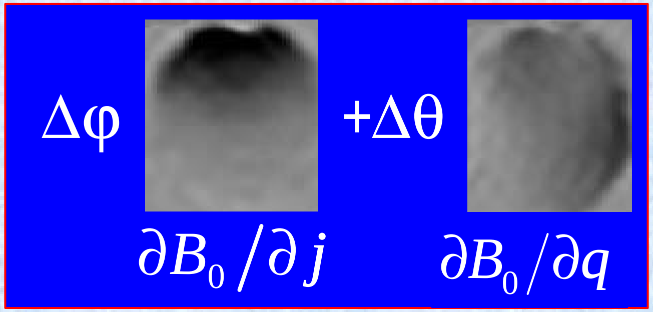


Estimate movement parameters.

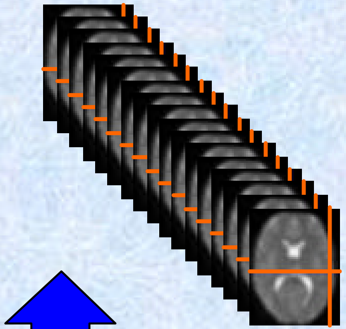
Estimate reference from mean of all scans.

Estimate new distortion fields for each image:

- estimate rate of change of field with respect to the current estimate of movement parameters in **pitch** and **roll**.



Unwarp time series.



Andersson et al, 2001

Contents

- Smoothing
- Rigid registration
- Spatial normalisation
  - Affine registration
  - Nonlinear registration
  - Regularisation

Spatial Normalisation - Reasons

- Inter-subject averaging
  - Increase sensitivity with more subjects
    - Fixed-effects analysis
  - Extrapolate findings to the population as a whole
    - Mixed-effects analysis
- Standard coordinate system
  - e.g., Talairach & Tournoux space

Spatial Normalisation - Objective

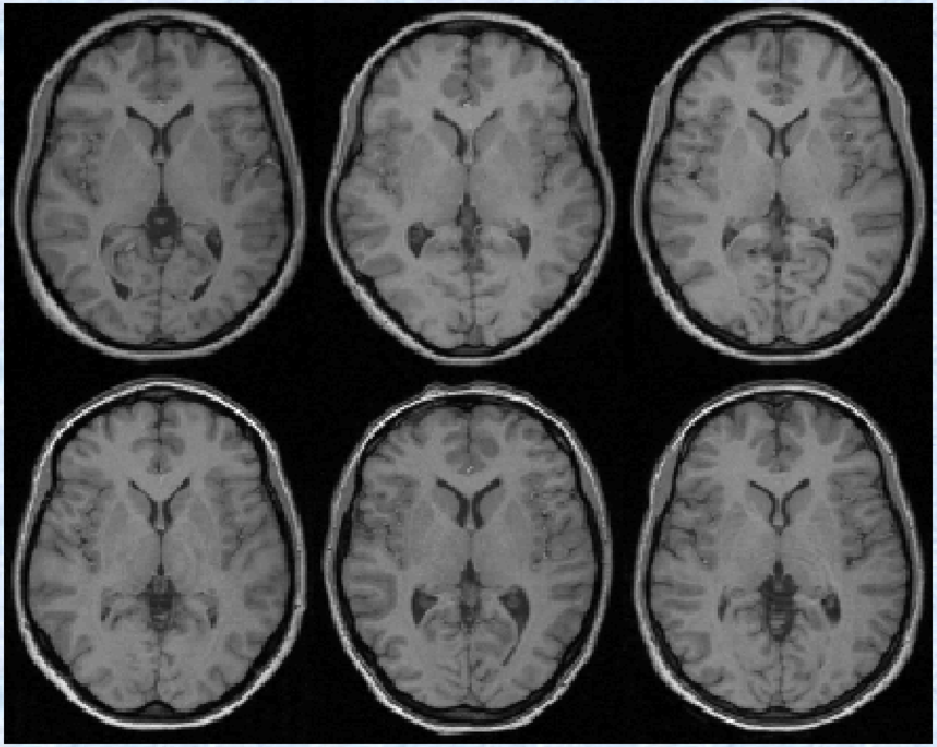
- Warp the images such that functionally homologous regions from different subjects are as close together as possible
  - Problems:
    - No exact match between structure and function
    - Different brains are organised differently
    - Computational problems (local minima, not enough information in the images, computationally expensive)
- Compromise by correcting gross differences followed by smoothing of normalised images

Very hard to define a one-to-one mapping of cortical folding

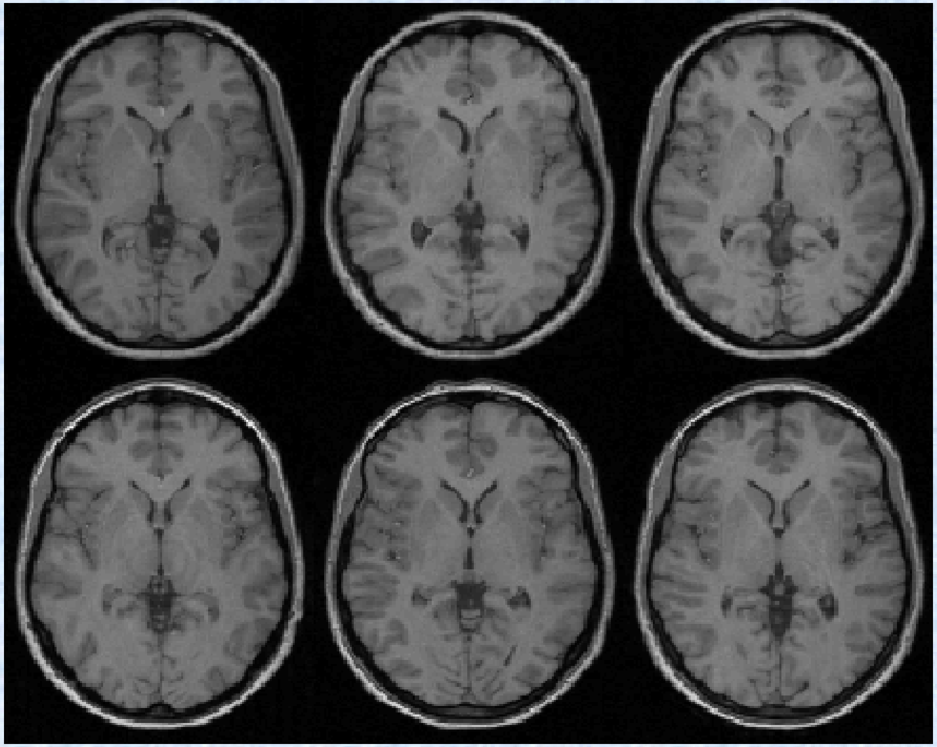
Use only approximate registration.

Spatial Normalisation - Procedure

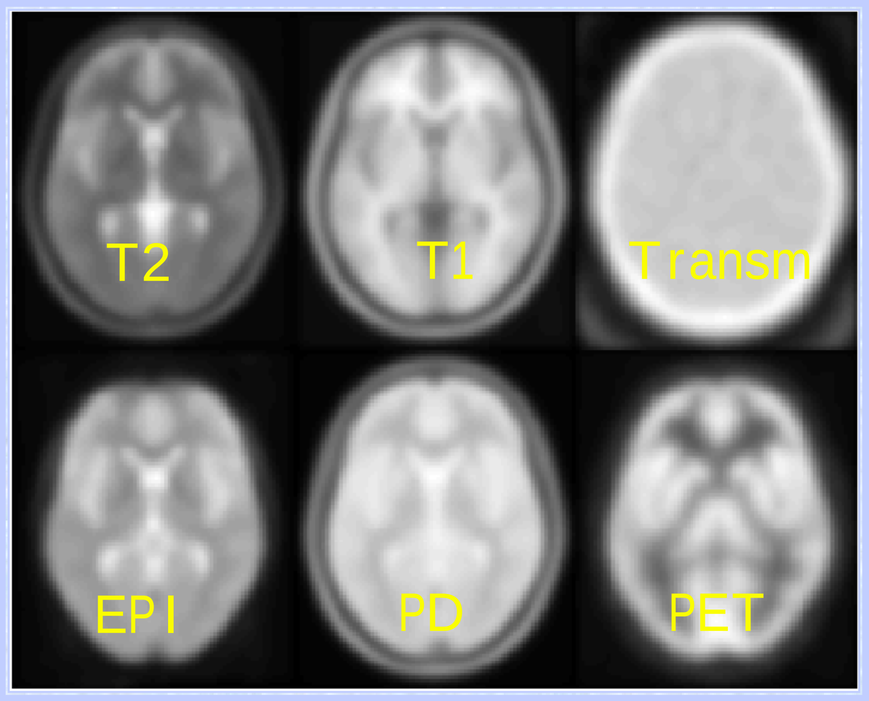
- Minimise mean squared difference from template image(s)



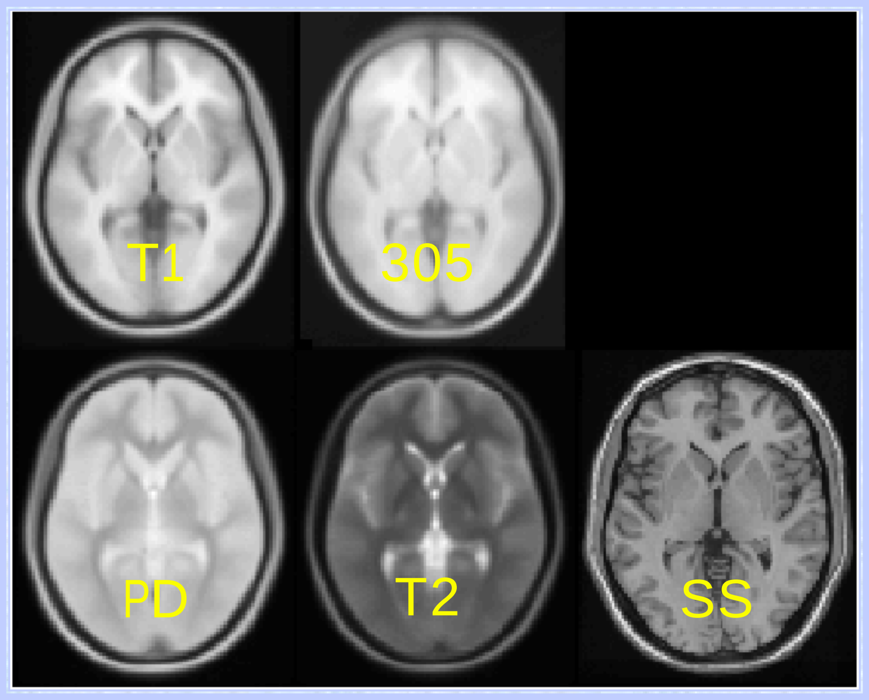
Affine registration



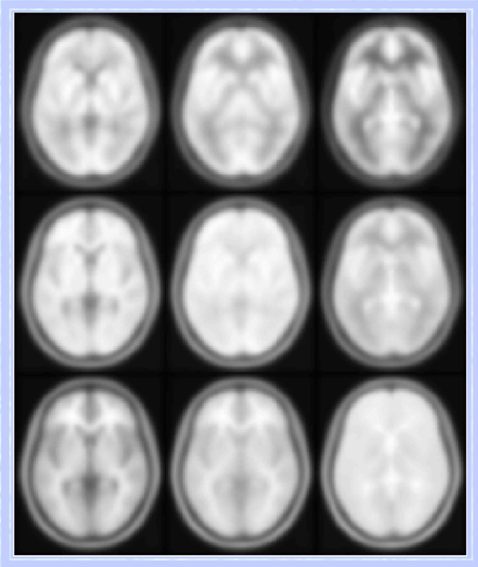
Non-linear registration



Template Images



“Canonical” images



A wider range of contrasts can be registered to a linear combination of template images.

Spatial normalisation can be weighted so that non-brain voxels do not influence the result.

Similar weighting masks can be used for normalising lesioned brains.

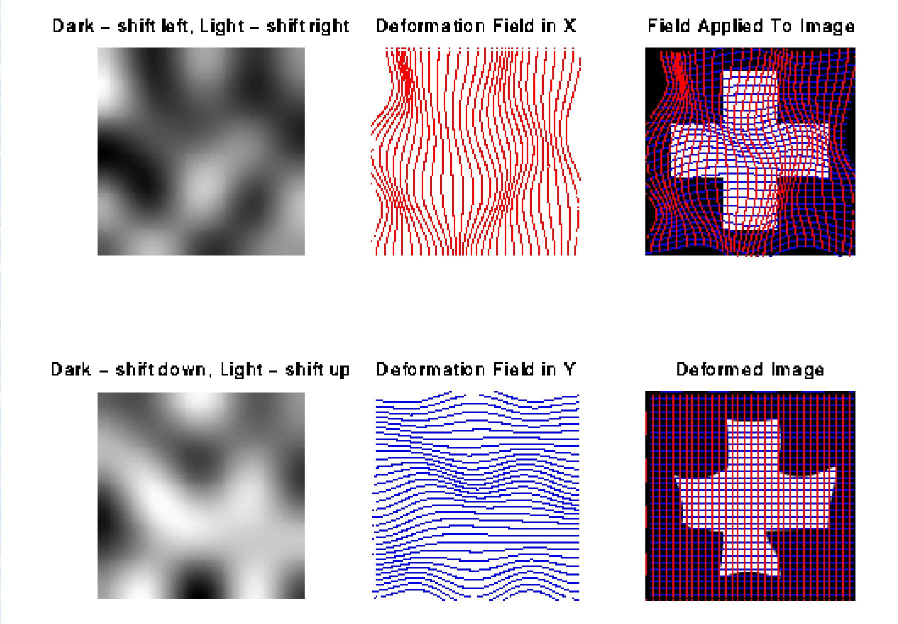
Spatial Normalisation - Templates

Spatial Normalisation - Affine

- The first part is a 12 parameter affine transform
  - 3 translations
  - 3 rotations
  - 3 zooms
  - 3 shears
- Fits overall shape and size

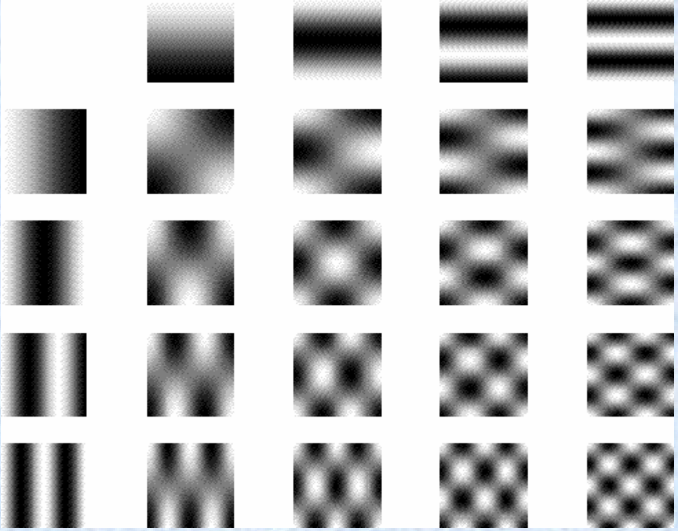
- Algorithm simultaneously minimises
  - Mean-squared difference between template and source image
  - Squared distance between parameters and their expected values (regularisation)

Spatial Normalisation - Non-linear



Deformations consist of a linear combination of smooth basis functions

These are the lowest frequencies of a 3D discrete cosine transform (DCT)

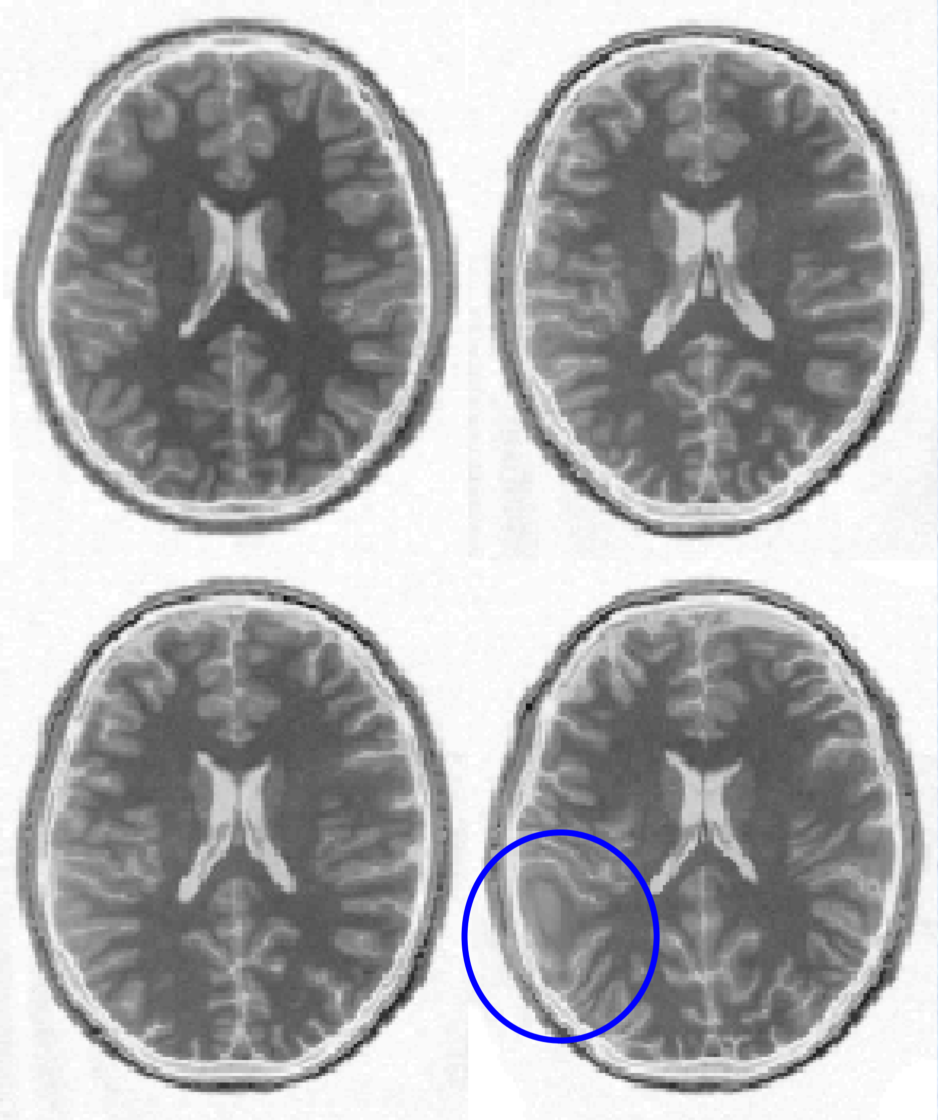


Algorithm simultaneously minimises

- Mean squared difference between template and source image
- Squared distance between parameters and their known expectation

Spatial Normalisation - Overfitting

Without regularisation, the non-linear spatial normalisation can introduce unnecessary warps.

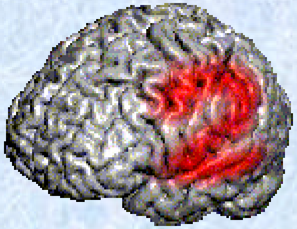


Template image

Affine registration. (χ^2 = 472.1)

Non-linear registration using regularisation. (χ^2 = 302.7)

Non-linear registration without regularisation. (χ^2 = 287.3)



References

Friston et al (1995): *Spatial registration and normalisation of images*. Human Brain Mapping 3:165-189

Collignon et al (1995): *Automated multi-modality image registration based on information theory*. IPMI'95 pp 263-274

Andersson et al (2001): *Modeling geometric deformations in EPI time series*. Neuroimage 13:903-919

Thévenaz et al (2000): *Interpolation revisited*. IEEE Trans. Med. Imaging 19:739-758.

Ashburner et al (1997): *Incorporating prior knowledge into image registration*. NeuroImage 6:344-352

Ashburner et al (1999): *Nonlinear spatial normalisation using basis functions*. Human Brain Mapping 7:254-266