Shape Representation and Similarity

Occlusion

Articulation

Shapes can be represented by contours, or by a set of interior points, or other ways. What is the shape representation that can be best used for shape similarity?

Shape similarities should be preserved under occlusions and articulation.

Two equal stretching along contours can affect shape similarity differently.
Symmetry Axis Representation

Shape

Shape Axis (SA)

SA-Tree
Shape Representation via Self-Similarity

Based on work of Liu, Kohn and Geiger

◆ A \textit{variational} shape representation model based on \textit{self-similarity} of shapes.

◆ For each shape contour, first compute its \textit{shape axis} then derive a unique \textit{shape-axis-tree (SA-tree)} or \textit{shape-axis-forest (SA-forest)} representation.
The Insight

- Use *two* different parameterizations to compute the representation of a shape.
- Construct a cost functional to measure the *goodness* of a match between the two parameterizations.
- The cost functional is decided by the self-similarity criteria of *symmetry*.
Parameterized Shapes

- Two different parameterizations:
  \[ \Gamma_1 = \{ x(s) : 0 \leq s \leq 1 \} \quad \text{counterclockwise} \]
  \[ \Gamma_2 = \{ \tilde{x}(t) : 0 \leq t \leq 1 \} \quad \text{clockwise} \]
  
  \( \text{Note: } \tilde{x}(t) = x(1 - t) \)

- When the curve is closed we have
  \[ x(0) = x(1) \quad \text{and} \quad \tilde{x}(0) = \tilde{x}(1) \]

- Matching the curves is described as the match of functions
  \[ s(\sigma) \leftrightarrow t(\sigma) ; \sigma = [0,1] \quad \text{or} \quad x(s(\sigma)) \leftrightarrow \tilde{x}(t(\sigma)) \]
A Global Optimization Approach

- We seek “good” matches over the possible correspondences

\[ s(\sigma) \leftrightarrow t(\sigma) \] or \[ x(s(\sigma)) \leftrightarrow \tilde{x}(t(\sigma)) \]

where \( \sigma = [0,1] \)
Similarity criteria: Symmetry

Mirror Symmetry

One point \( x \), its tangent \( \tau \), and another point \( \tilde{x} \) defines mirror symmetry/co-circularity. Given the extra tangent \( \tilde{\tau} \) we may no longer have perfect symmetry, but we can now define a symmetry measure/error.

Intuitively this error should give small values of
\[
\left( x(s(\sigma)) - \tilde{x}(t(\sigma)) \right) \left( \frac{dx(s(\sigma))}{d\sigma} + \frac{d\tilde{x}(t(\sigma))}{d\sigma} \right) \quad (a)
\]
\[
\left( x(s(\sigma)) - \tilde{x}(t(\sigma)) \right) \left( \frac{dx(s(\sigma))}{d\sigma} - \frac{d\tilde{x}(t(\sigma))}{d\sigma} \right) \quad (b)
\]

where
\[
\frac{dx(s(\sigma))}{d\sigma} = \frac{dx(s)}{ds} \frac{ds(\sigma)}{d\sigma} = \tau(s(\sigma)) s'(\sigma)
\]
\[
\frac{d\tilde{x}(t(\sigma))}{d\sigma} = \frac{d\tilde{x}(t)}{dt} \frac{dt(\sigma)}{d\sigma} = \tilde{\tau}(t(\sigma)) t'(\sigma)
\]
A Global Optimization Approach

Minimize

\[ E(s(\sigma), t(\sigma)) = \int_{\sigma=0}^{1} F(x(s(\sigma)), \tau(s(\sigma)), \tilde{x}(t(\sigma)), \tilde{\tau}(t(\sigma)), s'(\sigma), t'(\sigma)) d\sigma \]

over all possible correspondences \( s(\sigma) \leftrightarrow t(\sigma) \), where \( \tau(s(\sigma)) \) and \( \tilde{\tau}(t(\sigma)) \)
Constraints on the form of $F$

1. Symmetry of the match $F(x, \tau, \tilde{x}, \tilde{\tau}, s'(\sigma), t'(\sigma)) = F(\tilde{x}, \tilde{\tau}, x, \tau, t'(\sigma), s'(\sigma))$

   i.e., $\Gamma$ and $\tilde{\Gamma}$ play equivalent roles.

2. Scaling Condition $F(x, \tau, \tilde{x}, \tilde{\tau}, \lambda s'(\sigma), \lambda t'(\sigma)) = \lambda F(x, \tau, \tilde{x}, \tilde{\tau}, s'(\sigma), t'(\sigma))$

   To obtain invariance under any change of parametrization $: \sigma \rightarrow \tilde{\sigma}(\sigma)$

   $$\int_{\sigma=0}^{1} F(x, \tau, \tilde{x}, \tilde{\tau}, s'(\sigma), t'(\sigma))d\sigma =$$

   $$\int_{\sigma=0}^{1} F\left(x, \tau, \tilde{x}, \tilde{\tau}, \frac{ds}{d\sigma}, \frac{dt}{d\sigma}\right) d\sigma d\tilde{\sigma} =$$

   $$\int_{\sigma=0}^{1} F\left(x, \tau, \tilde{x}, \tilde{\tau}, \frac{ds}{d\sigma}, \frac{d\sigma}{d\tilde{\sigma}}, \frac{dt}{d\sigma}, \frac{d\sigma}{d\tilde{\sigma}}\right) d\sigma =$$

   $$\int_{\tilde{\sigma}=0}^{1} F(x, \tau, \tilde{x}, \tilde{\tau}, s'(\tilde{\sigma}), t'(\tilde{\sigma}))d\tilde{\sigma}$$

   assuring that the cost depends only on the match $s(\sigma) \leftrightarrow t(\sigma)$ and not on the choice of maps $\sigma \mapsto s(\sigma)$ and $\sigma \mapsto t(\sigma)$. 

© 2003 by Davi Geiger
Constraints on the form of $F$ (cont.)

3. Rotation Invariance

$$F(Rx, R\tau, R\tilde{x}, R\tilde{\tau}, s'(\sigma), t'(\sigma)) = F(x, \tau, \tilde{x}, \tilde{\tau}, s'(\sigma), t'(\sigma))$$

so that rigid transformations will not affect the cost.

4. Partial Scaling Invariance

$$F(\lambda x, \tau, \lambda \tilde{x}, \tilde{\tau}, s'(\sigma), t'(\sigma)) = g(\lambda) F(x, \tau, \tilde{x}, \tilde{\tau}, s'(\sigma), t'(\sigma))$$

for some function $g(\lambda)$. Thus, scaling $\Gamma$ and $\tilde{\Gamma}$ by the common factor $\lambda$ will preserve the matching and change the cost by a shape independent $t$ and known amount $g(\lambda)$.
Similarity criteria: Symmetry

We write the symmetry measure as

\[
S(\Gamma, \Gamma') = \min_{t(\sigma)} \int_{\sigma=0}^{1} F(x, \tau, \tilde{x}, \tilde{\tau}, s'(\sigma), t'(\sigma))d\sigma
\]

where for any arbitrary parameterization \(s(\sigma)\) (constraint 2) we minimize the cost over all possible mappings \(t(\sigma)\) (which automatically gives \(t(s)\)).

The symmetry cost for any correspondence is defined as

\[
\int_{\sigma=0}^{1} \left\{ \frac{\left[ (x(s(\sigma)) - \tilde{x}(t(\sigma))) \cdot (\tau(s(\sigma))s'(\sigma) + \tilde{\tau}(t(\sigma))t'(\sigma)) \right]^2}{s'(\sigma) + t'(\sigma)} \right\} \left\{ \frac{\left[ (x(s(\sigma)) - \tilde{x}(t(\sigma))) \cdot (\tau(s(\sigma))s'(\sigma) - \tilde{\tau}(t(\sigma))t'(\sigma))^\perp \right]^2}{s'(\sigma) + t'(\sigma)} \right\}d\sigma
\]
Similarity criteria: Symmetry II

We add a third term to bias correspondences between closed points, to bias (a) over (b) solutions

\[
\int_{\sigma=0}^{1} F(x, \tau, \tilde{x}, \tilde{\tau}, s'(\sigma), t'(\sigma)) d\sigma =
\]

\[
\left\{ \left[ (x(s(\sigma)) - \tilde{x}(t(\sigma))) \cdot (\tau(s(\sigma))s'(\sigma) + \tilde{\tau}(t(\sigma))t'(\sigma)) \right]^2 + \frac{s'(\sigma) + t'(\sigma)}{s'(\sigma) + t'(\sigma)} \right\} \int_{\sigma=0}^{1} d\sigma
\]

\[
\left\{ \left[ (x(s(\sigma)) - \tilde{x}(t(\sigma))) \cdot (\tau(s(\sigma))s'(\sigma) - \tilde{\tau}(t(\sigma))t'(\sigma)) \right]^2 + \frac{s'(\sigma) + t'(\sigma)}{s'(\sigma) + t'(\sigma)} \right\} \int_{\sigma=0}^{1} d\sigma
\]

\[
c |x(s(\sigma)) - \tilde{x}(t(\sigma))|^2 \frac{|s'(\sigma) - t'(\sigma)|^2}{s'(\sigma) + t'(\sigma)}
\]

The parameter \( c \) should be small. To avoid the creation of bifurcation due to shape irregularity and allow bifurcation for object parts, see (c) and (d), we add a cost for bifurcation (jump cost) of the solution

\[
E(t(\sigma), s(\sigma)) = \int_0^1 F(\text{self - similarity}) + \theta(\sigma) \times \text{JumpCost} \, d\sigma
\]
Summary: Cost Functional/Energy Density

- **Structural properties**
  - Symmetric
  - Parameterization Independent

- **Geometrical properties**
  - Translation invariant
  - Rotation invariant
  - Scale “almost” invariance

- **Self-Similarity properties**
A Dynamic Programming Solution

\[ t(1-t(s)) = 1-s \]

Graph \( G \) and its SA-tree transformation from \( G \):

- **TGN**
  - \( s \) axis from 0 to 1
  - \( t \) axis from 0 to 1

Graph transformation from \( G \):

- **SA-tree**
  - From \( A \) to \( B, C, E \)

Graph labeling:

- **Nodes**: A, B, C, D, E, F, G, H, I
- **Edges** with labels representing costs

Grid overlaying the graph for visualization.
A Dynamic Programming Solution

The recurrence is on the “square boxes”. Each box is either linked to one inside square box or linked to a pair of inside and well aligned square boxes (leading to bifurcations).

Assumption, that will not be needed later: The final state is a match (0,0).
A Dynamic Programming Solution

BestCost[s, t] =

\[
\min_{tt\in\{t, t+\frac{1}{N}, \ldots, 1-s-\frac{1}{N}, 1-s\}} \left[ \min_{ss\in\{s+\frac{1}{N}, \ldots, 1-t-\frac{1}{N}\}} \left[ \begin{array}{c}
\text{BestCost}[s, t + \frac{1}{N}] + F[(ss, t + \frac{1}{N}) \rightarrow (s, t)] \\
\text{BestCost}[s + \frac{1}{N}, tt] + F[(s + \frac{1}{N}, tt) \rightarrow (s, t)]
\end{array} \right] + \text{Jump} \right]
\]

© 2003 by Davi Geiger
A Dynamic Programming Solution

A few more steps …

Starting states (matching candidates) → "self-matches" → iteration direction

© 2003 by Davi Geiger

Computer Vision

November 2003  L1.17
A General Dynamic Programming Solution

No need to force a match at (0,0) or any particular node/match.

Extend the parameters $s$ and $t$ such that negative values $x \rightarrow s,t=1+x$.

Extended Graph
A Dynamic Programming Solution

\[
\begin{align*}
D_S & \quad j=\text{N-i}(t=1-s) \\
D_J & \quad j=\text{N-i}(t=1-s)
\end{align*}
\]
Experimental Results for Closed Shapes

Shape

Shape Axis

SA-Tree

© 2003 by Davi Geiger
Experimental Results for Open Shapes
(the first and last points are assumed to be a match)

Shape

Shape Axis (SA)

SA-Tree
Experimental Results for Open Shapes

Shape Axis (SA)

SA-Forest
Matching Trees with deletions and merges for articulation and occlusions
Experimental Results for Open Shapes
Convexity v.s. Symmetry

White Convex Region

Black Convex Region