Identifying Design Principles

Maneesh Agrawala

CS 294-10: Visualization
Fall 2008

Assignment 3: Visualization Software

Create an interactive visualization application – you choose data domain and visualization technique.

1. Describe data and storyboard interface
due Oct 1 (before class)

2. Implement interface and produce final writeup
due Oct 13 (before class)

3. Submit the application and a final writeup on the wiki

Can work alone or in pairs
Final write up due before class on Oct 13, 2008
Final project

Design new visualization method
- Pose problem, Implement creative solution

Deliverables
- Implementation of solution
- 8-12 page paper in format of conference paper submission
- 2 design discussion presentations

Schedule
- Project proposal: 10/27
- Initial problem presentation: 10/27, 10/29 or 11/3
- Midpoint design discussion: 11/19, 11/24 or 11/26
- Final paper and presentation: 12/10

Grading
- Groups of up to 3 people, graded individually
- Clearly report responsibilities of each member

Spatial Layout
Pros and cons

Pros

- Much more flexible than linear constraint solving systems

Cons

- Can be relatively slow to converge
- Need to set penalty function parameters (weights)
- Difficult to encode desired layout in terms of mathematical penalty functions
Design principles

Sometimes specified in design books

- Tufte, Few, photography manuals, cartography books …
- Often specified at a high level
- Challenge is to transform principles into constraints or penalties

Cartographer Eduard Imhof’s labeling heurists transformed into penalty functions for an optimization based point labeling system [Edmondson 97]

Example-Based Methods
Preference elicitation [Gajos and Weld 05]

Learn characteristics of good designs
- Generate designs based on a parameterized design space
- Ask designers if they are good or bad
- Learn good parameters values based on responses

Nonlinear Inverse Opt. [Vollick et al. 07]

Learn label layout style from single example

Horizontal/Vertical
Nonlinear Inverse Opt. [Vollick et al. 07]
Learn label layout style from single example

Parallel Leader Lines

Artistic Resizing
A Technique for Rich Scale-Sensitive Vector Graphics

Pierre Dragicevic
Stéphane Chatty
David Thevenin
Jean-Luc Vinot
The Resizing Problem

- Fixed size
- Naive scaling
- Artistic resizing

Expressing Artistic Resizing

- Commonly described using formulae
  \[ x_L = \frac{(w-w_L)}{2} \]
  \[ y_L = \frac{(h-h_L)}{2} \]
  \[ w_L = 20 \]
  \[ h_L = 10 \]
  \[ w_B = 5 \]
  \[ h_B = 5 \]
  \[ r = 20 \]

- These formulae are:
  - Translated into code by the programmer
  - Or used as an input to constraint-solving systems
Example-Based Approach

1. Designers produce variants using their authoring tool
2. System interprets the example set

Artistic Resizing
How does it work?

- Assumes the exclusive use of:
  - Copy & paste for adding new examples
  - Affine transformation tools (move, scale, rotate, shear)
- Based on local interpolation of transformations
Artistic Resizing
How does it work?

- Each variant of T1 is associated with the example’s bounding box.

Artistic Resizing
How does it work?

- Problem of multivariate interpolation.
Pros and cons

Pros
- Often much easier to specify desired layout via example

Cons
- Usually requires underlying model
- Model will constrain types of layouts possible
- Large design spaces likely to require lots of examples to learn parameters well

Identifying Design Principles
Good Design Improves Effectiveness

London Underground [Beck 33]  Geographic version of map

Design principle:
- Straighten lines to emphasize sequence of stops

Technique used to emphasize/de-emphasize information
**Approach**

**Identify design principles**
- Cognition and perception

**Instantiate design principles**
- Principles become constraints that guide an optimization process

---

**Route Maps**
Visualizing Routes

A Better Visualization
Cognition of Route Maps

**Essential information**
- Turning points
- Route topology

**Secondary context information**
- Local landmarks, cross streets, etc.
- Overview area landmarks, global shape

**Exact geometry less important**
- Not apprehended accurately
- Not drawn accurately

[Tversky 81] [Tufte 90] [Tversky 92] [MacEachren 95] [Denis 97] [Tversky 99]

---

**Design Principles**

- Exaggerate road length
- Regularize turning angles
- Simplify road shape
LineDrive

Hand-drawn route map  LineDrive route map

Map Design via Optimization

Set of graphic elements
- Roads, labels, cross-streets, …

Choose visual attributes
- Position, orientation, size, …
- Distortions increase flexibility

Develop constraints based on design principles

Simulated annealing
- Perturb: Form a layout
- Score: Evaluate quality
- Minimize score
Road Layout

Choose road lengths and orientations

Before road layout

After road layout
Road Layout

Choose road lengths and orientations

Road Layout Constraints

**Length**
- Ensure all roads visible
- Maintain ordering by length

\[ \left( \frac{L_{\min} \cdot l(v)}{L_{\text{min}}} \right)^2 \cdot W_{\text{mat}} \]

\[ W_{\text{shuffle}} \]

**Orientation**
- Maintain original orientation

\[ |\alpha_{\text{curr}}(v) - \alpha_{\text{orig}}(v)| \cdot W_{\text{orient}} \]

**Topological errors**
- Prevent false
- Prevent missing
- Ensure separation

\[ \min(d_{\text{orig}}, d_{\text{dest}}) \cdot W_{\text{false}} \]

\[ d \cdot W_{\text{missing}} \]

\[ \min(d_{\text{ext}}, E) \cdot W_{\text{ext}} \]

**Overall route shape**
- Maintain endpoint direction
- Maintain endpoint distance

\[ |\alpha_{\text{curr}}(v) - \alpha_{\text{orig}}(v)| \cdot W_{\text{enddir}} \]

\[ |d_{\text{curr}}(v) - d_{\text{orig}}(v)| \cdot W_{\text{enddist}} \]
Balancing the Constraints

Prioritize scores by importance
1. Prevent topological errors
2. Ensure all roads visible
3. Maintain original orientation
4. Maintain ordering by length
5. Maintain overall route shape

Priorities set based on usability tests
- Users given maps containing errors
- Rated which errors most confusing

Label Layout

Find overlap-free position for each label
Context Layout

Place cross-streets and exit signs if possible

Bellevue to Seattle

[Map showing the route from Bellevue to Seattle]
Cross-Country Route

System Performance

7727 routes  (sampled over 1 day at MapBlast!)

- Median distance  52.5 miles
- Median number turning points  13
- Median computation time  0.7 sec

- Short roads  5.4 %
- False intersections  0.3 %
- Missing intersections  0.2 %

- Label-label overlap  0.5 %
- Label-road overlap  11.7 %
Results

Beta version 6 months
- 150,000 maps served

2242 responses
- Replace standard 55.6%
- Use with standard 43.5%
- Prefer standard 0.9%

Current Status
- Deployed at: mappoint.com
- 750,000 maps/day

DEMO
mappoint.com
Limited Resolution PDA

Assembly Instructions
Previous Work

Planning
- Choose sequence of assembly operations
- Robotics / AI / Mechanical Engineering
  [Wolter 89], [de Mello 91], [Wilson 92], [Romney 95]

Presentation
- Visually convey assembly operations
- Visualization / Computer Graphics
  [Seligmann 91], [Rist 94], [Butz 97], [Strothotte 98]

Jointly optimize plan and presentation

Geometric Analysis [Romney 95]

Input Parts

Blocking Graph

A blocked by B

B blocked by A

B → A

Both parts free to move

A blocked by B

A → B

A → C

B → C

A → B

A → C

B → C

B → A
How do we choose the best sequence?

Many Geometrically Valid Sequences
Identifying Design Principles

**Stage 1:** Production
**Stage 2:** Preference
**Stage 3:** Comprehension

Spatial Ability Tests

Mental Rotation [Vandenburg 78]
Navigation [Money 78]

Separate high and low spatial ability
Stage 1: Production

- 43 Participants
- Assemble TV Stand without instructions
- Write instructions for novice assembler

Stage 1: Mean completion time

<table>
<thead>
<tr>
<th>Time to assemble (min)</th>
<th>Low spatial</th>
<th>High spatial</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12.76</td>
<td>7.29</td>
</tr>
</tbody>
</table>

Bar chart showing mean completion times for low and high spatial tasks.
Stage 1: Instructions produced

- Almost all contained diagrams: 98%
- Text redundant with diagrams: 62%

Stage 1: Errors in instructions

- Errors in low spatial instructions: 86%
- Errors in high spatial instructions: 12%
Stage 1: Errors in instructions

- Errors in low spatial instructions 86%
- Errors in high spatial instructions 12%

Stage 1: Classes of Diagrams

- Structural diagrams depict completed step
- Action diagrams show assembly action/operation

Parts menu to differentiate parts
Stage 1: Action diagrams

- High spatial
  - More action diagrams
  - More 3D diagrams
  - Less text

Mean number per set

<table>
<thead>
<tr>
<th>Low spatial</th>
<th>High spatial</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.64</td>
<td>2.67</td>
</tr>
</tbody>
</table>

Stage 2: Preference

- 21 Participants
- Assemble TV Stand without instructions
- Rated 39 sets of redrawn instructions
Stage 2: Highest Rated

- Ratings similar across all participants
- Spatial ability does not affect preference

Stage 3: Comprehension

- 44 Participants
- Given 1 of 4 instruction sets from Stage 2
- Assemble TV stand using instructions
Stage 3: Results

- No difference in assembly time by condition
- Instruction consultations: Low 8.9   High 7.1
- Box picture consultations: Low 9.1  High 3.4

Comments
- Should show relevant parts and attachments
- Structural diagrams and exploded view hard to use
- Text not very useful

Design Principles

Step-by-Step
Action diagrams
Good visibility

TV stand instructions generated by our system
**Input**

**Geometry:** Parts in assembled configuration

**Orientations:** Default viewpoint / orientation
- Preferred orientation for each part

**Groupings:** Fasteners, significant parts, similar actions, symmetry

Assembled geometry in default orientation

Parts grouped as fasteners and significant parts

---

**Find best assembly sequence**

- Planning: Geometric feasibility
- Presentation: Visibility

---

**Step-by-step assembly sequence**

Invalid

Valid

Valid
Computing Visibility

\[
\text{Area}(P,Q) = \# \text{ red pixels} \\
\text{Area}(P) = \# \text{ red pixels}
\]

\[
\text{Vis}(P,Q) = \text{Area}(P,Q) / \text{Area}(P)
\]

% pixels that remain visible when sides are included

Visibility Constraints

1. Parts being attached \( R \)
   - Check that each part is visible
   \[
   \min_{r \in R} (\text{Vis}(r, R-r)) \cdot W_R
   \]

2. Previously attached parts \( A \)
   - Check that context is visible
   \[
   \text{Vis}(A, R) \cdot W_A
   \]

3. Future unattached parts \( U \)
   - Check that future parts will be visible
   \[
   \min_{u \in U} (\text{Vis}(u, R)) \cdot W_U
   \]
Lego Car

Bookcase
Sequentially add parts

- Least visible to most visible
- Distance to viewer

Reorient

- Set preferred orientation
- If poor visibility try alternate orientations
Action Diagrams

- Choose Direction
- Build Stacks
- Place Guidelines

Step-by-step assembly diagrams

Search

Subdivide Steps

Reorientation

Step-by-step assembly sequence

**Bookcase**

1. Choose Direction
2. Build Stacks
3. Place Guidelines

9 Parts Design: 48s
Test Object

Evaluation

- 30 Participants
- Given 1 of 3 instruction sets: factory, hand-drawn, computer
- Assemble TV stand using instructions
Factory

Hand-drawn

1. Stand the following board up as shown in the picture. (You should have two of these boards, side by side for the box.)

2. Place the sides and bottom board on a flat surface such that the unmarked edges are facing up and the marked edges are facing down. (This is to make the side of the finished box have the same direction.)

3. Place a white paper in each of the holes of each side of the dimension board.

4. Place this board as shown in the picture. Again, make sure the marked portion of the same direction as the finished edge of the other board.

5. Take the matching long board and move it to the other side. Make sure it is in the same direction as the other board.

6. Place the board in the holes as indicated in the picture and then the side can come.

7. Place the board in the holes as indicated in the picture and then the side can come.
## Computer Generated

<table>
<thead>
<tr>
<th></th>
<th><img src="image1.png" alt="Diagram 1" /></th>
<th><img src="image2.png" alt="Diagram 2" /></th>
<th><img src="image3.png" alt="Diagram 3" /></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image4.png" alt="Diagram 4" /></td>
<td><img src="image5.png" alt="Diagram 5" /></td>
<td><img src="image6.png" alt="Diagram 6" /></td>
</tr>
</tbody>
</table>

## Results

<table>
<thead>
<tr>
<th></th>
<th>Factory</th>
<th>Hand-drawn</th>
<th>Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean time to assemble (min)</td>
<td>18.9</td>
<td>16.0</td>
<td>10.2</td>
</tr>
</tbody>
</table>

Errors: Factory 1.6 Hand-drawn 0.6 Computer 0.5
Task rated easiest in computer condition
Summary

Identification of design principles
- Production
- Preference
- Comprehension

Instantiation of design principles

Validation of design principles