Lecture 24: Multi-view stereo
Announcements

• HW 2 out today
  – Due end of month. Problems as we cover topics.

• PA 4 out later this week
Road map

• What we’ve seen so far:
  – Low-level image processing: filtering, edge detecting, feature detection
  – Geometry: image transformations, panoramas, single-view modeling Fundamental matrices

• What’s next:
  – Finishing up geometry
    • Today: multi view stereo, graph cuts for stereo
    • Wed: structure from motion
  – Then: Recognition
  – If we have time: computational photography
Readings

• Szeliski, Chapter 7.1 – 7.4
Multi-view stereo

Stereo

Multi-view stereo
Multi-view Stereo

Point Grey's Bumblebee XB3

Point Grey's ProFusion 25

CMU's 3D Room
Multi-view Stereo
Multi-view Stereo

Input: calibrated images from several viewpoints
Output: 3D object model
(Next time: sFM feeds MVS)

Figures by Carlos Hernandez
Stereo: another view
Choosing the stereo baseline

What’s the optimal baseline?
  – Too large: difficult search problem
  – Too small: large depth error
Depth from disparity

\[
(B + X' - X)/(Z - f) = B/Z
\]

\[
(X - X') / f = B / X
\]

\[
X - X' = (B \times f) / z
\]

\[
z = (B \times f) / (X - X')
\]
The Effect of Baseline on Depth Estimation

Figure 2: An example scene. The grid pattern in the background has ambiguity of matching.
pixel matching score
Fig. 5. SSD values versus inverse distance: (a) $B = b$; (b) $B = 2b$; (c) $B = 3b$; (d) $B = 4b$; (e) $B = 5b$; (f) $B = 6b$; (g) $B = 7b$; (h) $B = 8b$.

The horizontal axis is normalized such that $8bf = 1$.

Fig. 6. Combining two stereo pairs with different baselines.

Fig. 7. Combining multiple baseline stereo pairs.
Multibaseline Stereo

Basic Approach
  – Choose a reference view
  – Use your favorite stereo algorithm BUT
    • replace two-view SSD with SSSD over all baselines

Limitations
  – Won’t work for widely distributed views
Problem: visibility

Some Solutions
- Match only nearby photos [Narayanan 98]
- Use NCC instead of SSD, Ignore NCC values < threshold [Hernandez & Schmitt 03]
Popular matching scores

• SSD (Sum Squared Distance)

\[ \sum_{x,y} |W_1(x, y) - W_2(x, y)|^2 \]

• NCC (Normalized Cross Correlation)

\[ \sum_{x,y} \frac{(W_1(x, y) - \overline{W_1})(W_2(x, y) - \overline{W_2})}{\sigma_{W_1} \sigma_{W_2}} \]

— where

\[ \overline{W_i} = \frac{1}{n} \sum_{x,y} W_i \]
\[ \sigma_{W_i} = \sqrt{\frac{1}{n} \sum_{x,y} (W_i - \overline{W_i})^2} \]

— Benefits
Multi-view stereo from Internet Collections
Challenges

• Appearance variation

• Resolution

• Massive collections

82,754 results for photos matching *notre* and *dame* and *paris*.
Law of Large Image Collections
• Automatically select neighboring views for each point in the image
• Desiderata: good matches AND good baselines

[Goesele, Snavely, Curless, Hoppe, Seitz, ICCV 2007]
Local view selection

- Automatically select neighboring views for each point in the image
- Desiderata: good matches AND good baselines
Results
Notre Dame de Paris

653 images
313 photographers
merged model of Pisa Cathedral
Accuracy compared to laser scanned model:
90% of points within 0.25% of ground truth
PMVS

• Patch-based Multi-view Stereo

Patch-based Multi-view Stereo Software
(PMVS - Version 2)

Software developed and distributed by
Yasutaka Furukawa - University of Illinois at Urbana-Champaign, University of Washington
Jean Ponce - University of Illinois at Urbana-Champaign, Ecole Normale Supérieure

• Next time: sFM feeds PMVS
Aside: 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
Results with window search

Window-based matching

Ground truth
Aside: 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
Stereo as energy minimization

- Find disparity map $d$ that minimizes an energy function $E(d)$

- Simple pixel / window matching
  
  $$E(d) = \sum_{(x,y) \in I} C(x, y, d(x, y))$$

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

\[ I(x, y) \]

\[ J(x, y) \]

\[ y = 141 \]

\[ C(x, y, d); \text{the disparity space image (DSI)} \]
Stereo as energy minimization

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

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

- Better objective function

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

- match cost
- smoothness cost

Want each pixel to find a good match in the other image
Adjacent pixels should (usually) move about the same amount
Stereo as energy minimization

$$E(d) = E_d(d) + \lambda E_s(d)$$

match cost:
$$E_d(d) = \sum_{(x,y) \in I} C(x, y, d(x, y))$$

smoothness cost:
$$E_s(d) = \sum_{(p,q) \in E} V(d_p, d_q)$$

$$E$$ : set of neighboring pixels

4-connected neighborhood

8-connected neighborhood
Smoothness cost

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

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

“Potts model”
Dynamic programming

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

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

- Finds “smooth” path through DSI from left to right
Dynamic Programming
Coherent stereo on 2D grid

• Scanline stereo generates streaking artifacts

• Can’t use dynamic programming to find spatially coherent disparities/ correspondences on a 2D grid
Stereo matching as energy minimization

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

- Energy functions of this form can be minimized using graph cuts

Y. Boykov, O. Veksler, and R. Zabih,
Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001
Before

Graph cuts

Y. Boykov, O. Veksler, and R. Zabih,
*Fast Approximate Energy Minimization via Graph Cuts*, PAMI 2001

For the latest and greatest: [http://www.middlebury.edu/stereo/](http://www.middlebury.edu/stereo/)