RIGID IMAGE REGISTRATION

Duygu Tosun-Turgut, Ph.D.
Center for Imaging of Neurodegenerative Diseases
Department of Radiology and Biomedical Imaging

duygu.tosun@ucsf.edu
What is image registration?

• Image registration is the process of estimating an optimal geometric transformation to bring homologous points of two images as close as possible.
Why image registration?

- Longitudinal analysis
- Motion correction
- Distortion correction
- Multi-modal integration
- Atlas generation
- Kinematic analysis
- Morphometry
Image registration categories

- Feature-based
- Intensity-based
How is intensity-based image registration done?

- Fixed image
- Moving image
- Interpolator
- Comparison
- Optimizer
- Parameters
- Transform
Fixed image
Moving image

• The moving image is the image that will be transformed and resampled in to the fixed image coordinate system
How do we represent an image?

The intensity at a given voxel $p$ of a 3D image $I$ is usually expressed as

$$I(r,c,z)$$
How do we represent an image?

- Always ask for the spatial resolution
- Medical image registration is done in physical coordinates, not in voxel coordinates
How do we represent an image?

• The center of the physical coordinate system is usually given by the scanner.
• Be aware that still many images are not isotropic.
• In the physical coordinate system we usually express the position of a point \( p \) as \([x \ y \ z]\).
How do we go from one coordinate system to another?

- We need a geometric transformation
- A geometric transformation maps coordinates from one coordinate system to another
- The transformation model is often described by its Degrees of Freedom (DoF), which is the number of independent ways that the transformation can be changed
- In general, increasing the number of DoF allows the transformations greater scope to make one image match the other
Rigid transformation

- In 3D it has 6 DoF:
  - 3 translations

\[
\begin{bmatrix}
  x' \\
  y' \\
  z' \\
  1
\end{bmatrix} =
\begin{bmatrix}
  1 & 0 & 0 & t_x \\
  0 & 1 & 0 & t_y \\
  0 & 0 & 1 & t_z \\
  0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  z \\
  1
\end{bmatrix}
\]
Rigid transformation

- In 3D it has 6 DoF:
  - 3 translations
  - 3 rotations

\[
M_z = \begin{bmatrix}
x' \\
y' \\
z' \\
1
\end{bmatrix} = \begin{bmatrix}
\cos(\theta) & -\sin(\theta) & 0 & 0 & 0 \\
\sin(\theta) & \cos(\theta) & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
x \\
y \\
z \\
1
\end{bmatrix}
\]
Rigid transformation

- In 3D it has 6 DoF:
  - 3 translations
  - 3 rotations

\[
\begin{bmatrix}
    x' \\
    y' \\
    z' \\
    1 \\
\end{bmatrix} =
\begin{bmatrix}
    \cos(\varphi) & 0 & -\sin(\varphi) & 0 \\
    0 & 1 & 0 & 0 \\
    \sin(\varphi) & 0 & \cos(\varphi) & 0 \\
    0 & 0 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
    x \\
    y \\
    z \\
    1 \\
\end{bmatrix}
\]
Rigid transformation

- In 3D it has 6 DoF:
  - 3 translations
  - 3 rotations

\[
\begin{bmatrix}
x' \\
y' \\
z' \\
1
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & \cos(\omega) & -\sin(\omega) & 0 & 0 \\
0 & \sin(\omega) & \cos(\omega) & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
z \\
1
\end{bmatrix}
\]
Affine transformation

- In 3D it has 9 DoF:
  - 3 translations
  - 3 rotations
  - 3 scalings

$$\begin{bmatrix}
x' \\
y' \\
z' \\
1
\end{bmatrix} = \begin{bmatrix}
s_x & 0 & 0 & 0 & 0 \\
0 & s_y & 0 & 0 & 0 \\
0 & 0 & s_z & 0 & 0 \\
0 & 0 & 0 & 1 & 0
\end{bmatrix}\begin{bmatrix}
x \\
y \\
z \\
1
\end{bmatrix}$$
Affine transformation

- In 3D it has 12 DoF:
  - 3 translations
  - 3 rotations
  - 3 scalings
  - 3 skew

\[
\begin{bmatrix}
  x' \\
  y' \\
  z' \\
  1
\end{bmatrix} =
\begin{bmatrix}
  1 & 0 & a & 0 \\
  0 & 1 & b & 0 \\
  0 & 0 & 1 & 0 \\
  0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  z \\
  1
\end{bmatrix}
\]

Shear along any pair of axes is proportional to the third axis e.g., to shear along z:
x and y are altered by an amount proportional to z, leaving z unchanged.

\[
x' = x + az \\
y' = y + bz \\
z' = z
\]
Affine transformation

- Maps lines to lines
- Maps parallel lines to parallel lines
- Preserves ratios of distance along a line
- Does NOT preserve absolute distances and angles
Transformations
Transformations

• In medical imaging, you usually need some combination of these basic transformations
• This is accomplished by concatenating the transformations:
Transformations

Matrix multiplication is not commutative!

\[ M_z M_T \neq M_T M_z \]
Nonlinear transformation

- The non-linear transformation model includes all transformations that do not fit into the affine transformation model.

- In 3D, it can go from those that are nearly linear with fed DoF to the most general transformations which have a separate displacement (3 DoF) for each voxel, giving well over a million DoF for typical images.
Rigid transformation example

- Intra-subject longitudinal registration
Rigid transformation example

- Intra-subject longitudinal registration
Affine transformation example

- Inter-subject registration

Fixed image
Model – Young females (n=94)

Moving image
No registration

Moving image
Rigid registration
Affine transformation example

- Inter-subject registration

Fixed image
Model – Young females (n=94)

Moving image
No registration

Moving image
Affine registration
Affine transformation example

- Inter-subject registration

Fixed image
Model – Young females
(n=94)

Moving image
No registration

Moving image
Nonlinear registration
Affine transformation example

- Inter-subject registration
How do we apply the transformation to get an aligned moving image?

- How to change from voxel coordinates to physical coordinates…
How do we apply the transformation to get an aligned moving image?

- To go from voxel coordinates to the physical coordinates system we need a transformation matrix that includes scalings and translations so that:

  \[ x = c \times \text{Spacing}_x + O_x \]
  \[ y = r \times \text{Spacing}_y + O_y \]

After the geometric transformation has been completed, change from the physical coordinate system to the fixed voxel space.
How do we apply the transformation to get an aligned moving image?

• Once in the physical space we can apply the geometric transformation to align the moving image to the fixed image.

• However, remember that the final goal is to have the moving image aligned to the fixed image in the space of the fixed image, so we have to take into account the mapping of the voxel intensities.
Forward mapping

- Moving image voxels are mapped onto the fixed image
- Multiple moving voxels could hit the same fixed image voxel:
  - The transformation has to keep track and accumulate this overlapping
- The registered moving image could have holes
  - Fixed image voxels with no hits
Backward mapping

- Fixed image coordinates are mapped back onto the moving image.
- Output voxel values must be interpolated from a neighborhood in the moving image.
- All fixed coordinates are scanned sequentially thus avoiding holes and overlaps in the output image.
- This can sometimes become a problem if the mapping function is non-invertible.
Parameters

- Image registration in medical imaging is commonly done with backward mapping requiring intensity interpolation.

- The parameters of the transformation are thus those that map each voxel position in the fixed image to the moving image.
Intensity interpolation

- Nearest neighbor
- Linear
- Spline
- Sinc
Linear interpolation

- Intensity at a given point is computed based on a weighted average of the surrounding voxels (e.g., 4 in 2D and 8 in 3D)

- Weights are calculated based on the distances between the coordinates of the point of interest and those of the surrounding voxels
Linear interpolation

\[ I_M(x, y) \]

\[ I_M(x + 1, y) \]

\[ I_M(x, y + 1) \]

\[ I_M(x + 1, y + 1) \]

\[ I(p) = (1 - \Delta x)(1 - \Delta y)I_M(x, y) + \Delta x(1 - \Delta y)I_M(x + 1, y) + (1 - \Delta x)\Delta yI_M(x, y + 1) + \Delta x\Delta yI_M(x + 1, y + 1) \]
Optimizer

• The optimizer helps us to find the optimal set of transformation parameters

• Different optimization strategies, e.g. Gradient Descent

• Optimization is usually an iterative process

• The optimizer needs a metric to tell it how well the images are aligned
Optimization metrics

- Intra-model intensity metrics
  - Mean-squared differences (MSD; cost function)
  - Absolute differences (MAD; cost function)
  - Normalized correlation (NCC; similarity function)

- Inter-modal intensity metrics
  - Mutual information (MI; similarity function)
  - Normalized mutual information (NMI; cost function)
  - Correlation ratio (CR; similarity function)
Optimization metrics

MSD

\[ MSD = \frac{1}{N} \sum_j \left( I_{F_j} - I_{T_j} \right)^2 \]

- F = Fixed image
- T = Transformed moving image
- N = Number of voxels in the overlapping region
- H(.) = Marginal entropy
- H(.,.) = Joint entropy
Optimization metric MI

\[ MI = H(I_F) + H(I_T) - H(I_F, I_T) \]

\[ H(I_F) = - \sum p_F(f) \log[p_F(f)] \]

\[ H(I_T) = - \sum p_T(f) \log[p_T(f)] \]

\[ H(I_F, I_T) = - \sum \sum p_{FT}(f, t) \log[p_{FT}(f, t)] \]

- F = Fixed image
- T = Transformed moving image
- N = Number of voxels in the overlapping region
- H(.) = Marginal entropy
- H(.,.) = Joint entropy
Multi-resolution

• Improves speed
• Improves robustness
• Improves accuracy
Nonlinear registration

- Basis functions
- Splines
- Elastic-solid
- Viscous fluid
- Smoothed displacement fields
Nonlinear registration - FFD
Nonlinear registration - FFD