Stereo

CSE P 576
Larry Zitnick (larryz@microsoft.com)
Many slides courtesy of Steve Seitz

Thomas Edison

Mark Twain, 1908
Why do we perceive depth?

What do humans use as depth cues?

**Motion**

**Convergence**

When watching an object close to us, our eyes point slightly inward. This difference in the direction of the eyes is called convergence. This depth cue is effective only on short distances (less than 10 meters).

**Binocular Parallax**

As our eyes see the world from slightly different locations, the images sensed by the eyes are slightly different. This difference in the sensed images is called binocular parallax. Human visual system is very sensitive to these differences, and binocular parallax is the most important depth cue for medium viewing distances. The sense of depth can be achieved using binocular parallax; even if all other depth cues are removed.

**Monocular Movement Parallax**

If we close one of our eyes, we can perceive depth by moving our head. This happens because human visual system can extract depth information in two similar images sensed after each other, in the same way it can combine two images from different eyes.

**Focus**

Accommodation is the tension of the muscle that changes the focal length of the lens of eye. Thus it brings into focus objects at different distances. This depth cue is weak, and it is effective only at short viewing distances (less than 2 meters) and with other cues.

**Image cues**

**Retinal Image Size**

When we know the size of the object is known, our brain compares the sensed size of the object to the real size, and thus acquires information about the distance of the object.

**Linear Perspective**

When looking down a straight level road we see the parallel sides of the road meet in the horizon. This effect is often called linear perspective.

**Texture Gradient**

The closer we are to an object the more detail we sense of its surface texture. This is especially true if the texture spans all the way from near to far.

**Overlapping**

When objects block each other out of our sight, we learn that the object that blocks the other one is closer to us. The object whose outline pattern looks more continuous is felt to be closer.

**Shadows and Shadows**

When we know the location of a light source and see objects casting shadows on other objects, we learn about the distance of these objects. For most illumination scenes dominant light sources are suns, stars, lights, etc. The three-dimensional world consists of surfaces, and surfaces cast shadows on other surfaces.


**Stereo**

Jonathan Chiu

scene point

optical center

image plane
Stereo

Basic Principle: Triangulation
- Gives reconstruction as intersection of two rays
- Requires
  - camera pose (calibration)
  - point correspondence

Stereo correspondence

Determine Pixel Correspondence
- Pairs of points that correspond to same scene point

Epipolar Constraint
- Reduces correspondence problem to 1D search along conjugate epipolar lines

Fundamental matrix

Let $p$ be a point in left image, $p'$ in right image

Epipolar relation
- $p$ maps to epipolar line $l'$
- $p'$ maps to epipolar line $l$

Epipolar mapping described by a 3x3 matrix $F$

$$l' = FP$$
$$l = p'F$$

It follows that
$$p' F p = 0$$

Fundamental matrix

This matrix $F$ is called
- the “Essential Matrix”
  - when image intrinsic parameters are known
- the “Fundamental Matrix”
  - more generally (uncalibrated case)

Can solve for $F$ from point correspondences
- Each $(p, p')$ pair gives one linear equation in entries of $F$

$$p' F p = 0$$

- 8 points give enough to solve for $F$ (8-point algorithm)
- see [Marc Pollefeys’s notes](http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html) for a nice tutorial
Stereo image rectification

- Reproject image planes onto a common plane parallel to the line between optical centers.
- Pixel motion is horizontal after this transformation.
- Two homographies (3x3 transform), one for each input image reprojection.


Example

Unrectified

Rectified

Depth from disparity

\[
\text{disparity} = x - x' = \frac{\text{baseline} \times f}{z}
\]
Stereo matching algorithms

Match Pixels in Conjugate Epipolar Lines
- Assume brightness constancy
- This is a tough problem
- Numerous approaches
  - A good survey and evaluation: [http://www.middlebury.edu/stereo/](http://www.middlebury.edu/stereo/)

Your basic stereo algorithm

For each epipolar line
For each pixel in the left image
  - compare with every pixel on same epipolar line in right image
  - pick pixel with minimum match cost
Improvement: match windows
  - This should look familiar...

Stereo as energy minimization

- Find disparities \( d \) that minimize an energy function \( E(d) \)

Simple pixel / window matching

\[
E(d) = \sum_{(x,y) \in I} C(x, y, d(x, y))
\]

\[
C(x, y, d(x, y)) = \text{SSD distance between windows } I(x, y) \text{ and } J(x, y + d(x, y))
\]
Stereo as energy minimization

\[ y = 14 \]

Simple pixel/window matching: choose the minimum of each column in the DSI independently:

\[ d(x, y) = \arg \min_{d'} C(x, y, d') \]

Matching windows

<table>
<thead>
<tr>
<th>Similarity Measure</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum of Absolute Differences (SAD)</td>
<td>[ \sum_{i=0}^{W-1}</td>
</tr>
<tr>
<td>Sum of Squared Differences (SSD)</td>
<td>[ \sum_{i=0}^{W-1} (f(x+i,y) - f'(x+i,y))^2 ]</td>
</tr>
<tr>
<td>Zero-mean SAD</td>
<td>[ \sum_{i=0}^{W-1} (f(x+i,y) - \bar{f}(x+i,y))^2 ]</td>
</tr>
<tr>
<td>Locally scaled SAD</td>
<td>[ \sum_{i=0}^{W-1} \frac{(f(x+i,y) - \bar{f}(x+i,y))^2}{\sum_{j=0}^{W-1} (f(x+j,y) - \bar{f}(x+j,y))^2} ]</td>
</tr>
<tr>
<td>Normalized Cross Correlation (NCC)</td>
<td>[ \frac{\sum_{i=0}^{W-1} f(x+i,y) f'(x+i,y)}{\sqrt{\sum_{i=0}^{W-1} f(x+i,y)^2 \sum_{i=0}^{W-1} f'(x+i,y)^2}} ]</td>
</tr>
</tbody>
</table>

Matching windows with SAD, SSD, NCC, and ground truth.

More window techniques

- Bilateral filtering
- Adaptive weighting

Window size

Effect of window size
- Smaller window
  - Better results with adaptive window
- Larger window

- W = 3
- W = 20

Effect of window size
- Smaller window
Stereo as energy minimization

What defines a good stereo correspondence?

1. **Match quality**
   - Want each pixel to find a good match in the other image

2. **Smoothness**
   - If two pixels are adjacent, they should (usually) move about the same amount

Better objective function

\[
E(d) = E_d(d) + \lambda E_s(d)
\]

- **Match cost:**
  \[E_d(d) = \sum_{(x,y) \in I} C(x, y, d(x, y))\]

- **Smoothness cost:**
  \[E_s(d) = \sum_{(p,q) \in \mathcal{E}} V(d_p, d_q)\]

\[\mathcal{E}: \text{set of neighboring pixels}\]

4-connected 8-connected neighborhood

**Smoothness cost**

\[
E_s(d) = \sum_{(p,q) \in \mathcal{E}} V(d_p, d_q)
\]

\[V(d_p, d_q) = \begin{cases} 
1 & \text{if } d_p = d_q \\
0 & \text{if } d_p \neq d_q
\end{cases}\]

“Potts model”
Dynamic programming

\[ E(d) = E_d(d) + \lambda E_s(d) \]

Can minimize this independently per scanline using dynamic programming (DP)

\[ D(x, y, d) : \text{minimum cost of solution such that } d(x, y) = d \]

\[ D(x, y, d) = C(x, y, d) + \min_{d'} \{ D(x-1, y, d') + \lambda |d - d'| \} \]

Energy minimization via graph cuts

Labels (disparities)

- Graph Cut
  - Delete enough edges so that
    - each pixel is connected to exactly one label node
  - Cost of a cut: sum of deleted edge weights
  - Finding min cost cut equivalent to finding global minimum of energy function

Computing a multiway cut

- With 2 labels: classical min-cut problem
  - Solvable by standard flow algorithms
    - polynomial time in theory, nearly linear in practice
  - More than 2 terminals: NP-hard
    - Dahlhaus et al., STOC ’92
- Efficient approximation algorithms exist
    - Within a factor of 2 of optimal
    - Computes local minimum in a strong sense
      - even very large moves will not improve the energy

**Move examples**

Idea: convert multi-way cut into a sequence of binary cut problems

- Red-blue swap move
- Green expansion move

**The swap move algorithm**

1. Start with an arbitrary labeling
2. Cycle through every label pair \((A,B)\) in some order
   2.1 Find the lowest \(E\) labeling within a single \(AB\)-swap
   2.2 Go there if it’s lower \(E\) than the current labeling
3. If \(E\) did not decrease in the cycle, we’re done
   Otherwise, go to step 2

**Alpha-expansion**

Similar to swap move algorithm, except it’s one label vs. all others.

**Other energy functions**

Can optimize other functions (exactly or approximately) with graph cuts

\[
V(d_p, d_q) = (d_p - d_q)^2
\]

\[
V(d_p, d_q) = |d_p - d_q|
\]

But many functions are much harder…
Markov Random Fields

Allows rich probabilistic models for images.
But built in a local, modular way. Learn local relationships, get global effects

Network joint probability

\[ P(x, y) = \frac{1}{Z} \prod_{i, j} \Psi(x_i, x_j) \prod_i \Phi(x_i, y_i) \]

Belief Propagation

**BELIEFS:** Approximate posterior marginal distributions

\[ \tilde{p}(x_i | y) \propto \psi_i(x_i, y) \prod_{k \in \Gamma(i)} m_{ki}(x_i) \]

**MESSAGES:** Approximate sufficient statistics

\[ m_{ij}(x_j) \propto \int \psi_{ij}(x_j, x_i) \psi_i(x_i, y) \prod_{k \in \Gamma(i) \setminus j} m_{ki}(x_i) dx_i \]

I. Belief Update (Message Product)
II. Message Propagation (Convolution)

Justifications for BP

- Gives exact marginals for trees
  - Optimal estimates
  - Confidence measures
- For general graphs, _loopy BP_ has excellent empirical performance in many applications
- Recent theory provides some guarantees:
  - Statistical physics: _variational method_ (Yedidia, Freeman, & Weiss)
  - BP as reparameterization: _error bounds_ (Wainwright, Jaakkola, & Willsky)
  - Many others…
References on BP and GBP

J. Pearl, 1985
- classic

Y. Weiss, NIPS 1998
- Inspires application of BP to vision

W. Freeman et al learning low-level vision, IJCV 1999
- Applications in super-resolution, motion, shading/paint discrimination

H. Shum et al, ECCV 2002
- Application to stereo

M. Wainwright, T. Jaakkola, A. Willsky
- Reparameterization version

J. Yedidia, AAAI 2000
- The clearest place to read about BP and GBP.

MRF results

Segmentation approaches

Real-time stereo

Nomad robot searches for meteorites in Antartica
http://www.frc.ri.cmu.edu/projects/meteorobot/index.html

Used for robot navigation (and other tasks)
- Several software-based real-time stereo techniques have been developed (most based on simple discrete search)
Using more than two images

Why does stereo fail?

Mono-stereo. Ordering: Points along an epipolar scanline appear in the same order in both stereo images.

Occlusion: All points are visible in each image.

Why does stereo fail?

Fronto-parallel Surfaces: Depth is constant within the region of local support.

Image Brightness Constancy: Assuming Lambertian surfaces, the brightness of corresponding points in stereo images are the same.
Why does stereo fail?

- Match uniqueness: For every point in one stereo image, there is at most one corresponding point in the other image.

- Camera calibration errors
- Poor image resolution
- Occlusions
- Violations of brightness constancy (specular reflections)
- Large motions
- Low-contrast image regions

Stereo reconstruction pipeline

Steps
- Calibrate cameras
- Rectify images
- Compute disparity
- Estimate depth

What will cause errors?
- Camera calibration errors
- Poor image resolution
- Occlusions
- Violations of brightness constancy (specular reflections)
- Large motions
- Low-contrast image regions