Deformable Templates

Pedro F. Felzenszwalb
University of Chicago
Motivation

• Similar objects can be deformed into each other.

• We can represent a family of objects in terms of deformations of an ideal template.

• Compact representation for highly variable objects classes.

[Images of fish and rabbit with labels for template and deformations]
Examples

Deformations of a leaf shape

Deformations of a face image
Deformable matching

- Given a template and an image.
- Find an “optimal” map taking the template to the image.

\[ f : \quad \rightarrow \]

- Quality of map depends on
  - How much the template is deformed.
  - Correlation between deformed template and image.
  - Interpretation in terms of statistical inference...
Example: finding a butternut squash

template

optimal match
Object recognition

- Simple nearest neighbor approach:
  - Find minimum deformation necessary to transform object into one of the stored examples.

Using a good (shared) deformation model we get high classification accuracy with few training examples.
Main issues

• How to represent the template?
  - Boundary curve, grayscale image, constellation of parts.
  - Want to capture both boundary and interior of objects.

• How to model deformations?
  - Need to capture which are “good” deformations.
  - Can we learn deformation model from examples?

• Efficient matching algorithms.
  - Find optimum deformations between objects.
  - No initialization, invariant to rigid motions and scaling.
Deformable boundaries

- Polygonal boundary model.
- Consider deformations which preserve local properties.
  - angles, ratio of lengths.
- Measure deformation cost in terms of bending and stretching.
Part-based models

- Pictorial structures / Constellation of parts
- Rigid parts arranged in a deformable configuration.
  - Each part represents local visual properties.
  - Spatial configuration captured by statistical model or spring-like connections.
- Good matching algorithms
  - Using dynamic programming and distance transforms.

(template image)
Body template
2D objects

- Represent object using a polygon P.
- A triangulation gives a decomposition of P into parts.
  - Captures both boundary and region information.
- Triangles are connected together in a tree structure.
  - Shape of object “factors” in a good way.
  - Good computational properties (Amit and Kong, 1996).
Deformation model

- Piecewise affine deformations, taking triangles to triangles.
- Measure deformation of template in terms of a sum of deformation costs per triangle.
- Use a generic deformation model for each triangle, or learn model using multiple training examples.
Matching

There is an efficient dynamic programming algorithm for finding optimal deformations.
Coarse-to-fine matching

Detection of optimal convex objects around reference points


<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Running time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic programming</td>
<td>5936 seconds</td>
</tr>
<tr>
<td>coarse-to-fine DP (Raphael, 2001)</td>
<td>15 seconds</td>
</tr>
<tr>
<td>hierarchical A*</td>
<td>10 seconds</td>
</tr>
</tbody>
</table>
Discussion

- Deformable templates give a simple and compact representation of highly variable objects.
- We can use a generic deformation model to represent large families of objects using a few examples from each class.
- Efficient matching algorithms.
  - Exploit structure in classes of models.
  - Can find optimal match of model to image quickly.
  - Robust to occlusion, noise, etc.