# Neuroimage Processing

Instructor: Moo K. Chung

mkchung@wisc.edu

Lecture 10-11.

Deformation-based morphometry (DBM)

Tensor-based morphometry (TBM)

November 13, 2009

# Image Registration

- Process of transforming one image to another.
- Transformation types: linear (rigid-body), affine (non rigid-body), nonlinear.
- Type of data obtained: 3D vector fields (displacement, deformation).

# Affine Transformation matrix for (rigid, non rigid)

- *R*: 3 x 3 matrix of rotation, scaling and shear
- *p'=Rp+c, where*

$$p' = \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}, p = \begin{pmatrix} x \\ y \\ z \end{pmatrix}, c = \begin{pmatrix} c_x \\ c_y \\ c_z \end{pmatrix}$$

# Linear transform

- T is a linear transform if

$T(ap+bq) = aT(p)+bT(q)$ for all numbers $a, b$.

- Checking if the affine transform is linear. Let $T(p)=Rp+c$.

Note

$T(ap+bq)=R(ap+bq)+c=aT(p)+bT(q)+c(1-a-b)$.

This shows the affine transform is nonlinear.

# Rigid-body:

- Rotation and Translation only

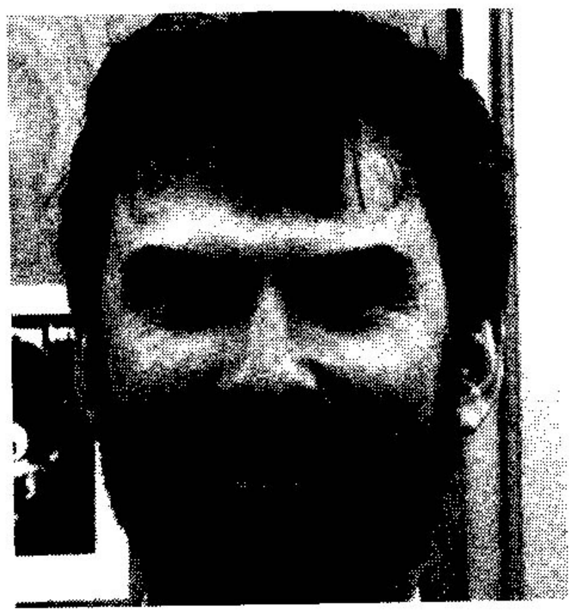
Original Image

Shape doesn’t change

A. Chowdhury,
University of Georgia

# Non rigid-body

Scaling, Translation, Rotation, Reflection, Shear

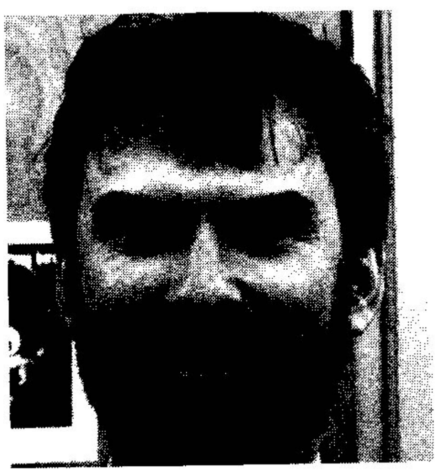


Original Image

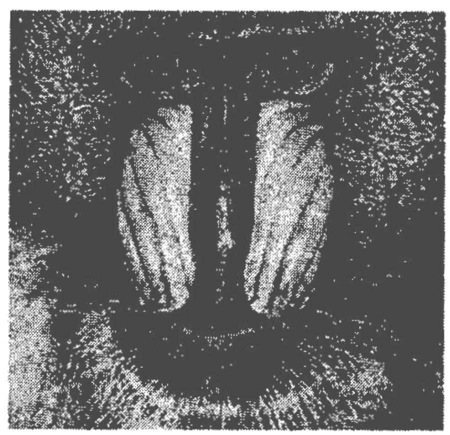
Non rigid-body

# Nonlinear transform

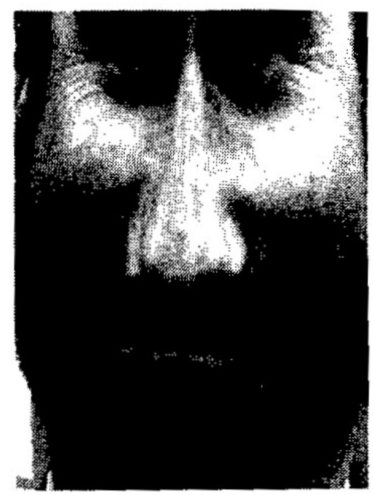
- Anatomical variability is encoded in the nonlinear transform.



Original image

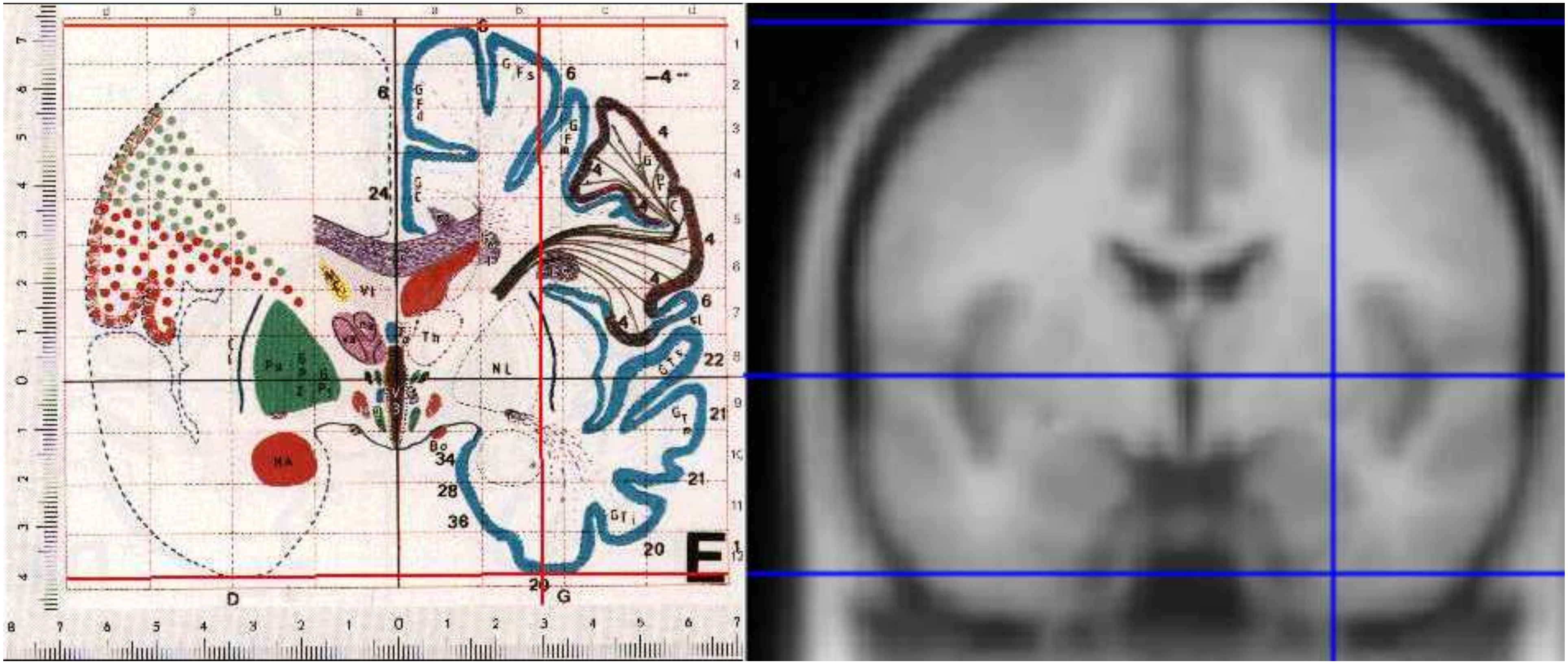


Target Image



Result of warping

# Talairach Template



Talairach coordinate system has been widely used to describe the location of brain structures.

http://www.talairach.org/

# Piecewise Affine Transform to Talairach

- The standard Talairach normalization approach
- It uses a different matrix transformation for each of the 12 pieces of the Talairach Grid
- This is a really old technique based on a single old subject.
- It provides an easy reference for comparing different study results.
- Other than for comparing results against literature, the Talairach approach is outdated.

# Talairach Definition

- Interhemispheric plane (3+ landmarks) $\Rightarrow 2$ rotations and 1 translation
- Anterior and posterior commissure (AC, PC) $\Rightarrow 3$rd rotation, 2 translations
- Scale to anterior, posterior, left, right, inferior, superior landmarks (7 parameters)
- Each cerebral hemispheres divided into *six associated blocks* (interhemispheric plane, AC-PC axial plane, 2 coronal planes through AC and PC.

# NIfTI image format

http://nifti.nimh.nih.gov/

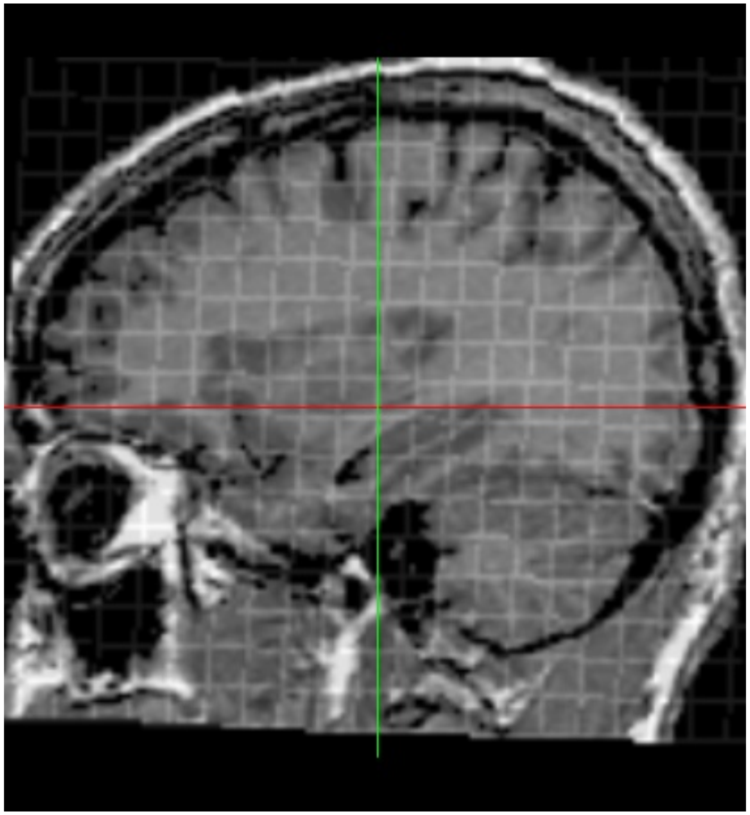
NIH initiated effort for standardizing brain imaging file format.
It supersedes the analyze and MNI file format.

MATLAB codes for reading/writing
http://www.rotman-baycrest.on.ca/~jimmy/NIFTI/

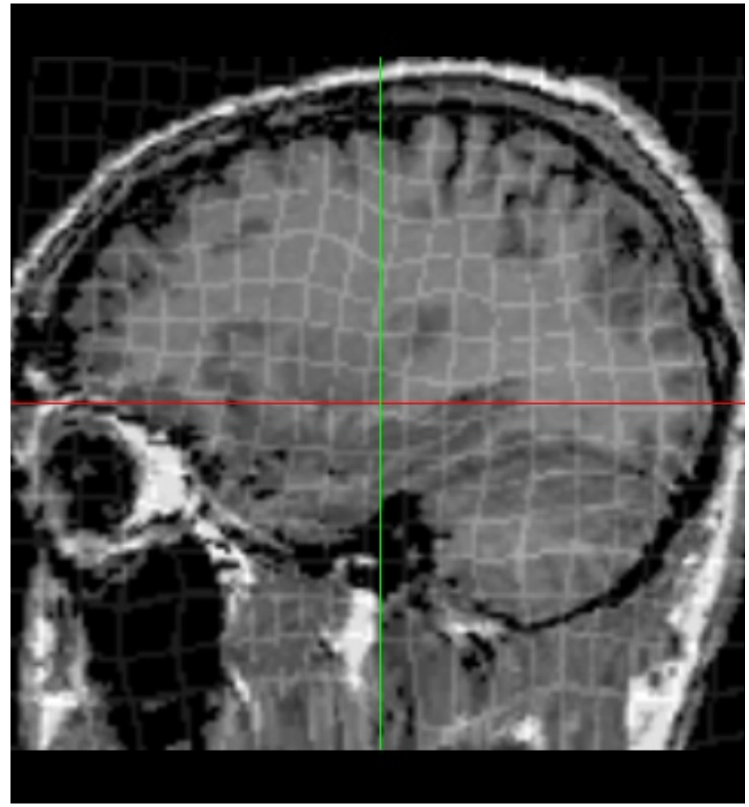
Siemens DICOM sort and conversion to NIfTI format
http://www.mathworks.com/matlabcentral/fileexchange/22508

# MATLAB Demonstration

# Affine vs. Nonlinear



Affine – 12 parameters



Non-Rigid ~ 2000 parameters

Lawrence H. Staib
Yale University

# Nonlinear registration tools

A*utomated image registration* (AIR) package (http://bishopw.loni.ucla.edu/AIR5/index.html) uses polynomial basis.

SPM package uses cosine basis.

Advanced Normalization Tools (ANTs)
http://picsl.upenn.edu/ANTS/

# Deformation field obtained from AIR

- The deformation field d is a transformation from a subject image to a target image:

$$d : (x, y, z) \rightarrow (x', y', z')$$

$$(x', y', z') = d(x, y, z)$$

- Example: AIR 3rd degree warping

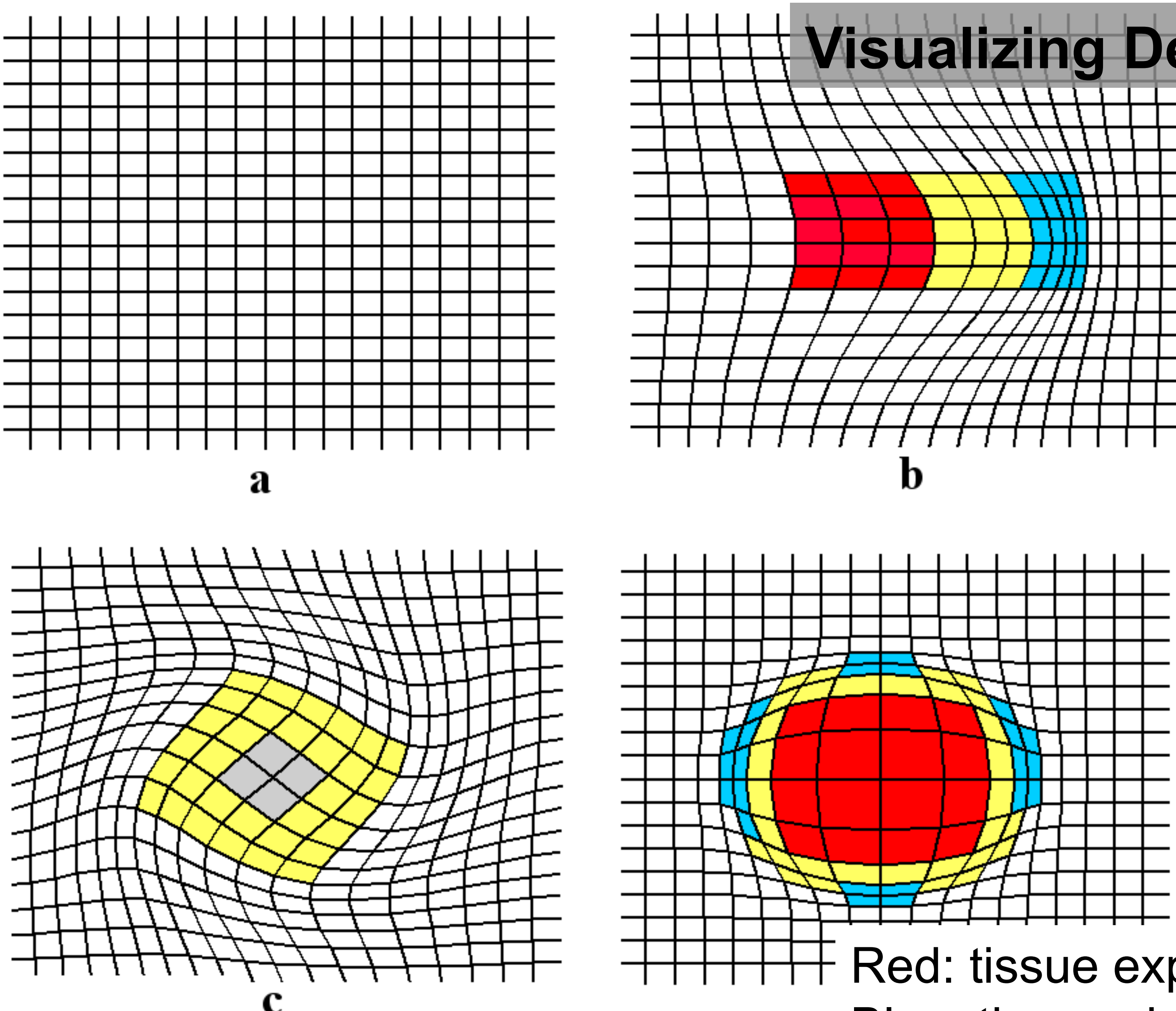
$$\begin{aligned} x' = &-1.24 - 0.47x - 1.02\times10^{-2}y + 0.99z + 1.25\times10^{-5}x^2 - 2.66\times10^{-5}xy + 4.04\times10^{-5}y^2 + 2.35\times10^{-6}xz \\ &+9.23\times10^{-5}yz + 5.39\times10^{-5}z^2 + 3.60\times10^{-8}x^3 + 1.04\times10^{-7}x^2y - 9.3\times10^{-10}xy^2 + 1.17\times10^{-7}y^3 \\ &-2.2\times10^{-8}x^2z + 1.24\times10^{-7}xyz - 6.82\times10^{-7}y^2z - 1.24\times10^{-7}xz^2 - 1.69\times10^{-8}yz^2 - 2.76\times10^{-7}z^3. \end{aligned}$$

# Deformation based Morphometry (DBM)

- It uses deformation fields obtained by nonlinear registration of brain images.

Read M.K.Chung.Book…
Chapter 7 upto section 7.2

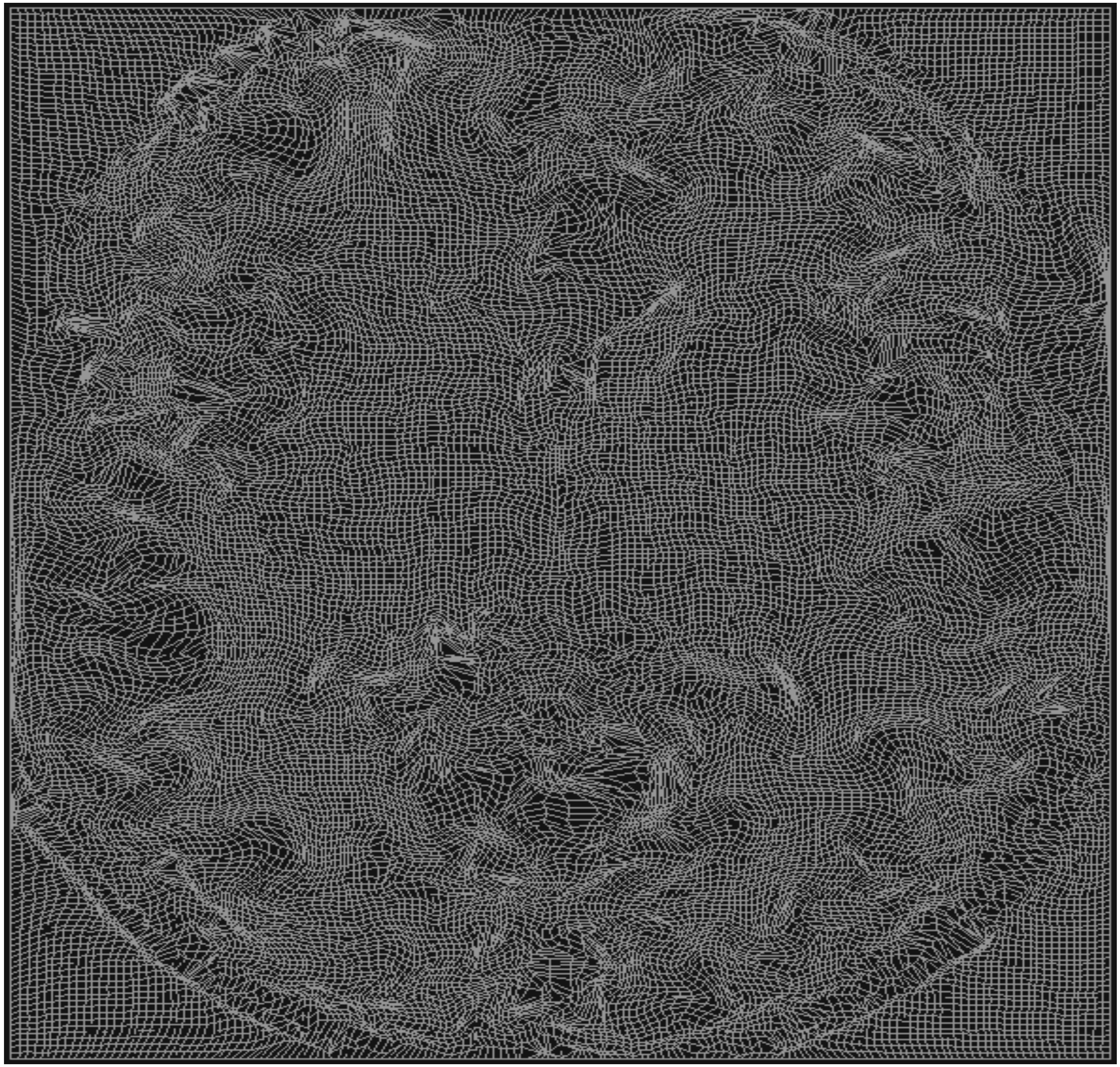
# Visualizing Deformation




Red: tissue expansion
Blue: tissue shrinking
Yellow: deformation change

Deformation field visualization

SPM result



## Displacement vector fields

•The vector difference between the final position and the original position.

target initial voxel position

$$(x', y', z') - (x, y, z)$$

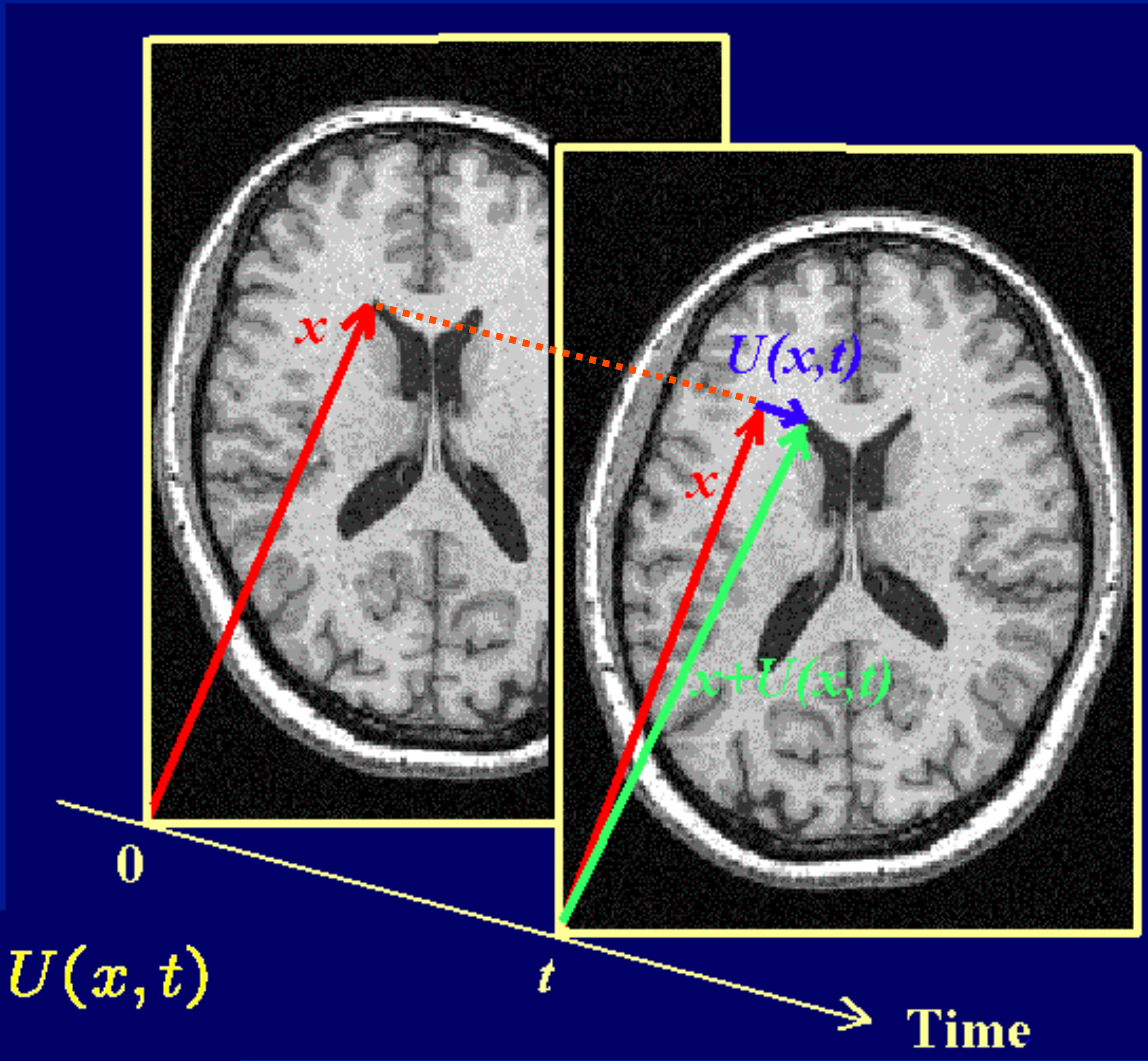
$$U(x, y, z) = d(x, y, z) - (x, y, z)$$

displacement deformation

•Relation between displacement and deformation

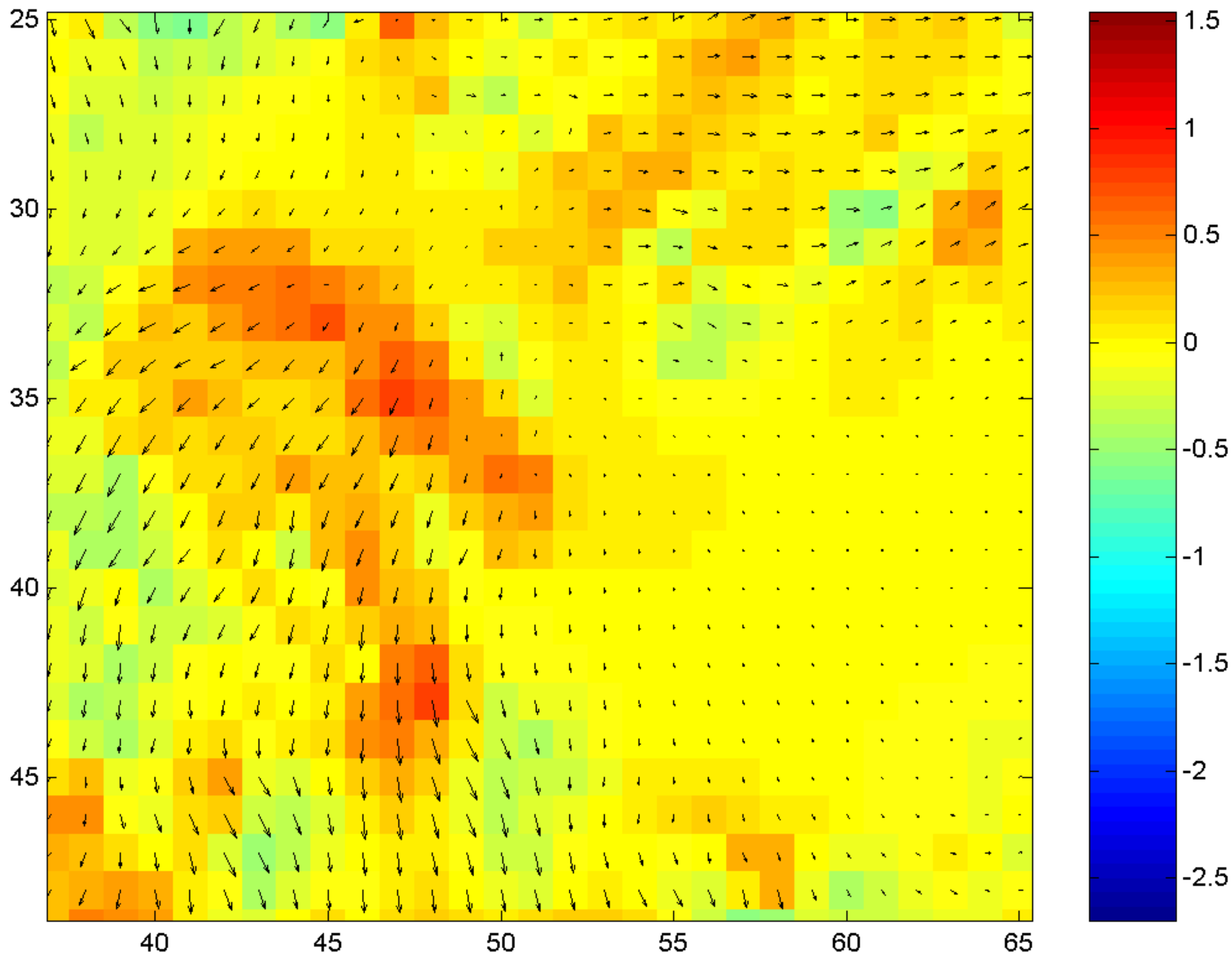
$$d(x, y, z) = (x, y, z) + U(x, y, z)$$

# Displacement vector




$$x \rightarrow x + U(x,t)$$

# Visualizing Displacement Vector Field




Displacement is easier to model statistically than deformation

NeuroImage
doi:10.1006/nimg.2001.0862, available online at http://www.idealibrary.com on IDEAL®

# A Unified Statistical Approach to Deformation-Based Morphometry

M. K. Chung,* K. J. Worsley,*·† T. Paus,† C. Cherif,‡ D. L. Collins,† J. N. Giedd,§
J. L. Rapoport,§ and A. C. Evans†

*Department of Mathematics and Statistics and †Montreal Neurological Institute, McGill University, Montréal, Québec, Canada; ‡Département de Mathématiques, École Polytechnique Fédérale de Lausanne, Switzerland; and §Child Psychiatry Branch, National Institute of Mental Health, NIH, Bethesda, Maryland 20892

Received August 14, 2000

28 normal subjects, 1.5T MRI, 2 scans per subjects

First scan: 11.5±3.1 year, second scan: 17.8 ±3.2 year

*Problem: localize the regions of anatomical change over time*

Modeling on the rate of displacement change
**Why?** Easier than modeling on deformation itself.

$$\frac{\partial U}{\partial t}(x,t) = \mu_0(x) + \Sigma^{1/2}(x)\epsilon(x)$$

$\Sigma$ : Covariance matrix

$\epsilon$ : Gaussian random vector field

$$H_0 : \mu_0 = 0 \text{ vs. } H_1 : \mu_1 \neq 0$$

# Estimating the rate of change

- For subject *j*, displacement is given by $U^j$

- Finite difference estimation:

$$\frac{\partial U^j}{\partial t} = \frac{U^j}{\text{age difference}}$$

# Hotelling's T-square statistic

- Rate of displacement $V^j$
- Sample mean $$\overline{V} = \sum_{j=1}^{n} V^j$$
- Sample covariance $$\hat{\Sigma} = \frac{1}{n-1} \sum_{j=1}^{n} (V^j - \overline{V})(V^j - \overline{V})^T$$
- Hotelling's T-square (related to Mahalanobis distance) $$H = \overline{V}^T \hat{\Sigma}^{-1} \overline{V} \approx cF_{3,n-3}$$

# Hotelling's T-square for two sample test

Group index *j*, subject index *i*

$$V_{ij} = \mu_j + \Sigma^{1/2} e_{ij}$$

$$H_0 : \mu_1 = \mu_2 \text{ vs. } H_1 : \mu_1 \neq \mu_2$$

n and m samples each

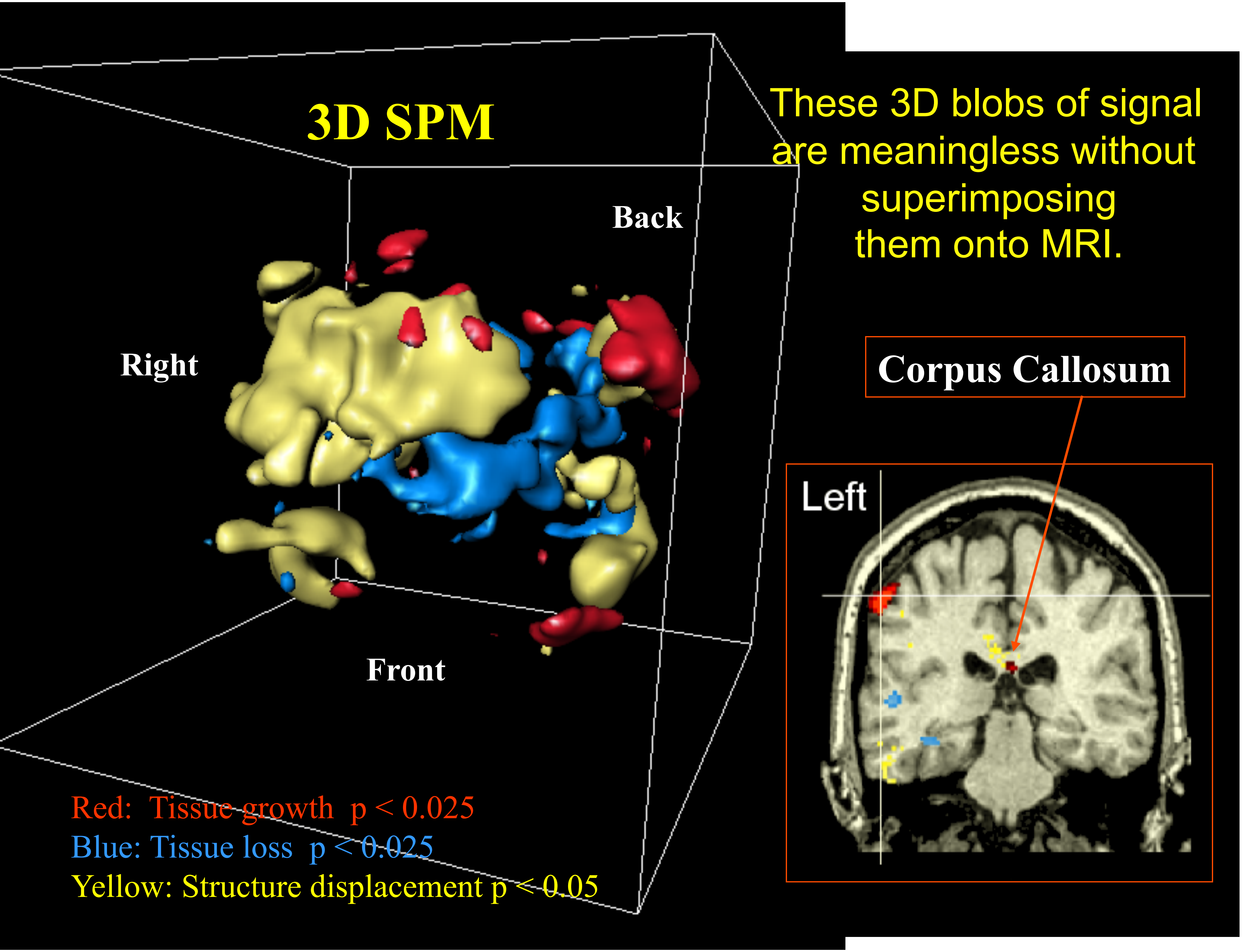
- Sample means: $\overline{V}_1, \quad \overline{V}_2$

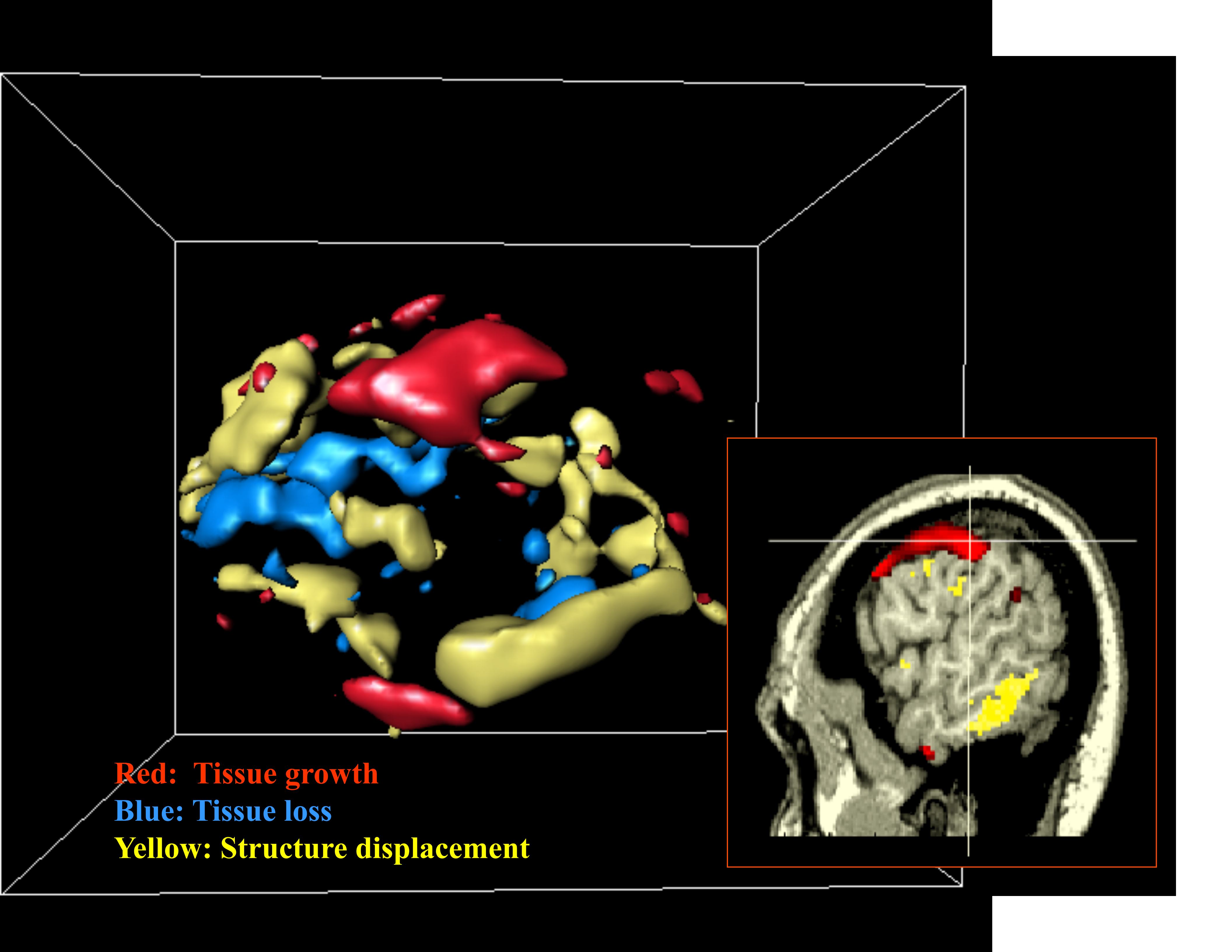
- Sample covariance: pool the variance across groups

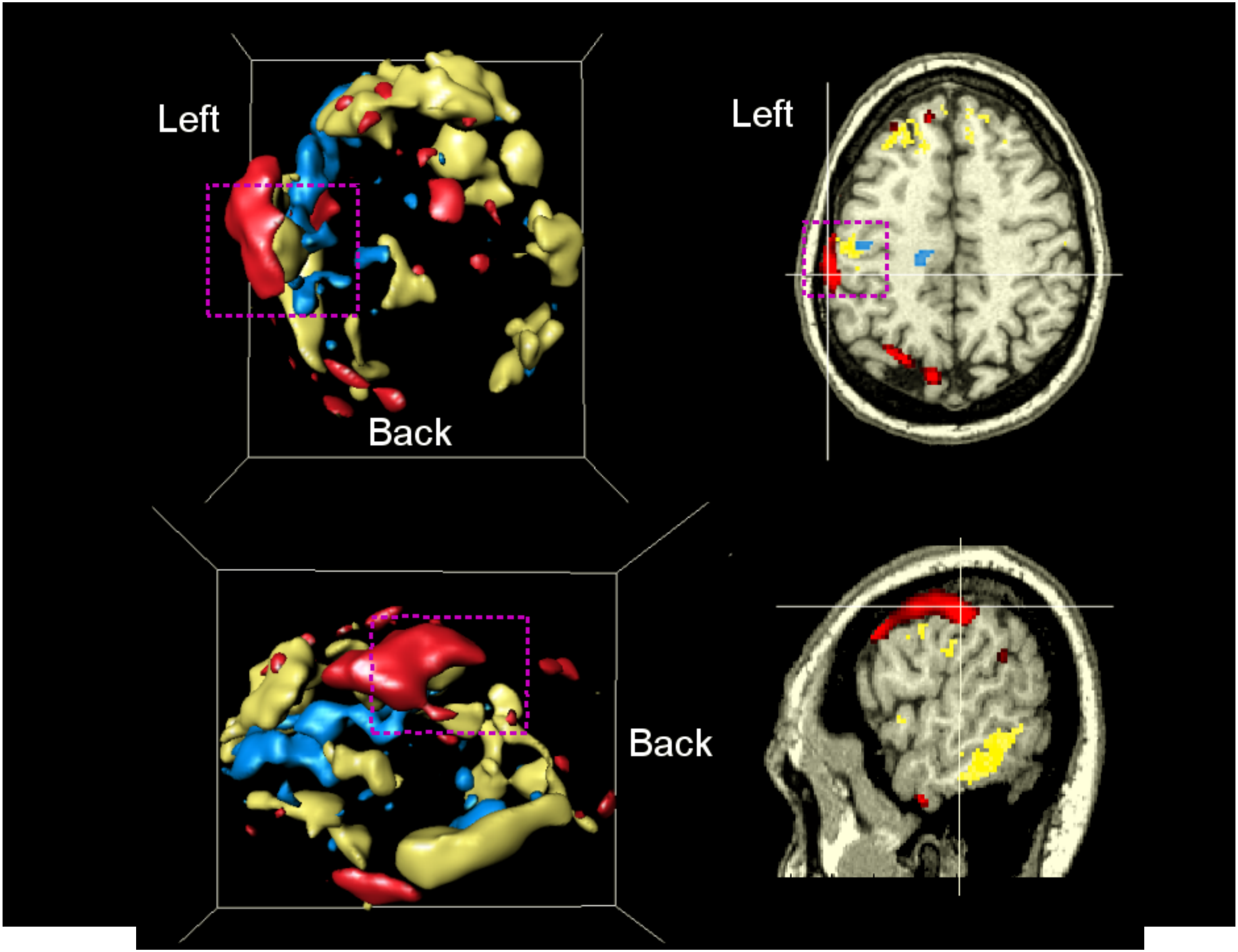
$$\hat{\Sigma} = \frac{1}{n+m-2}\left[\sum_{i=1}^{n}(V_{i1}-\overline{V}_1)(V_{i1}-\overline{V}_1)^T + \sum_{i=1}^{m}(V_{i2}-\overline{V}_2)(V_{i2}-\overline{V}_2)^T\right]$$

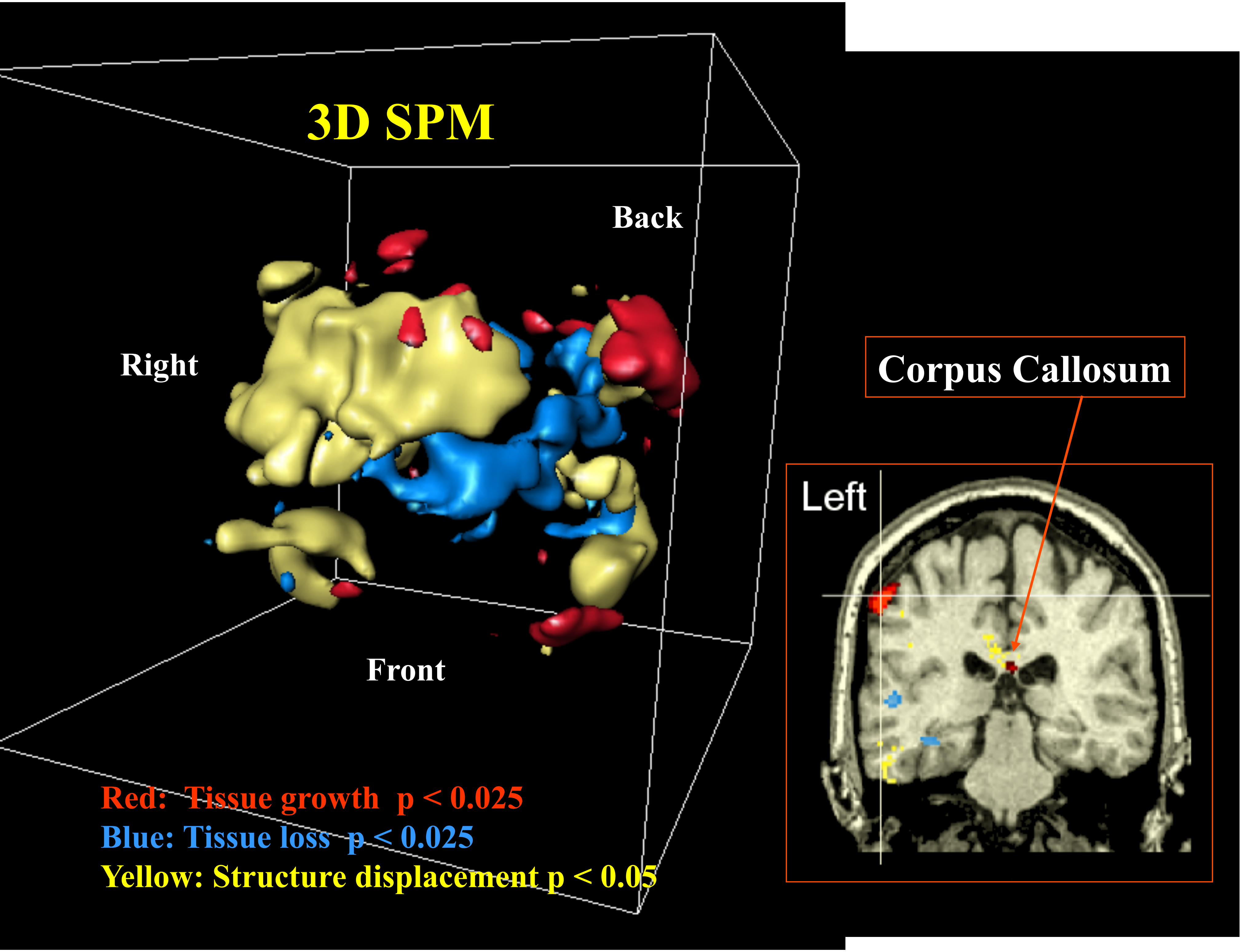
- Hotelling's T-square statistic for two samples

$$H = \left(\overline{V}_2 - \overline{V}_1\right)^T \hat{\Sigma}^{-1}\left(\overline{V}_2 - \overline{V}_1\right) \approx cF_{3,n+m-4}$$

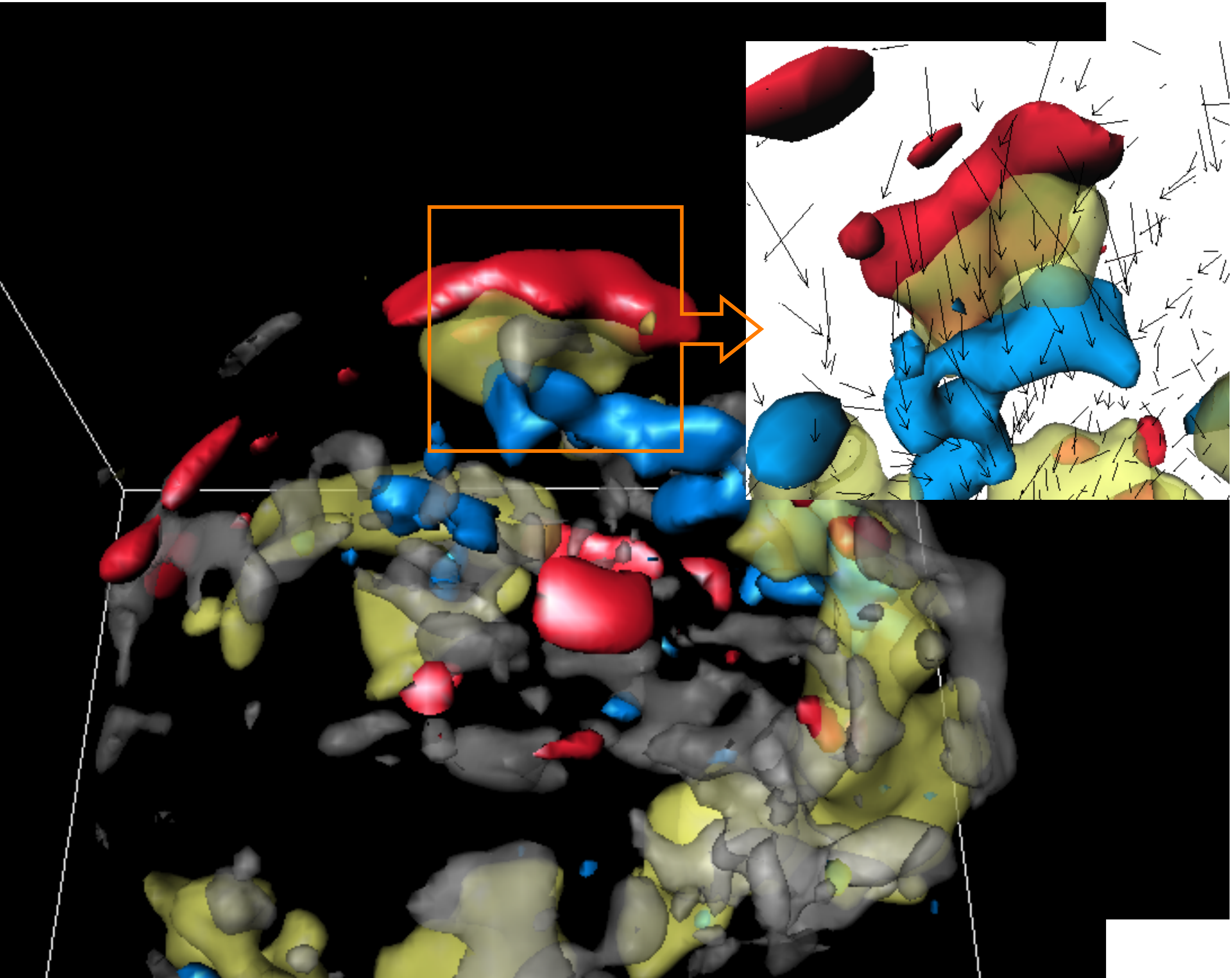

3D SPM
Back
Right
Front
Red: Tissue growth p < 0.025
Blue: Tissue loss p < 0.025
Yellow: Structure displacement p < 0.05
These 3D blobs of signal are meaningless without superimposing them onto MRI.
Corpus Callosum
Left


Red: Tissue growth
Blue: Tissue loss
Yellow: Structure displacement


Left
Back
Left
Back

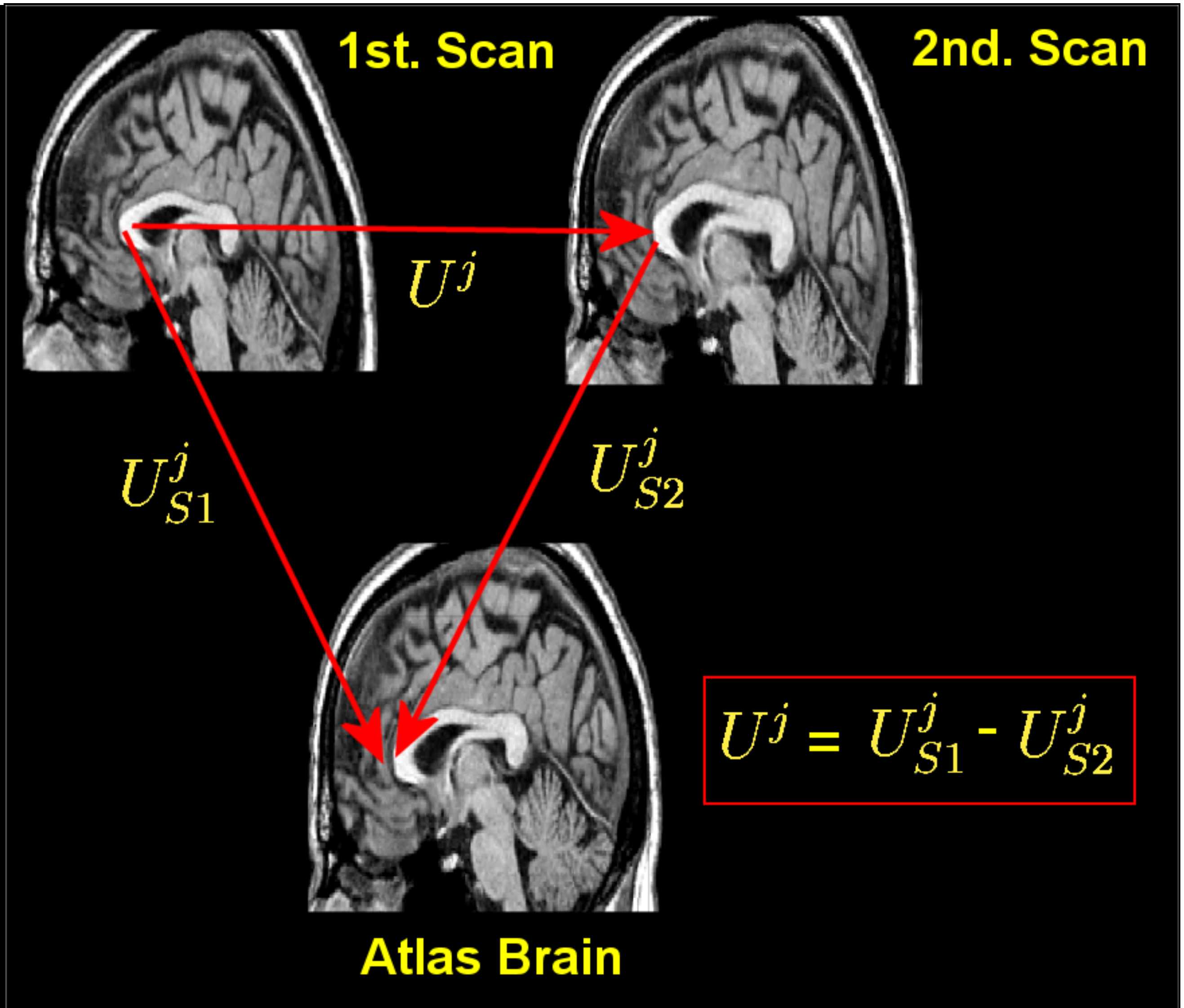


3D SPM
Back
Right
Front
Corpus Callosum
Left
Red: Tissue growth p < 0.025
Blue: Tissue loss p < 0.025
Yellow: Structure displacement p < 0.05



# Limitation of Chung et al., 2001

Registration procedures in Chung et al. (2001) is not optimal.




How it should be done correctly? To reduce the total amount of registration errors, 1st scans should be registered to the 2nd scans first.

# Tensor-based Morphometry (TBM)

- It uses higher order spatial derivatives of deformation fields to construct morphological tensor maps.
- From these tensor maps, 3D statistical parametric maps (SPM) are created to quantify the variations in the higher order change of deformation.

$$\frac{\partial U}{\partial x} = \begin{pmatrix} \frac{\partial U_1}{\partial x_1} & \frac{\partial U_1}{\partial x_2} & \frac{\partial U_1}{\partial x_3} \\ \frac{\partial U_2}{\partial x_1} & \frac{\partial U_2}{\partial x_2} & \frac{\partial U_2}{\partial x_3} \\ \frac{\partial U_3}{\partial x_1} & \frac{\partial U_3}{\partial x_2} & \frac{\partial U_3}{\partial x_3} \end{pmatrix}.$$

Determinant out of this

The component $\partial U_j/\partial x_i$ is called the *displacement tensor* and, in tensor-based morphometry (Ashburner and Friston, 2000), these nine components form scalar fields used to measure the second-order morphological variabilities. Note that local translation captures the

# Jacobian determinant (JD)

- The Jacobian determinant *J* of the deformation field is mainly used to detect volumetric changes.

Voxel position $x = (x_1, x_2, x_3)'$

Deformation $d = (d_1, d_2, d_3)'$

Jacobian determinant

$$J(x) = \det \frac{\partial d(x)}{\partial x} = \det\left(\frac{\partial d_j}{\partial x_i}\right)$$

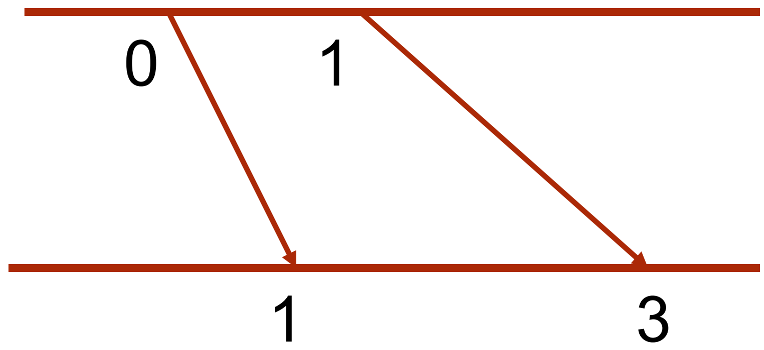
## Interpretation of Jacobian determinant (JD)

- JD measures the volume of the deformed unit-cube after registration.
- In images, a voxel can be considered a unit-cube
- JD measures how voxel volume changes after registration.

# JD Computation

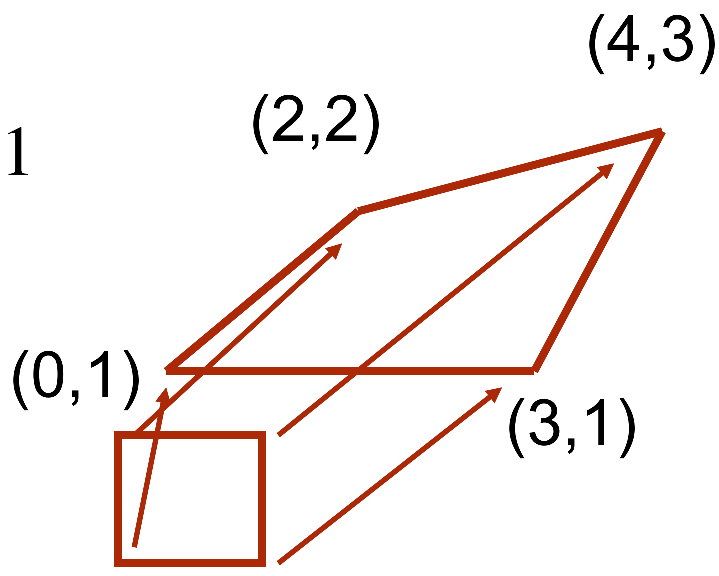
- 1D example:

$$x' = 2x + 1$$
$$J(x) = 2$$




- 2D example:

$$x' = 2x + y + 1$$
$$y' = x + 2y$$
$$J(x,y) = 3$$

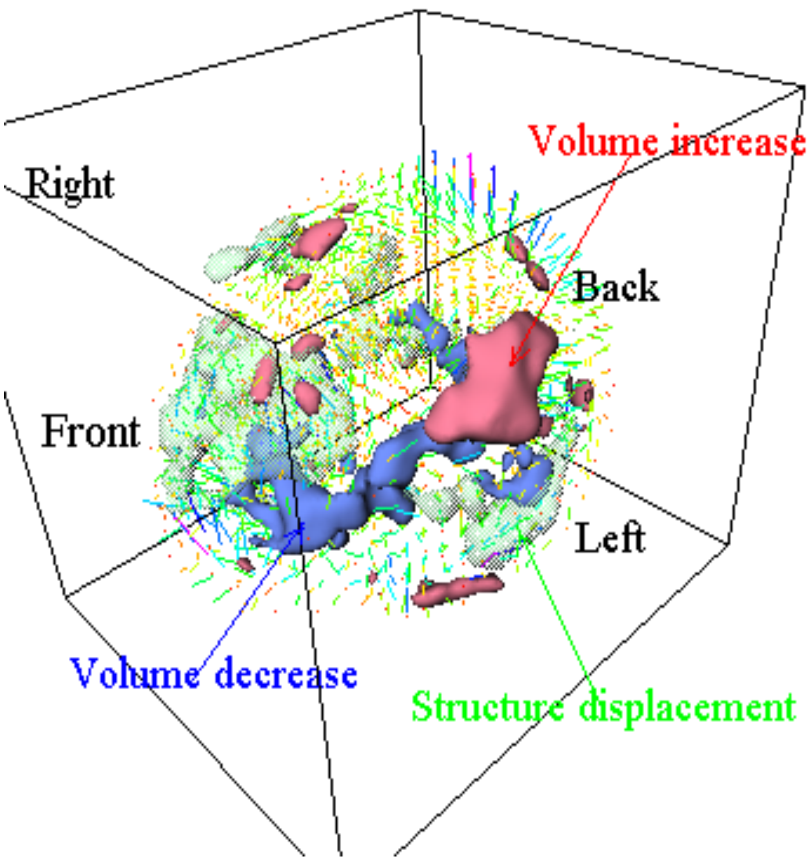


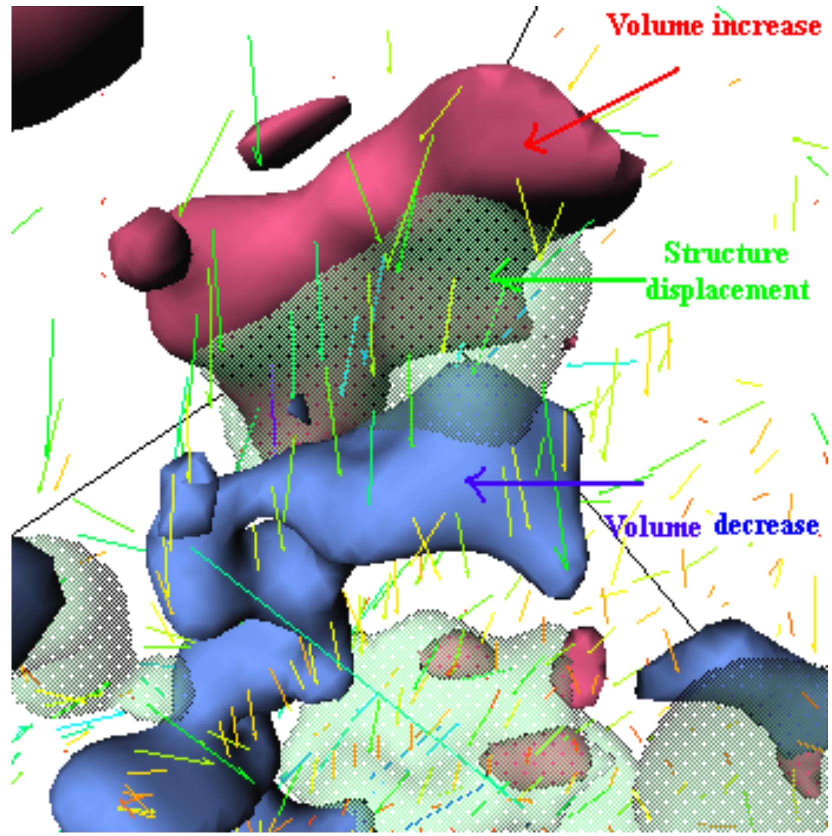
# Is it possible to perform TBM using only an affine registration?

- *For affine transformation p'=Ap+B*, the Jacobian determinant is det(*A*) at every vovels.
- Every voxel will have the same scalar value.
- Affine registration based TBM only detect global size difference.

TBM requires really good high order registration technique to work properly.

Statistically significant regions of local volume change
JD > 1 volume increase, JD < 1 volume decrease over time

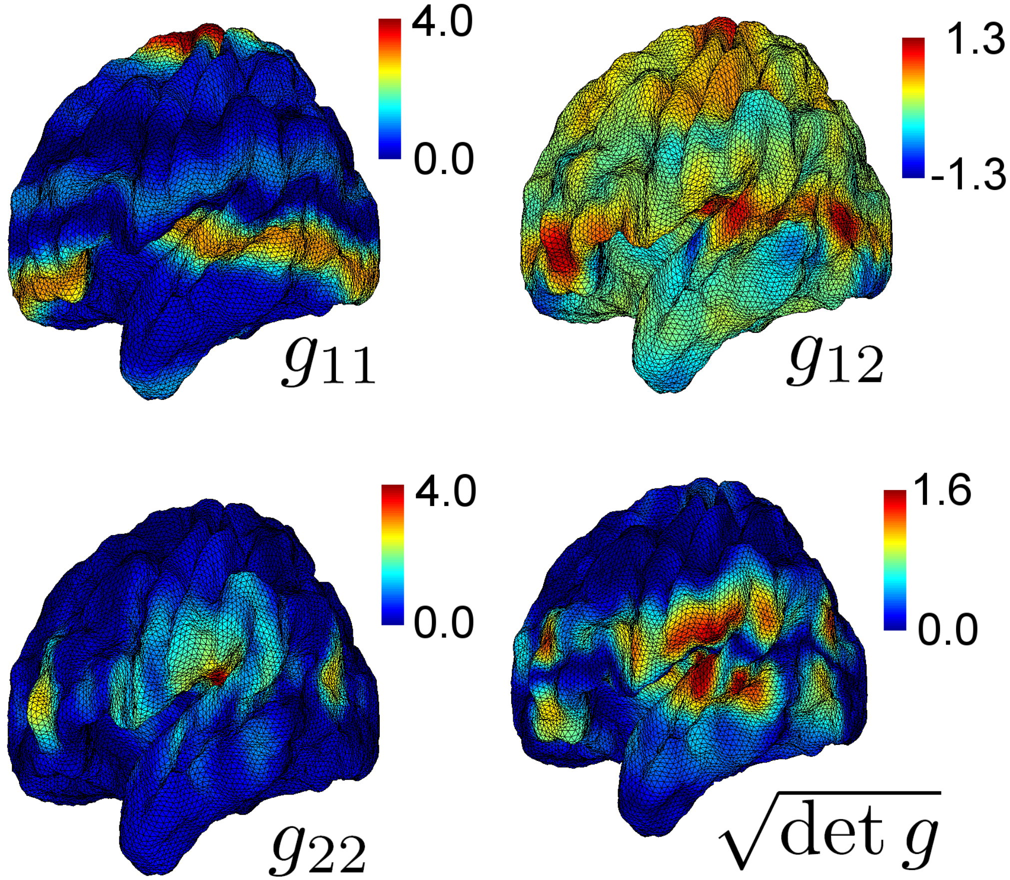



My first paper ☺ Chung et al. 1999. HBM meeting

# Surface-based Jacobian determinant

Generalization of Jacobian determinant in arbitrary manifold
= determinant of Riemannian metric tensors
= local volume (surface area) expansion



# Distributional assumption: Normality of JD

$$d(p) = p + U(p)$$

$$\frac{\partial d(p)}{\partial p'} = I + \frac{\partial U(p)}{\partial p'}$$

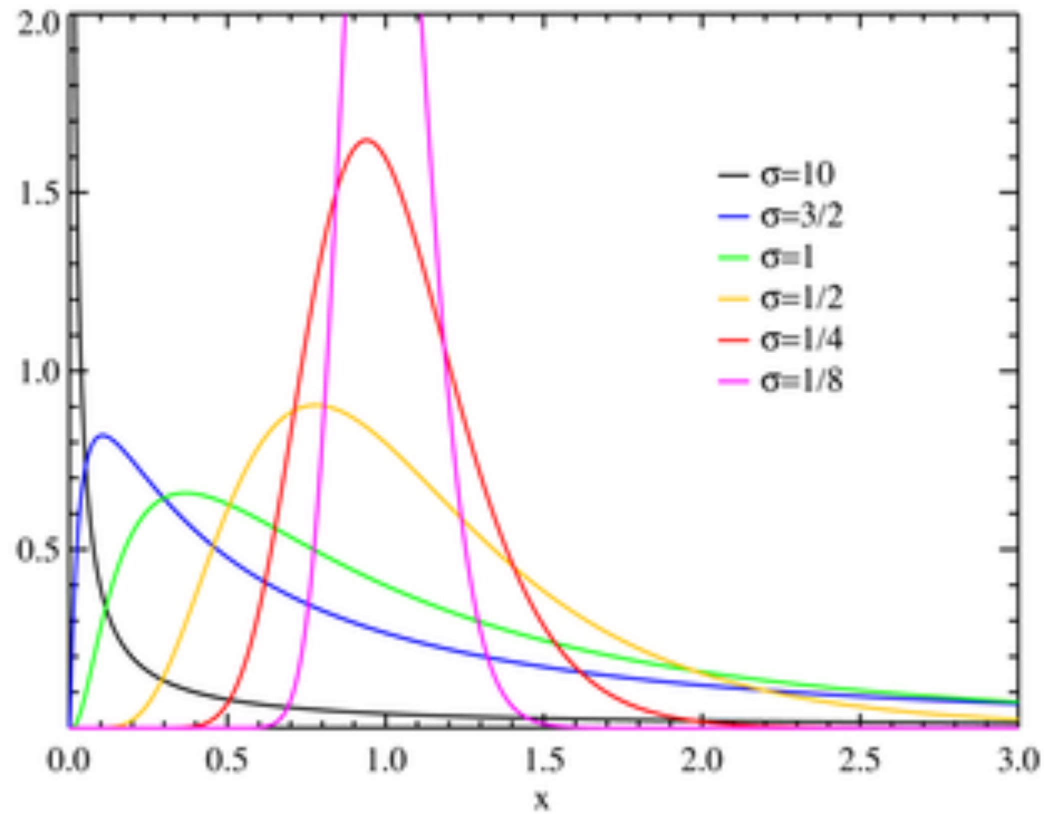
$$J(p) \cong 1 + tr\left(\frac{\partial U}{\partial p'}\right) = 1 + \frac{\partial U_1}{\partial p_1} + \frac{\partial U_2}{\partial p_2} + \frac{\partial U_3}{\partial p_3}$$

Note that we modeled displacement *U* to be a Gaussian random field. Any linear operation (derivative) on a Gaussian random field is again Gaussian. So *J* is approximately a Gaussian random field.

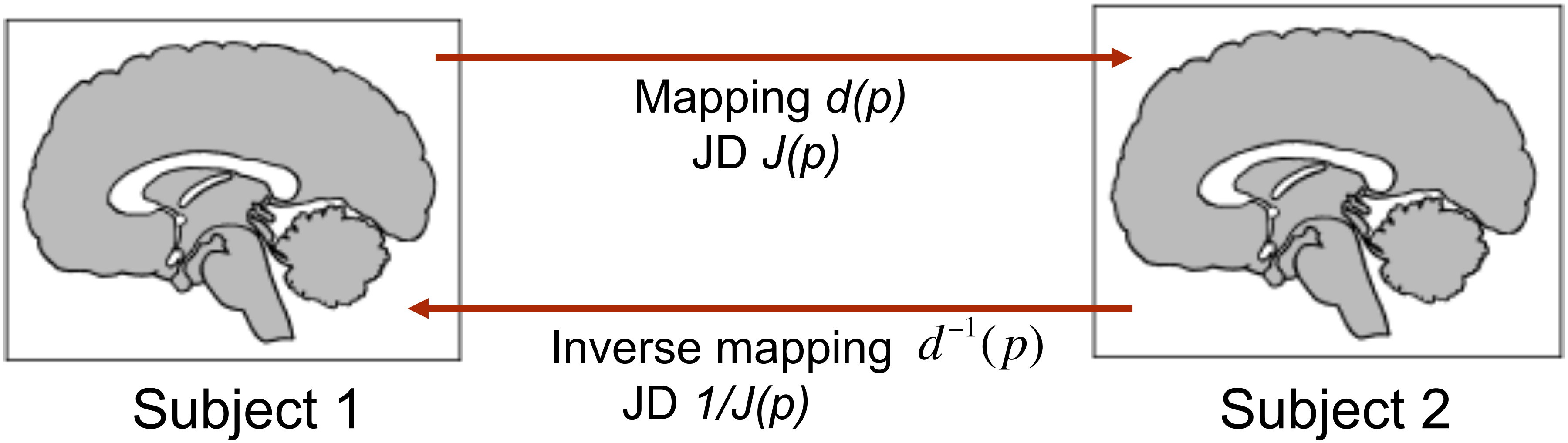
# Lognormal distribution

- Random variable *X* is log-normally distributed if *log X* is normally distributed N(mu, sigma^2)
- For E *log X=0*, the shape of density:

Some lognormal distribution looks normal so how do we check if data follows normal or lognormal?



# Properties of JD




- *J(p) >0* for one-to-one mapping
- *J(p) > 1* volume increase; *J(p) <1* volume decrease

•Due to symmetry, the statistical distribution of J(p) and 1/J(p) should be identical.

# Distributional assumption: Lognormality of JD

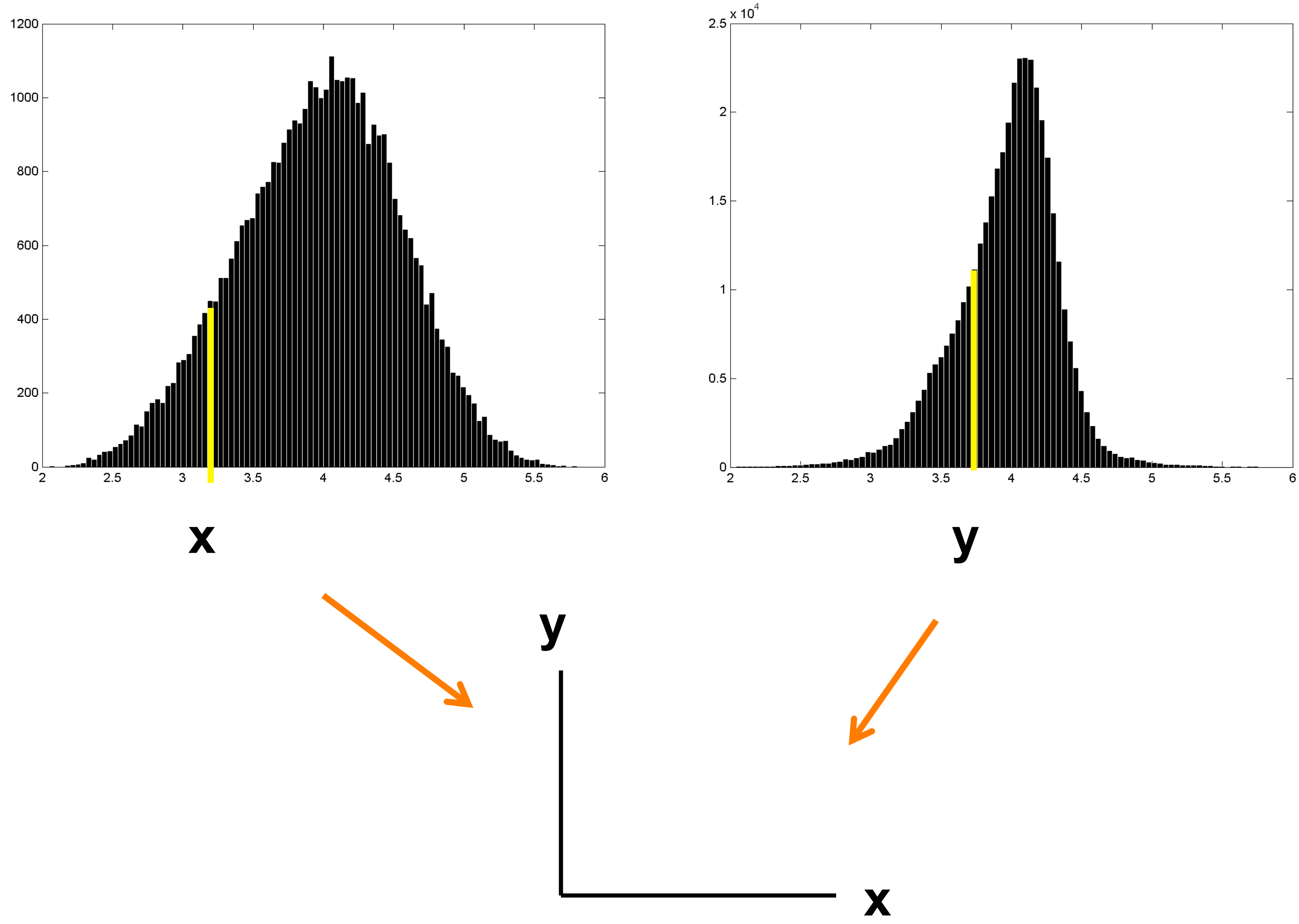
- Domain $-\infty < \log J(p) < \infty$
- If *J(p)=1*, $\log J(p) = 0$
- Symmetry: $\log\left[J^{-1}(p)\right] = -\log J(p)$
- These 3 properties show that JD can be modeled as *lognormal distribution*.

# Testing normality of data

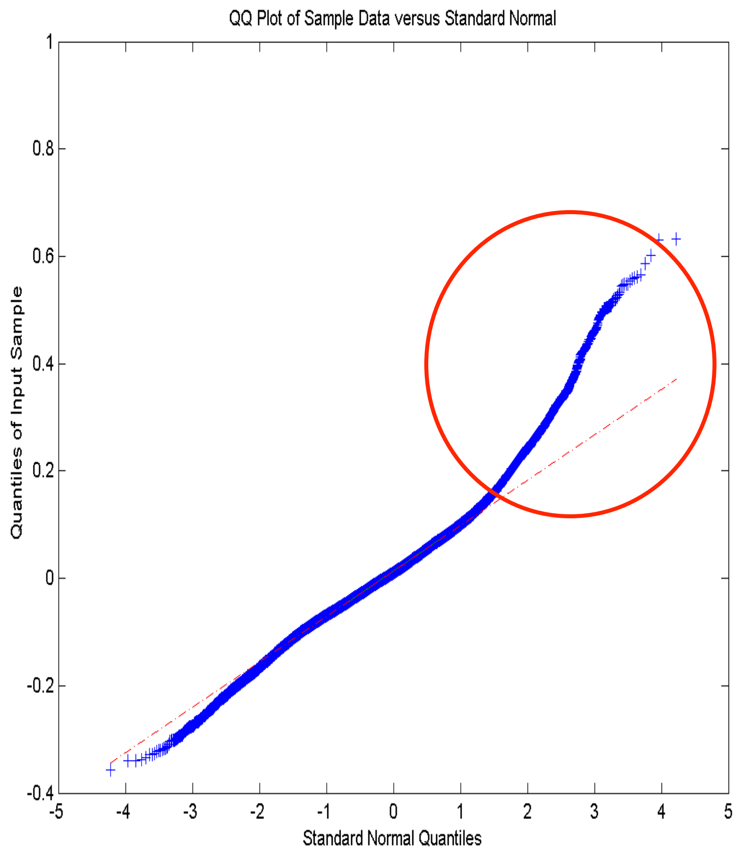
- How do we check if JD is normal or lognormal emphatically?
- Quantile-quantile (QQ) plot can be used. For given probability *p*, the *p*-th quantile of random variable *X* is the point *q* that satisfies

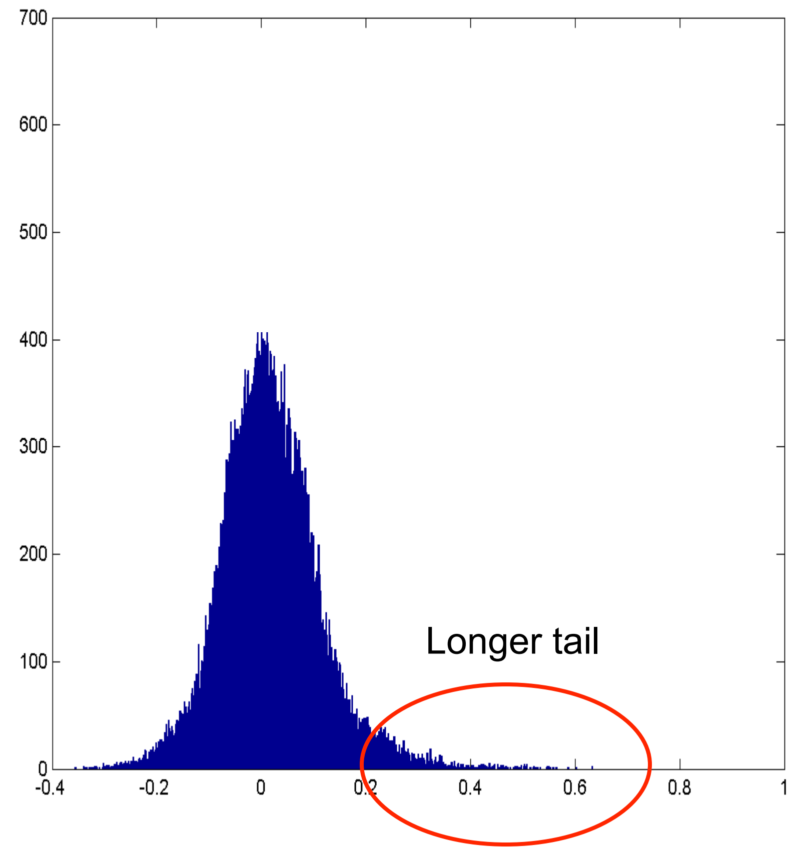
$$P(X < q) = p.$$

# QQ-plot compares quantiles

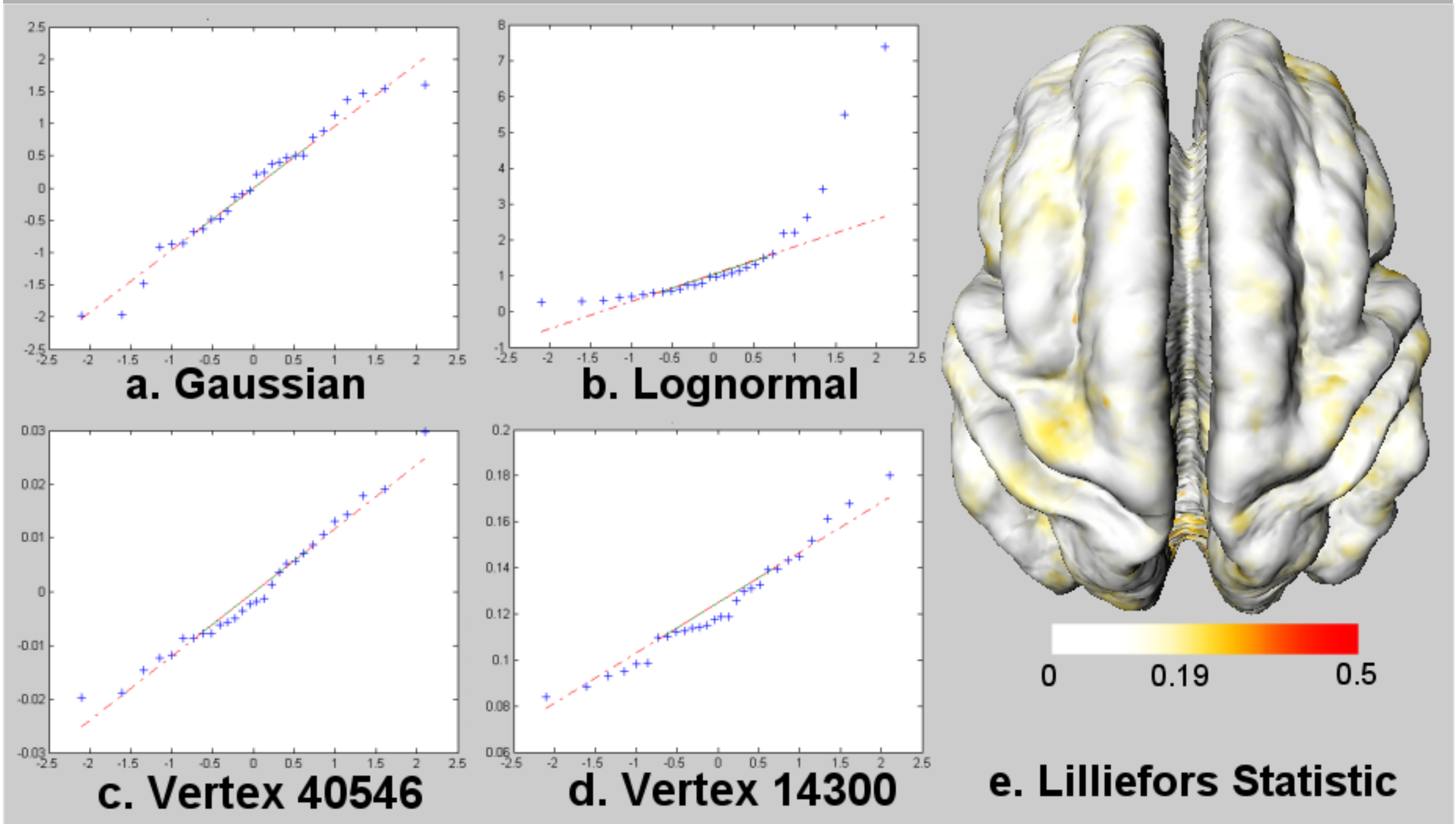


# Normal probability plot showing asymmetric distribution



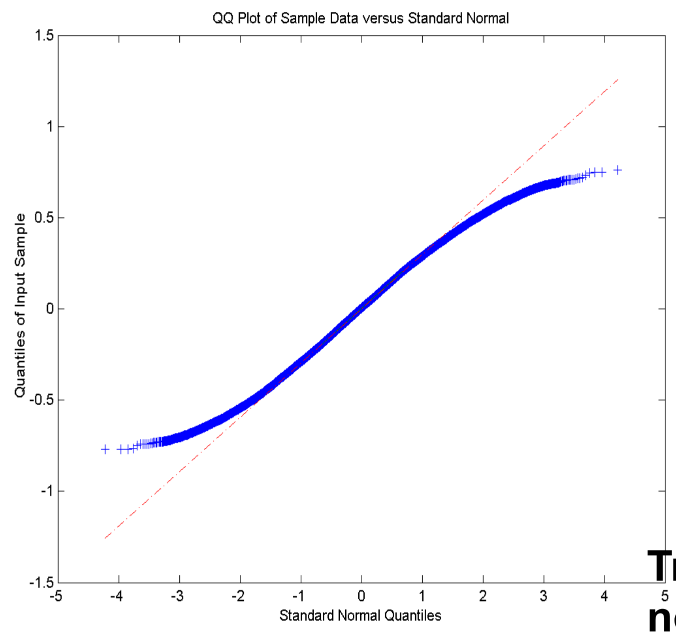



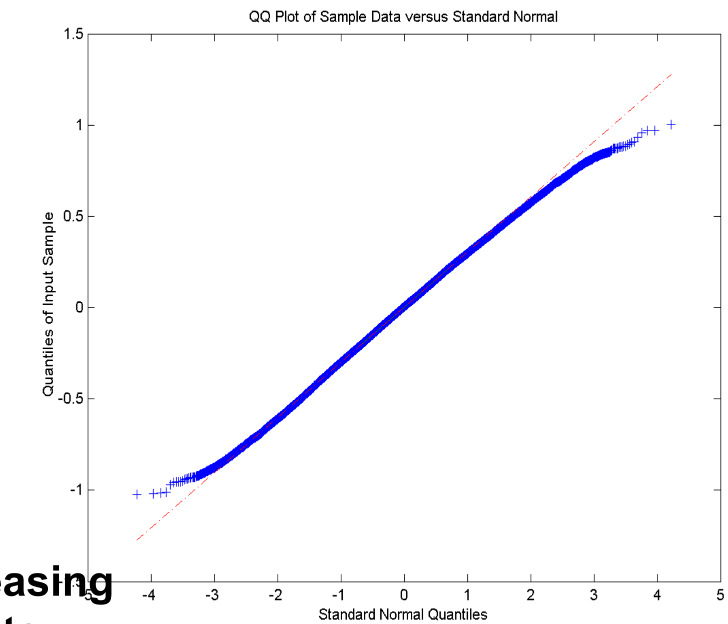
# Checking normality across subjects on cortical measure

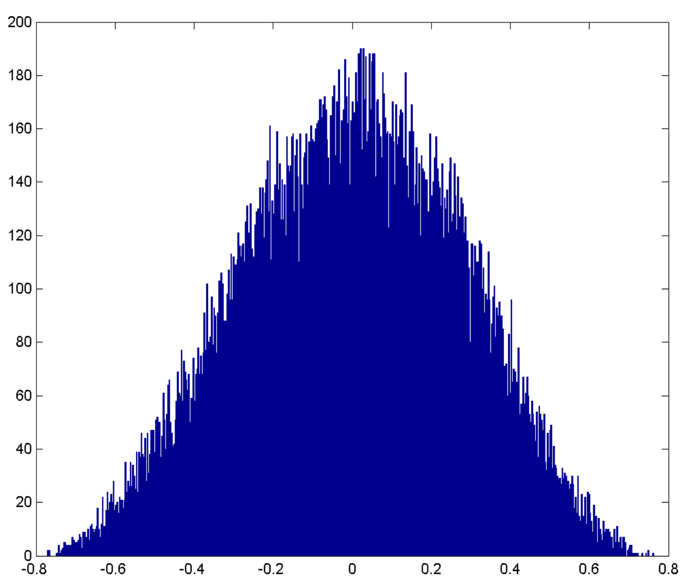



Chung et al., 2003 NeuroImage

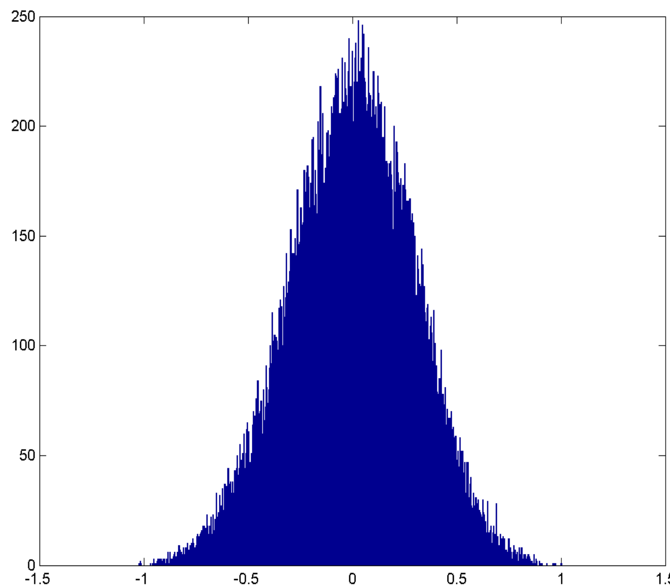
# Tricks for increasing normality of data



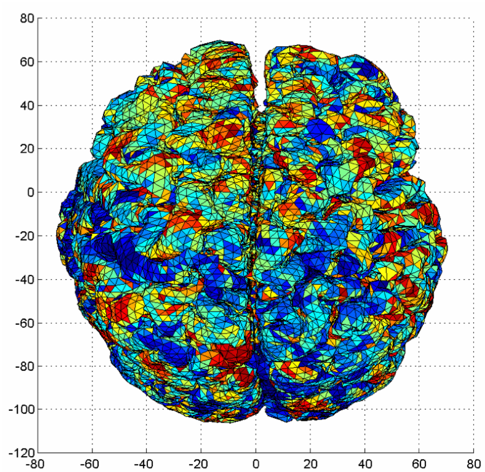



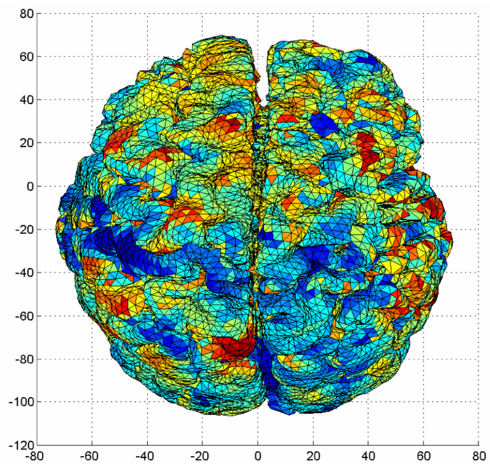
Fisher's
Z transform
on correlation



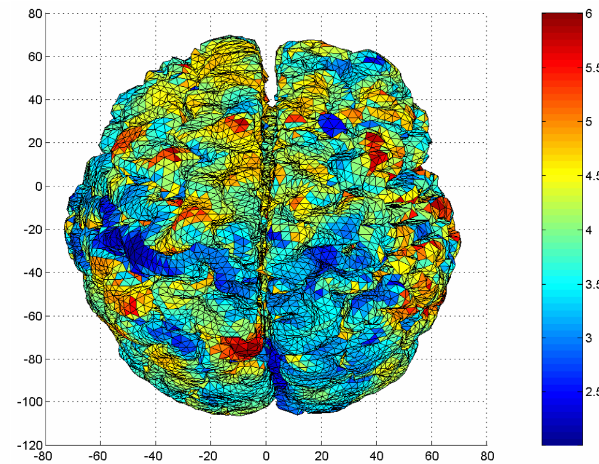
# Increasing normality in heat kernel smoothing



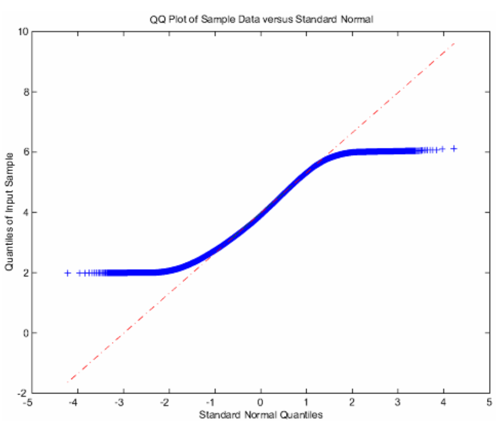
Thickness



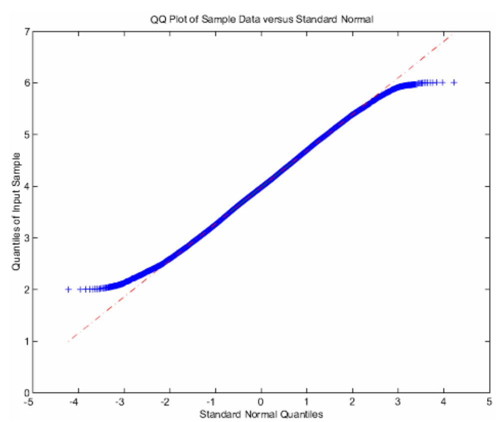
50 iterations



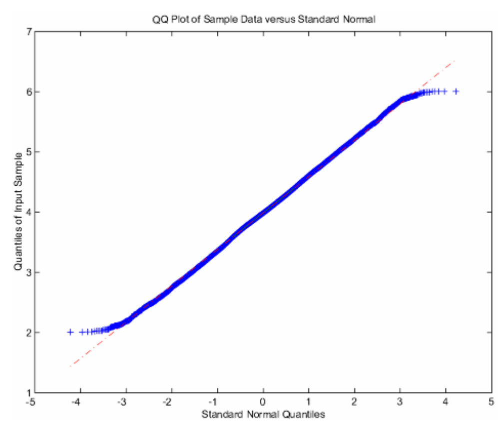
100 iterations




QQ-plot




QQ-plot




QQ-plot