Using 3D cues

Martial Hebert

Carnegie Mellon University
Set of feature vectors → Learning/inference → Training data

Input image → Set of feature vectors

[Example from D. Hoiem]

Questions

- How to estimate geometric properties from an image?
- How to incorporate geometric constraints?
- How to combine reasoning tools with statistical classification/regression tools?
• Guzman (*SEE*), 1968
• Yakimovsky & Feldman, 1973
• Hansen & Riseman (*VISIONS*), 1978
• Barrow & Tenenbaum 1978
• Brooks (*ACRONYM*), 1979
• Ohta & Kanade, 1978

**Chronology**

Bottom up classifiers  More explicit constraint+reasoning

Qualitative  Explicit/Quantitative
Qualitative

Region labels

First attempt: Estimate surface labels

Training data

Classification

Multiple segmentations

Example from Hoiem

Example

Support | Vertical | Sky

V-Left | V-Center | V-Right | V-Porous | V-Solid
• Learning from image features to depth + MRF: A. Saxena, S. H. Chung, and A. Y. Ng. 3-D depth reconstruction from a single still image. IJCV, 76, 2007.

MRF on superpixels $\rightarrow$ Depth map
• Unary potentials:
  • Depth prediction from local features + co-occurrence of superpixels over multiple segmentations
• Binary potentials:
  • Colinearity along edges
  • Connectivity along neighboring superpixels
  • Co-planarity along neighboring superpixels

• Is a more precise representation possible?

• For example:
  – We would like to include reasoning about interposition (relations between object relative to a viewpoint induced by occlusion boundaries)
  – We would like to include constraints about object semantics (when known)
Qualitative

Region labels + Boundaries and objects

Using occlusion cues: Depth ordering and depth estimation

Depth estimate from ground intersection

Depth ordering from occlusion relations
Surfaces

Occlusions

Objects and Viewpoint

Separate cues

Example from D. Hoiem
Combined reasoning

Input
Surfaces
Occlusion Boundaries
Objects and Horizon

Example from D. Hoiem
Semantic labels provide strong constraints on local surface orientation
Semantic labels also provide an estimate of relative depth ordering and occlusion relations

Outline:
1. Estimate labels from image features
2. Estimate point-wise depth (with depth constraints from labels)
3. Estimate local orientations (with orientation constraints from labels)
Comments

• Plus:
  – Scene geometry (surface geometry and object relations) estimated from image data
  – Scene geometry used explicitly in scene understanding

• Minus:
  – Still mostly bottom-up classification approach
  – No use of domain constraints or constraints governing the physical world

Qualitative
More quantitative
more precise
Example

• Using constraints induced by man-made environments in interpreting images
• Examples: Manhattan world, limited vocabulary of object configurations, etc.

Constraint: Manhattan world assumption

• Three dominant directions corresponding to three “orthogonal” vanishing points

\[ n_i = K^{-1}v_i \]

\[ n_j \cdot n_i = v_j^T K^{-T} K^{-1}v_i = 0 \]
We need to design 4 things

• Parameterization: \( y \)

• Features: \( x \)

• Scoring/hypothesis evaluation:
  \[
  y_o = \arg\max_y f(x, y, w)
  \]

• A way to sample, or generate hypotheses \( y \)

\[
\begin{align*}
  y &= [y_1 \ y_2 \ y_3 \ y_4] \\
  y_1 &= v_1 \\
  y_2 &= v_2 \\
  y_3 &= v_3 \\
  y_4 &= v_4
\end{align*}
\]

Features

- Features: $x$
  - Surface layout (see earlier)
  - Lines, regions, …
  - Orientation maps
  - Junctions

Orientation maps: Sweep algorithm

$p$ is of orientation if it is in one $\mathcal{S}$
It is in one $\mathcal{S}$
It is in none of $\mathcal{S}$

**Scoring the hypotheses**

- **Structured prediction**

$$y_o = \arg\max_y w^T \varphi(x, y)$$


Vanishing points in input image \( x \)

Many initial layout hypotheses \( y \)

Score = output of predictor \( f(x,y,w) \)

Data for evaluating \( y \)

Lines

Faces

Label confidences from classifier

<table>
<thead>
<tr>
<th>Surface labels</th>
<th>Floor</th>
<th>Left</th>
<th>Middle</th>
<th>Right</th>
<th>Ceiling</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Floor</td>
<td>7/4</td>
<td>0/0</td>
<td>0/1</td>
<td>0/1</td>
<td>0/0</td>
<td>24/30</td>
</tr>
<tr>
<td>Left</td>
<td>1/0</td>
<td>75/43</td>
<td>14/44</td>
<td>0/0</td>
<td>1/1</td>
<td>9/12</td>
</tr>
<tr>
<td>Middle</td>
<td>1/0</td>
<td>3/5</td>
<td>60/82</td>
<td>4/6</td>
<td>2/1</td>
<td>1/99</td>
</tr>
<tr>
<td>Right</td>
<td>1/1</td>
<td>0/0</td>
<td>14/48</td>
<td>3/42</td>
<td>3/2</td>
<td>10/7</td>
</tr>
<tr>
<td>Ceiling</td>
<td>0/0</td>
<td>4/3</td>
<td>28/47</td>
<td>3/25</td>
<td>66/45</td>
<td>0/0</td>
</tr>
<tr>
<td>Objects</td>
<td>16/12</td>
<td>1/1</td>
<td>5/10</td>
<td>1/2</td>
<td>0/0</td>
<td>76/76</td>
</tr>
</tbody>
</table>
**Definition of mapping function**

\[ f(x, y, w) = w^T \varphi(x, y) \]

**Learned weight vector**

**Feature vector measuring agreement between lines, faces and labels**

\( \varphi(x, y) = \text{Relative sum of lengths of line segments in each face agreeing with labels from appearance-based classifiers} \)

**Learning**

\[
\min_{w, \xi} \frac{1}{2} \|w\|^2 + C \sum_i \xi_i \\
\text{s.t. } \xi_i \geq 0 \forall i, \text{ and } w^T \psi(x_i, y_i) - w^T \psi(x_i, y) \geq \Delta(y_i, y) - \xi_i,
\]

Loss function:
- Distance between centroids of true and estimated faces
- Overlap between true and estimated faces
- Number of missing faces

True: \( y_i \)  

Estimated: \( y \)  

\( \Delta(y_i, y) \)
More detailed interpretation: Clutter vs. free space

Example from H. Wang

\[ f(x, y, h; w) = w^T \varphi(x, y, h) \]

- Clutter mask

- 228-dim feature vector
  - Measures how the fraction of each face not in clutter agrees with \( y \)
  - Measures the fraction of the faces not in clutter

We need to design 4 things

- Parameterization: \( y \)

- Features: \( x \)

- Scoring/hypothesis evaluation:
  \[ y_o = \arg\max_y f(x, y, w) \]

- A way to sample, or generate hypotheses \( y \)
\[
w^T \varphi(x, y) = \sum_{f=1}^{f=5} w_f^T \varphi_f(x, y)
\]

4th order potential

\[
y = [y_1 \ y_2 \ y_3 \ y_4]
\]

Integral geometry trick

\[
\varphi_f(x, y) = \text{Sum of features over facet } f \text{ (geometric context, orientation map, edges, junctions, …)}
\]
\[ \varphi_1(x, y) = A(x, y_2, y_4) - A(x, y_1, y_4) \]
Qualitative + Boundaries and objects

Stronger geometric constraints from domain knowledge + more constraints

Region labels

Explicit

More quantitative more precise

Integrating more constraints

- **Constraints**
  - Volumetric constraints

- **Techniques**
  - Structured prediction
Constraints: Solid objects must satisfy volumetric/physical constraints

- Finite volume
- Spatial exclusion
- Containment


\[ y = [r_1 \ldots r_n o_1 \ldots o_m] \]

\[ f(x, y) \]

Labeling: Indicator vector of scene configurations + object hypothesis
\[ f(x, y) = w^T \psi(x, y) + w_\varphi^T \varphi(y) \]

**Compatibility of image data with geometric configuration**

**Penalty term for incompatible configurations**

**Features from image (surface labels, vanishing points, etc.)**

**Features of the scene configuration to evaluate constraint violations**

- **Inference:**
  \[ y^* = \arg \max_y f(x, y) \]

- **Training**
  - Use structured SVM to estimate \( w \)
Integrating more constraints

• **Constraints**
  – Volumetric constraints

• **Techniques**
  – Structured prediction
  – Sampling
Representation

• \( \theta = \) set of all parameters describing scene and camera parameters

Scene model: \( \theta = (r,c) \)

Camera model: \( c = (\psi, \varphi, f) \)

Scene layout:
One block + orientation

\( r = (x, y, z, w, h, l, \gamma) \)

Object model: block

\( o = (x, y, z, w, h, l) \)


Score function

• \( E = \) edge points in input image

• \( \theta = \) hypothesis

\[ p(E | \theta) = e_{bg}^{N_{bg}} e_{miss}^{N_{miss}} \prod_k e(x_k) \]

• Sampling:
Sample hypothesis by MCMC sampling of

\[ P(\theta | E) \sim P(E | \theta)P(\theta) \]
Sampling: Diffusion moves
- Diffusion moves change some part of $\theta$
- Multiple types of moves used in random order

Sample room boundary
Change $r = (x, y, z, w, h, l, \gamma)$

Sample camera
Change $c = (\psi, \phi, f)$

Sample object parameters
Change $o = (x, y, z, w, h, l)$

Sample over a block edge

[Example from DelPero et al.]

Sampling: Jump moves
- Continuous parameters only so far
- Number of objects is fixed in $\theta$
- We need to sample over the possible number of objects

- Jump proposal generated based on corner features
- A corner feature can generate a new block or a new layout

Propose layout
Propose frame
Propose block

[Example from DelPero et al.]
• Shows how to search through hypotheses using MCMC sampling
• No training
• Samples through continuous (sizes, etc.) and discrete (# objects) parameters
• Recovers camera parameters as well

• http://civs.ucla.edu/old/MCMC/MCMC_tutorial.htm

[Example from DelPero et al.]

3D geometry refinement

• Two (related) problems:
  – Discrepancy between 2D and 3D error evaluation
  – Large errors in object placement

• Possible approach:
  – Error evaluation directly in 3D: Free space estimation
  – Pose refinement by contact estimation
    • For each object
      – Search through micro-hypothesis by varying location of vertices
      – Score using SVM to classify contact/non-contact

[Example from Hoiem et al.]
Integrating more constraints

- **Constraints**
  - Volumetric constraints
  - Physical constraints

- **Techniques**
  - Structured prediction
  - Sampling
  - Search through hypothesis space

Physical constraints


Integrating more constraints

- **Constraints**
  - Volumetric constraints
  - Physical constraints
  - Relative placement

- **Techniques**
  - Structured prediction
  - Sampling
  - Search through hypothesis space
Using relative placement statistics

Layout: statistics on relative size
\[ r_1 = \frac{\max(w, l)}{\min(w, l)} \]
\[ r_2 = \frac{\max(w, l)}{h} \]

Objects: statistics on relative size and contact for each object type \( i \)
\[ r_{i1} = \frac{h_i}{\max(w_i, l_i)} \]
\[ r_{i2} = \frac{\max(w_i, l_i)}{\min(w_i, l_i)} \]
\[ r_{i3} = \frac{h}{h_i} \quad d_i = 1 \text{ if surface contact} \]

Gaussian distribution estimated from prior data
Sampling technique as before but incorporating the priors

Using relative placement statistics

• Is it worth it?

| only edge likelihood (no blocks) | 26.0 % |
| + camera and room prior (no blocks) | 24.7 % |
| + orientation likelihood (no blocks) | 21.3 % |
| + random blocks | 19.7 % |
| + objects | 16.3 % |

• Yes but still simplified representation of the prior distribution (independent distributions of a few parameters)

Generalization to groups of objects (geometric phrases)

Integrating more constraints

- **Constraints**
  - Volumetric constraints
  - Physical constraints
  - Relative placement
  - Functional constraints

- **Techniques**
  - Structured prediction
  - Sampling
  - Search through hypothesis space
Even more constraints: Functional


Summary

- Estimating qualitative geometry from input image
- Combining geometric cues with interpretation
- Incorporating more and more constraints
  - Volumetric
  - Physical
  - Relative placement
  - Functional

- Classifiers/regressors + multiple segmentations
- Sampling techniques
- Search through discrete hypothesis space
- Structured prediction
- Grammars
- Using (large) prior data
Summary

• Estimating qualitative geometry from input image
• Combining geometric cues with interpretation
• Incorporating more and more constraints
  – Volumetric
  – Physical
  – Relative
  – Functional
• Classifiers/regressors + multiple segmentations
• Sampling techniques
• Search through discrete hypothesis space
• Structured prediction
• Grammars
• Using (large) prior data

How to represent (3D, imprecise) spatial information?
How to generate hypotheses?
How to score hypotheses?
How to search through hypotheses?