Depth and Shape Inference (II)

Introduction to Computational and Biological Vision

CS 202-1-5261

Computer Science Department, BGU

Ohad Ben-Shahar
**Edge and boundary Interpretation**

A (desired) edge detection mapping (labeling)

\[ E[I(x, y)] : \mathbb{R}^2 \rightarrow \Lambda = \{0, 1, 2, 3, 4, 5, \ldots \} \]

\[ E[I(x, y)] = \begin{cases} 
1 & (x,y) \text{ is a boundary point} \\
2 & (x,y) \text{ is a surface discontinuity point} \\
3 & (x,y) \text{ is an illumination discontinuity point} \\
4 & (x,y) \text{ is a reflectance discontinuity point} \\
\vdots \\
0 & \text{Otherwise (non edge point)} 
\end{cases} \]
Edge and boundary Interpretation

Edge/Contour interpretation = Determination of physical cause

Surface (normal) discontinuity

Depth discontinuity
## Edge and boundary Interpretation

<table>
<thead>
<tr>
<th>Year</th>
<th>Author</th>
<th>Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1968</td>
<td>Guzman</td>
<td>- The SEE program</td>
</tr>
<tr>
<td>1971</td>
<td>Huffman / Clowes</td>
<td>- Formal analysis of block world</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- polygonal planar surfaces</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Depth and surface discontinuities only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- trihedral vertices</td>
</tr>
<tr>
<td>1972</td>
<td>Waltz</td>
<td>- Shadow boundaries</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Cracks</td>
</tr>
<tr>
<td>1974</td>
<td>Turner</td>
<td>- Opaque solids with limited class of smooth surfaces</td>
</tr>
<tr>
<td>1987</td>
<td>Malik</td>
<td>- Curved objects with C³ faces</td>
</tr>
</tbody>
</table>
**Edge and boundary Interpretation**

**Block worlds**

- **polygonal objects**
  - Planar faces
  - trihedral vertices

- **Allowed discontinuities**
  - Depth discontinuities
  - Surface normal discontinuities

- **Ignored discontinuities**
  - Illumination discontinuities (shadows)
  - Reflectance (pigmentation/color) discontinuities

- **General viewpoint assumption:**
**Edge and boundary Interpretation**

**General viewpoint assumption**

A viewpoint is general (as opposed to accidental) if a small perturbation of it would not affect the configuration of the line drawing (no qualitative change).
Edge and boundary Interpretation

Edge types

Convex orientation edge

Concave orientation edge

Depth edge

Depth edge
**Edge and boundary Interpretation**

**Edge types**

- Convex orientation edge
- Concave orientation edge
- Depth edge
- Depth edge
Edge and boundary Interpretation

Edge interpretation = Edge labeling

$\lambda \in \{+, -, \uparrow, \downarrow\}$
**Edge and boundary Interpretation**

**Edge intersections (vertices)**

- V-junction (L-junction)
- Y-junction
- W-junction (arrows)
- T-junction
Edge and boundary Interpretation

Physical constraints on labeling of edge intersections

Physical constraints on arrow junctions

[Huffman 1971, Clowes 1971]
Edge and boundary Interpretation

Huffman and Clowes complete catalog (1971)
Edge and boundary Interpretation

Constancy of interpretation along edges
Edge and boundary Interpretation

Constancy of interpretation along edges

Not valid for curved objects
Edge and boundary Interpretation

Using constraints to label line drawings
Edge and boundary Interpretation

Consistent labeling is a necessary condition

...but not a sufficient one
Edge and boundary Interpretation

Consistent line drawing labeling via sequential search and backtracking

1. Form a graph representation $G=\left(V,E\right)$ of the line drawing with vertices as nodes line drawing edges as graph edges

2. Assume some (arbitrary) ordering between edges $E=\{e_1,e_2,\ldots,e_n\}$

3. Set $i=1$

4. Label $e_i$ with the next untried label from $\{+,-,<,>\}$

5. Using the catalog, check consistency of the new label with all adjacent labeled edges.

6. If consistent then increase $i$ and go to step 4
   else backtrack
Edge and boundary Interpretation

Consistent line drawing labeling via relaxation labeling

1. Initial the label set for each line drawing label to \{+,-,<,>\}

2. Repeat at all edges concurrently until no label set decreases in size
   - If label \(L\) at edge \(e\) cannot form a consistent junction using available labels at edges intersecting \(e\) in a common vertex, filter \(L\) from \(e\)’s label set
Edge and boundary Interpretation

Consistent line drawing labeling via relaxation labeling

Iteration #1
Edge and boundary Interpretation

Consistent line drawing labeling via relaxation labeling
Edge and boundary Interpretation

Consistent line drawing labeling via relaxation labeling

Iteration #3
Edge and boundary Interpretation

Consistent line drawing labeling via relaxation labeling

Iteration #3
**Edge and boundary Interpretation**

Consistent line drawing labeling via relaxation labeling

[Diagram of consistent line drawing labeling via relaxation labeling with arrows indicating positivity and negativity at each vertex, labeled as Iteration #4 and Convergence state!!]
Edge and boundary Interpretation

Consistent line drawing labeling via relaxation labeling

Possible interpretations
**Edge and boundary Interpretation**

Consistent line drawing labeling via relaxation labeling

\[ B = \{ b_1, \ldots, b_n \} \quad \text{set of objects to be labeled} \]

\[ \Lambda = \{ 1, 2, \ldots, m \} \quad \text{set of possible labels} \]

\[ p_i^0(\lambda) \quad \text{the measured confidence that } b_i \text{ should be labeled } \lambda. \]

\[ p_i^0(\lambda) \geq 0 \]

\[ \sum_{\lambda=1}^{m} p_i^0(\lambda) = 1 \quad \forall i \]

\[ r_{ij}(\lambda, \mu) \quad \Leftrightarrow \quad \text{the strength of compatibility between the hypotheses “} b_i \text{ has label } \lambda \text{” and “} b_j \text{ has label } \mu \text{”} \]
**Edge and boundary Interpretation**

Consistent line drawing labeling via relaxation labeling

Nodes = line drawing intersections

Labels = Vertex catalog
Edge and boundary Interpretation

Consistent line drawing labeling via relaxation labeling

\[ r_{ij} \rightarrow -1 \]

\[ r_{ij} \rightarrow 1 \]