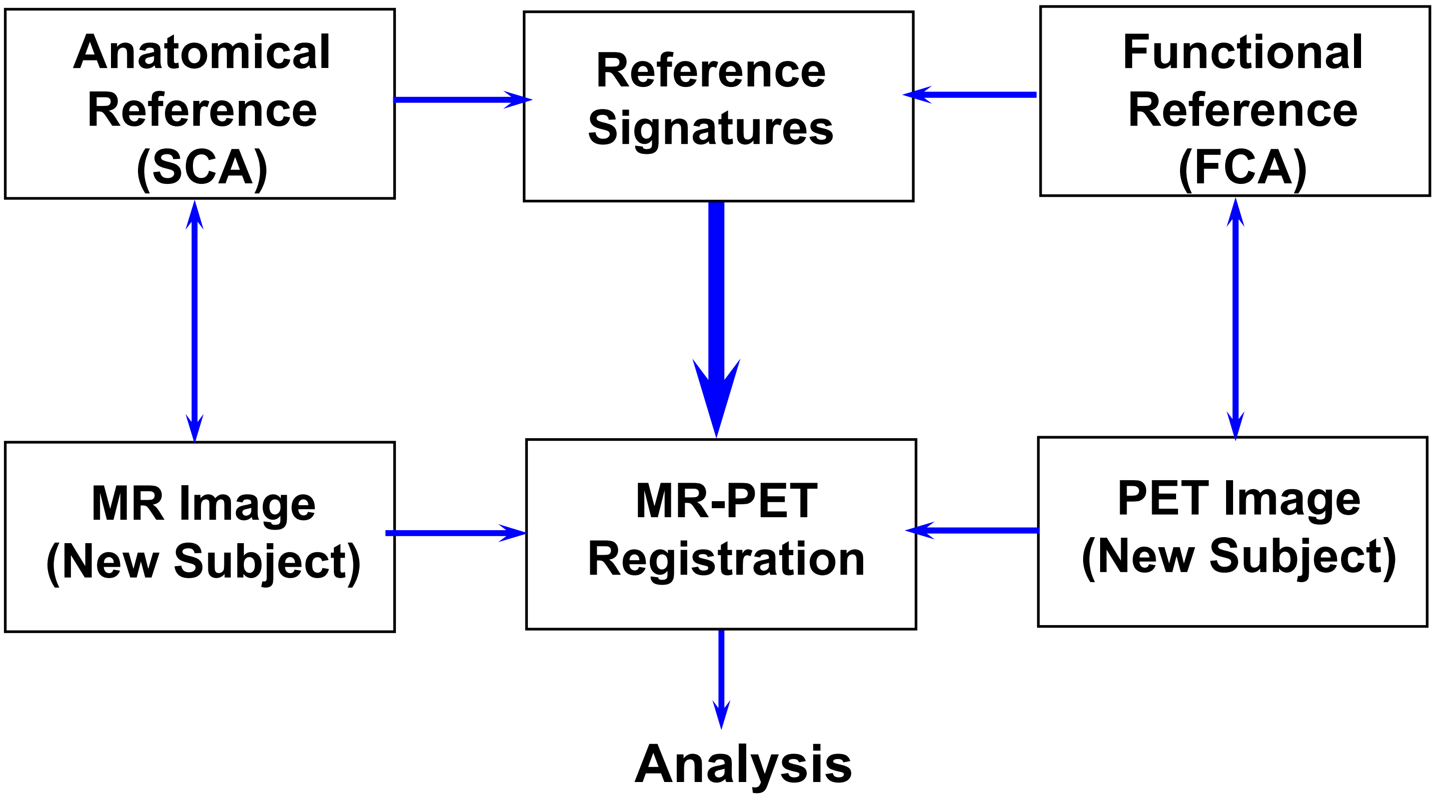
ENG4BF3
Medical Image Processing

Image Registration

McMaster
University

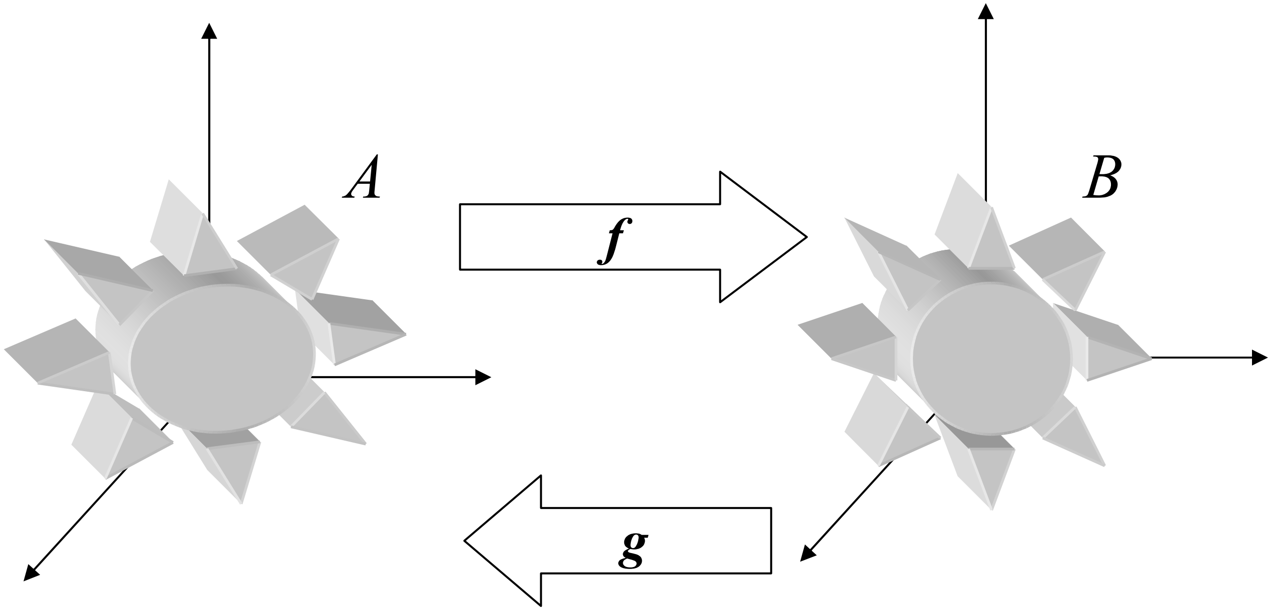
What is image registration?

- Image Registration is the process of estimating an optimal transformation between two images.
- Sometimes also known as “Spatial Normalization”.
- Applications in medical imaging
  - Matching PET (metabolic) to MR (anatomical) Images
  - Atlas-based segmentation/brain stripping
  - fMRI Specific
    - Motion Correction
    - Correcting for Geometric Distortion in EPI
    - Alignment of images obtained at different times or with different imaging parameters
    - Formation of Composite Functional Maps



A schematic diagram of multi-modality MR-PET image analysis using computerized atlases.

Image Registration Through Transform



- Image registration provides transformation of a source image space to the target image space.
- The target image may be of different modalities from the source one.

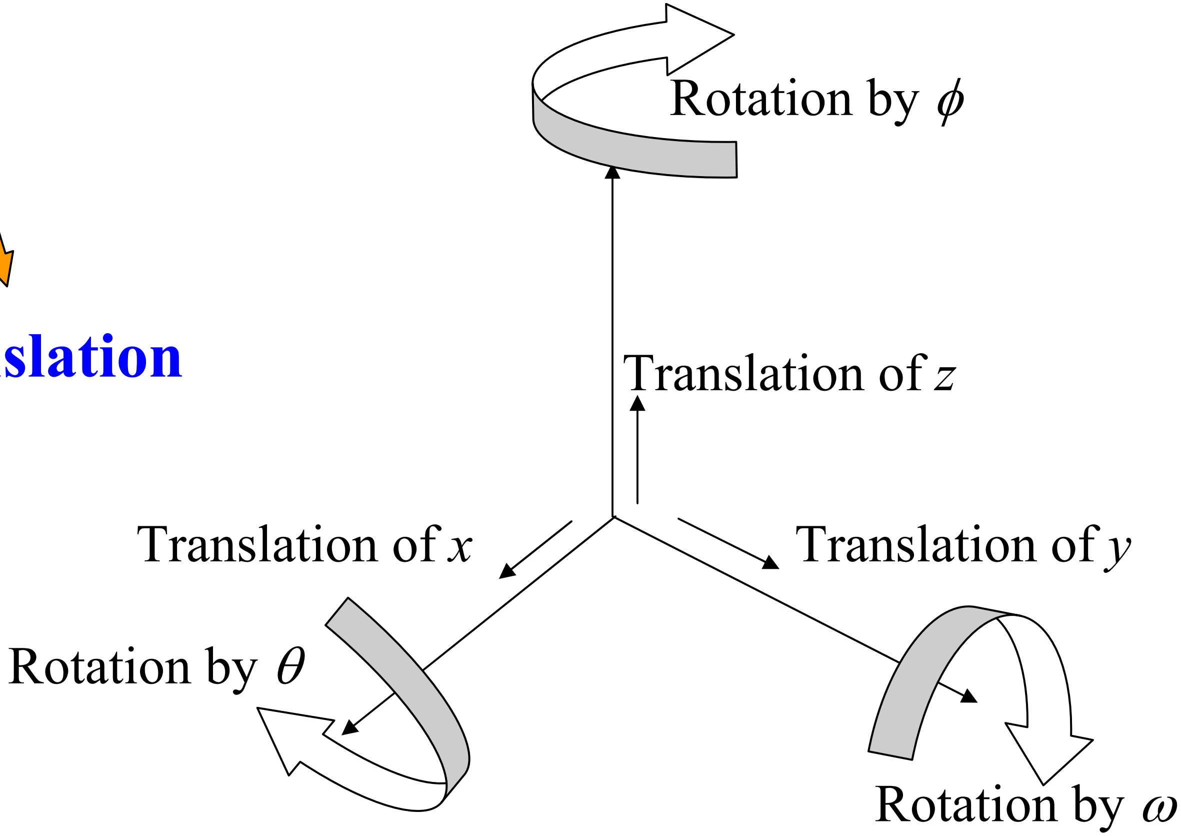
Multi-modality brain image registration

- External Markers and Stereotactic Frames Based Landmark Registration.
- Rigid-Body Transformation Based Global Registration.
- Image Feature Based Registration.
    - Boundary and Surface Matching Based Registration
    - Image Landmarks and Features Based Registration

Rigid-Body Transformation

$$\boldsymbol{x}' = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}$$

Rotation **Translation**



Rigid-Body Transformation

(a) Translation along x-axis by p

$$x' = x + p$$
$$y' = y$$
$$z' = z$$

(b) Translation along y-axis by q

$$x' = x$$
$$y' = y + q$$
$$z' = z$$

(c) Translation along z-axis by r

$$x' = x$$
$$y' = y$$
$$z' = z + r$$

McMaster University

Rigid-Body Transformation

(a) Rotation about x-axis by θ

$$x' = x + p$$
$$y' = y\cos\theta + z\sin\theta$$
$$z' = -y\sin\theta + z\cos\theta$$

(b) Rotation about y-axis by ω

$$x' = x\cos\omega - z\sin\omega$$
$$y' = y$$
$$z' = x\sin\omega + z\cos\omega$$

(c) Rotation about z-axis by φ

$$x' = x\cos\varphi + y\sin\varphi$$
$$y' = -x\sin\varphi + y\cos\varphi$$
$$z' = z$$

Rigid-Body Transformation

$$\boldsymbol{x}^{'} = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}$$

$$\boldsymbol{R} = R_\theta R_\omega R_\varphi =$$

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \cos\omega & 0 & -\sin\omega \\ 0 & 1 & 0 \\ \sin\omega & 0 & \cos\omega \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{bmatrix}$$

$$\boldsymbol{t} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Rigid-Body Transformation

$$\boldsymbol{x}^{'} = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}$$

$$\Downarrow$$

$$\begin{bmatrix} \boldsymbol{x}^{'} \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}$$

Rigid-Body Transformation: Affine Transform

Affine transform: translation + rotation + scaling

Scaling:

$$x' = ax$$

$$y' = by$$

$$z' = cz$$

***a*, *b* and *c* are scaling parameters in the three directions.**

Affine transform can be expressed as

$$\begin{bmatrix} \boldsymbol{x}' \\ 1 \end{bmatrix} = \boldsymbol{A} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}$$

Rigid-Body Transformation: Affine Transform

$$\begin{bmatrix} \boldsymbol{x}' \\ 1 \end{bmatrix} = \boldsymbol{A} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}$$

A: The affine transform matrix that integrates translation, rotation and scaling effects.

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{S} & 0 \\ \boldsymbol{0} & 1 \end{bmatrix} \quad \textbf{where} \quad \boldsymbol{S} = \begin{bmatrix} a & & \\ & b & \\ & & c \end{bmatrix}$$

Rigid-Body Transformation: Affine Transform

The overall mapping can be expressed as

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & p \\ 0 & 1 & 0 & q \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\omega & 0 & -\sin\omega & 0 \\ 0 & 1 & 0 & 0 \\ \sin\omega & 0 & \cos\omega & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi & 0 \\ 0 & -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rigid-Body Transformation: Principal Axes Registration

Principal axes registration (PAR) is used for global matching of binary volumes from CT, MR or PET images.

$$B(x,y,z) = \begin{cases} 1 & (x,y,z) \in object \\ 0 & (x,y,z) \notin object \end{cases}$$

$(x_g, y_g, z_g)^T$ → **centroid of** $B(x,y,z)$

$$x_g = \frac{\sum_{x,y,z} xB(x,y,z)}{\sum_{x,y,z} B(x,y,z)} \quad y_g = \frac{\sum_{x,y,z} yB(x,y,z)}{\sum_{x,y,z} B(x,y,z)} \quad z_g = \frac{\sum_{x,y,z} zB(x,y,z)}{\sum_{x,y,z} B(x,y,z)}$$

Rigid-Body Transformation: Principal Axes Registration

The principal axes of $B(x,y,z)$ are the eigenvector of inertia matrix I:

$$I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix}$$

$$I_{xx} = \sum_{x,y,z} \left[(y - y_g)^2 + (z - z_g)^2 \right] B(x,y,z)$$

$$I_{yy} = \sum_{x,y,z} \left[(x - x_g)^2 + (z - z_g)^2 \right] B(x,y,z)$$

$$I_{zz} = \sum_{x,y,z} \left[(x - x_g)^2 + (y - y_g)^2 \right] B(x,y,z)$$

$$I_{xy} = I_{yx} = \sum_{x,y,z} \left[(x - x_g)(y - y_g) \right] B(x,y,z)$$

$$I_{xz} = I_{zx} = \sum_{x,y,z} \left[(x - x_g)(z - z_g) \right] B(x,y,z)$$

$$I_{yz} = I_{zy} = \sum_{x,y,z} \left[(y - y_g)(z - z_g) \right] B(x,y,z)$$

Rigid-Body Transformation: Principal Axes Registration

- ☐ The inertia matrix *I* is diagonal when computed with respect to the principal axes.
- ☐ The centroid and principal axes can describe the orientation of a volume.
- ☐ The principal axes registration can resolve six degrees of freedom of an object (three rotation and three translation).
- ☐ It can compare the orientations of two binary volumes through rotation, translation and scaling.

Rigid-Body Transformation: Principal Axes Registration

Normalize the principal axes (the eigenvectors of *I*) and define:

$$E = \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix}$$

The rotation matrix is

$$\boldsymbol{R} = R_\theta R_\omega R_\varphi =$$

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\omega & 0 & -\sin\omega \\ 0 & 1 & 0 \\ \sin\omega & 0 & \cos\omega \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{bmatrix}$$

Rigid-Body Transformation: Principal Axes Registration

Let

$$E = \boldsymbol{R}$$

We can resolve the rotation angles as

$$\omega = \arcsin(e_{31})$$

$$\theta = \arcsin(-e_{21} / \cos\omega)$$

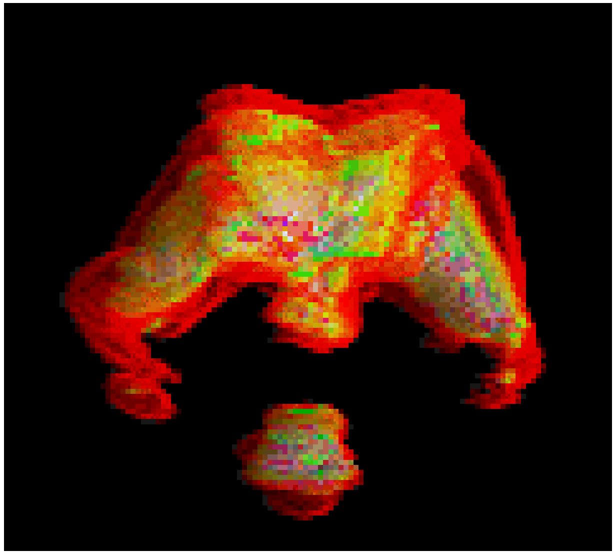
$$\varphi = \arcsin(-e_{32} / \cos\omega)$$

Rigid-Body Transformation: Principal Axes Registration

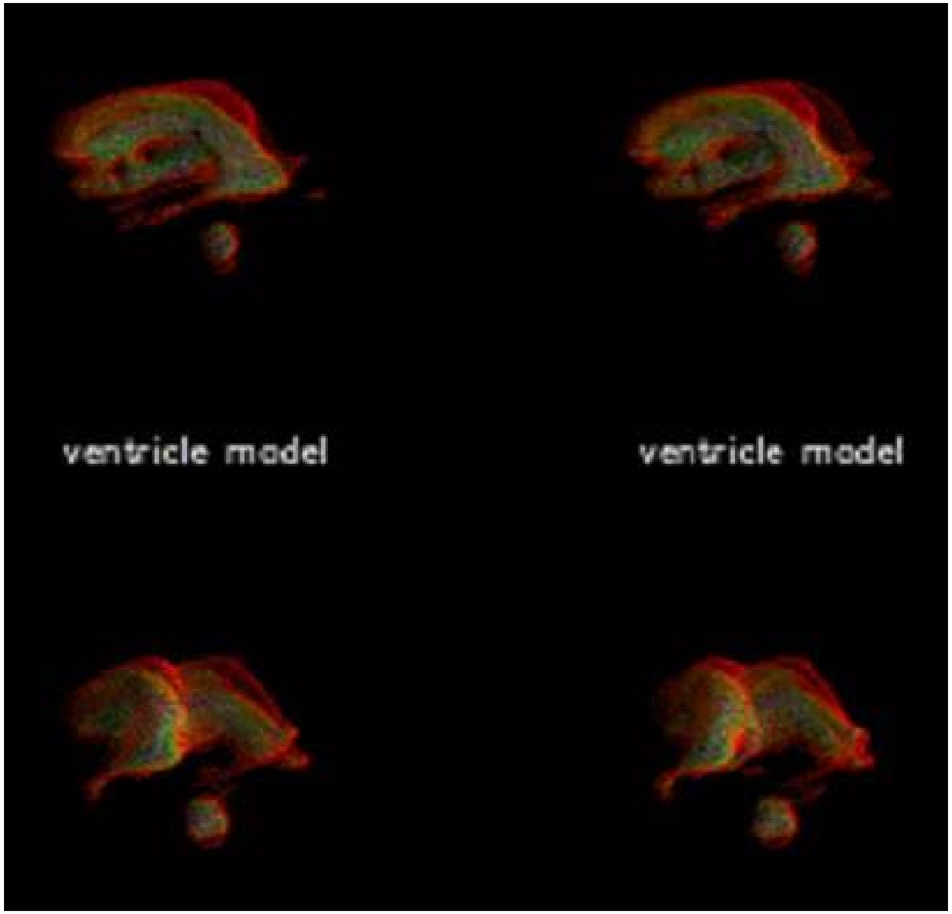
The PAR algorithm to register V1 and V2:

- 1. Compute the centroid of V1 and translate it to the origin.
- 2. Compute the principal axes (by using ***E*=*R***) of V1 and rotate V1 to coincide with the x, y and z axes.
- 3. Compute the principal axes of V2 and rotate the x, y and z axes to coincide with it.
- 4. Translate the origin to the centroid of V2.
- 5. Scaling V2 to match V1 by using factor $F_s = \sqrt[3]{V1/V2}$.

Rigid-Body Transformation: Principal Axes Registration



A 3-D model of brain ventricles obtained from registering 22 MR brain images using the PAR method.



Rotated views of the 3-D brain ventricle model in the left image.

McMaster University

Iterative Principal Axes Registration (IPAR)

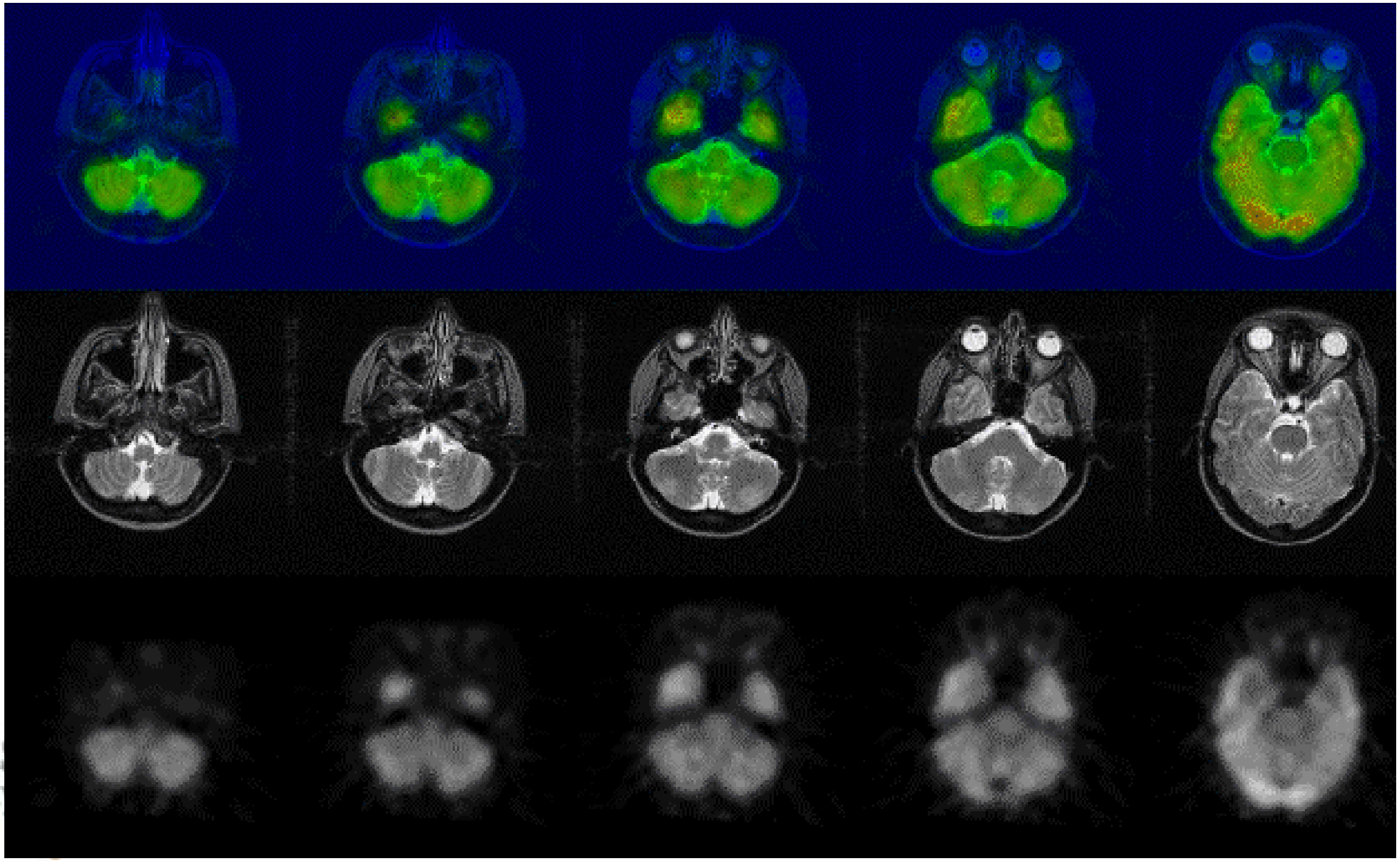
Advantages over PAR: IPAR can be used with partial volumes.

Assumption made in IPAR: the field of view (FOV) of a functional image (such as PET)) is less than the full brain volume while the other volume (such as MR image) covers the entire brain.

Algorithm: refer to pages 259 ~ 264 of the textbook.

Iterative Principal Axes Registration (IPAR)

Sequential slices of MR (middle rows) and PET (bottom rows) and the registered MR-PET brain images (top row) using the IPAR method.



Landmarks and Features Based Registration

- With the corresponding landmarks/features identified in the source and target images, a transformation can be computed to register the images.
- Non-rigid transformations have been used in landmarks/features based registration by exploiting the relationship of corresponding points/features in source and target images.
- Two point-based algorithms will be introduced.

Similarity Transformation

X: source image

Y: target image

***x*: landmark points in X**

***y*: landmark points in Y**

***T*(*x*): non-rigid transformation**

$$\boldsymbol{x}^{'} = T(\boldsymbol{x}) = \underset{\substack{\swarrow \\ \textbf{scaling}}}{s} \cdot \underset{\substack{\downarrow \\ \textbf{rotation}}}{\boldsymbol{r}} \cdot \boldsymbol{x} + \underset{\substack{\downarrow \\ \textbf{translation}}}{\boldsymbol{t}}$$

Similarity Transformation

The registration error:

$$E(\boldsymbol{x}) = T(\boldsymbol{x}) - \boldsymbol{y} = s\boldsymbol{r}\boldsymbol{x} + \boldsymbol{t} - \boldsymbol{y}$$

The optimal transform is to find s, $\boldsymbol{r}$ and $\boldsymbol{t}$ to minimize:

$$\sum_{i=1}^{N} w_i^2 \left| s\boldsymbol{r}\boldsymbol{x}_i + \boldsymbol{t} - \boldsymbol{y}_i \right|^2$$

w_i: the weighting factor for landmark $\boldsymbol{x}_i$

N: the number of landmark points

Similarity Transformation: Algorithm

1. Set s=1.

2. Find $\boldsymbol{r}$ through the following steps:

2a) Compute the weighted centroids of $\boldsymbol{x}$ and $\boldsymbol{y}$.

$$\bar{\boldsymbol{x}} = \frac{\sum_{i=1}^{N} w_i^2 \boldsymbol{x}_i}{\sum_{i=1}^{N} w_i^2} \qquad \bar{\boldsymbol{y}} = \frac{\sum_{i=1}^{N} w_i^2 \boldsymbol{y}_i}{\sum_{i=1}^{N} w_i^2}$$

2b) Compute the distance of each landmark from the centroid.

$$\bar{\boldsymbol{x}}_i = \boldsymbol{x}_i - \bar{\boldsymbol{x}} \qquad \bar{\boldsymbol{y}}_i = \boldsymbol{y}_i - \bar{\boldsymbol{y}}$$

Similarity Transformation: Algorithm (continued)

2c) Compute the weighted co-variance matrix.

$$\boldsymbol{Z} = \sum_{i=1}^{N} w_i^2 \overline{\boldsymbol{x}}_i \overline{\boldsymbol{y}}_i^t$$

Singular value decomposition

$$\boldsymbol{Z} = \boldsymbol{U} \Lambda \boldsymbol{V}^t$$

where $\boldsymbol{U}\boldsymbol{U}^t = \boldsymbol{V}\boldsymbol{V}^t = \boldsymbol{I}$

$$\Lambda = diag(\lambda_1, \lambda_2, \lambda_3) \quad \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$$

2d) Compute

$$\boldsymbol{r} = \boldsymbol{V} \cdot diag(1, 1, \det(\boldsymbol{V}\boldsymbol{U})) \cdot \boldsymbol{U}^t$$

Similarity Transformation: Algorithm (continued)

3. Compute the scaling factor:

$$s = \frac{\sum_{i=1}^{N} w_i^2 \boldsymbol{r}\overline{\boldsymbol{x}}_i \overline{\boldsymbol{y}}_i}{\sum_{i=1}^{N} w_i^2 \boldsymbol{r}\overline{\boldsymbol{x}}_i \overline{\boldsymbol{x}}_i}$$

4. Compute

$$\boldsymbol{t} = \overline{\boldsymbol{y}} - s\boldsymbol{r}\overline{\boldsymbol{x}}$$

Iterative Features Based Registration (WFBR)

X_i: a data set representing a shape in source image

Y_i: the corresponding data set in target image

x_{ij}: points in X_i **y_{ij}: points in Y_i**

$T(x)$: transform operator

$$\boldsymbol{x}' = T(\boldsymbol{x}) = s \cdot \boldsymbol{r} \cdot \boldsymbol{x} + \boldsymbol{t}$$

Disparity function to be minimized:

$$d(T) = \sqrt{\sum_{i=1}^{N_s} \sum_{j=1}^{N_i} w_{ij}^2 \left\| T(\boldsymbol{x}_{ij}) - \boldsymbol{y}_{ij} \right\|^2}$$

N_s: number of shapes

N_i: number of points in shape X_i

WFBR: Iterative Algorithm

1. Determine T by the similarity transformation method.

2. Let $T^{(0)}=T$ and initialize the optimization loop for k=1 as

$$\boldsymbol{x}_{ij}^{(0)} = \boldsymbol{x}_{ij}$$

$$\boldsymbol{x}_{ij}^{(1)} = T^{(0)}(\boldsymbol{x}_{ij}^{(0)})$$

3. For the points in shape X_i, find the closest points in Y_i as

$$\boldsymbol{y}_{ij}^{(k)} = C_i(\boldsymbol{x}_{ij}^{(k)}, Y_i) \quad j = 1, 2, 3, \ldots, N_i$$

where C_i is the operator to find the closest point.

WFBR: Iterative Algorithm (continued)

4.Compute the transformation $T^{(k)}$ between $\{\boldsymbol{x}_{\mathrm{ij}}^{(0)}\}$ and $\{\boldsymbol{y}_{\mathrm{ij}}^{(k)}\}$ with the weights $\{w_{\mathrm{ij}}\}$ using the similarity transformation method.

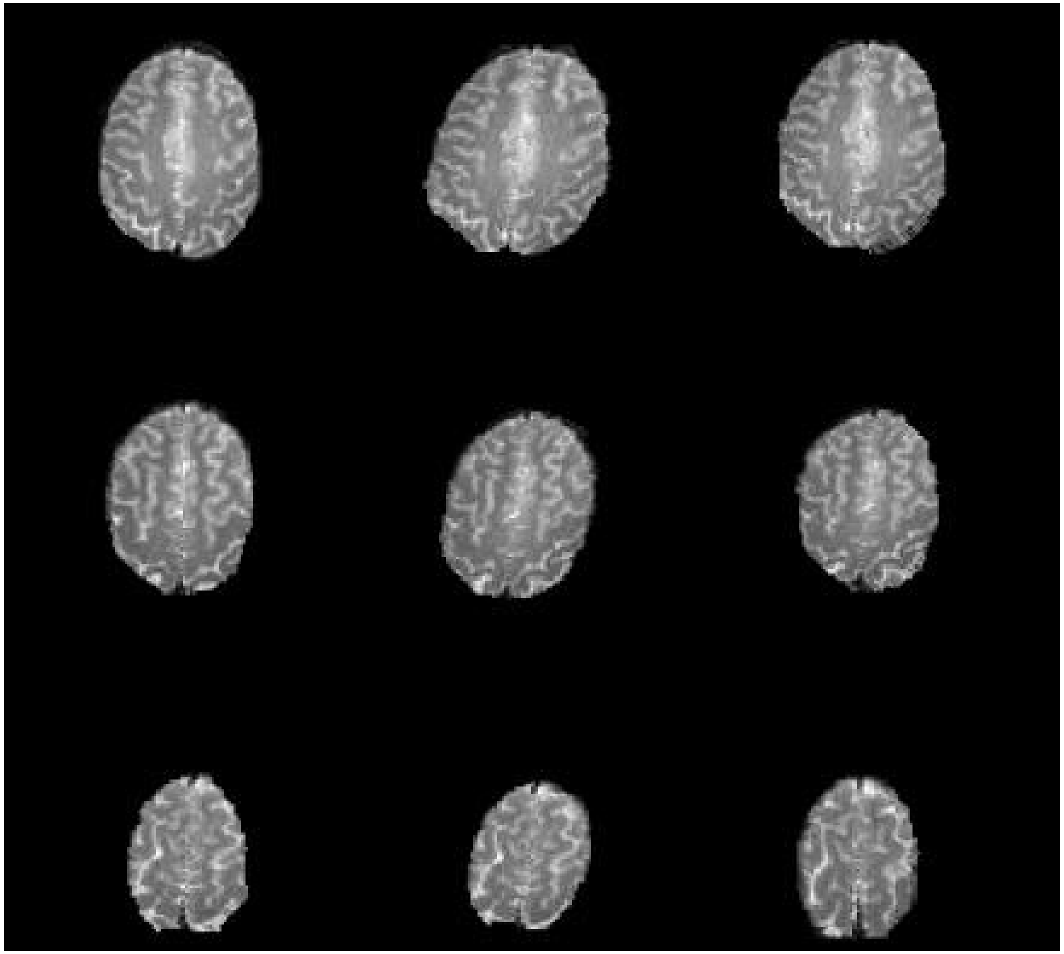
5. Compute

$$\boldsymbol{x}_{ij}^{(k+1)} = T^{(k)}(\boldsymbol{x}_{ij}^{(0)})$$

6. Compute $d(T^{(k)}) - d(T^{(k+1)})$. If the convergence criterion is met, stop; otherwise go to step 3 for the next iteration.

Elastic Deformation Based Registration

- ☐ One of the two volume is considered to be made of elastic material while the other serves as a rigid reference. Elastic matching is to map the elastic volume to the reference volume.
- ☐ The matching starts in a coarse mode followed by fine adjustments.
- ☐ The constraints in the optimization include smoothness and incompressibility.
- ☐ The smoothness ensures that there is continuity in the deformed volume while the incompressibility guarantee that there is no change in the total volume.
- ☐ For detail algorithm, refer to pp. 269 ~ 272 in the textbook.



Results of the elastic deformation based registration of 3-D MR brain images: The left column shows three reference images, the middle column shows the images to be registered and the right column shows the registered brain images.

End of Lecture