Deformable Image Registration
Using B-Splines

Gregory C Sharp
Massachusetts General Hospital

February 10, 2011
“No conflict of interest”
Acknowledgements

• Thanks to James Shackleford for slides
• NIH U54 EB005149
• NIH 6-P01 CA 21239
• NIH/MGH C06CA059267
• Ira J Spiro Translational Research Grant
• NSF EEC-0946463
Outline

• Theory of B-spline registration
  • What are B-splines?
  • How are they optimized in registration?

• Using B-splines and open-source for ART
  • Plastimatch – registration
  • 3D Slicer and vv – visualization
  • CERR – plan review
B-splines

- B-spline is short for “basis spline”
- A function is represented as a linear combination of basis functions

\[ v(x) = \sum_{i} p_i \beta_i(x) \]
Cubic B-splines

\[ \beta(u) \] is a piecewise cubic polynomial

\[
\begin{align*}
\beta(u) &= \frac{3u^3 - 6u^2 + 4}{6} \\
&\quad \text{for } u \in (0, 1) \\
\beta(u) &= \frac{-3u^3 + 3u^2 + 3u + 1}{6} \\
&\quad \text{for } u \in (1, 2) \\
\beta(u) &= \frac{u^3}{6} \\
&\quad \text{for } u \in (2, 3)
\end{align*}
\]
Cubic B-splines

$P_i$ is a scaling factor
Uniform cubic B-splines

\[ v(x) = \sum_{i} p_i \beta_i(x) \]
Uniform cubic B-splines

$$v(x) = \sum_{i} p_i \beta_i(x)$$
B-splines for vector fields
Optimizing B-splines for image registration
Fixed and moving images

- Vector field is defined on **fixed image**
  - A.k.a. Reference image, static image
- Vector field maps fixed image to **moving image**
  - A.k.a. Test image, target image
Fixed and moving images

- Vector field is defined on \textit{fixed image}
  - A.k.a. Reference image, static image
- Vector field maps fixed image to \textit{moving image}
  - A.k.a. Test image, target image
Image similarity metric

- Sum of Squared Difference (SSD)
  - A.k.a. MSE, RMS, ...
- F is fixed image, M is moving image
- \( i = (x,y,z) \) is a voxel location in fixed image
- \( v \) is the vector field

\[
C = \sum_{i=(x,y,z)} [F(i) - M(i + v(i))]^2
\]
**INPUTS**

- Static Image
- Moving Image

**ITERATIVE REGISTRATION PROCESS**

- Cost ($C$)
- Image Difference
- Deformation Field ($v$)
- B-Spline Coefficients ($P$)

Legend:
- Control Point Quantity
- Voxelized Quantity
- Optimizer
- Function
Cost function gradient

- **Sum of Squared Difference (SSD)**

\[ C = \sum_{i=(x, y, z)} \left[ F(i) - M(i + v(i)) \right]^2 \]

- We want to optimize the B-spline coefficients:

\[ v(x) = \sum_i p_i \beta_i(x) \]

- Need the gradient of C with respect to \( P \):

\[ \frac{\partial C}{\partial P} = \frac{\partial C}{\partial v} \frac{\partial v}{\partial P} \]
INPUTS

- Static Image
- Moving Image

ITERATIVE REGISTRATION PROCESS

- Warped Image
- Deformation Field ($\nu$)
- B-Spline Coefficients ($P$)

Cost ($C$)

$\frac{\partial C}{\partial \nu}$

Voxelized Quantity

Control Point Quantity

Optimizer

Function
INPUTS

- Static Image
- Moving Image
- Moving Image Spatial Gradient

ITERATIVE REGISTRATION PROCESS

- Cost ($C$)
- Image Difference
- Warped Image
- Deformation Field ($\nu$)
- B-Spline Coefficients ($P$)

- Control Point Quantity
- Voxelized Quantity
- Optimizer
- Function
INPUTS

- Static Image
- Moving Image
- Moving Image Spatial Gradient

ITERATIVE REGISTRATION PROCESS

- Cost ($C$)
- Image Difference
- Deformation Field ($v$)

- B-Spline Coefficients ($P$)
- Voxelized Quantity
- Control Point Quantity
- Optimizer

$\frac{\partial C}{\partial v}$
$\frac{\partial C}{\partial P}$
$\frac{\partial v}{\partial P}$
INPUTS

Static Image
Moving Image
Moving Image Spatial Gradient

ITERATIVE REGISTRATION PROCESS

Warped Image
Deformation Field (v)
B-Spline Coefficients (P)

Cost (C)

∂C/∂v

∂C/∂P

Cubic B-Splines

Image Difference

Control Point Quantity
Voxelized Quantity
Optimizer
Function
INPUTS

- Static Image
- Moving Image
- Moving Image Spatial Gradient

ITERATIVE REGISTRATION PROCESS

- Cost ($C$)
- Image Difference
- $\frac{\partial C}{\partial v}$
- $\frac{\partial C}{\partial P}$

- Warped Image
- Deformation Field ($v$)
- B-Spline Coefficients ($P$)

- L-BFGS-B

- Cubic B-Splines

Legend:
- Control Point Quantity
- Voxelized Quantity
- Optimizer
- Function
Using B-Splines with open source tools for adaptive planning
Open source tools

• Plastimatch http://plastimatch.org
• 3D Slicer http://slicer.org
• VV http://www.creatis.insa-lyon.fr/rio/vv
• CERR http://radium.wustl.edu/CERR

• These tools are for research only, and have not been reviewed or approved by the FDA or any other agency
B-Spline registration

ct1 is original plan
ct2 is boost plan
Data is DICOM-RT
B-Spline registration

Fixed image → warping → Moving Image

Dose
Structure sets
B-Spline registration

Fixed image

CT2

warping

Moving Image
Dose
Structure sets...

CT1
B-Spline registration

GLOBAL
fixed=ct2
moving=ct1
img_out=warped.mha
xform_out=bspline.txt

STAGE
xform=translation
max_its=50
res=4 4 2

STAGE
xform=bspline
max_its=50
grid_spac=100 100 100
res=4 4 2
B-Spline registration

[GLOBAL]
fixed=ct2
moving=ct1
img_out=warped.mha
xform_out=bspline.txt

[STAGE]
xform=translation
max_its=50
res=4 4 2

[STAGE]
xform=bspline
max_its=50
grid_spac=100 100 100
res=4 4 2
B-Spline registration

- Run registration command: `plastimatch parms.txt`
- Output files are generated
Visualize the results
Visualize the results

- **3D Slicer Tips & Tricks**
  - Load all images CT1, CT2, warped image, warped dose, structure sets, etc. at the same time
  - Use color addition instead of alpha blending
Fixing bad registration results

- What to do if registration results are bad?
  - Check rigid registration stage
    - Consider manual rigid alignment
  - Use stiffer registration (coarser B-spline grid)
  - Use more registration stages (coarse → fine)
  - Post-process with landmark-based registration
    - http://plastimatch.org
Testing vector field smoothness

- Run command
  plastimatch stats vf.mha
- Output
  Jacobian: MINJAC 0.823 MAXJAC 1.20 MINABSJAC 0.82
- Jacobian measures local increase/decrease in volume
- Jacobian = 1 means volume exactly preserved
- Jacobian < 0 means vector field folding
Warping the dose

• Run command

   plastimatch warp --input ct2/dose.dcm
   --xf bspline.txt
   --output-dose-img warped-dose.mha
Dose manipulation

- Convert to DICOM-RT → Slicer
  `plastimatch convert --input ct1/dose.dcm --fixed warped-dose.mha --output dose1.mha`

- Adding doses
  `plastimatch add dose1.mha warped-dose.mha combined-dose.mha`

- Convert to Slicer → DICOM-RT
  `plastimatch convert --input-dose-img combined-dose.mha --output-dicom ct2`
Plan review in CERR

Tra: 80/160
z: 0.85cm
Plan review in CERR
Advertisement

- 3D Slicer user group at AAPM/COMP
  - Talk to me during the break
  - Or email gcsharp@partners.org
THANK YOU