Learning Based Coarse-to-fine Image Registration

J. Jiang, S. Zheng, A. W. Toga, Z. Tu, LONI, UCLA

CHEN-RUI CHOU
October 20, 2009
Outline

- Image registration overview
- Outline of the method
  - Rigid Registration
  - Salient structure extraction
  - Non-rigid registration

- Results
- Image registration overview
- Outline of the method
  - Rigid Registration
  - Salient structure extraction
  - Non-rigid registration
- Results
What is Image Registration?
What is Image Registration?

- To estimate a geometric transform so that two images can be aligned properly. [Jiang et al.]

Source Image

Target Image

D'Arcy Thompson
What is rigid / non-rigid registration?

- Rigid registration
  
  =>
  
  ⊡
  
  ⊡
  
  ⊡

- Non-Rigid registration

  =>
What is rigid / non-rigid registration?

- Rigid registration
  => move globally
  - Translation
  - Rotation
  - Scaling
- Non-Rigid registration
  =>
  - 
What is rigid / non-rigid registration?

- Rigid registration
  - move globally
  - Translation
  - Rotation
  - Scaling
- Non-Rigid registration
  - move locally
  - Fluid flow
What is multi-modal image registration?
What is multi-modal image registration?

(a) MRI T1 images; (b) MRI T2 images

(a) 2D brain image slice (b) template
Problem definition

- Register the source image (left) to the template (right) with a different modality.

(a) 2D brain image slice       (b) template
Outline

- Image registration overview
- Outline of the method
  - Rigid Registration
  - Salient structure extraction
  - Non-rigid registration
- Results
Outline of the method

- **Rigid Registration**
  - (1) register the image by computing the learned global similarity measure at coarse level
    - rotate and rescale to roughly the same size as template
    - similarity measure learned by Boosting
- **Salient structure extraction**
  - (2) extract edge to have the map points
    - learned by boosting edge learner (BEL)
- **Non-rigid Registration**
  - (3) computing the learned local similarity measure at fine level
    - based on edge points (image) and landmarks (template)
    - similarity measure learned by Boosting
  - (4) register by finding out the transformation
Outline of the method

- **Rigid Registration**
  - (1) register the image by computing the learned global similarity measure at coarse level
    - rotate and rescale to roughly the same size as template
    - similarity measure learned by Boosting
- **Salient structure extraction**
  - (2) extract edge to have the map points
    - learned by boosting edge learner (BEL)
- **Non-rigid Registration**
  - (3) computing the learned local similarity measure at fine level
    - based on edge points (image) and landmarks (template)
    - similarity measure learned by Boosting
  - (4) register by finding out the transformation
Rigid Registration

- **Input:** a source image and a template
- **Output:** an roughly aligned source image
- **Method:**
  - Search over many locations, scales, and orientations which best match the template
Input: a source image and a template
Output: an roughly aligned source image
Method:
- Search over many locations, scales, and orientations which best match the template
- Need a similarity measure
Rigid Registration

- Input: a source image and a template
- Output: an roughly aligned source image
- Method:
  - Search over many locations, scales, and orientations which best match the template
  - Need a similarity measure
  - Can be learned from Probabilistic boosting-tree (PBT) which is based on Boosting
    - Boosting [Freund and Schapire]  Next: what is boosting?
What is Boosting

- boosting = general method of converting rough rules of thumb into highly accurate prediction rule

Technically:
- assume given “weak” learning algorithm that can consistently find classifiers (“rules of thumb”) at least slightly better than random, say, accuracy $\geq 55\%$ (in two-class setting)
- given sufficient data, a boosting algorithm can provably construct single classifier with very high accuracy, say, 99%
- resistant to overfitting problem

\[
H(x) = \sum_{t=1}^{T} \alpha_t h_t(x)
\]
A Formal Description of Boosting

- given training set \((x_1, y_1), \ldots , (x_m, y_m)\)
- \(y_i \in \{-1, +1\}\) correct label of instance \(x_i \in X\)
- for \(t = 1, \ldots , T:\)
  - construct distribution \(D_t\) on \(\{1, \ldots , m\}\)
  - find weak classifier (“rule of thumb”) \(h_t : X \rightarrow \{-1, +1\}\)
    with small error \(\epsilon_t\) on \(D_t\):
      \[\epsilon_t = \Pr_{D_t}[h_t(x_i) \neq y_i]\]
  - output final classifier \(H_{\text{final}}\)

Let’s compare the parameters in our rigid registration
In our problem:  I₁: source’  I₂: template

- X: image pair (I₁, I₂)
- Y: \{+1,-1\}  
  - +1: similar; -1: not similar
  - ground truths
- weak classifiers from 40,000 features
  - Ex: [Viola] the first and the second features
- weak classifiers: h=?

\[
F_i(I_1, I_2) = f_i(I_1) - f_i(I_2) \\
h(F_i(I_1, I_2), tr) = \begin{cases} 
+1, & \text{if } F_i(I_1, I_2) \geq tr \\
-1, & \text{otherwise}
\end{cases}
\]
**AdaBoost**

- **constructing $D_t$:**
  - $D_1(i) = 1/m$
  - given $D_t$ and $h_t$:
    \[
    D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \left\{ \begin{array}{ll}
    e^{-\alpha_t} & \text{if } y_i = h_t(x_i) \\
    e^{\alpha_t} & \text{if } y_i \neq h_t(x_i)
    \end{array} \right.
    \]
    
    \[
    = \frac{D_t(i)}{Z_t} \exp(-\alpha_t y_i h_t(x_i))
    \]

  where $Z_t = \text{normalization constant}$

  \[
  \alpha_t = \frac{1}{2} \ln \left( \frac{1 - \epsilon_t}{\epsilon_t} \right) > 0
  \]

- **final classifier:**
  - $H_{\text{final}}(x) = \text{sign} \left( \sum_t \alpha_t h_t(x) \right)$

Next slides: Examples!
Toy Example

weak classifiers = vertical or horizontal half-planes
Detour: Boosting

Round 1

\[ \alpha_1 = 0.42 \]

\[ \epsilon_1 = 0.30 \]

\[ D_2 \]

\[ h_1 \]

\[ \alpha_t = \frac{1}{2} \ln \left( \frac{1 - \epsilon_t}{\epsilon_t} \right) > 0 \]

[credit: Schapire]
Detour: Boosting

Round 2

\[ \varepsilon_2 = 0.21 \]
\[ \alpha_2 = 0.65 \]

\[ \alpha_t = \frac{1}{2} \ln \left( \frac{1 - \varepsilon_t}{\varepsilon_t} \right) > 0 \]

[credit: Schapire]
Round 3

$$\alpha_t = \frac{1}{2} \ln \left( \frac{1 - \epsilon_t}{\epsilon_t} \right) > 0$$

$$\varepsilon_3 = 0.14$$
$$\alpha_3 = 0.92$$
Detour: Boosting

**Final Classifier**

\[ H_{\text{final}} = \text{sign} \left( 0.42 + 0.65 + 0.92 \right) \]

That’s boosting, but what is Probabilistic boosting-tree?
An extension of Boosting

Build a learning tree
- Each node is a strong classifier which combines a set of weak classifiers (boosting)

Divide and conquer strategy
- Easy classifiers on the top levels
- Hard classifiers on the lower levels

Better performance

Next: illustration: how it works
Detour: PBT

Strong classifier

Feature 1,..,k

Feature k+1,…m

Uses feature differences as weak classifiers

To learn:
The location of yellow node of our desired result => need positive/negative training sets
- Examples.
- Positive Training set for the similarity measure
Rigid Registration

- **Input:** a source image and a template
- **Output:** an roughly aligned source image
- **Method:**
  - Search over many locations, scales, and orientations to match the template.
  - Computing the similarity measure and choose the images with the highest score

NEXT: Results
Rigid Registration

source img  template img  registered source img
Outline of the method

- **Rigid Registration**
  - (1) register the image by computing the learned global similarity measure at coarse level
    - rotate and rescale to roughly the same size as template
    - similarity measure learned by Boosting

- **Salient structure extraction**
  - (2) extract edge to have the map points
    - learned by boosting edge learner (BEL)

- **Non-rigid Registration**
  - (3) computing the learned local similarity measure at fine level
    - based on edge points (image) and landmarks (template)
    - similarity measure learned by Boosting
  - (4) register by finding out the transformation
- Take edges as salient structure
  - Input: an image
  - Output: edges of the image
  - How to do the edge extraction?
    - 
Take edges as salient structure

- Input: an image
- Output: edges of the image
- How to do the edge extraction?
  - Canny’s
Take edges as salient structure

- Input: an image
- Output: edges of the image
- How to do the edge extraction?
  - Canny’s
    - Problems?
- **Take edges as salient structure**
  - **Input:** an image
  - **Output:** edges of the image
  - **How to do the edge extraction?**
    - Canny’s
      - Problems—
        - Too many non-informative edges
        - =>try an supervised learning method
        - Boosted Edge Learner
[Dollar, et al.]

- the algorithm selects and combines a large number of features across different scales in order to learn a discriminative model using Probabilistic Boosting Tree.
PBT flash back!

Strong classifier

Feature 1,...,k

Feature k+1,...,m

Now!
Features are designed to determine whether or not there is an edge in the center of the patch.
BEL learns in small patches

For each patch, a strong classifier uses several weak classifiers (features) to report if the edges is in the center.

negatives

positives

NO

YES
Result
Outline of the method

- Rigid Registration
  - (1) register the image by computing the learned global similarity measure at coarse level
    - rotate and rescale to roughly the same size as template
    - similarity measure learned by Boosting
- Salient structure extraction
  - (2) extract edge to have the map points
    - learned by boosting edge learner (BEL)
- Non-rigid Registration
  - (3) computing the learned local similarity measure at fine level
    - based on edge points (image) and landmarks (template)
    - similarity measure learned by Boosting
  - (4) register by finding out the transformation
Non-rigid registration

- **Input:** coarsely registered source image and template
- **Output:** correspondence points
- **Method:**
  - Detect salient edge points - BEL
  - Analyze the patches centered on the edge points
  - Compute the similarity measure on those patches
    - Similarity measure learned from PBT
Non-rigid registration

Patches centered at edge points

Positive training set

Negative training set
Non-rigid registration

PBT flash back!

Strong classifier

Feature 1,…,k

Feature k+1,…,m

Again, Features are like Haar feature, angle of derivative,…

\[ F_i(I_1, I_2) = f_i(I_1) - f_i(I_2) \]

\[ h(F_i(I_1, I_2), tr) = \begin{cases} 
+1, & \text{if } F_i(I_1, I_2) \geq tr \\
-1, & \text{otherwise}
\end{cases} \]
Non-rigid registration

- **Results**

**Input:** Given an point on template

**Output:** a patch with the highest similarity score

Next: more results
Non-rigid registration

- Result--Match pairs based on learned similarity measure

Source | Template | mapping
Non-rigid registration

- Result--Match pairs based on learned similarity measure

Source  Template  mapping

what's wrong?
Because we measure in patches—locally they look similar to positive training pairs

Solutions?
Non-rigid registration

- Because we measure in patches—locally they look similar to positive training pairs

- Solutions?
  - use RANSAC to find the affine transform in order to remove outliers

Next: How RANSAC works?
RANSAC

Do Voting

Source image

Template
Source image

Template

Outlier
Results

Before | after RANSAC | Before | after RANSAC

![Images showing results before and after RANSAC]
Non-rigid registration

- How to find out the registration transformation $\phi$ by matched pair of patches?
Non-rigid registration

- How to find out the registration transformation $\phi$ by matched pair of patches?
  - Thin plate splines (two dim. cubic splines)
  - Non-rigid
Outline

- Image registration overview
- Outline of the method
  - Rigid Registration
  - Salient structure extraction
  - Non-rigid registration
- Results
Review

- **Rigid Registration**
  - (1) register the image by computing the learned global similarity measure at coarse level
    - rotate and rescale to roughly the same size as template
    - similarity measure learned by Boosting
  - **Salient structure extraction**
    - (2) extract edge to have the map points
      - learned by boosting edge learner (BEL)
  - **Non-rigid Registration**
    - (3) computing the learned local similarity measure at fine level
      - based on edge points (image) and landmarks (template)
      - similarity measure learned by Boosting
    - (4) register by finding out the transformation
Results

<table>
<thead>
<tr>
<th>Sources</th>
<th>Templates</th>
<th>After Registration</th>
<th>Overlay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example 1</td>
<td>Example 2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(a)  (b)  (c)  (d)
Results

(a) source    (b) template    (c) registered    (d)
Questions?